A doctoral thesis submitted to the Department of Applied Statistics, Operations Research and Quality

NOVEL CHEMOMETRIC PROPOSALS FOR ADVANCED MULTIVARIATE DATA ANALYSIS, PROCESSING AND INTERPRETATION

RAFFAELE VITALE

Supervisors: Alberto J. Ferrer-Riquelme
Onno E. de Noord

Valencia, September 2017

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Ph.D. Dissertation

Novel chemometric proposals for advanced multivariate data analysis, processing and interpretation

Author
Raffaele Vitale

Ph.D. supervisors
Alberto J. Ferrer-Riquelme
Onno E. de Noord

A doctoral thesis submitted to
Department of Applied Statistics, Operations Research and Quality

Valencia, September 2017

A Marica

Acknowledgements

As my experience in Valencia started on September 2012, I did not imagine the list of people to acknowledge would have become so long. But luckily, all over the latest 5 years I have had the chance to meet colleagues, companions, and friends, who have left a clear and deep mark in my life from both a personal and professional point of view. Therefore, if this manuscript has finally seen the light of day, the merit goes to many beloved people. First of all, to whom I consider my academic and scientific father, Alberto. I still remember with affection our meeting after having received the decision letter for the first paper we wrote together. 2 hours, 46 minutes and 39 seconds, as the chronometer of my mobile phone still reminds me. 2 hours, 46 minutes and 39 seconds of pure scientific, but informal, warm and kind cooperation, that taught me the three principles on which I would like to base the future (I hope) of my career: dedication, open-mindedness, respect. Let me quote Benjamin Franklin: *Tell me and I forget. Teach me and I remember. Involve me and I learn.* You have definitely involved me. Thank you very much. To Onno, whose continuous suggestions and incessant passion showed me how to unconditionally love chemometrics without ever losing contact with the problems of everyday life, but trying to find an easy, effective and constructive answer to them. I have extremely matured under your committed supervision. To Johan, Age and Harald. Being hosted by you was an enormous pleasure, having your names in this manuscript is an inestimable honour. To Federico and Cyril, for having never stopped believing in me. To those with whom I shared some of the most fantastic adventures of my life: Alessandro, because Valencia would not have been the same without you, and because if these years will always remain a vivid and unforgettable memory in my heart is mainly thanks to your priceless friendship; José Maria, for being the older brother I have never had; Eric and Dani, for being the best officemates I could have dreamt of. A special mention goes to José Manuel and Pili. The tunnel of life was darker sometimes. You were never scared to take my hands and join me to the next brighter spot. To Pepe and the *fellas* of the bar, because coming back home has never been so pleasant before I met you. To my best Valencian friends: Natxo, Alex, Quique and Abel. Natxo and Alex, both of you represent the main reason Valencia will forever be one of the destinations of my flights. Quique and Abel, we are currently spread all over the

world, but be sure that the endless thread linking us will continue being impossible to break. To mum, dad and Enea. Your support would deserve much more than a line on this page, because it was fundamental not only for the achievement of this important objective, but for all those I fulfilled during my entire life. To my grandmothers, for always being by my side no matter the distance that now separates us. And finally to you, Marica. This Ph.D. thesis is dedicated to you, because it is you who actually taught me the most beautiful lesson. We can sweat, we can make efforts and suffer, but at the end all our dreams may come true. You are the living proof my dream is now reality.

Abstract

The present Ph.D. thesis, primarily conceived to support and reinforce the relation between academic and industrial worlds, was developed in collaboration with Shell Global Solutions (Amsterdam, The Netherlands) in the endeavour of applying and possibly extending well-established latent variable-based approaches (i.e. Principal Component Analysis - PCA - Partial Least Squares regression - PLS - or Partial Least Squares Discriminant Analysis - PLSDA) for complex problem solving not only in the fields of manufacturing troubleshooting and optimisation, but also in the wider environment of multivariate data analysis. To this end, novel efficient algorithmic solutions are proposed throughout all chapters to address very disparate tasks, from calibration transfer in spectroscopy to real-time modelling of streaming flows of data.
The manuscript is divided into the following six parts, focused on various topics of interest:

Part I - Preface, where an overview of this research work, its main aims and justification is given together with a brief introduction on PCA, PLS and PLSDA;

Part II - On kernel-based extensions of PCA, PLS and PLSDA, where the potential of kernel techniques, possibly coupled to specific variants of the recently rediscovered pseudo-sample projection, formulated by the English statistician John C. Gower, is explored and their performance compared to that of more classical methodologies in four different applications scenarios: segmentation of Red-Green-Blue (RGB) images, discrimination of on-/off-specification batch runs, monitoring of batch processes and analysis of mixture designs of experiments;

Part III - On the selection of the number of factors in PCA by permutation testing, where an extensive guideline on how to accomplish the selection of PCA components by permutation testing is provided through the comprehensive illustration of an original algorithmic procedure implemented for such a purpose;

Part IV - On modelling common and distinctive sources of variability in multi-set data analysis, where several practical aspects of two-block common and distinctive component analysis (carried out by methods like Simultaneous Component Analysis - SCA - DIStinctive and COmmon Simultaneous Component Analysis - DISCO-SCA - Adapted Generalised Singular Value Decomposition - Adapted GSVD - ECO-POWER, Canonical Correlation Analysis - CCA - and 2-block Orthogonal Projections to Latent Structures - O2PLS) are discussed, a new computational strategy for determining the number of common factors underlying two data matrices sharing the same row- or column-dimension is described, and two innovative approaches for calibration transfer between near-infrared spectrometers are presented;

Part V - On the on-the-fly processing and modelling of continuous high-dimensional data streams, where a novel software system for rational handling of multi-channel measurements recorded in real time, the *On-The-Fly Processing* (OTFP) tool, is designed;

Part VI - Epilogue, where final conclusions are drawn, future perspectives are delineated, and annexes are included.

Resumen

La presente tesis doctoral, concebida principalmente para apoyar y reforzar la relación entre la academia y la industria, se desarrolló en colaboración con Shell Global Solutions (Amsterdam, Países Bajos) en el esfuerzo de aplicar y posiblemente extender los enfoques ya consolidados basados en variables latentes (es decir, Análisis de Componentes Principales - PCA - Regresión en Mínimos Cuadrados Parciales - PLS - o PLS discriminante - PLSDA) para la resolución de problemas complejos no sólo en los campos de mejora y optimización de procesos, sino también en el entorno más amplio del análisis de datos multivariados. Con este fin, en todos los capítulos proponemos nuevas soluciones algorítmicas eficientes para abordar tareas dispares, desde la transferencia de calibración en espectroscopia hasta el modelado en tiempo real de flujos de datos.
El manuscrito se divide en las seis partes siguientes, centradas en diversos temas de interés:

Parte I - Prefacio, donde presentamos un resumen de este trabajo de investigación, damos sus principales objetivos y justificaciones junto con una breve introducción sobre PCA, PLS y PLSDA;

Parte II - Sobre las extensiones basadas en kernels de PCA, PLS y PLSDA, donde presentamos el potencial de las técnicas de kernel, eventualmente acopladas a variantes específicas de la recién redescubierta proyección de pseudo-muestras, formulada por el estadista inglés John C. Gower, y comparamos su rendimiento respecto a metodologías más clásicas en cuatro aplicaciones a escenarios diferentes: segmentación de imágenes Rojo-Verde-Azul (RGB), discriminación y monitorización de procesos por lotes y análisis de diseños de experimentos de mezclas;

Parte III - Sobre la selección del número de factores en el PCA por pruebas de permutación, donde aportamos una guía extensa sobre cómo conseguir la selección de componentes de PCA mediante pruebas de permutación y una ilustración completa de un procedimiento algorítmico original implementado para tal fin;

Parte IV - Sobre la modelización de fuentes de variabilidad común y distintiva en el análisis de datos multi-conjunto, donde discutimos varios aspectos prácticos del análisis de componentes comunes y distintivos de dos bloques de datos (realizado por métodos como el Análisis Simultáneo de Componentes - SCA - Análisis Simultáneo de Componentes Distintivos y Comunes - DISCO-SCA - Descomposición Adaptada Generalizada de Valores Singulares - Adapted GSVD - ECO-POWER, Análisis de Correlaciones Canónicas - CCA - y Proyecciones Ortogonales de 2 conjuntos a Estructuras Latentes - O2PLS). Presentamos a su vez una nueva estrategia computacional para determinar el número de factores comunes subyacentes a dos matrices de datos que comparten la misma dimensión de fila o columna y dos planteamientos novedosos para la transferencia de calibración entre espectrómetros de infrarrojo cercano;

Parte V - Sobre el procesamiento y la modelización en tiempo real de flujos de datos de alta dimensión, donde diseñamos la herramienta de Procesamiento en Tiempo Real (OTFP), un nuevo sistema de manejo racional de mediciones multi-canal registradas en tiempo real;

Parte VI - Epílogo, donde presentamos las conclusiones finales, delimitamos las perspectivas futuras, e incluimos los anexos.

Resum

La present tesi doctoral, concebuda principalment per a recolzar i reforçar la relació entre l'acadèmia i la indústria, es va desenvolupar en col·laboració amb Shell Global Solutions (Amsterdam, Països Baixos) amb l'esforç d'aplicar i possiblement estendre els enfocaments ja consolidats basats en variables latents (és a dir, Anàlisi de Components Principals - PCA - Regressió en Mínims Quadrats Parcials - PLS - o PLS discriminant - PLSDA) per a la resolució de problemes complexos no solament en els camps de la millora i optimització de processos, sinó també en l'entorn més ampli de l'anàlisi de dades multivariades. A aquest efecte, en tots els capítols proposem noves solucions algorítmiques eficients per a abordar tasques dispars, des de la transferència de calibratge en espectroscopia fins al modelatge en temps real de fluxos de dades.
El manuscrit es divideix en les sis parts següents, centrades en diversos temes d'interès:

Part I - Prefaci, on presentem un resum d'aquest treball de recerca, es donen els seus principals objectius i justificacions juntament amb una breu introducció sobre PCA, PLS i PLSDA;

Part II - Sobre les extensions basades en kernels de PCA, PLS i PLSDA, on presentem el potencial de les tècniques de kernel, eventualment acoblades a variants específiques de la recentment redescoberta projecció de pseudo-mostres, formulada per l'estadista anglés John C. Gower, i comparem el seu rendiment respecte a metodologies més clàssiques en quatre aplicacions a escenaris diferents: segmentació d'imatges Roig-Verd-Blau (RGB), discriminació i monitorització de processos per lots i anàlisi de dissenys d'experiments de mescles;

Part III - Sobre la selecció del nombre de factors en el PCA per proves de permutació, on aportem una guia extensa sobre com aconseguir la selecció de components de PCA a través de proves de permutació i una il·lustració completa d'un procediment algorítmic original implementat per a la finalitat esmentada;

Part IV - Sobre la modelització de fonts de variabilitat comuna i distintiva en l'anàlisi de dades multi-conjunt, on discutim diversos aspectes pràctics de l'anàlisis de components comuns i distintius de dos blocs de dades (realitzat per mètodes com l'Anàlisi Simultània de Components - SCA - Anàlisi Simultània de Components Distintius i Comuns - DISCO-SCA - Descomposició Adaptada Generalitzada en Valors Singulars - Adapted GSVD - ECO-POWER, Anàlisi de Correlacions Canòniques - CCA - i Projeccions Ortogonals de 2 blocs a Estructures Latents - O2PLS). Presentem al mateix temps una nova estratègia computacional per a determinar el nombre de factors comuns subjacents a dues matrius de dades que comparteixen la mateixa dimensió de fila o columna, i dos plantejaments nous per a la transferència de calibratge entre espectròmetres d'infraroig proper;

Part V - Sobre el processament i la modelització en temps real de fluxos de dades d'alta dimensió, on dissenyem l'eina de Processament en Temps Real (OTFP), un nou sistema de tractament racional de mesures multi-canal registrades en temps real;

Part VI - Epíleg, on presentem les conclusions finals, delimitem les perspectives futures, i incloem annexos.

Contents

General notation **1**

Abbreviations and acronyms **3**

I Preface **7**

1 Justification, objectives and contributions **9**
1.1 An overview of the research project 9
1.2 Objectives of the thesis 10
1.2.1 Kernel-based methodologies for statistical process monitoring, improved fault diagnosis, translation of *out-of-control* signals to operator actions, and analysis of mixture designs of experiments 10
1.2.2 Rational approaches for selecting the optimal amount of information to be modelled for data exploration and understanding 11
1.2.3 Model transfer between manufacturing units or workstations 11
1.2.4 Real-time data processing 12
1.3 Contributions. 12

2 On latent variable-based multivariate data analysis **17**
2.1 Introduction 17
2.2 Latent variable-based multivariate data analysis techniques 19
2.2.1 Principal Component Analysis (PCA) 19
2.2.2 Partial Least Squares regression (PLS) 20
2.2.3 Partial Least Squares Discriminant Analysis (PLSDA) 21
2.3 Some important additional notions: *cross-validation*, *jackknifing* and *permutation testing* 21

3 Materials and methods **23**
3.1 Hardware 23
3.2 Software 23
3.3 Datasets and methods 23

II On kernel-based extensions of PCA, PLS and PLSDA **25**

4 Preliminary considerations **27**
4.1 Introduction . . . 28
4.1.1 Kernel-based techniques: basic principles . . . 29
4.1.2 Pseudo-samples and pseudo-sample projection . . . 29

5 K-PLSDA for RGB image segmentation **33**
5.1 Introduction . . . 34
5.2 Methods . . . 36
5.2.1 Colour analysis-based segmentation techniques . . . 36
5.2.2 Texture analysis-based segmentation techniques . . . 37
5.2.3 Graph-based segmentation techniques . . . 38
5.2.4 Multivariate Image Analysis (MIA)-based segmentation techniques . . . 38
5.3 Dataset . . . 39
5.4 Comparative study . . . 40
5.5 Results . . . 41
5.6 Illustration case . . . 47
5.7 Concluding remarks . . . 51

6 K-PLSDA and pseudo-sample projection for batch run discrimination **53**
6.1 Introduction . . . 54
6.2 Methods . . . 54
6.3 Datasets . . . 55
6.4 Results and discussion . . . 56
6.4.1 Simulated dataset . . . 56
6.4.2 VWU/K-PLSDA (polymerisation process dataset) . . . 61
6.4.3 BWU/K-PLSDA (polymerisation process dataset) . . . 63
6.4.4 LFE/K-PLSDA (pharmaceutical batch drying process dataset) . . . 67
6.5 Comparison between K-PLSDA and classical PLSDA models . . . 70
6.6 Conclusions . . . 71

7 K-PCA and pseudo-sample projection for batch process monitoring **73**
7.1 Introduction . . . 74
7.2 Adaptation of the pseudo-sample projection strategy to batch process monitoring . . . 74
7.3 Datasets . . . 77
7.4 Results and discussion . . . 78
7.4.1 Simulated dataset - Variability increase detection case study . . . 78
7.4.2 Simulated dataset - Variability decrease detection case study . . . 89
7.4.3 Chemical process dataset . . . 94
7.4.4 Pharmaceutical batch drying process dataset . . . 100

7.5 Conclusions . . . 101

8 K-PLS and pseudo-sample trajectories for mixture data analysis 103
8.1 Introduction . . . 104
8.2 Methods . . . 105
8.2.1 Scheffé and Cox models . . . 105
8.2.2 Pseudo-sample trajectories for mixture data . . . 106
8.2.3 Pseudo-sample-based response surfaces . . . 108
8.3 Datasets . . . 109
8.3.1 Simulated data . . . 109
8.3.2 Tablet data . . . 109
8.3.3 Bubbles data . . . 110
8.3.4 Colorant data . . . 110
8.3.5 Photographic paper data . . . 110
8.4 Results . . . 110
8.4.1 Simulated data . . . 110
8.4.2 Tablet data . . . 112
8.4.3 Bubbles data . . . 114
8.4.4 Colorant data . . . 117
8.4.5 Photographic paper data . . . 120
8.5 Conclusions . . . 120

III On the selection of the number of factors in PCA by permutation testing 125

9 A novel permutation test-based approach for PCA component selection 127
9.1 Introduction . . . 128
9.1.1 Strategies for principal component selection . . . 128
9.2 Methodology . . . 129
9.3 Theoretical and practical aspects of the algorithm . . . 132
9.3.1 Permutations . . . 133
9.3.2 Deflation . . . 133
9.3.3 Projection . . . 134
9.3.4 The rationale behind F_a . . . 138
9.4 Performance of the algorithm . . . 139
9.4.1 Synthetic datasets . . . 139
9.4.2 Real case studies . . . 143
9.5 Conclusions . . . 148

IV On modelling common and distinctive sources of variability in multi-set data analysis 149

10 Some considerations on two-block common and distinctive component analysis 151

10.1 Introduction . . . 152
10.2 Methods . . . 154
10.2.1 Simultaneous Component Analysis (SCA) . . . 154
10.2.2 DIStinctive and COmmon Simultaneous Component Analysis (DISCO-SCA) . . . 155
10.2.3 Adapted Generalised Singular Value Decomposition (Adapted GSVD) 155
10.2.4 ECO-POWER . . . 156
10.2.5 Canonical Correlation Analysis (CCA) . . . 156
10.2.6 2-block Orthogonal Projections to Latent Structures (O2PLS) . . . 157
10.3 Datasets . . . 158
10.3.1 Gene expression data . . . 158
10.3.2 Simulated pseudo-spectral data . . . 158
10.3.3 Industrial batch process data . . . 159
10.4 Results and discussion . . . 159
10.4.1 Is the variation accounted for reliable? . . . 159
10.4.2 Determining the number of common and distinctive components: a novel strategy . . . 161
10.4.3 Modelling common and distinctive components in regression scenarios . . . 169
10.5 Conclusions . . . 174

11 Calibration transfer between near-infrared spectrometers 175

11.1 Introduction . . . 176
11.2 Methods . . . 177
11.2.1 Piecewise Direct Standardisation (PDS) . . . 177
11.2.2 Maximum Likelihood Principal Component Analysis (MLPCA) . . . 177
11.2.3 Trimmed Scores Regression (TSR) . . . 178
11.2.4 JYPLS-based approaches . . . 179
11.3 Modelling procedure . . . 180
11.4 Datasets . . . 182
11.5 Results . . . 183
11.5.1 Gasoline dataset . . . 183
11.5.2 Corn dataset . . . 187
11.6 Discussion . . . 190
11.7 Conclusions . . . 191

V On the on-the fly processing and modelling of continuous high-dimensional data streams **193**

12 The On-The-Fly Processing tool **195**
12.1 Introduction . . . 196
12.1.1 Data compression strategies . . . 196
12.1.2 Subspace compression . . . 198
12.1.3 PCA as a multivariate series expansion of the underlying data generation mechanism . . . 198
12.1.4 Algorithms for PCA decomposition . . . 199
12.2 System overview . . . 200
12.2.1 Input . . . 202
12.2.2 Fit to already established model subspace . . . 203
12.2.3 Bilinear model expansion . . . 204
12.2.4 Model updating . . . 204
12.3 Datasets . . . 205
12.4 Results and discussion . . . 206
12.4.1 High-speed multi-channel monitoring of the Belousov-Zhabotinsky reaction . . . 206
12.4.2 Detailed remote characterisation of orange samples . . . 208
12.4.3 Environmental surveillance by airborne hyperspectral imaging . . . 211
12.4.4 Analysis of an industrial manufacturing process . . . 213
12.5 Comparison with classical PCA . . . 215
12.6 Discussion . . . 216
12.7 Conclusions and perspectives . . . 217

VI Epilogue **219**

13 Conclusions and perspectives **221**
13.1 Accomplishment of the objectives . . . 221
13.1.1 Objective I - Exploring the potential of kernel-based methodologies for statistical process monitoring, improved fault diagnosis, translation of *out-of-control signals* to operator actions, and analysis of mixture designs of experiments . . . 221
13.1.2 Objective II - Proposing rational approaches for selecting the optimal amount of information to be modelled for data exploration and understanding . . . 222
13.1.3 Objective III - Enhancing model transfer between manufacturing units or workstations . . . 222
13.1.4 Objective IV - Implementing new computational strategies for real-time data processing . . . 223
13.2 Future research lines . . . 224

14 Appendices **227**
14.1 Annex to Part II 227
14.1.1 Relationship between the Euclidean distance matrix, **D**, and the inner product matrix, $\mathbf{XX}^{\mathrm{T}}$ 227
14.1.2 Practical meaning of the pseudo-samples in the feature space 228
14.1.3 Relationship between Scheffé and Cox model coefficients 229
14.2 Annex to Part III 231
14.2.1 Horn's parallel analysis 231
14.2.2 Dray's method 231
14.3 Annex to Part IV 233
14.3.1 The JYPLS algorithm 233
14.3.2 Principal Component Regression (PCR) 235
14.4 Annex to Part V 235
14.4.1 Multivariate Curve Resolution-Alternating Least Squares (MCR-ALS) 235

Bibliography **237**

General notation

In the present Ph.D. thesis, matrices and column vectors are represented by upper-case and lower-case bold characters, respectively (with $\times$ denoting their dimensions and $^{\mathrm{T}}$ the transpose operator). Italic capital letters define constants, while italic lower-case font is used for scalars and variables. Functions are identified by bold italic lower-case alphabetical symbols. Plain subscripts indicate fixed indices, italic subscripts varying ones. Square brackets connote scalar, vector or matrix concatenation, while curly brackets symbolise intervals.
Additional information about more specific mathematical notation is provided in every single chapter of this manuscript.

Abbreviations and acronyms

ANOVA: ANalysis Of VAriance
BWU: Batch-Wise Unfolding
B-Z: Belousov-Zabhotinsky
CA: Correspondence Analysis
CCA: Canonical Correlation Analysis
CPU: Central Processing Unit
CR: Compression Ratio
CV: Cross-Validation
CVS: Computer Vision Systems
***DD*:** Discriminant Distance
DISCO-SCA: DIStinctive and COmmon Simultaneous Component Analysis
DWL1: Dish-Washing Liquid 1
DWL2: Dish-Washing Liquid 2
EFA: Evolving Factor Analysis
EMSC: Extended Multiplicative Signal Correction
EV: Explained Variance
FFT: Fast Fourier Transform
F-H: Felzenszwalb-Huttenlocher approach
FN: False Negatives
FP: False Positives
GCCA: Generalised Canonical Correlation Analysis
GLS: Generalised Least Squares
GSVD: Generalised Singular Value Decomposition
HSE: Health, Safety and Environment
ICA: Independent Component Analysis
ISL: Imposed Significance Level
JPEG: Joint Photographic Experts Group
JYPLS: Joint-Y Partial Least Squares
***K*-M:** K-Means
K-PCA: Kernel-Principal Component Analysis
K-PLS: Kernel-Partial Least Squares
K-PLSDA: Kernel-Partial Least Squares Discriminant Analysis

KS: Kennard-Stone
LAIV: Live Attenuated Influenza Vaccine
LFE: Landmark Feature Extraction
LSD: Least Significant Difference
LV: Latent Variable
MCR-ALS: Multivariate Curve Resolution-Alternating Least Squares
MIA: Multivariate Image Analysis
MLPCA: Maximum Likelihood Principal Component Analysis
MLR: Multiple Linear Regression
MPEG: Moving Picture Experts Group
MSC: Multiplicative Scatter Correction
MVI: Manual Visual Inspection
NC: Nearest Centroid
NIPALS: Non-linear Iterative PArtial Least Squares
NIR: Near-InfraRed
NOC: Normal Operating Conditions
O2PLS: 2-block Orthogonal Projections to Latent Structures
OLS: Ordinary Least Squares
OTFP: On-The-Fly Processing
***OTI*:** Overall Type I
***OTII*:** Overall Type II
PARAFAC: PARAllel FACtor analysis
PC: Principal Component
PCA: Principal Component Analysis
PCR: Principal Component Regression
PDS: Piecewise Direct Standardisation
PLS: Partial Least Squares
PLSDA: Partial Least Squares Discriminant Analysis
RGB: Red-Green-Blue
RMSECV: Root Mean Square Error in Cross-Validation
RMSEP: Root Mean Square Error in Prediction
RMSRE: Root Mean Square Reconstruction Error
RSS: Residuals Sum-of-Squares
SCA: Simultaneous Component Analysis
SIMCA: Soft Independent Modelling of Class Analogy
SIMPLISMA: SIMPLe-to-use Interactive Self-modeling Mixture Analysis
SNV: Standard Normal Variate
SPE: Squared Prediction Error
SVD: Singular Value Decomposition
TCS: Trajectory Centring and Scaling
***TI*:** Type I
***TII*:** Type II
TIIV: Trivalent Inactivated Influenza Vaccine
TN: True Negatives

TP: True Positives
TSR: Trimmed Scores Regression
UV: UltraViolet
***VAF*:** Variation Accounted For
VCS: Variable Centring and Scaling
Vis: Visible
***VR*:** Variance Ratio
VWU: Variable-Wise Unfolding

Part I

Preface

Chapter 1

Justification, objectives and contributions

1.1 An overview of the research project

The research project associated to the present Ph.D. thesis, entitled *Advanced multivariate methods for the analysis and monitoring of chemical processes* (contract number PT13698), resulted from the cooperation involving the Department of Applied Statistics and Operation Research and Quality of the Technical University of Valencia (Valencia, Spain) and Shell Global Solutions (Amsterdam, The Netherlands). Its main aim was to promote the synergy between academia and industry by the development of novel algorithmic technologies for *real world* applications in particular fields of interest, including:

1. innovative methodologies for statistical process monitoring, improved fault diagnosis and translation of *out-of-control* signals to operator actions;
2. rational approaches for selecting the optimal amount of information to be modelled for data exploration and understanding;
3. data-driven (empirical) model transfer between e.g. manufacturing units or workstations;
4. real-time data processing.

Such technologies would potentially guarantee:

- increased plant utilisation (cycle time and production loss reduction) and reliability;

- improved HSE (*Health, Safety and Environment*) performance;
- more consistent product quality;
- reduced costs (by e.g. reduced energy consumption and reduced storage).

For all these reasons, the outcomes of the studies reported here may lead to breakthroughs not only for industrial troubleshooting and manufacturing optimisation, but also in the more general contexts of BIG DATA analytics and computational statistics.

1.2 Objectives of the thesis

The objectives of this Ph.D. thesis and the proposals to accomplish them will now be described in more detail.

1.2.1 Kernel-based methodologies for statistical process monitoring, improved fault diagnosis, translation of *out-of-control* signals to operator actions, and analysis of mixture designs of experiments

Nowadays, huge quantities of data are routinely collected by automated sensors during e.g. the evolution of industrial processes to preserve the quality of the final products and prevent abnormal events to affect their manufacturing. Although several algorithmic strategies exist to properly model such data, their degree of complexity might be so high that the appropriate detection of *out-of-control* situations and the identification of the root causes of these potential failures are sometimes unfeasible. More advanced statistical techniques are then required to address these tasks. Coupling the well-known kernel-based methods and the so-called pseudo-sample projection may represent a valid alternative in similar contingencies. Part II (Chapters 4-8) will be devoted to the description of their theoretical and practical aspects. Four different applications (not only strictly industrial) will also permit to evaluate their performance for RGB (Red-Green-Blue) image analysis and segmentation, fault detection and diagnosis and analysis of mixture designs of experiments.
This work was partially carried out in collaboration with Prof. Lutgarde M.C. Buydens and Dr. Geert J. Postma from Radboud Universiteit Nijmegen, Dr. José Blasco from the Instituto Valenciano de Investigaciones Agrarias and Prof. Fernando López-García from the Instituto de Automática e Informática Industrial of the Technical University of Valencia.

1.2.2 Rational approaches for selecting the optimal amount of information to be modelled for data exploration and understanding

Correctly setting the complexity of multivariate empirical models is a critical step to attain a reasonable and reliable interpretation of the data one is investigating. To this end, in the last decades, much work has been devoted to techniques like statistical tests and cross-validation, but scarce attention has been paid to the possibility of resorting to permutation testing for the same purpose. A novel and efficient permutation test-based algorithm for multivariate data model complexity determination will be illustrated in Part III (Chapter 9) of this Ph.D. thesis. Its most relevant theoretical and practical aspects will be discussed and clarified and its pros over more standard solutions highlighted.
This work was partially carried out in collaboration with Prof. Age K. Smilde and Dr. Johan A. Westerhuis from Universiteit van Amsterdam and Prof. Tormod Næs from Nofima AS.

1.2.3 Model transfer between manufacturing units or workstations

One of the possible ways of achieving a model transfer is to identify and describe the common data variation generated e.g. by the same industrial process run in two distinct reactors or by various instrumentations used to characterise a certain set of samples so that calibrations or monitoring schemes originally developed for a specific manufacturing unit or analytical platform can be exploited for prediction or control on a second one. In this respect, attempting to distinguish the shared and distinctive sources of variability underlying multiple blocks of measurements has recently become of intriguing interest in many research domains. In Part IV (Chapters 10-11) of this Ph.D. thesis, a new methodology to perform such a disentanglement will be proposed and tested in simulated and real case studies. Along the same lines, two novel strategies for calibration transfer between near-infrared spectrometers will also be presented and compared with classical reference ones.
This work was partially carried out in collaboration with Prof. Age K. Smilde and Dr. Johan A. Westerhuis from the Universiteit van Amsterdam.

1.2.4 Real-time data processing

Today's society and industry are undoubtedly flooded by constantly streaming data, typical of the modern world of multimedia communication. Unfortunately, most of the traditional computing systems are generally not capable of analysing them in real time as they flow. A novel approach to simultaneously compress and model continuous high-dimensional data streams is introduced in Part V (Chapter 12) of this Ph.D. thesis. Four examples will show how it can ease their storage, transfer, retrieval, visualisation, interpretation and quantitative utilisation.
This work was partially carried out in collaboration with Prof. Harald Martens and Mr. Joao F. Fortuna from Norges Teknisk-Naturvitenskapelige Universitet and Ms. Anna Zhyrova from Jihočeská Univerzita v Českých Budêjovicích.

1.3 Contributions

Here is a list of all the contributions authored by the candidate during the progress of the aforementioned research project:

Peer-reviewed publications

1. Vitale, R., de Noord, O. & Ferrer, A. A kernel-based approach for fault diagnosis in batch processes. *J. Chemometr.* **28**, 697-707 (2014).

2. Vitale, R., de Noord, O. & Ferrer, A. Pseudo-sample based contributions plots: innovative tools for fault diagnosis in kernel-based batch process monitoring. *Chemometr. Intell. Lab.* **149**, 40-52 (2015).

3. Vitale, R., Prats-Montalbán, J., López-García, F., Blasco, J. & Ferrer, A. Segmentation techniques in image analysis: a comparative study. *J. Chemometr.* **30**, 749-758 (2016).

4. Vitale, R., Zhyrova, A., Fortuna, J., de Noord, O., Ferrer, A. & Martens, H. On-The-Fly Processing of continuous high-dimensional data streams. *Chemometr. Intell. Lab.* **161**, 118-129 (2017).

5. Folch-Fortuny, A., Vitale, R., de Noord, O. & Ferrer, A. Calibration transfer between NIR spectrometers: new proposals and a comparative study. *J. Chemometr.* **31**, e2874 (2017).
   V Siemens Process Analytics Prize for Young Scientists

6. Vitale, R., Westerhuis, J., Næs, T., Smilde, A., de Noord, O. & Ferrer, A. Selecting the number of factors in Principal Component Analysis by permutation testing - Numerical and practical aspects. *J. Chemometr.*, In press.

7. Vitale, R., Palací-López, D., Kerkenaar, H., Postma, G., Buydens, L. & Ferrer, A. Kernel-Partial Least Squares regression coupled to pseudo-sample trajectories for the analysis of mixture designs of experiments. *Submitted*.

Poster communications in conferences

1. Vitale, R., de Noord, O. & Ferrer, A. Kernel-based Batch Multivariate Statistical Process Monitoring: better discrimination with a better understanding. *XIII Scandinavian Symposium on Chemometrics - SSC13*, 17-20/06/2013, Djurönäset, Sweden.

2. Vitale, R., de Noord, O. & Ferrer, A. Advanced multivariate methods for the analysis and monitoring of chemical processes. *I Encuentro de Estudiantes de Doctorado de la Universitat Politècnica de València*, 12/06/2014, Valencia, Spain.

3. Kerkenaar, H., Vitale, R., Palací-López, D., Postma, G., Buydens, L. & Ferrer, A. Kernel-Partial Least Squares regression coupled to pseudo-sample projection for the analysis of mixture designs of experiments. *XVI Chemometrics in Analytical Chemistry - CAC 2016*, 06-10/06/2016, Barcelona, Spain.
   XVI Chemometrics in Analytical Chemistry (CAC 2016) Conference Best Poster Award - Session 1: Theory and Method Development & DOE, SPC and PAT

Oral communications in conferences

1. Vitale, R., de Noord, O. & Ferrer, A. Kernel-based Batch Multivariate Statistical Process Monitoring: better discrimination with a better understanding. *XXIV Congresso della Divisione di Chimica Analitica della Società Chimica Italiana*, 15-19/09/2013, Sestri Levante, Italy.

2. Vitale, R., de Noord, O. & Ferrer, A. A kernel-based approach for fault diagnosis in batch process monitoring. *V Workshop de Quimiometría para Jóvenes Investigadores*, 17-18/10/2013, Badajoz, Spain.

3. Vitale, R., de Noord, O. & Ferrer, A. Kernel-based models for batch process monitoring: an industrial case study. *III European Conference on Process Analytics and Control Technology - EuroPACT 2014*, 06-09/05/2014, Barcelona, Spain.

4. Vitale, R., de Noord, O. & Ferrer, A. "Through the kernels": pseudo-sample based fault diagnosis in batch process monitoring. *VI International Chemo-*

metrics Research Meeting - ICRM 2014, 14-18/09/2014, Berg en Dal/Nijmegen, The Netherlands.

5. Vitale, R., de Noord, O. & Ferrer, A. A novel proposal for the identification of common and distinctive sources of variability in multi-set data. *VIII ThRee-Way Methods in Chemistry and Psychology - TRICAP 2015*, 31/05-05/06/2015, Pecol-Val di Zoldo, Italy.

6. Vitale, R., de Noord, O., González-Martínez, J. & Ferrer, A. Improving batch process fault diagnosis by combining kernel-based methods and pseudo-sample projection. *XIV Scandinavian Symposium on Chemometrics - SSC14*, 14-17/06/2015, Chia, Italy.

7. Folch-Fortuny, A., Vitale, R., de Noord, O. & Ferrer, A. Fast and efficient calibration transfer between near infrared instruments imputing unmeasured spectra. *XIV Scandinavian Symposium on Chemometrics - SSC14*, 14-17/06/2015, Chia, Italy.

8. Vitale, R., Westerhuis, J., de Noord, O., Smilde, A. & Ferrer, A. Identifying and retrieving common and distinctive components underlying two object-wise linked data blocks. *VI Chemometrics Workshop for Young Researchers*, 01-02/10/2015, Valencia, Spain.

9. Vitale, R., Westerhuis, J., Smilde, A., de Noord, O., Næs, T. & Ferrer, A. Permutation testing in PCA - Theoretical and practical aspects. *XVI Chemometrics in Analytical Chemistry - CAC 2016*, 06-10/06/2016, Barcelona, Spain.

10. Vitale, R., Prats-Montalbán, J., López-García, F., Blasco, J. & Ferrer, A. A comprehensive comparison of different RGB image segmentation techniques. *VI Conference in Spectral Imaging - IASIM 2016*, 03-06/07/2016, Chamonix-Mont Blanc, France.
International Association of Spectral Imaging (IASIM) Student Award

11. Vitale, R. Calibration transfer between near-infrared spectrometers: a comprehensive overview. *XXVI Congresso della Divisione di Chimica Analitica della Società Chimica Italiana*, 18-22/09/2016, Giardini Naxos, Italy.
Invited keynote presentation

12. Vitale, R., Westerhuis, J., Smilde, A., de Noord, O. & Ferrer, A. A novel proposal for the identification of the number of common and distinctive components in multi-set data analysis. *Nordic Arctic Workshop*, 10-11/11/2016, Groningen, The Netherlands.

Additional contributions not related to the content of the Ph.D. thesis

1. Vitale, R., Bevilacqua, M., Bucci, R., Magrì, A.D., Magrì, A.L. & Marini, F. A rapid and non-invasive method for authenticating the origin of pistachio samples by NIR spectroscopy and chemometrics. *Chemometr. Intell. Lab.* **121**, 90-99 (2013).
   ScienceDirect TOP25 Hottest Articles
   The content of this article constituted part of the Master's thesis of the candidate, which was awarded as the 2012 Best Master's Thesis in Analytical Chemistry by the Divisione di Chimica Analitica della Societá Chimica Italiana

2. González-Martínez, J., Vitale, R., de Noord, O. & Ferrer, A. Effect of synchronization on bilinear batch process modeling. *Ind. Eng. Chem. Res.* **53**, 4339-4351 (2014).

3. Vitale, R. Practical Three-Way Calibration - Book Review. *J. Chemometr.* **29**, 323 (2015).

4. Sales, R., Vitale, R., de Lima, S., Pimentel, M., Stragevitch, L. & Ferrer, A. Multivariate statistical process control charts for batch monitoring of transesterification reactions for biodiesel production based on near-infrared spectroscopy. *Comput. Chem. Eng.* **94**, 343-353 (2016).

5. Debus, B., Vitale, R., Sasaki, S., Asahi, T., Sliwa, M. & Ruckebusch, C. A multivariate curve resolution approach to separate UV-visible scattering and absorption contributions for organic nanoparticles. *Chemometr. Intell. Lab.* **160**, 72-76 (2017)

6. Vidal-Puig, S., Vitale, R. & Ferrer, A. Data-driven supervised fault diagnosis methods based on latent variable models: a comparative study. *In preparation*

7. González-Martínez, J., Vitale, R., de Noord, O. & Ferrer, A. Does synchronization matter in Batch Multivariate Statistical Process Control? *XIII Scandinavian Symposium on Chemometrics - SSC13*, 17-20/06/2013, Djurönäset, Sweden.

8. Sales, R, Vitale, R., Lima, S., Pimentel, M., Stragevitch, L. & Ferrer, A. Multivariate control charts based on near infrared spectroscopy for monitoring transesterification reactions for biodiesel production. *XVI Chemometrics in Analytical Chemistry - CAC 2016*, 06-10/06/2016, Barcelona, Spain.

9. Vidal-Puig, S., Ferrer, A. & Vitale, R. A comparative study of different methodologies for supervised fault diagnosis in multivariate statistical pro-

cess control. *XII Annual Conference of the European Network for Business and Industrial Statistics - ENBIS 2014*, 21-25/09/2014, Linz, Austria.

Chapter 2

On latent variable-based multivariate data analysis

2.1 Introduction

Nowadays, many measurement technologies can generate massive amounts of data in a very short time and commonly via one-step analytical procedures. For instance:

- advanced spectrometers can deliver hundreds of informative, high-dimensional spectra per second;
- innovative hyperspectral cameras produce multivariate spatially resolved images. In addition, when configured in a time-lapse mode, they can yield continuous streams of high-dimensional spatiotemporal recordings;
- industrial monitoring for condition-based maintenance, as well as the control of complex dynamic processes, requires high-dimensional inputs to be sufficiently informative;
- computer experiments, needed in order to study the behaviour of complex mathematical models, involve advanced workstations performing thousands of simulations, each one possibly characterised by just as many input and output properties.

Hence, a measurement revolution (recently termed *data tsunami* [1]) is currently taking place in numerous fields of applied science, ranging from analytical chemistry and medicine to environmental surveillance and informatics. However, these incredibly quick advances run the risk of being practically useless if the analysis

and the interpretation of such a huge volume of information is carried out in an inappropriate manner. Consider, e.g., a present-day industrial environment. A very high number of statistical process control schemes, which are ordinarily resorted to for keeping the manufacture *in-control*, are based on charting one or small numbers of measured variables (temperatures, pressure, flow rates *etc.*) in a univariate way. Even if this strategy could be effective in the past, it is now proven to be completely inadequate for coping with the huge quantities of instrumental responses routinely and automatically registered in modern plants, especially because undesired special events, which may jeopardise the quality of the final products, affect not only their magnitude, but also their relationships. As pointed out by Figure 2.1, these changes are difficult to detect by classical statistical approaches due to the fact they are only able to handle independent or, at most, slightly correlated variables (rarely met when processes are monitored by many automated sensors),

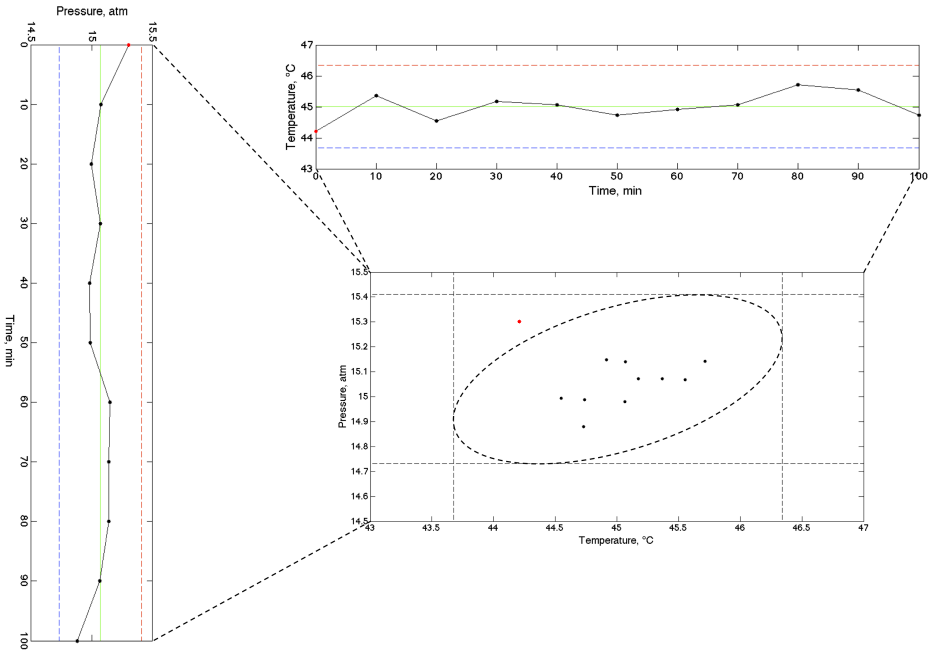

Figure 2.1 Univariate *vs* bivariate control charts. By monitoring two different engineering variables (i.e. temperature and pressure) separately and independently, no *out-of-control* signal is detected. On the contrary, by representing one of them as function of the other, an abnormal event is clearly detected (red dot). This supports the claim that any process control scheme should capture the correlation among variables to be capable of identifying ongoing failures affecting the multivariate nature of the data under study. Notice that the blue and red dotted lines denote the lower and upper control limits, respectively, associated to the temperature and pressure registered values, while the dotted ellipses embraces the corresponding bivariate *in-control* process behaviour region

and, thus, often impossible to correct for maintaining the so-called Normal Operating Conditions (NOC). Conversely, there exist alternative methods, based on the so-called *latent variables*, which take advantage of such a correlation for modelling the structure of the process space and describing the typically few independent events occurring during its evolution. In the last decades, these techniques have been found to be extremely powerful in case large numbers of measured variables, possibly showing low content of useful information (i.e. low signal-to-noise ratio) and/or the presence of missing entries, have to be dealt with.
In this Ph.D. thesis, algorithmic extensions of these methodologies (presented in the distinct chapters of the manuscript) will be proposed and/or exploited to address complex problems in advanced multivariate data analysis scenarios, from digital image treatment to the on-the-fly processing of information streaming in real time.

2.2 Latent variable-based multivariate data analysis techniques

Multivariate latent variable-based methods (widely employed in *chemometrics*) reduce the dimensionality of the data under study by projecting the high-dimensional space spanned by the original measured variables onto a low-dimensional subspace defined by orthogonal (uncorrelated) *latent* ones, which describe their underlying sources of variation. Principal Component Analysis (PCA), Partial Least Squares regression (PLS) and Partial Least Squares Discriminant Analysis (PLSDA) are undoubtedly the most representative and widespread members of this class of techniques.

2.2.1 Principal Component Analysis (PCA)

PCA [2, 3] is probably the most commonly used multivariate statistical tool to compress, describe and interpret large sets of data. Its basic principle can be summarised as follows: let $\mathbf{X}$ be a $N \times J$ matrix with J denoting the number of variables (e.g. J *sensor* responses monitored during an industrial process or J wavelengths of light scanned in a spectroscopy experiment) registered for each of N measurements performed, for instance, at N time instants or for N different individuals. As aforementioned, when J is very large, the useful and meaningful information in $\mathbf{X}$ is usually intercorrelated among various of these variables over the whole set of recordings. Then, for a chosen degree of acceptable accuracy, it is possible to reduce the J-dimensional space of the original descriptors to an A-dimensional subspace where data mostly vary and onto which all the N objects under study can be projected and represented as new points. Mathematically speaking, PCA is based on the bilinear structure model in Equation 2.1:

$$\mathbf{X} = \mathbf{1}\mathbf{m}^{\mathrm{T}} + \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} \tag{2.1}$$

where $\mathbf{1}$ $(N \times 1)$ is a vector of ones, $\mathbf{m}$ $(J \times 1)$ contains the mean values of the J variables in $\mathbf{X}$, $\mathbf{P}$ $(J \times A)$ is an array of so-called *loadings*, which determine the A basis vectors (*principal components* or *factors*) of the PCA subspace, $\mathbf{T}$ $(N \times A)$ defines the projection coordinates or *scores* of all the N rows of $\mathbf{X}$ on this lower-dimensional space and $\mathbf{E}$ $(N \times J)$ stands for the matrix of unmodelled residuals, i.e. the portion of $\mathbf{X}$ not *explained* at the chosen rank, A.
The PCA solution may be formulated in many equivalent ways and attained by a variety of algorithms, among which the most widespread and popular one is certainly Singular Value Decomposition (SVD) [4][i]. Here, it is assumed to show the following properties:

$$\mathbf{P}^{\mathrm{T}}\mathbf{P} = \mathbf{I} \tag{2.2}$$

$$\mathbf{T}^{\mathrm{T}}\mathbf{T} = \mathrm{diag}(\boldsymbol{\lambda}_A) \tag{2.3}$$

where $\mathbf{I}$ is an identity matrix of dimensions $A \times A$, while the a-th element of $\boldsymbol{\lambda}_A$ $(A \times 1)$ corresponds to the eigenvalue of the a-th PCA component.

2.2.2 Partial Least Squares regression (PLS)

PLS is a latent variable-based regression approach for modelling the intrinsic relationships between a matrix of predictors, say $\mathbf{X}$ $(N \times J)$, and a set of response variables, $\mathbf{Y}$ $(N \times M)$. The basic idea behind this technique is estimating such responses from the A-dimensional subspace of $\mathbf{X}$ which maximises its covariance with $\mathbf{Y}$. Thus, the PLS structure model can be written as:

$$\begin{aligned} \mathbf{X} &= \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} \\ \mathbf{Y} &= \mathbf{T}\mathbf{Q}^{\mathrm{T}} + \mathbf{F} \\ \mathbf{Y} &= \mathbf{X}\mathbf{B} + \mathbf{F} \\ \mathbf{B} &= \mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1}\mathbf{Q}^{\mathrm{T}} = \mathbf{W}^{*}\mathbf{Q}^{\mathrm{T}} \end{aligned} \tag{2.4}$$

being $\mathbf{T}$ $(N \times A)$, $\mathbf{P}$ $(J \times A)$ and $\mathbf{E}$ $(N \times J)$ the so-called $\mathbf{X}$-scores, $\mathbf{X}$-loadings and $\mathbf{X}$-residuals matrices, respectively; $\mathbf{Q}$ $(M \times A)$ and $\mathbf{F}$ $(N \times M)$ the so-called $\mathbf{Y}$-loadings and $\mathbf{Y}$-residuals matrices, respectively; $\mathbf{W}$ $(J \times A)$ an array of weights; and $\mathbf{B}$ $(J \times M)$ an array of regression coefficients.
By PLS one does not need to assume linearly independent regressors as for most classical statistical predictive methods like Classical Least Squares (CLS).

[i] By SVD $\mathbf{X}$ is decomposed as:

$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}} + \mathbf{E}$$

with the columns of $\mathbf{U}$ $(N \times A)$ and $\mathbf{V}$ $(J \times A)$ being the left and right singular vectors of $\mathbf{X}$, respectively, and $\mathbf{S}$ $(A \times A)$ a square diagonal array whose diagonal elements are its first A non-zero singular values. Then, it holds that $\mathbf{T} = \mathbf{U}\mathbf{S}$ and $\mathbf{P}$=$\mathbf{V}$.

2.2.3 Partial Least Squares Discriminant Analysis (PLSDA)

PLSDA [5, 6] is the direct extension of PLS, developed for classification problem solving: $\mathbf{X}$ is regressed via PLS on a dummy binary-coded response matrix, say again $\mathbf{Y}$, made up by a set of piled Z-dimensional row vectors[ii], constructed so that, if their corresponding objects/samples are members of the z-th class, they have a 1-value in their z-th entry and 0-values in all the other ones. Whenever new objects/samples become available, the *a posteriori* probabilities that each one of them belongs to the Z categories are calculated. The assignation is finally carried out according to either the highest-probability or the higher-than-a-threshold probability rule. Notice that the model structure in Equation 2.4 applies here as well.

2.3 Some important additional notions: *cross-validation*, *jackknifing* and *permutation testing*

All over the manuscript, the reader will often encounter references to three key-concepts related to both uni- and multivariate statistics: *cross-validation*, *jackknifing* and *permutation testing*. Cross-validation refers to a model validation technique commonly resorted to for determining the number of latent variables to extract in predictive (e.g. PLS and PLSDA) models. Concretely, i) the data under study are split into several complementary subsets, ii) the analysis is carried out on all but 1 of these subsets, and iii) the left out one is exploited for testing purposes. To reduce variability, multiple rounds of cross-validation are performed under different partitions and the final results are averaged over such rounds[iii]. Some reflections on the use of cross-validation when PCA is concerned can be found in Part III.

Jackknifing is a resampling approach by which a particular statistic (usually variance or bias) is quantified by i) systematically removing an observation from a data matrix, ii) computing iteratively the estimate of interest, and iii) calculating the mean or a certain percentile of the resulting distribution of values.

Finally, a *permutation test* (also called a *randomisation test*) permits to evaluate the statistical significance of a specific parameter by empirically defining a *null*-distribution H_0 rearranging the labels associated to the various considered objects. For instance, in a classification scenario, this would imply randomly assigning samples to the distinct categories taken into account.

[ii] Z equals the number of categories to be discriminated.

[iii] Generally, the evolution of the prediction error with the model complexity (number of latent variables) is assessed.

Chapter 3

Materials and methods

3.1 Hardware

All the computations executed for the elaboration of this Ph.D. thesis were run on a MacBook Pro equipped with a 2.3 GHz Intel Core i7 and 8 GB 1600 MHz DDR3 RAM.

3.2 Software

The software packages exploited here are:

- macOS Sierra, Version 10.12.2;
- MATLAB R2012b, Version 8.0.0.783.

The whole set of scripts and functions resorted to for data treatment, processing, analysis and interpretation was self-coded and is available on request.

3.3 Datasets and methods

Owing to the high number of investigated datasets and compared methods and due to their different nature and properties, the reader can refer to every single chapter of this manuscript for an overview of the materials and methods employed and presented in it.

Part II

On kernel-based extensions of PCA, PLS and PLSDA

Chapter 4

Preliminary considerations

Part of the content of this chapter has been included in:

1. Vitale, R., de Noord, O. & Ferrer, A. A kernel-based approach for fault diagnosis in batch processes. *J. Chemometr.* **28**, 697-707 (2014).

2. Vitale, R., de Noord, O. & Ferrer, A. Pseudo-sample based contributions plots: innovative tools for fault diagnosis in kernel-based batch process monitoring. *Chemometr. Intell. Lab.* **149**, 40-52 (2015).

3. Vitale, R., Prats-Montalbán, J., López-García, F., Blasco, J. & Ferrer, A. Segmentation techniques in image analysis: a comparative study. *J. Chemometr.* **30**, 749-758 (2016).

4. Vitale, R.[i], Palací-López, D.[i], Kerkenaar, H., Postma, G., Buydens, L. & Ferrer, A. Kernel-Partial Least Squares regression coupled to pseudo-sample trajectories for the analysis of mixture designs of experiments. *Submitted*.

[i]These authors had equal contributions

4.1 Introduction

Although PCA, PLS and PLSDA are currently recognised as some of the most powerful techniques for the analysis and interpretation of multivariate data, the presence of strong non-linear relationships among objects and/or variables may represent a difficult issue to solve when one tries to model them by means of these methods. In fact, in principle such tools are not able to directly describe the underlying structure of datasets affected by severe non-linearities, since they *a priori* assume this structure to be linear [7]. In the last decades, many novel approaches have been proposed to handle similar situations like non-linear PLS [8–15] or artificial neural networks [16]. Nevertheless, these strategies often require the optimisation of many adjustable parameters and may show overfitting and local minima. A good alternative is represented by the so-called kernel-based techniques [17], which also comprehend support vector machines [18] and have already been broadly used in chemistry [19, 20], biology [21], informatics [22, 23] and continuous process monitoring [24, 25]. The first aim of this chapter is to explore their potential in contexts where the high complexity of the concerned problems might drastically complicate troubleshooting by classical PCA, PLS or PLSDA. Specifically, the power of kernel-based methodologies will be assessed in 4 different application scenarios:

1. segmentation of RGB images;

2. discrimination of on-/off-specification batch runs;

3. monitoring of industrial batch processes;

4. analysis of mixture designs of experiments.

Even if kernel-based approaches permit to easily cope with strong non-linear relationships in data, their main disadvantage is associated to the fact that the information about the importance or the influence of the original variables to the final models is lost. Many possibilities to recover this information exist, but authors commonly abstain from resorting to them for three reasons: i) their implementation is not straightforward; ii) most of them do not permit to graphically and intuitively interpret the resulting outcomes; iii) when they are used for fault diagnosis in continuous process monitoring, their employment cannot prescind from comparing the possible detected failures with a database of previously observed ones, which is rarely available when dealing with real industrial case studies [20, 26–29]. Recently, the principles of non-linear bi-plots and so-called pseudo-sample projection, originally described by Gower and Hardings in [30], have been extended

to overcome all these limitations [31–34]. Here, they will be adapted and exploited, when needed[ii], to enable model interpretation and get useful insights into the *black box* typical of kernel-based methods.

4.1.1 Kernel-based techniques: basic principles

The framework of all the kernel-based data analysis techniques is common and based on the so-called *kernel transformation*, given by:

$$K(\mathbf{x}_n, \mathbf{x}_{n^*}) =< \phi(\mathbf{x}_n), \phi(\mathbf{x}_{n^*}) > \tag{4.1}$$

where $\mathbf{x}_n^{\mathrm{T}}$ and $\mathbf{x}_{n^*}^{\mathrm{T}}$ are two row vectors of the original data matrix to which a specific mapping function ϕ is applied, while $<$ and $>$ denote the inner product. If one applies this transformation to every possible couple of vectors constituting a generic array, $\mathbf{X}$, with dimensions $N \times J$, it will be converted into a squared symmetric $N \times N$ kernel matrix, $\mathbf{K}$, whose elements constitute dissimilarity or distance measurements between two different observations.
When dealing with kernel-based approaches, it is not needed to know the mapping function *a priori*. There are many generic kernel functions one can resort to for obtaining $\mathbf{K}$ and all of them exhibit two fundamental properties: i) they allow the original data to be projected onto a higher-dimensional space, the feature space; ii) they provide a way to calculate the inner product between observations in such a feature space. The former permits to describe in a linear way possible non-linear relationships in $\mathbf{X}$. The latter makes all the algorithms of classical multivariate linear methodologies, which are based on the calculation of the inner product matrix of $\mathbf{X}$ (e.g. PCA, PLS and PLSDA), suitable for being applied in this higher-dimensional feature space [7]. In this part of the thesis, only three types of kernel functions will be taken into account: the linear (first-order), the q^{th}-order polynomial and the Gaussian. Their mathematical formulations are listed in Table 4.1 together with the indication of their possible adjustable parameters. Therefore, once the kernel matrix, $\mathbf{K}$, has been computed, a classical bilinear technique can be applied to it: in case one uses PCA, PLS or PLSDA, a Kernel-PCA (K-PCA), a Kernel-PLS (K-PLS) or a Kernel-PLSDA (K-PLSDA) model is generated, respectively.

4.1.2 Pseudo-samples and pseudo-sample projection

A pseudo-sample corresponds to a particular observation, which carries all the weight in one single variable. For example, the vector $\mathbf{g}^{\mathrm{T}} = [0, 0, \ldots, 1, 0, \ldots, 0]$ represents one of the possible pseudo-samples associated to the j-th variable of a generic dataset. By projecting an observation like this onto the latent structure of

[ii] This will be the case for all the applications under study except for RGB image segmentation.

Table 4.1 Kernel functions referred to in this part of the thesis and list of their adjustable parameters

Kernel type	Kernel function	Adjustable parameters
Linear (first-order)	$\mathbf{x}_n^{\mathrm{T}}\mathbf{x}_{n^*}$	None
q^{th}-order polynomial	$(\mathbf{x}_n^{\mathrm{T}}\mathbf{x}_{n^*})^q$	None
Gaussian	$\exp(-\frac{\|\|\mathbf{x}_n-\mathbf{x}_{n^*}\|\|^2}{2\sigma})$	σ

a classical 1-component PCA model[iii], the score for this new sample is calculated by Equation 4.2:

$$t_{\mathbf{g}^{\mathrm{T}}} = \mathbf{g}^{\mathrm{T}}\mathbf{p} = [0, 0, \ldots, 1, 0, \ldots, 0]\mathbf{p} = p_j \tag{4.2}$$

This score is equal to the j-th value of the loadings vector $\mathbf{p}$, and, thus, gives information about the contribution of the variable x_j to the model. Creating a pseudo-sample matrix, $\mathbf{V}_j$, which contains in its j-th column an arbitrary number (e.g. V) of values ranging from the minimum to the maximum of the variable x_j and 0 in all the other entries, as:

$$\mathbf{V}_j = \begin{bmatrix} 0, & 0, & \ldots, & \min(x_j), & 0, & \ldots, & 0 \\ & & & \cdots & & & \\ & & & \cdots & & & \\ 0, & 0, & \ldots, & \max(x_j), & 0, & \ldots, & 0 \end{bmatrix} \tag{4.3}$$

and projecting it onto the latent space, a trajectory is obtained according to the following equation:

$$\mathbf{V}_j\mathbf{p} = \begin{bmatrix} \min(x_j)p_j \\ \cdots \\ \cdots \\ \max(x_j)p_j \end{bmatrix} \tag{4.4}$$

In a higher-dimensional model space (i.e. when more than one latent variable is considered), the matrix resulting from the previous operation would define the geometrical locus of all the points lying along the direction determined by the origin of the latent space and the point, whose coordinates represent the weights of x_j on the A calculated components. In a classical PCA/PLS/PLSDA model, representing such a trajectory does not provide any additional information, but, as will be shown later, it is possible to get an idea from this kind of plot about how the original variable evolves in the latent space when kernel-based methods are applied. In this regard, Postma *et al.* [32] have recently demonstrated pseudo-sample projection permits to recover the information related to the contribution

[iii] It is straightforward to extend the following mathematical derivation to PLS and PLSDA models.

of the original variables when dealing with a Euclidean distance matrix, say **D**. Moreover, as **D** (double-centred) is directly generated applying a linear kernel transformation to a generic mean-centred dataset (see Section 14.1.1 for further details), it is possible to resort to this strategy even when one uses K-PCA, K-PLS or K-PLSDA. In this circumstance, it is only needed to transform each pseudo-sample array into a pseudo-sample kernel one by the same transformation as for **X**. The result is a $V \times N$ array, which contains information about the dissimilarity between the V pseudo-samples and the N original observations. The mathematical derivation of this extension is described in Section 14.1.2. In addition, this is valid not only in case one is exploiting a linear function to transform the analysed data. The pseudo-sample projection can in fact be utilised when dealing with all the kernel transformations, provided that they generate sets of distances which may be embedded in a Euclidean space [30].

Chapter 5

K-PLSDA for RGB image segmentation

In this chapter K-PLSDA is compared to some of the most commonly used RGB image segmentation approaches in an orange quality control case study.

Part of the content of this chapter has been included in:

1. Vitale, R., Prats-Montalbán, J., López-García, F., Blasco, J. & Ferrer, A. Segmentation techniques in image analysis: a comparative study. *J. Chemometr.* **30**, 749-758 (2016).

5.1 Introduction

The quality evaluation of products and processes mostly depends on the identification of features which are distinct from standard patterns. Nowadays, such an identification is more and more often carried out by Computer Vision Systems (CVS) comparing images collected along the production chain with reference ones. Nevertheless, if these references are not available, the aforementioned quality assessment cannot be addressed by direct comparison or pattern recognition. This is generally the case in e.g. fruit industry, whose products may exhibit completely different shapes, colours and/or defects even if collected in the same area or from the same tree. Considering in addition that a single inspection line commonly controls about 9 tons of fruits per hour flowing at a very high speed, the problem seems far from being simple to solve. For this reason, CVS have had limited success and diffusion in this specific field, where Manual Visual Inspection (MVI) still plays a predominant role. On the other hand, MVI clearly lacks objectivity and is deeply influenced by the mood and/or the fatigue of the operators. Furthermore, it is biased by both between- and within-inspector variability. Therefore, it is important to develop automatic image processing techniques, capable of coping with these kinds of issues.

One of the first steps in image processing is the so-called *segmentation*. In the computer vision field, segmentation is usually defined as the process of partitioning a digital image into multiple segments (sets of pixels also known as super-pixels) and is aimed at distinguishing the different objects or regions of interest present in a picture. More precisely, it permits to assign each one of its pixels to a specific class or category so that those belonging to the same subgroup share certain characteristics [35]. Such a task can be accomplished by numerous disparate methods, which are commonly classified according to various criteria [36, 37], for instance whether they are supervised (they take advantage on the *a priori* knowledge on the class-belonging of a specific set of pixels for the assignation of new ones) or unsupervised (they look for clusters or groups of pixels, which are not known beforehand) [38]. Here, a classification based on the nature of the information exploited for assigning each pixel to a specific class or region and on the data modelling approach is proposed. Specifically, i) colour analysis-based, ii) texture analysis-based, iii) graph-based and iv) Multivariate Image Analysis (MIA)-based techniques are distinguished.

The main objective of this chapter is to compare some of the segmentation strategies representative of these four categories (comprising K-PLSDA - see Table 5.1) and, concretely, determine which ones enable a correct identification of sound and green areas as well as of scale blemishes on the surface of several orange samples. Such strategies, namely Nearest Centroid (NC), K-Means (K-M), Standard Deviation, Range, Entropy, Felzenszwalb-Huttenlocher approach (F-H), PLSDA, K-PLSDA, Q-statistic and D-statistic, are probably the most popular, widespread and commonly used (in the forms and configurations described in the Section 11.2) to tackle problems of this type [36, 46, 47] and can be found coded in many

Table 5.1 Overview of the image segmentation techniques under comparison

Colour analysis-based segmentation techniques	Texture analysis-based segmentation techniques	Graph-based segmentation techniques	Multivariate Image Analysis (MIA)-based segmentation techniques
Nearest Centroid (NC) [39] ***K*-Means (*K*-M)** [40]	Standard deviation [41] Range [41] Entropy [41]	**Felzenszwalb-Huttenlocher approach (F-H)** [42]	PLSDA [5, 6] K-PLSDA [32] *Q*-statistic [43, 44] *D*-statistic [43–45]

Bold text indicates unsupervised segmentation techniques

ready-to-use routines for programming suites like MATLAB [48] and R [49]. This study does, however, not aim at identifying which of them constitute the best segmentation methods to be applied in fruit quality control case studies, but is rather an attempt of determining their most relevant differences and highlighting their pros and cons by means of classical statistical approaches, i.e. ANalysis Of VAriance (ANOVA) and Correspondence Analysis (CA). No similar works have been reported in the scientific literature before.

5.2 Methods

As specified before, segmentation consists of assigning each pixel of an image to a specific category or class, which defines its nature. In this case, pixels belonging to sound, green or blemished orange peel areas are to be discriminated. This can be achieved according to different classification rules, depending on the algorithmic procedure underlying the adopted methodology, and exploiting different kinds of information that can be extracted directly from the picture.

5.2.1 Colour analysis-based segmentation techniques

These techniques directly deal with the three colour intensity values (red, green and blue) of the pixels of the various images under study. These images are unfolded into two-dimensional arrays so that each one of their rows contains the intensity values associated to a single specific pixel and subsequently analysed as such [37].

Nearest Centroid (NC)

The Nearest Centroid (NC) classification, also known as Nearest Prototype or Rocchio classification, is a non-parametric approach usually exploited for pattern recognition purposes [39]. Unlabelled pixels are classified as belonging to the category to whose centroid (estimated from a set of training data) their distance (here, euclidean) is minimum.

K-Means (K-M)

K-Means (K-M) is a vector quantisation algorithm, popular for unsupervised cluster analysis [40]. It aims at partitioning all the pixels of an image into K classes in an iterative fashion:

1. The centroids of the K classes are initialised;
2. Every pixel of the image is assigned to the class with the nearest centroid;

3. The centroids of the K classes are recalculated after the classification step;

4. The procedure is repeated until a convergence criterion is met.

Clearly, there is no need of training data to carry out a segmentation by K-M[i].

5.2.2 Texture analysis-based segmentation techniques

Texture analysis provides information about pixel intensity changes within predefined spatial domains [41]. In this work, the original RGB images are converted to grey-scale ones. Afterwards, each new intensity value is substituted by a first-order statistic, derived from a neighbouring window surrounding its corresponding pixel and capturing the aforementioned local variability[ii]. If an unlabelled pixel is found to be characterised by a value of this first-order statistic within a particular interval, determined based on training data and typical of one of the considered classes, it is assigned to such a category.

Standard deviation

Probably the best known index to perform a textural transform of RGB images is standard deviation. Generally, it applies that the smoother (more homogeneous) the examined image area, the lower its respective standard deviation values, and *vice versa*.

Range

As for standard deviation, also range permits to catch the textural information contained in RGB images. Smooth textures (homogeneous image regions) usually result in lower range values and *vice versa*.

Entropy

Entropy relates to the non-homogeneity of a scene [50]. It is estimated as:

$$E = -\sum_{i \in H} p_i \log p_i \qquad \sum_i p_i = 1 \tag{5.1}$$

being p_i the relative frequency of the i-th intensity value in the neighbouring window of interest and H its global intensity range.
Homogeneous image regions are known to feature low entropy, and *vice versa*.

[i] Each image is handled separately and independently from the others.

[ii] Specifically, 3×3 pixel windows were circumscribed to compute standard deviation, range and entropy.

5.2.3 Graph-based segmentation techniques

Graph-based segmentation techniques look at RGB images as kinds of networks, also known as *graphs*, i.e. sets of edges connecting certain pairs of adjacent pixels (vertices or nodes). Each one of these edges is associated to a weight, which is function of the dissimilarity between the pixels it connects (e.g. the difference in their intensity, location or some other spatial attribute). Segmentation is then addressed by finding a subgroup of meaningful edges, separating image regions encompassing pixels with different properties. Several decision criteria can be taken into account to achieve the identification of such a subgroup. Here, the algorithmic procedure proposed by Felzenszwalb and Huttenlocher in [42] (F-H) is applied, which attains the selection of the significant edge weights by adaptively assessing both pixel intensity differences across boundaries (to be maximised) and intensity differences between neighbouring pixels within single delimited areas (to be minimised)[iii]. Once performed a single segmentation per colour channel separately, the three of them are intersected in order to achieve the final pixel discrimination.

5.2.4 Multivariate Image Analysis (MIA)-based segmentation techniques

Multivariate Image Analysis (MIA) stands for the study of the image-intrinsic information through multivariate models. The basic principle of MIA is to unfold the investigated images into a matrix, say $\mathbf{X}$, whose row and column dimensions relate to the pixel mode and the colour and/or texture mode, respectively, and analyse the resulting data structure by means of e.g. PCA [51] or PLS [52]. Details about the various ways of accomplishing this unfolding step[iv] are provided in [37]. As the basic principles of PLSDA and K-PLSDA have already been presented in Sections 2.2.3 and 4.1.1, only Q-statistic and D-statistic will now be thoroughly described.

Q- and D-statistic

If PCA is applied to every single subset of $\mathbf{X}$ (namely $\mathbf{X}_z$), including exclusively the information associated to the pixels belonging to the z-th concerned category, Z independent class models are subsequently built. Thus, unlabelled pixels can be discriminated according to two distance indices:

- Q-statistic, reflecting their perpendicular distance to each model hyperplane [43, 44];

[iii] Not even this segmentation strategy requires training data.

[iv] Here, both colour and textural information were combined to conduct the comparison.

- D-statistic, reflecting the distance from the origin of each model hyperplane to their projection onto it [43–45].

For a fair comparison, such pixels are assigned to the class, for whose model they show the lowest $\frac{Q_z}{Q_{z,95}}$ or $\frac{D_z}{D_{z,95}}$ ratio, where $Q_{z,95}$ and $D_{z,95}$ denote an empirical 95% confidence threshold for Q and D, respectively, estimated from training data.

5.3 Dataset

30 RGB images of sweet oranges, collected from the Citrus Germplasm Bank at the Instituto Valenciano de Investigaciones Agrarias, were recorded by a Canon EOS 550D digital camera with a resolution of 0.0625 mm/pixel, installed in an inspection chamber (see Figure 5.1) internally illuminated by eight fluorescent tubes (OSRAM L 18W/965 BIOLUX, colour temperature = 6500 K, colour rendering index > 90%).

The samples were collected at different ripening stages and, therefore, their

Figure 5.1 Image capturing computer vision system

colour varied from green to orange depending on their maturity status. Most of them were also characterised by particular dark spots, mainly generated by California red scale (*Aonidiella aurantii*), long mussel scale (*Lepidosaphes gloverii*) and purple mussel scale (*Lepidosaphe backii*) infestations, which depreciated their commercial value and whose detection is then of utmost importance from a quality control perspective. Hence, three distinctive features were observed: sound orange-coloured, green-coloured and blemished peel areas (see Figure 5.2).
Except for K-M and Graphs, 3330 training pixels per each one of the aforementioned regions were randomly selected from 5 images to establish the segmentation

criteria or calibrate the segmentation/classification models[v]. The potential of the single methods illustrated in Section 11.2 was evaluated by exploiting the remaining 25 test pictures.

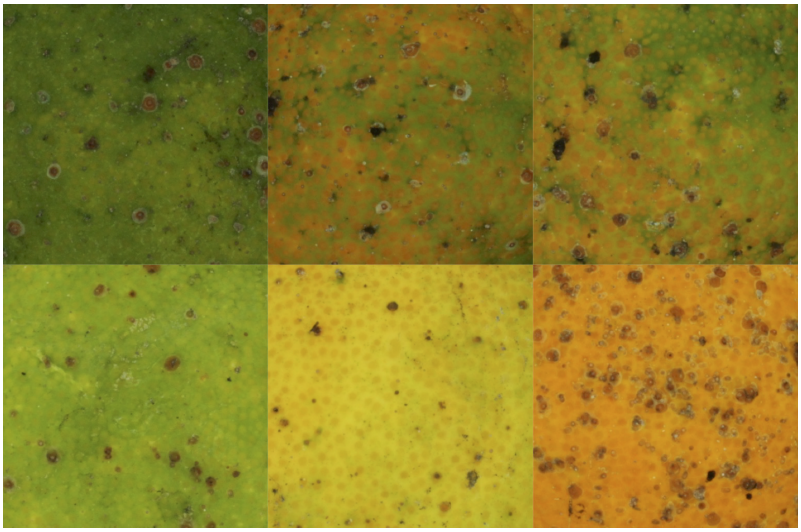

Figure 5.2 Details of 6 different orange images highlighting sound, green-coloured and blemished peel areas

5.4 Comparative study

Once determined the membership class of all the test pixels, the image segmentations led to by the various adopted approaches were compared to reference ones manually elaborated by an expert operator. Segmentation accuracy degree was assessed in terms of F − score, computed for the n-th image and the z-th category as:

$$\mathrm{F-score}_{n,z} = 2 \times \frac{\mathrm{precision}_{n,z} \times \mathrm{recall}_{n,z}}{\mathrm{precision}_{n,z} + \mathrm{recall}_{n,z}} \qquad \forall n = 1, 2, \ldots, 25 \qquad \forall z = 1, 2, 3 \tag{5.2}$$

being

$$\mathrm{precision}_{n,z} = \frac{\mathrm{TP}_{n,z}}{\mathrm{TP}_{n,z} + \mathrm{FP}_{n,z}} \tag{5.3}$$

[v]For K-PLSDA, several functions with a growing degree of non-linearity were tested: linear, 2^{nd}-9^{th}-order polynomial and Gaussian. Notice that the last one embraces a supplementary parameter, σ, which needs to be adjusted. Values of σ varying from 0.5 (yielding an extremely non-linear transformation) to 5000 (yielding an approximately linear transformation) were spanned.

$$\text{recall}_{n,z} = \frac{\text{TP}_{n,z}}{\text{TP}_{n,z} + \text{FN}_{n,z}} \tag{5.4}$$

where $\text{TP}_{n,z}$, $\text{FP}_{n,z}$ and $\text{FN}_{n,z}$ stand for *True Positives* (the number of pixels of the n-th image correctly identified as belonging to the z-th category), *False Positives* (the number of pixels of the n-th image mistakenly identified as belonging to the z-th category) and *False Negatives* (the number of pixels of the n-th image mistakenly identified as not belonging to the z-th category), respectively.
For every z-th class, statistically significant differences among the F-score indices associated to the techniques under study and constituting measures of their general performance were detected by a two-way ANalysis of VAriance (ANOVA) taking into account one fixed factor (i.e. segmentation method) and one blocking factor (i.e. orange sample). For a more comprehensive overview of their power, pros and cons, the contingency tables made up by the total number per class of True Positives, True Negatives (TN, the number of pixels correctly identified as not belonging to the z-th category), False Positives and False Negatives, resulting from the distinct methodologies, were investigated by means of Correspondence Analysis (CA) [53], which is conceptually similar to PCA, but was proposed for categorical rather than continuous data processing.

5.5 Results

Tables 5.2, 5.3 and 5.4 list the global values of TP, FP, FN, TN, precision, recall and F-score, related to the 25 test images, yielded by the different segmentation strategies for the sound peel area, green peel area and surface blemishes class. As the effect of the two factors included in the ANOVA models was found to be statistically significant ($p-$values $\ll 0.05$), the 95% Least Significant Difference (LSD) intervals were calculated for each single considered category. They are displayed in Figures 5.3a, 5.3b and 5.3c[vi]. Clearly, PLSDA and K-PLSDA outmatched the other approaches under study (F-H was as effective as them only when detecting surface blemishes). Colour analysis- (NC and K-M) and texture analysis-based techniques (Standard deviation, Range and Entropy) were not able to satisfactorily identify green and blemished peel areas. NC worked generally better than K-M while D-statistic always guaranteed a statistically significantly larger F-score than Q-statistic. It is also important to notice that the outcomes associated to the surface blemishes class are on average negatively biased: according to Table 5.4 most of the methods generated an acceptable quantity of TP and FN (high recall) but too many FP (low precision). This is due to the extreme variability affecting the intensity of the pixels belonging to it.
By looking at the CA bi-plots in Figures 5.4a, 5.4b and 5.4c, some additional insights can be gleaned. As the first two CA dimensions relate to either TP/FN

[vi]Notice that the LSD intervals are centred around the average F-scores calculated over the 25 test images, which do not necessarily correspond to the global values shown in Tables 5.2, 5.3 and 5.4.

Table 5.2 Global TP, FP, FN, TN, precision, recall and F-score yielded by the various approaches under study for the sound peel area class. Bold characters indicate the highest precision, recall and F-score values

Sound peel area							
Segmentation method	True Positives (TP)	False Positives (FP)	False Negatives (FN)	True Negatives (TN)	Precision	Recall	F-score
NC	1310877	127934	480670	648080	0.9111	0.7317	0.8116
K-M	615968	269077	1175579	506937	0.6960	0.3438	0.4603
Standard deviation	825091	299396	966456	476618	0.7337	0.4605	0.5659
Range	540394	197470	1251153	578544	0.7324	0.3016	0.4273
Entropy	854303	324538	937244	451476	0.7247	0.4769	0.5752
F-H	1568223	583715	223324	192299	0.7287	**0.8753**	0.7953
PLSDA	1562519	20410	229028	755604	**0.9871**	**0.8722**	**0.9261**
K-PLSDA	156610	20506	224937	755508	**0.9871**	**0.8744**	**0.9274**
Q-statistic	733050	88726	1058497	687288	0.8920	0.4092	0.5610
D-statistic	1148340	27108	643207	748906	0.9769	0.6410	0.7741

Table 5.3 Global TP, FP, FN, TN, precision, recall and F-score yielded by the various approaches under study for the green peel area class. Bold characters indicate the highest precision, recall and F-score values

Green peel area							
Segmentation method	True Positives (TP)	False Positives (FP)	False Negatives (FN)	True Negatives (TN)	Precision	Recall	F-score
NC	280877	158046	413699	1476671	0.6399	0.4044	0.4956
K-M	230189	567488	464387	1067229	0.2886	0.3314	0.3085
Standard deviation	277198	598843	417378	1035874	0.3164	0.3991	0.3530
Range	347590	778806	346986	855911	0.3086	0.5004	0.3818
Entropy	282777	529609	411799	1105108	0.3481	0.4071	0.3753
F-H	509836	1276225	184740	358492	0.2855	0.7340	0.4111
PLSDA	634050	124372	60526	1510345	**0.8360**	**0.9129**	**0.8727**
K-PLSDA	639258	121954	55318	1512763	**0.8398**	**0.9204**	**0.8782**
Q-statistic	328220	210111	366356	1424606	0.6097	0.4725	0.5324
D-statistic	449446	155983	245130	1478734	0.7424	0.6471	0.6915

N.B. 2 out of the 25 test images did not contain pixels belonging to the green peel area class and were then excluded from the comparative study for this particular category.

Table 5.4 Global TP, FP, FN, TN, precision, recall and F-score yielded by the various approaches under study for the surface blemishes class. Bold characters indicate the highest precision, recall and F-score values

Surface blemishes							
Segmentation method	True Positives (TP)	False Positives (FP)	False Negatives (FN)	True Negatives (TN)	Precision	Recall	F-score
NC	63125	625429	18300	1860707	0.0917	0.7753	0.1640
K-M	29949	703116	51476	1783020	0.0409	0.3678	0.0735
Standard deviation	66787	393741	14638	2092395	0.1450	0.8202	0.2465
Range	70302	492197	11123	1993939	0.1250	0.8634	0.2184
Entropy	58780	400583	22645	2085553	0.1280	0.7219	0.2174
F-H	50767	224394	30658	2261742	0.1845	0.6235	0.2847
PLSDA	69446	156762	11979	2329374	**0.3070**	0.8529	**0.4515**
K-PLSDA	69602	152471	11823	2333665	**0.3134**	0.8548	**0.4587**
Q-statistic	78198	1118405	3227	1367731	0.0653	**0.9604**	0.1224
D-statistic	76330	709570	5095	1776566	0.0971	**0.9374**	0.1760

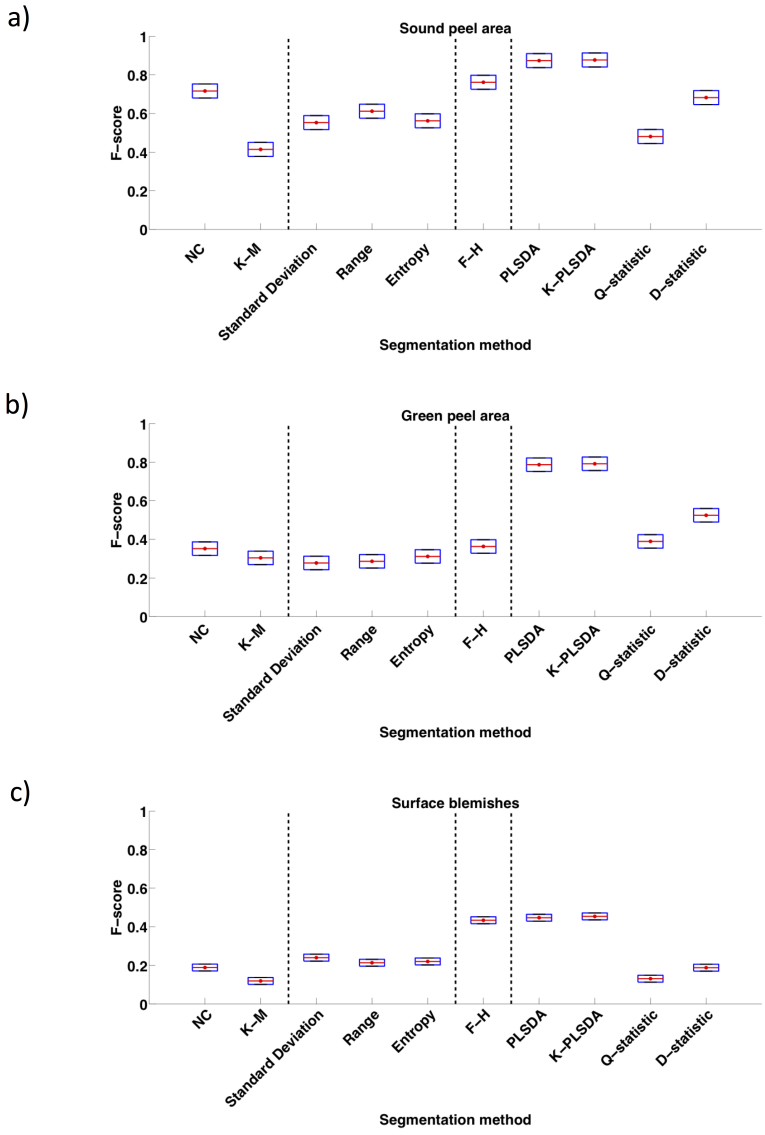

Figure 5.3 95% Least Significant Difference (LSD) intervals resulting from the ANOVA models built on the F-score indices yielded by the compared segmentation approaches for a) the sound peel area class, b) the green peel area class and c) the surface blemishes class. The vertical dotted lines separate the different categories of methods (see Table 5.1)

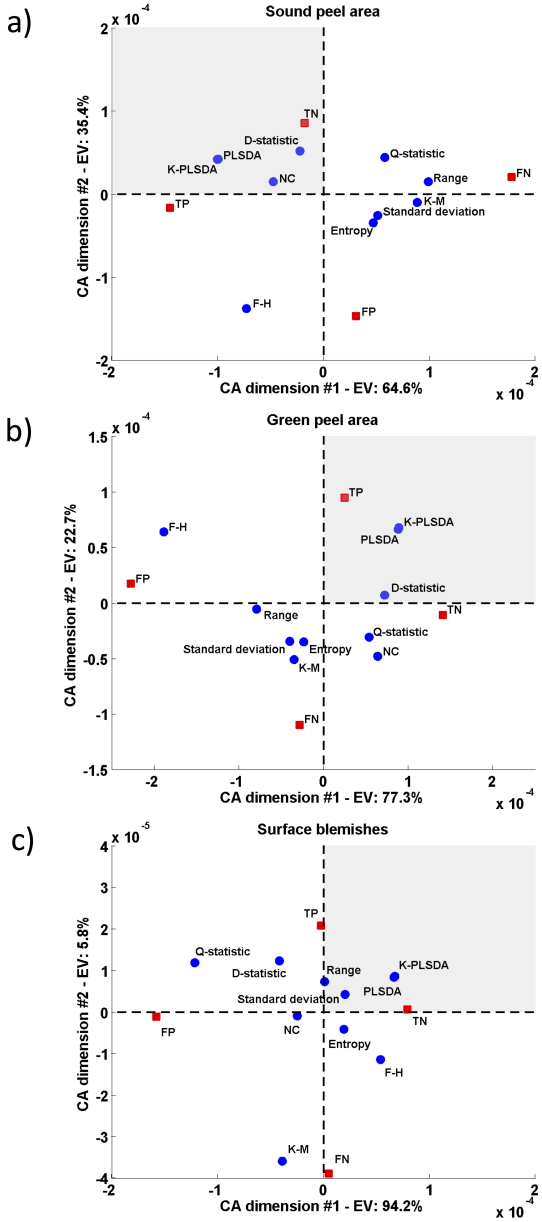

Figure 5.4 Correspondence Analysis (CA) bi-plots for a) the sound peel area class, b) the green peel area class and c) the surface blemishes class. Each blue dot represents one of the segmentation techniques under study, while the red squares refer to the number of True Positives, True Negatives, False Positives and False Negatives, respectively. EV stands for Explained Variance. The TP/TN quadrant is highlighted

or TN/FP, respectively, the best segmentation strategies can be found lying in or close to the TP/TN quadrant (highlighted). This is always the case for both PLSDA and K-PLSDA. In comparison with the other approaches, F-H produced a very large number of FP for the sound and the green peel area categories, but not for the surface blemishes one. That means F-H cannot adequately discriminate the pixels of the first two classes. Regarding the surface blemishes, Q-statistic, D-statistic, NC and K-M delivered the highest number of FP. Conversely, K-M returned a more considerable amount of FN than the other techniques for all the three categories as well as Q-statistic and Range for the sound peel area class and NC for the green peel area class.

5.6 Illustration case

The practical consequences of the previous results can be easily visualised by an illustrative example. Figures 5.5, 5.6 and 5.7 show the segmentations accomplished manually and by the different concerned methodologies for the sound peel areas, the green peel areas and the surface blemishes of one of the 25 test images. In addition to what remarked before, it is worth pointing out that:

- although K-PLSDA is the only non-linear statistical technique applied here, a strict similarity between it and classical (linear) PLSDA was observed. This is justified by the fact that, for the former, an approximately linear kernel function (Gaussian with a σ-parameter value around 500) was selected as the most appropriate to transform the analysed data structure, giving rise to a discrimination among the pixels of the three classes analogous to that enabled by PLSDA. It is then possible to conclude that no strong non-linear relationships needed to be modelled for a good classification, probably due to the limited number of handled categories;

- generally, K-M exhibited a similar segmentation performance to NC. However, across all the test images, the three recognised clusters were not always corresponding to the same pixel classes. I.e. in the various images the first cluster sometimes captured the sound peel area category, sometimes the green peel area category and sometimes the blemished area category. This is a consequence of the unsupervised nature of the iterative algorithmic procedure K-M is based on, which makes the segmentation of series of different images not univocal. That might constitute a critical issue when image segmentation has to be performed automatically and sequentially. The reported outcomes also highlight this effect: Figures 5.5c and 5.6c clearly show that in such an illustrative example the first and the second cluster of pixels are inverted;

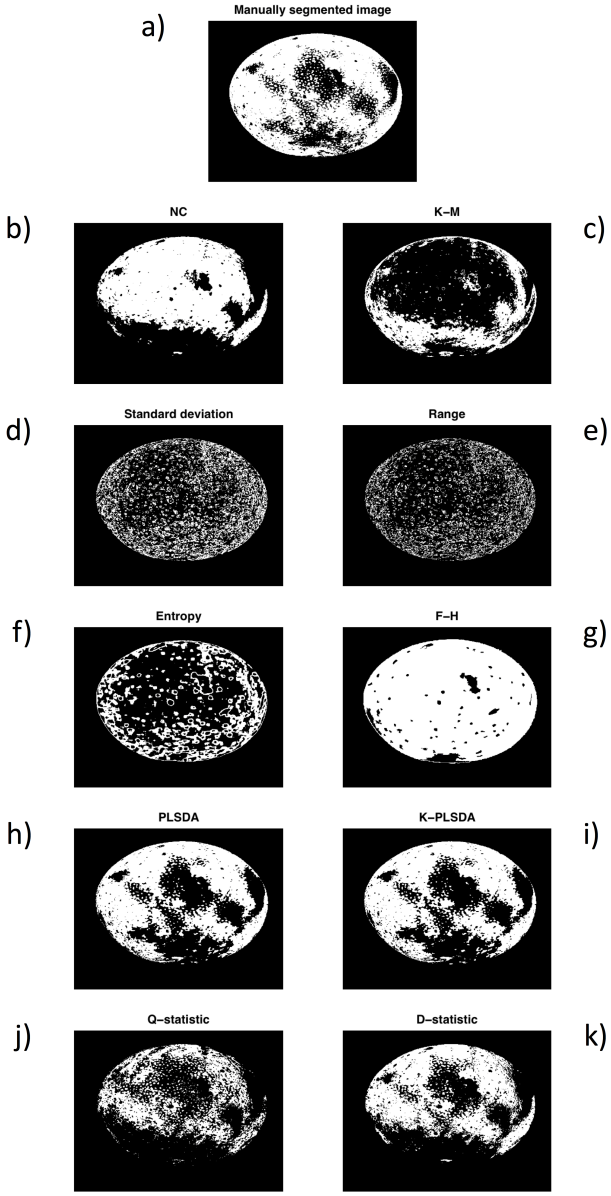

Figure 5.5 Sound peel area: segmentations accomplished a) manually (reference) and by b) NC, c) K-M, d) Standard deviation, e) Range, f) Entropy, g) F-H, h) PLSDA, i) K-PLSDA, j) Q-statistic and k) D-statistic. White colour identifies the pixels assigned to the concerned class

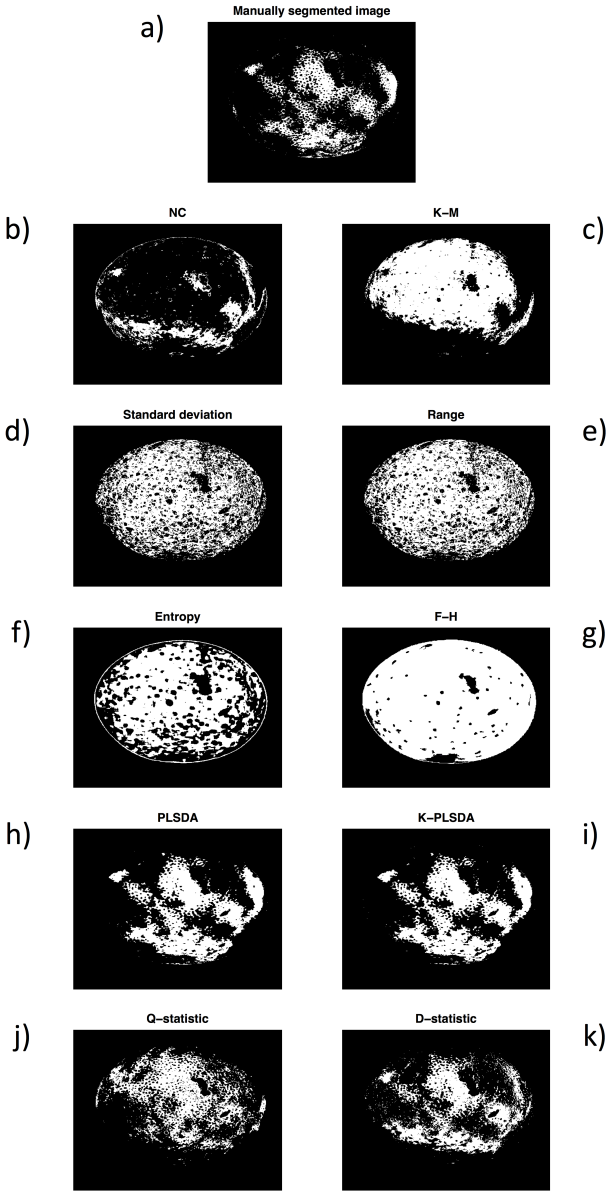

Figure 5.6 Green peel area: segmentations accomplished a) manually (reference) and by b) NC, c) K-M, d) Standard deviation, e) Range, f) Entropy, g) F-H, h) PLSDA, i) K-PLSDA, j) Q-statistic and k) D-statistic. White colour identifies the pixels assigned to the concerned class

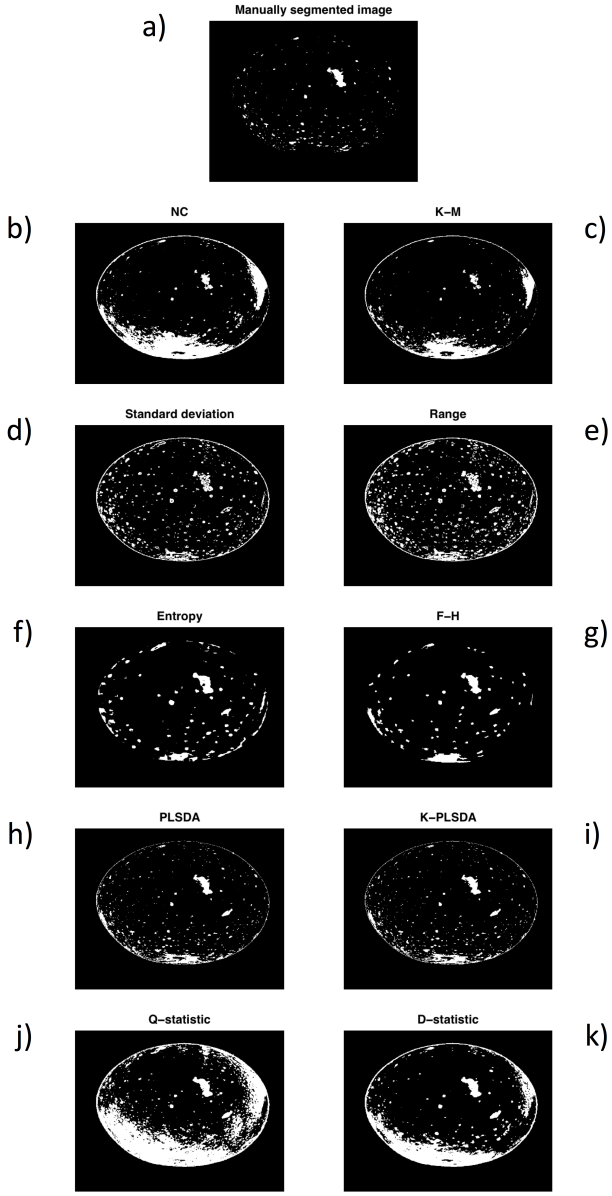

Figure 5.7 Surface blemishes: segmentations accomplished a) manually (reference) and by b) NC, c) K-M, d) Standard deviation, e) Range, f) Entropy, g) F-H, h) PLSDA, i) K-PLSDA, j) Q-statistic and k) D-statistic. White colour identifies the pixels assigned to the concerned class

- solely for F-H sound and green peel areas could not be discriminated at all (Figures 5.5g and 5.6g are identical). This was found to be the main limitation of F-H and is the root cause of the high amount of FP it led to;
- all the methods (even PLSDA and K-PLSDA) suffered from the same issue when dealing with the surface blemishes class: too many pixels were incorrectly classified as belonging to it (low precision), confirming its rather large internal variability, which prevented a more satisfactory segmentation;
- it can also be said that both techniques which performed the best in this particular application (i.e. PLSDA and K-PLSDA) are supervised. Notice this has to be looked at as a circumscribed rather than general conclusion. In fact, in many other fields of interest (e.g. remote sensing) that could easily not be the case.

5.7 Concluding remarks

A comprehensive comparative study among various segmentation methodologies (namely Nearest Centroid, K-Means, Standard Deviation, Range, Entropy, Felzenszwalb-Huttenlocher approach, PLSDA, K-PLSDA, Q-statistic and D-statistic) was carried out to determine which strategies could enable a correct discrimination of sound orange-coloured, green-coloured and blemished areas on the peel of several orange samples. ANOVA-based LSD intervals highlighted that PLSDA and K-PLSDA outmatched the other techniques in terms of F-score, a general measure of segmentation accuracy and precision. Furthermore, CA permitted to more specifically appraise their pros and cons and recognise in a very simple and direct graphical way those strategies yielding higher/lower than average quantities of TP, FP, FN and TN, respectively. The final outcomes revealed that resorting to both colour and textural information in combination with Multivariate Image Analysis (MIA)-based segmentation approaches (i.e. PLSDA and K-PLSDA) might represent a suitable option when dealing with complex problems like the one at hand in this specific case.

Chapter 6

K-PLSDA and pseudo-sample for batch run discrimination

This chapter explores the potential of K-PLSDA coupled to pseudo-sample projection for discriminating on-specification and off-specification batch runs.

Part of the content of this chapter has been included in:

1. Vitale, R., de Noord, O. & Ferrer, A. A kernel-based approach for fault diagnosis in batch processes. *J. Chemometr.* **28**, 697-707 (2014).

6.1 Introduction

Industrial batch processes generate massive amounts of data, which are collected for online treatment or posterior analysis. During a batch run, $r = 1,\ 2,\ \ldots,\ R$ process variables are commonly recorded at $t = 1,\ 2,\ \ldots,\ T$ time points. Measurements registered for $b = 1,\ 2,\ \ldots,\ B$ batches can then be gathered in a three-way array with dimensions $B \times R \times T$. Although many methods exist for directly analysing similar three-dimensional structures, the most widely used approach to retrieve exploitable information from them consists in unfolding such three-way arrays into matrices and then fitting empirical models by means of PCA, PLS or PLSDA [54]. In general, this rearrangement is carried out by i) Variable-Wise Unfolding (VWU), ii) Batch-Wise Unfolding (BWU) or iii) Landmark Feature Extraction (LFE). VWU separates the single sub-matrices associated to the evolution of each batch and reorders them preserving the variable direction, which leads to an $BT \times R$ two-dimensional array. This strategy does not take into account the dynamics of the process under study and has been found to be valid only when its correlation structure is more or less constant along every batch run [54]. BWU unfolds the original three-way structure so that all the sub-matrices related to the evolution of a single variable in all the batches are disposed side by side (the final two-way array has dimensions $B \times RT$). It therefore permits to model the linear dynamics of the analysed process and capture changing relationships among the original variables. By the third, F landmark features of the evolution of each run are extracted and organised in a new data structure with dimensions $B \times F$. A good survey of these techniques can be found in [55].
Unfortunately, due to their intrinsic complexity, batch processes usually result in strong data non-linearities, which may jeopardise the assessment of their quality if this is addressed by classical bilinear methodologies. In these circumstances, kernel-based techniques may represent feasible alternative solutions. For this reason, this chapter will explore the potential of K-PLSDA for the classification of on- and off-specification batch runs. Pseudo-sample projection will be exploited here to recognise the measured variables having the highest discriminant power between the two concerned classes and supposedly corresponding to those exhibiting an *out-of-control* behaviour.

6.2 Methods

Given the availability of measurements collected during both on-specification and off-specification process runs, the whole procedure for building K-PLSDA models and recovering the information about the importance of the original variables comprises the following steps: i) unfold the batch three way array by VWU, BWU

or LFE; ii) auto-scale[i] the unfolded data matrix, say $\mathbf{X}$ $(N \times J)$[ii]; iii) transform the auto-scaled dataset into a kernel matrix, $\mathbf{K}$ $(N \times N)$, by a specific kernel function; iv) double-centre $\mathbf{K}$ so that:

$$\mathbf{K}_{\mathrm{c}} = \mathbf{K} - \overline{\mathbf{K}}_j - \overline{\mathbf{K}}_n + \overline{\mathbf{K}}_{n,j} \tag{6.1}$$

where $\overline{\mathbf{K}}_j$ $(N \times N)$, $\overline{\mathbf{K}}_n$ $(N \times N)$ and $\overline{\mathbf{K}}_{n,j}$ $(N \times N)$ consist of the column means, the row means, and the overall mean of $\mathbf{K}$, respectively; v) Fit a PLSDA model on $\mathbf{K}_{\mathrm{c}}$; vi) construct a pseudo-sample matrix, $\mathbf{V}_j$ $(V \times J)$, for each one of the J measured variables as in Equation 4.3; vii) execute on every $\mathbf{V}_j$ the same kernel transformation as for $\mathbf{X}$ to compute a pseudo-sample kernel matrix, $\mathbf{V}_j^{\mathbf{K}}$ $(V \times N)$; viii) double-centre each $\mathbf{V}_j^{\mathbf{K}}$ so that:

$$\mathbf{V}_{j,\mathrm{c}}^{\mathbf{K}} = \mathbf{V}_j^{\mathbf{K}} - \overline{\mathbf{K}}_j - \mathbf{V}_{j,v}^{\mathbf{K}} + \overline{\mathbf{K}}_{n,j} \tag{6.2}$$

where the v-th row of $\mathbf{V}_{j,v}^{\mathbf{K}}$ $(V \times N)$ contains the mean of the v-th row of $\mathbf{V}_j^{\mathbf{K}}$. Notice that $\overline{\mathbf{K}}_n$ is substituted by the term $\mathbf{V}_{j,v}^{\mathbf{K}}$ because the total number of rows of $\mathbf{V}_j^{\mathbf{K}}$ is usually different from the number of rows of $\mathbf{K}$; ix) project every j-th pseudo-sample kernel matrix onto the model subspace as:

$$\mathbf{T}_{j,\mathrm{ps}} = \mathbf{V}_{j,\mathrm{c}}^{\mathbf{K}} \mathbf{W}^{*\mathbf{K}} \tag{6.3}$$

where $\mathbf{W}^{*\mathbf{K}}$ $(N \times A)$ denotes the array of weights resulting from K-PLSDA (see Equation 2.4); x) plot the predicted pseudo-sample scores, $\mathbf{T}_{j,\mathrm{ps}}$ $(V \times A)$, to get insights into the contribution of each original variable to the final classification.

6.3 Datasets

Three different datasets will be considered. The first refers to a simulated process and includes 10 process variables sampled at 25 time points for 30 distinct batch runs (15 NOC and 15 off-specification owing to an increment in the variance of some of the 10 original variables). The second one related to a polymerisation process (whose engineering details were given in [56]) and consists of 23 batches (18 NOC and 5 off-specification) during which 10 variables (mainly temperatures, pressures and flow-rates) were measured at 100 sampling times. The third dataset contains the values of 8 landmark features of the time evolution of 71 runs of a pharmaceutical batch drying process (33 NOC, 10 on-specification but exhibiting an abnormally high quantity of residual solvent, and 28 off-specification). In contrast with [57], where these data were first described, the second group of 10 runs was excluded from the analysis in order to enable a simpler discrimination between on- and off-specification batches.

[i] Auto-scaling corresponds to variable centering and scaling to unit variance.

[ii] For the sake of simplicity, from now on N and J will denote the number of rows and columns of the unfolded batch data matrix, respectively.

6.4 Results and discussion

6.4.1 Simulated dataset

A 750×2 set of scores featuring trimmed circular profiles was simulated, generating two classes of 15 trajectories of 25 observations each (see Figure 6.1). Such profiles

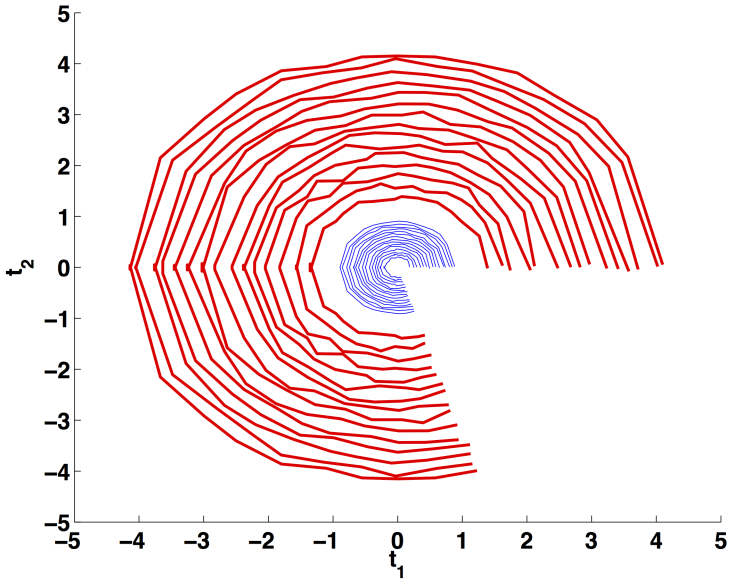

Figure 6.1 Simulated dataset: on- (blue thin lines) and off-specification (red thick lines) simulated batch score trajectories

represent the proper evolution of 30 process runs in a generic latent variable space and might have been obtained by e.g. applying PCA on a batch three-way array unfolded using VWU. A 750 × 10 matrix was finally constructed by multiplying this set of scores by a 2×10 transposed array of loadings computed building a PCA model on real process data. As shown in Figure 6.2, this results in two different classes of 15 batches, characterised by differences in the variance of 10 measured variables (sampled at 25 time points) but not in their mean values (e.g. due to sensor or controller faults). To verify whether pseudo-sample projection enables the correct identification of the most discriminant variables, three columns of such a matrix were substituted by three white noise vectors.
The whole array was then divided into a training and a test set, containing 500 (20 complete batches) and 250 (10 complete batches) observations, respectively. Batch selection was randomly performed class-wise.

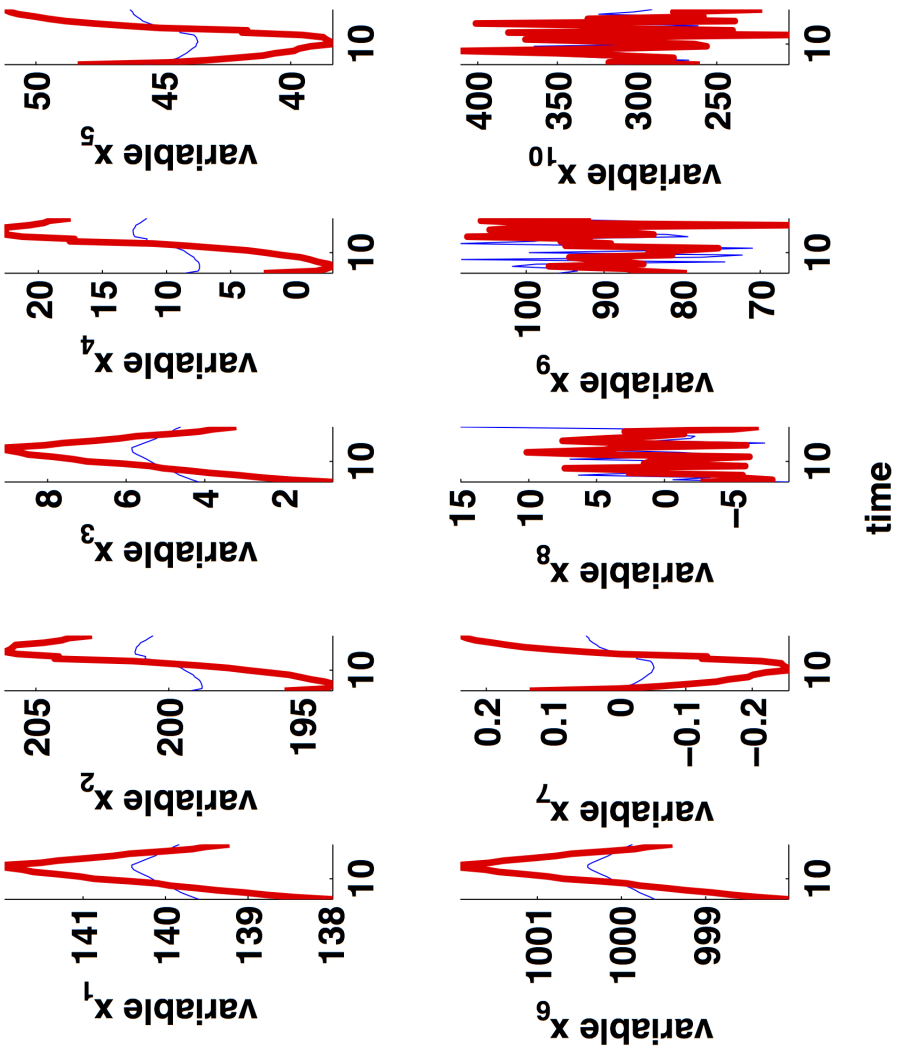

Figure 6.2 Simulated dataset: time evolution of the variables of the simulated dataset for an on- (blue thin line) and an off-specification (red thick line) batch run

Four cross-validated discrimination models, with a growing degree of non-linearity, were subsequently built. Their performance indices are listed in Table 6.1. Clearly, the two classes of runs cannot be satisfactorily separated by classical PLSDA and K-PLSDA encompassing a linear kernel transformation. However, resorting to non-linear kernel functions permits to correctly distinguish most of the observations belonging to the two categories under study for both training and test sets. The best results are evidently yielded by the second-order polynomial K-PLSDA model, whose scores and predicted class value plots are reported in Figure 6.3, where each point corresponds to a specific time instant of a particular batch. Here, it can be said that non-linear K-PLSDA is able to correctly classify most

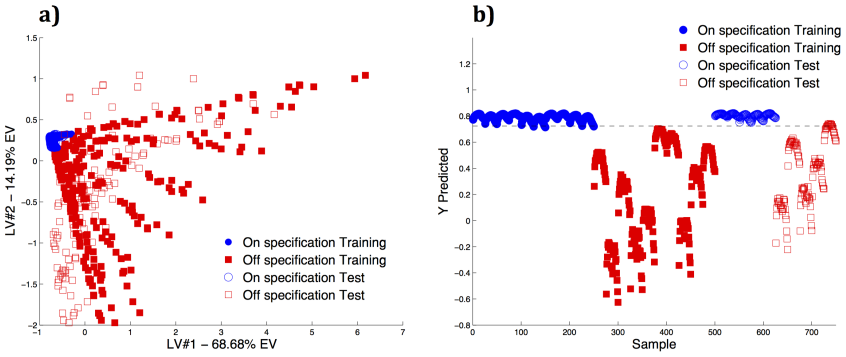

Figure 6.3 Simulated dataset (VWU): a) scores and b) predicted y-values for both training and test set resulting from a second-order K-PLSDA model. The black dotted line represents the class belonging probability threshold calculated according to the Bayes' theorem. EV and LV stand for *explained variance* and *latent variable*, respectively

of the time samples in which the process is progressing under NOC or not. The highest discrimination ability of the second-order K-PLSDA model is reasonable, considering that the differences between the two classes of runs are associated to the variance of the measured variables, which is, indeed, a quadratic transformation of the original data.

Concerning pseudo-sample projection, a 20×10 pseudo-sample matrix was built for every column of the simulated dataset, transformed, and projected onto the K-PLSDA model space as described in Section 6.2. Figure 6.4 shows the resulting outcomes. The displayed trajectories represent the predicted pseudo-sample scores associated to the concerned variables (numbered from 1 to 10). They were graphed so that the font size of the numerical characters constituting them increases in correspondence of regions of the latent space where the respective variables assume higher values and *vice versa*. On the other hand, the blue dotted line coincides with the discriminant direction between the centres of gravity of the two classes of observations, derived from Figure 6.3a. So, comparing Figures 6.4 and 6.3a, it can be inferred that the second category (red squares) embraces batch runs exhibiting

Table 6.1 Simulated dataset (VWU): complexity and performance of the 4 compared classification models

Model	LV number	Correct classification rate (%)			
		On-specification class (calibration)	Off-specification class (calibration)	On-specification class (validation)	Off-specification class (validation)
PLSDA	2	96.0	46.0	100.0	42.4
K-PLSDA (linear transformation function)	2	95.6	45.2	100.0	42.4
K-PLSDA (second-order transformation function)	2	98.4	100.0	100.0	92.8
K-PLSDA (Gaussian transformation function, $\sigma = 0.5$)	2	100.0	99.2	100.0	87.2

either higher or lower values of variables x_1-x_7 than those belonging to the first group (blue dots).
In order to define an objective criterion for evaluating the single variable influence on the classification, the cosine of the angle formed by the blue dotted line and the linear interpolation of each pseudo-sample trajectory was calculated (see Table 6.2). As one can easily notice, only x_1-x_7 feature a high discriminant power (i.e., angle cosines close to 1), which is perfectly coherent with the way the data simulation was carried out (see also Figure 6.2).

Table 6.2 Simulated dataset (VWU): values of the cosine of the angles formed by the pseudo-sample trajectories associated to the 10 variables under study (linearly interpolated) and the class discriminant direction. No definite trajectories were observed for x_8, x_9 and x_{10}, which hindered the univocal definition of such an angle for them

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0.88	0.97	0.89	0.92	0.80	0.87	0.81	-	-	-

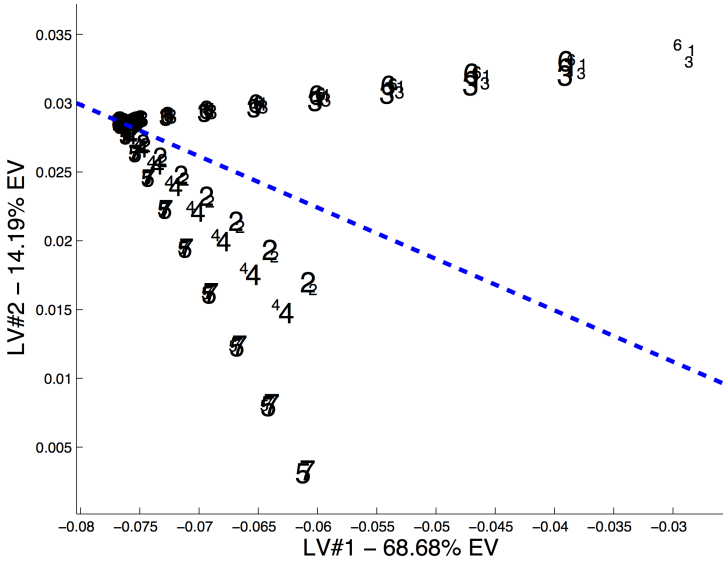

Figure 6.4 Simulated dataset (VWU): second-order K-PLSDA pseudo-sample trajectories. The blue dotted line represents the discriminant direction connecting the centres of gravity of the two classes under study. EV and LV stand for *explained variance* and *latent variable*, respectively

6.4.2 VWU/K-PLSDA (polymerisation process dataset)

The polymerisation process dataset contains observations related to on- and off-specification batches, but only during the first 15 time points of their evolution they actually differed. Therefore, after VWU, the unfolded data matrix was reduced to a 345×10 one and then divided into a training and a test set, containing 225 (15 batches, 12 on- and 3 off-specification) and 120 (8 batches, 6 on- and 2 off-specification) observations, respectively. Afterwards, a linear kernel transformation was applied to the training data, and a cross-validated 2-latent variable PLSDA model ($R^2 = 0.95$, $Q^2 = 0.94$) was built on the resulting 225×225 kernel matrix. In order to assess its prediction ability, the test set was transformed in the same way as the training set (generating a kernel test matrix with dimensions 120×225), projected onto its latent structure and, according to their predicted y-values, assigned to one of the two considered classes. The outcomes are displayed in Figure 6.5. A perfect separation between the observations belonging to the

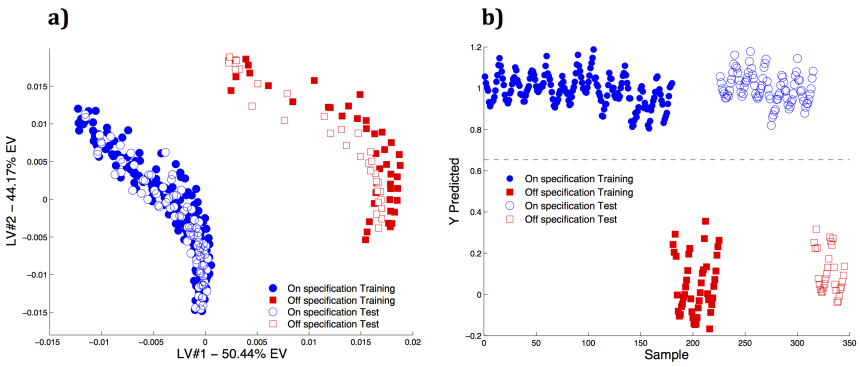

Figure 6.5 Polymerisation process dataset (VWU): a) scores and b) predicted y-values for both training and test set resulting from a first-order K-PLSDA model. The black dotted line represents the class belonging probability threshold calculated according to the Bayes' theorem. EV and LV stand for *explained variance* and *latent variable*, respectively

different categories is observed (see Figure 6.5a). The good discrimination is corroborated by the plot of the predicted class values (see Figure 6.5b) highlighting that, for both of them, a 100% correct classification rate was obtained in calibration and external validation. As for the simulated case study, each represented symbol corresponds to a specific time point of a particular batch run.

To enable model interpretation, for every column of the unfolded data matrix, a 20×10 pseudo-sample array was built, transformed, and projected onto the K-PLSDA subspace. Their respective pseudo-sample trajectories are graphed in Figure 6.6.

Variables x_4, x_7, x_9, and x_{10} were found to be the most discriminant ones, which

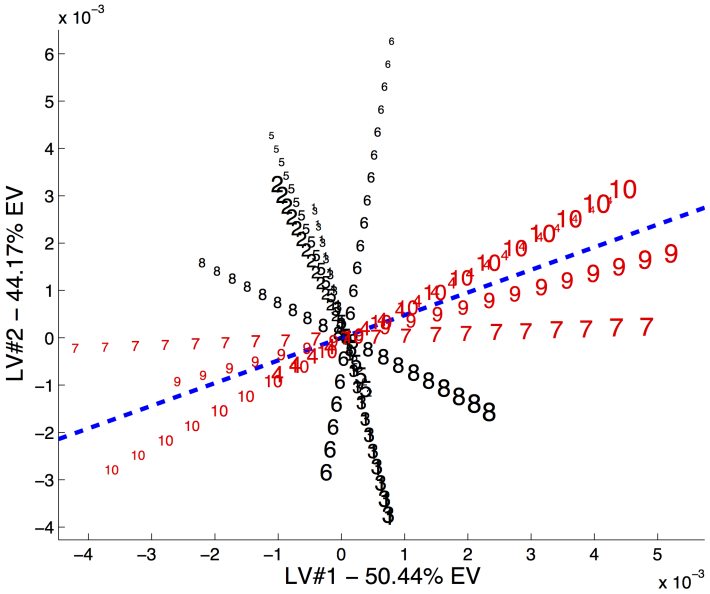

Figure 6.6 Polymerisation process dataset (VWU): first-order K-PLSDA pseudo-sample trajectories. The blue dotted line represents the discriminant direction connecting the centres of gravity of the two classes under study. EV and LV stand for *explained variance* and *latent variable*, respectively. Remind that the font size of the numerical characters constituting each trajectory increases in correspondence of regions of the latent space where the respective variable assumes higher values and *vice versa*

is additionally confirmed by the cosine values in Table 6.3. Furthermore, from the comparison of Figures 6.6 and 6.5a it is clear that the off-specification batches are characterised by higher values of x_7, x_9, and x_{10} and lower values of x_4 than those evolved under NOC (as also proven by Figure 6.7).

Table 6.3 Polymerisation process dataset (VWU): values of the cosine of the angles formed by the pseudo-sample trajectories associated to the 10 variables under study and the class discriminant direction. Bold characters indicate those closest to 1

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0.27	0.11	0.27	**0.99**	0.18	0.53	**0.92**	0.49	**0.99**	**0.98**

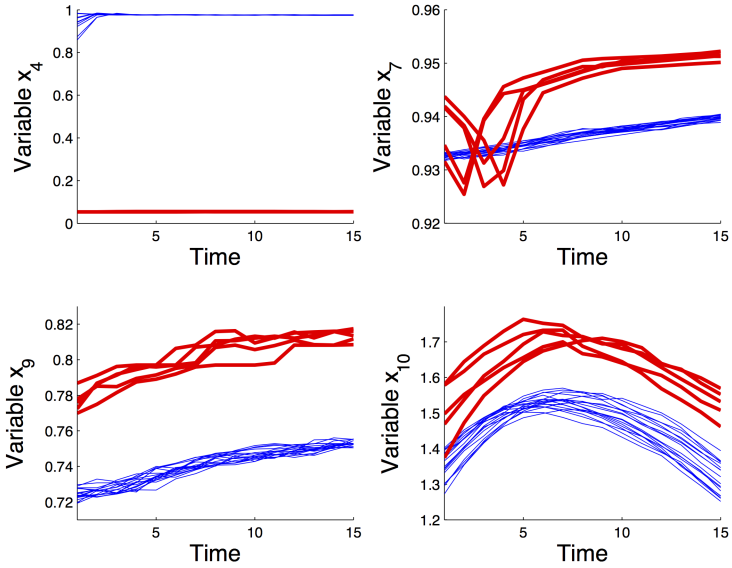

Figure 6.7 Polymerisation process dataset: time evolution of variables x_4, x_7, x_9 and x_{10}. The blue thin lines and the red thick lines relate to the on- and the off-specification batch runs, respectively

6.4.3 BWU/K-PLSDA (polymerisation process dataset)

The same modelling procedure was then applied to the polymerisation process dataset after BWU. A linear kernel function was again chosen for the transformation of the complete unfolded array (23×1000). As only few observations (batches in this case) were available, it was not possible to assess the predictive ability of the final model via an external test set. Permutation tests were resorted to for overcoming this limitation.
Still, as shown in Figure 6.8, a perfect separation between the two concerned categories was yielded by K-PLSDA in both calibration and cross-validation (2 latent variables were extracted, which led to R^2 and Q^2 values of 0.97 and 0.96, respectively and to a correct classification rate of 100%). In this circumstance, it is important to notice that each displayed point denotes an entire process run: therefore, existing dissimilarities between on- and off-specification batches are here spotted.
1000 pseudo-sample trajectories were subsequently constructed to evaluate the importance of every variable at a specific time point. However, plotting all these trajectories would have made their interpretation not straightforward. For this reason, only those associated to the period, during which the difference in the

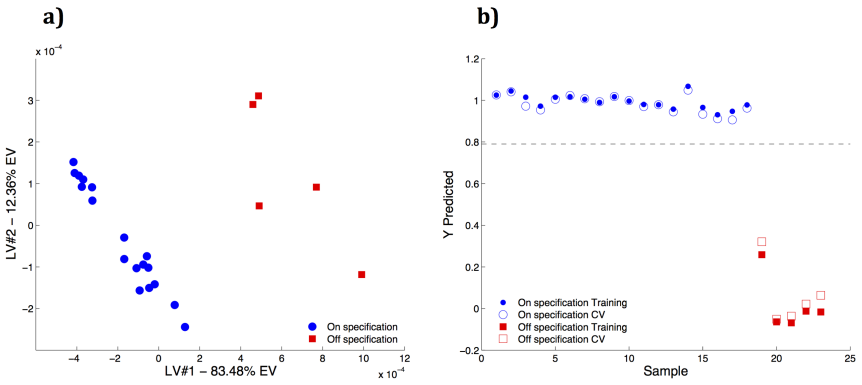

Figure 6.8 Polymerisation process dataset (BWU): a) scores and b) y-values (predicted in calibration and cross-validation) resulting from a first-order K-PLSDA model. The black dotted line represents the class belonging probability threshold calculated according to the Bayes' theorem. EV, LV and CV stand for *explained variance*, *latent variable* and *cross-validation*, respectively

evolution of the batches was detected, were included in Figure 6.9. The graph is divided into 10 subsections. Each subsection contains the pseudo-sample trajectories for a single variable at the various considered time points. Only those constituting with the class discriminant direction (the blue dotted line) an angle of amplitude lower than 30° are black-coloured. The cosine of the angles formed by every trajectory and such a direction is also represented in Figure 6.10 as a function of the batch time. Variables x_4, x_9, and x_{10} were found to have high contributions to the K-PLSDA model approximately for the whole period under study, while variable x_7 only in part of it. Variables x_2, x_5 and x_6 also proved to have certain influence on the classification in short time intervals. As for the previous examples, regarding the pseudo-sample trajectories in Figure 6.9, the larger the bullet size the higher the value the labelled variable assumes in the corresponding area of the latent space. Hence, it can be said that on-specification batches are characterised by smaller values of x_7, x_9, and x_{10} and larger values of x_4 compared to the off-specification runs, which is in agreement with what discussed before.
A permutation test for determining the statistical significance of the R^2 and Q^2 indices of the computed K-PLSDA model was finally carried out. For both of them, p-values much lower than 0.005 were estimated (see also Figure 6.11). This indicates the observed separation between categories of batches is consistent and not due to random data variation.

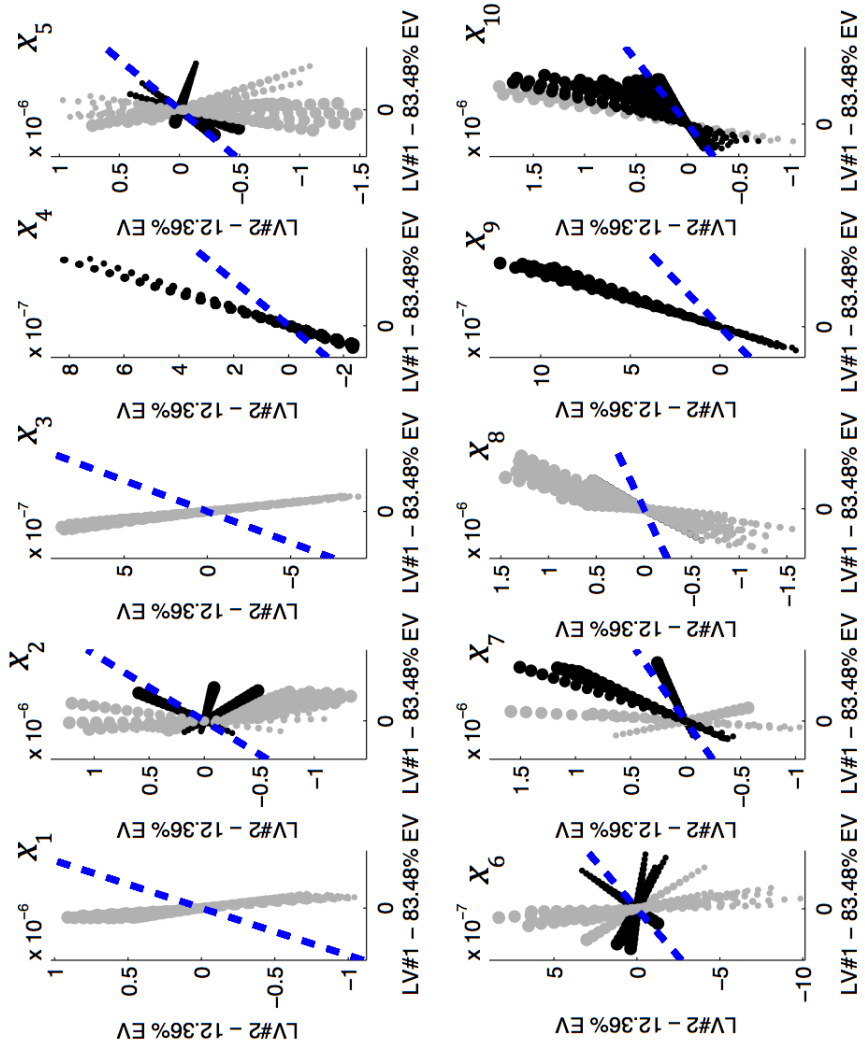

Figure 6.9 Polymerisation process dataset (BWU): first-order K-PLSDA pseudo-sample trajectories. Every subplot contains a pseudo-sample trajectory per variable per time point. The blue dotted line represents the discriminant direction connecting the centres of gravity of the two classes under study. EV and LV stand for *explained variance* and *latent variable*, respectively

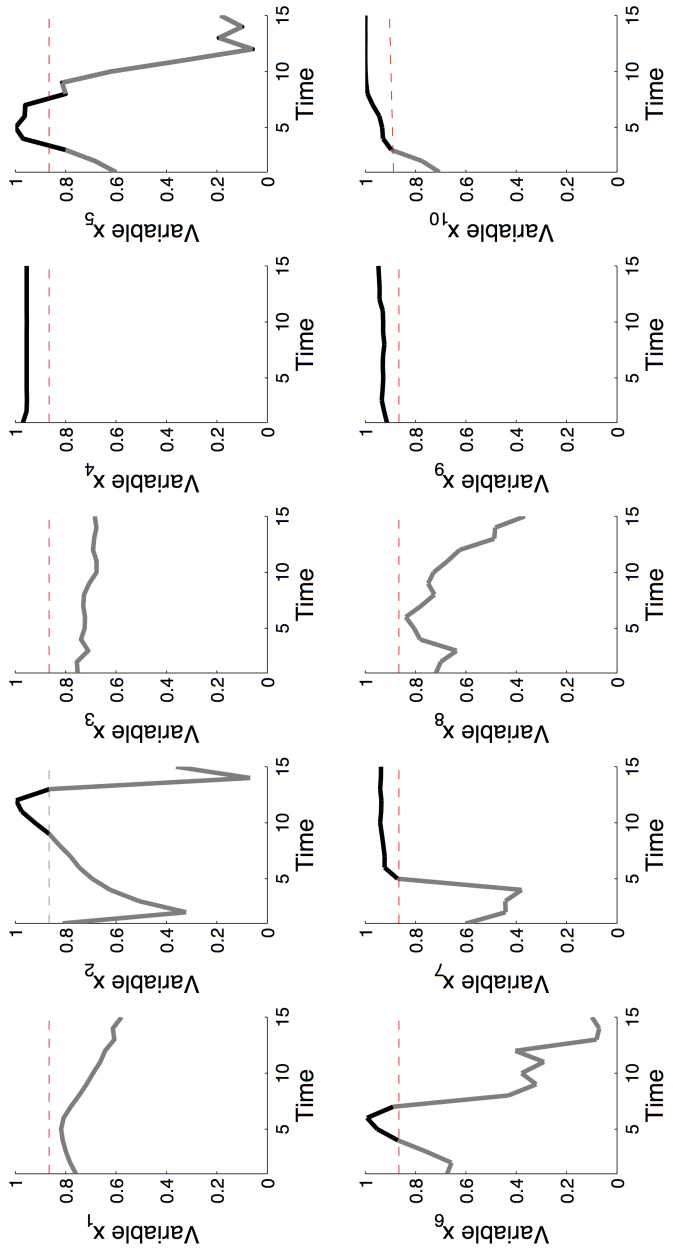

Figure 6.10 Polymerisation process dataset (BWU): values of the cosine of the angles formed by the pseudo-sample trajectories and the discriminant direction as a function of the batch time. Each subplot relates to a single measured variable. The red dotted line represents an empirical threshold set as $\cos(30°) = \frac{\sqrt{3}}{2}$

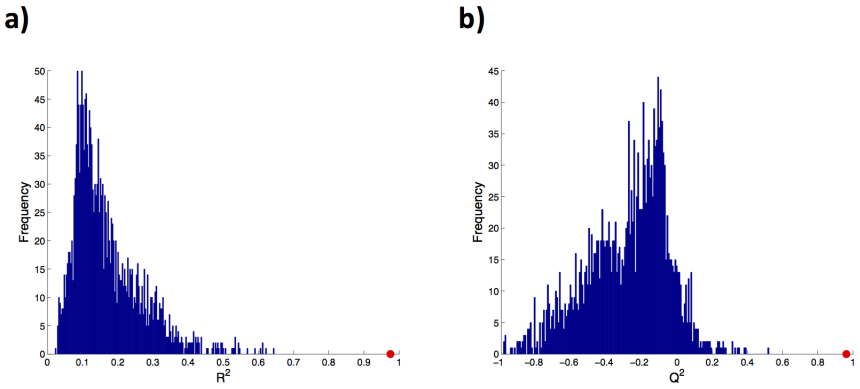

Figure 6.11 Polymerisation process dataset (BWU): permutation test-based R^2 and Q^2 validation plots. The red dots represent the R^2 and Q^2 values of the original K-PLSDA model. The blue bars denote the R^2 and Q^2 values resulting from 2000 data randomisation rounds.

6.4.4 LFE/K-PLSDA (pharmaceutical batch drying process dataset)

The 61×8 matrix resulting from LFE of the pharmaceutical batch drying process dataset was also subjected to K-PLSDA. Among the original observations, 12 (7 on-specification and 5 off-specification) exhibited abnormally high residuals and were afterwards excluded from the final classification. As a linear kernel transformation did not guarantee a satisfactory performance, a Gaussian function was applied to the reduced 49×8 array. The σ parameter was optimised by cross-validation and fixed at a value of 0.8[iii]. A 2-latent variable PLSDA model ($R^2 = 0.74$ and $Q^2 = 0.44$) was then built to address the discrimination of the two different categories of runs. Its scores and the predicted y-values for all the observations in calibration and cross-validation are displayed in Figure 6.12[iv]. Here, as the selected landmark features are not good indicators of the quality of the batches [57], the class separation was not as satisfactory as those obtained in the other illustrated cases (73.1% and 91.3% correct classification rate in cross-validation for on- and off-specification process runs, respectively).
In order to recover the information about the discriminant power of such features, 8 pseudo-sample matrices of dimensions 20×8 were constructed, transformed and projected onto the K-PLSDA latent structure. The outcomes are graphed in Figure 6.13. Since univocally defining an angle between the discriminant direction and so strongly curved lines is unfeasible, the interpretation of Figure 6.13 is not

[iii] Smaller values would have led to overfitting and hardly interpretable pseudo-sample trajectories.

[iv] Also in these plots each represented point refers to a whole batch.

straightforward. However, at a first glance, the trajectories of variables x_1, x_2 and x_8 seem to be the most correlated to the blue dotted line. All the others, in fact, cover circular paths (see those associated to x_4, x_5 and x_6) or show nearly linear trends approximately orthogonal to it (see those associated to x_3 and x_7). Reminding that the larger the font size of the numerical characters the higher the values the labelled variables assume in the corresponding area of the K-PLSDA subspace, it is clear that the off-specification batches are characterised by lower values of x_8 and higher values of x_1 and x_2. This is completely coherent with the conclusions drawn in [57].
The permutation test-based validation plots in Figure 6.14 confirm the final classification model is statistically significant (R^2 p-value $= 0.003$, Q^2 p-value < 0.005). Nevertheless, its R^2 was found to be sometimes lower than that computed after data randomisation. This aspect might be a further evidence that the quality of the batches under study only slightly depends on the nature of the extracted landmark features [58]. In general, defining in a proper way a good subset of descriptors summarising the differences of on-specification and off-specification runs may not be obvious [55], which may often lead to poorer discriminations than when directly operating on the evolution of the measured variables over time.

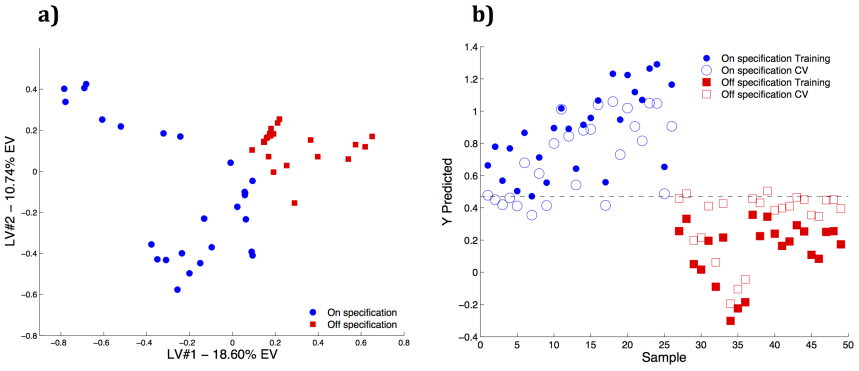

Figure 6.12 Batch drying process dataset (LFE): a) scores and b) y-values (predicted in calibration and cross-validation) resulting from a Gaussian K-PLSDA model ($\sigma = 0.8$). The black dotted line represents the class belonging probability threshold calculated according to the Bayes' theorem. EV, LV and CV stand for *explained variance*, *latent variable* and *cross-validation*, respectively

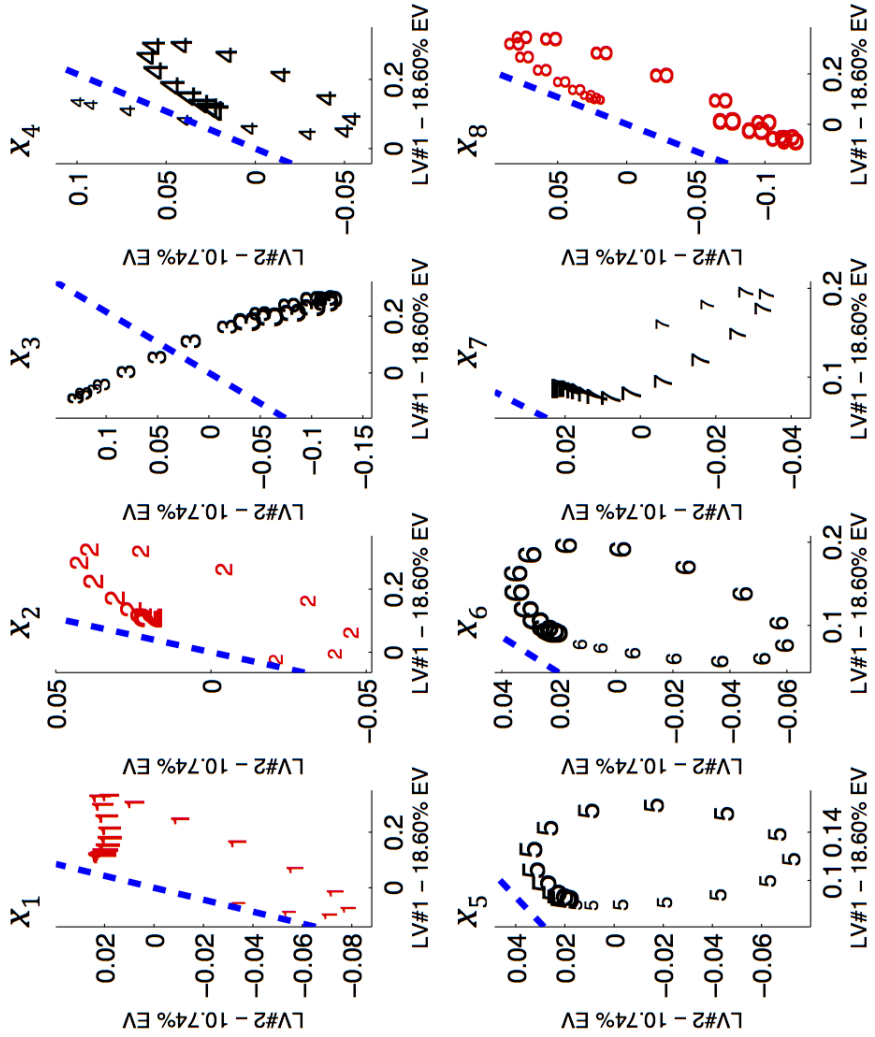

Figure 6.13 Batch drying process dataset (LFE): Gaussian K-PLSDA ($\sigma = 0.8$) pseudo-sample trajectories. Every subplot contains the pseudo-sample trajectory associated to a single measured variable. The blue dotted line represents the discriminant direction connecting the centres of gravity of the two classes under study. EV and LV stand for *explained variance* and *latent variable*, respectively

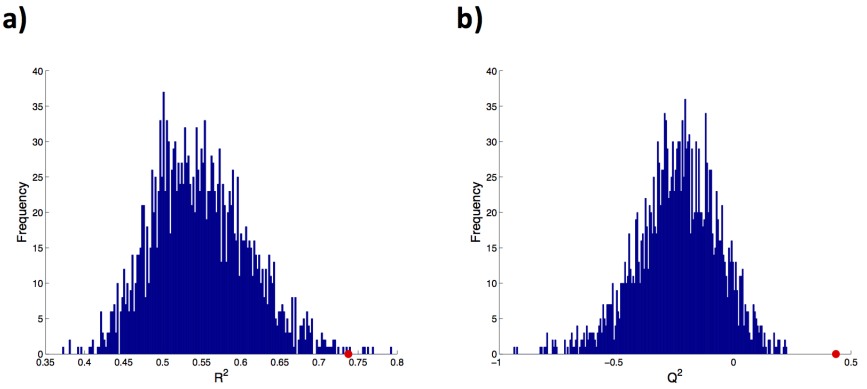

Figure 6.14 Batch drying process dataset (LFE): permutation test-based R^2 and Q^2 validation plots. The red dots represent the R^2 and Q^2 values of the original K-PLSDA model. The blue bars denote the R^2 and Q^2 values resulting from 2000 data randomisation rounds.

6.5 Comparison between K-PLSDA and classical PLSDA models

The simulated example highlighted the main advantage of using non-linear kernel-based classification methods over classical PLSDA. In fact, when complex data structures are dealt with, this latter may return unsatisfactory correct classification rate values, thus jeopardising the identification of an *out-of-control* situation. On the contrary, exploiting non-linear classifiers in such circumstances may radically improve the quality of the classification and the recognition of the process runs which did not progress under NOC.
In the first real case study, for both VWU and BWU, resorting to K-PLSDA did not yield a significantly better performance than when standard PLSDA was applied (results not shown). This similarity is a consequence of the fact that no strongly non-linear relationships had to be modelled here, and, therefore, a simple linear kernel transformation permitted to accomplish a similar discrimination to that achieved by means of a conventional approach.
Conversely, when the batch drying process dataset was handled, a Gaussian K-PLSDA model provided a clearer separation of the two classes of observations compared to a classical PLSDA one (see Figures 6.12 and 6.15).

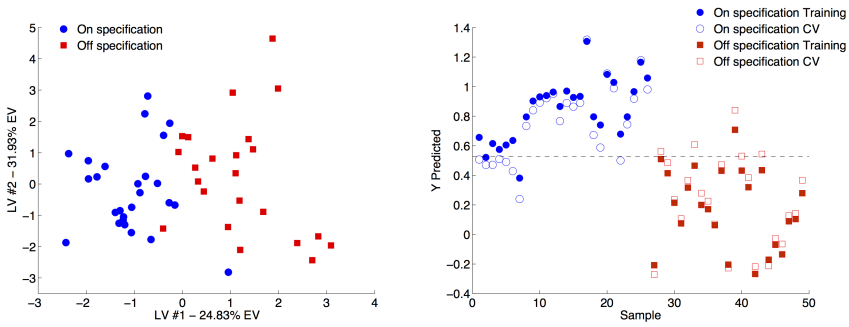

Figure 6.15 Batch drying process dataset (LFE): a) scores and b) y-values (predicted in calibration and cross-validation) resulting from a classical PLSDA model. The black dotted line represents the class belonging probability threshold calculated according to the Bayes' theorem. EV, LV and CV stand for *explained variance*, *latent variable* and *cross-validation*, respectively

6.6 Conclusions

In this chapter, a novel computational strategy for on- and off-specification batch run classification was proposed. It couples the ability of kernel-based techniques (concretely K-PLSDA) of dealing with non-linear data structures to the power of pseudo-sample projection for the interpretation of the final models.
K-PLSDA showed a similar performance to standard PLSDA, when linear transformations were appropriate for the analysis of the concerned datasets, but led to a better separation between classes in case non-linear functions were needed to account for more complex relationships. In both scenarios, pseudo-sample projection enabled a correct identification of the most discriminant variables.
Moreover, it was observed that the described approach may constitute a powerful method for detecting differences in the variance of the variable trajectories measured during the considered batch runs and could then represent an important crossroad for such a specific application of statistical process monitoring and control.

Chapter 7

K-PCA and pseudo-sample projection for batch process monitoring

This chapter explores the potential of K-PCA coupled to pseudo-sample projection for fault detection and diagnosis in batch process monitoring.

Part of the content of this chapter has been included in:

1. Vitale, R., de Noord, O. & Ferrer, A. Pseudo-sample based contributions plots: innovative tools for fault diagnosis in kernel-based batch process monitoring. *Chemometr. Intell. Lab.* **149**, 40-52 (2015).

7.1 Introduction

In Chapter 6 an example of how to combine K-PLSDA and pseudo-sample projection for finding the original variables with the highest discriminant power between two different classes of batches (on- and off-specification) and, then, solving a particular supervised problem was reported. Nonetheless, as also remarked in Section 4.1, historical fault databases are rarely available in real-world situations. In these circumstances, in order to guarantee and preserve the high quality of the final products and to minimise the number of failures, most of the manufacturing companies in the world generally design batch monitoring schemes which allow abnormal events to be quickly, easily and efficiently recognised (*fault detection*) and their possible root causes to be correctly identified (*fault diagnosis*)[i]. Again, these monitoring schemes are usually constructed through empirical approaches [54, 59, 60] based on unfolding a typical three-way batch array into a matrix and subsequently analysing it by means of PCA or PLS [54]. More specifically, i) data resulting from runs which evolved under NOC are used to calibrate a so-called *in-control* model, and afterwards ii) incoming batch data are projected onto its space for the assessment of new process runs. However, as detailed in Section 6.1, due to the intrinsic complexity of batch processes, such data are usually affected by strong non-linear relationships. Thus, also in similar scenarios, kernel-based techniques may represent feasible alternative solutions to be resorted to when classical methodologies exhibit unsatisfactory performance. Hence, the first aim of this chapter is to evaluate their potential for this concrete application of interest.
Ideally, pseudo-sample projection could constitute a valuable option for spotting the measured variables responsible for the deviations from NOC in these cases, but, at least in the form it was originally conceived and has always been implemented until now, it is not suitable to be directly utilised in a semi-supervised context like the one which has just been sketched. For this reason and considering the fundamental role fault diagnosis plays when industrial process monitoring is concerned, the algorithmic strategy proposed in Chapter 6 is here first adapted and then exploited for the development of novel diagnostic tools that can be employed when a kernel-based batch monitoring scheme is built, the so-called pseudo-sample based contribution plots.

7.2 Adaptation of the pseudo-sample projection strategy to batch process monitoring

The whole procedure for building kernel-based batch monitoring schemes and recovering the information about the contribution of the original variables, enabling fault diagnosis, comprises two phases (*Model building* and *Model exploitation*).

[i] After that, *ad hoc* countermeasures can be adopted for recovering NOC.

Phase I - Model building: i) Preprocess[ii] and unfold by VWU, BWU or LFE the three-way dataset containing the observations resulting from the batches which evolved under NOC into the training matrix $\mathbf{X}$ ($N \times J$); ii) Convert $\mathbf{X}$ into a kernel matrix $\mathbf{K}$ ($N \times N$) by using a specific kernel function; iii) Double-centre $\mathbf{K}$ as:

$$\mathbf{K}_{\mathrm{c}} = \mathbf{K} - \overline{\mathbf{K}}_j - \overline{\mathbf{K}}_n + \overline{\mathbf{K}}_{n,j} \tag{7.1}$$

where $\overline{\mathbf{K}}_j$, $\overline{\mathbf{K}}_n$ and $\overline{\mathbf{K}}_{n,j}$ are squared matrices ($N \times N$) containing the column means, the row means and the overall mean of $\mathbf{K}$, respectively; iv) Fit a PCA model on $\mathbf{K}_{\mathrm{c}}$:

$$\mathbf{K}_{\mathrm{c}} = \mathbf{T}^{\mathbf{K}}\mathbf{P}^{\mathbf{K}^{\mathrm{T}}} + \mathbf{E}^{\mathbf{K}} \tag{7.2}$$

where $\mathbf{T}^{\mathbf{K}}$ ($N \times A$), $\mathbf{P}^{\mathbf{K}}$ ($N \times A$) and $\mathbf{E}^{\mathbf{K}}$ ($N \times N$) are the score, loading and residual matrices of the fitted K-PCA model. v) Calculate the D-statistic and the Squared Prediction Error (SPE) for every observation of $\mathbf{K}$ as follows:

$$D_n^{\mathbf{K}} = \mathbf{t}_n^{\mathbf{K}^{\mathrm{T}}}\mathbf{S}^{\mathbf{K}^{-1}}\mathbf{t}_n^{\mathbf{K}} \tag{7.3}$$

$$\mathrm{SPE}_n^{\mathbf{K}} = \sum_{n=1}^{N} e_{n,n}^{\mathbf{K}^2} \tag{7.4}$$

and their corresponding control limits as in [61] and [62] or [63], respectively. $\mathbf{t}_n^{\mathbf{K}^{\mathrm{T}}}$ ($1 \times A$) is here the n-th row of $\mathbf{T}^{\mathbf{K}}$, $\mathbf{S}^{\mathbf{K}}$ ($A \times A$) is the K-PCA scores covariance matrix[iii], while $e_{n,n}^{\mathbf{K}}$ represents the $n \times n$-th element of $\mathbf{E}^{\mathbf{K}}$; vi) Provided that all the training process runs are *in-control*, proceed with phase II. Otherwise, remove outliers and return to step ii).

Phase II - Model exploitation i) Preprocess[ii] and unfold, as for the training set, the three-way dataset including the test process runs into the matrix $\mathbf{X}_{\mathrm{test}}$ ($N' \times J$); ii) Convert $\mathbf{X}_{\mathrm{test}}$ into a kernel matrix $\mathbf{K}_{\mathrm{test}}$ ($N' \times N$) by using the same kernel function as for $\mathbf{X}$. By this operation, the dissimilarity/distance values between the N' test and the N training observations are calculated, which allows $\mathbf{K}_{\mathrm{test}}$, after appropriate preprocessing, to be projected onto the K-PCA model space defined by the loadings matrix $\mathbf{P}^{\mathbf{K}}$; iii) Double-centre $\mathbf{K}_{\mathrm{test}}$ as follows:

$$\mathbf{K}_{\mathrm{c,test}} = \mathbf{K}_{\mathrm{test}} - \overline{\mathbf{K}}_{j,\mathrm{test}} - \overline{\mathbf{K}}_{n',\mathrm{test}} + \overline{\mathbf{K}}_{n,j,\mathrm{test}} \tag{7.5}$$

where $\overline{\mathbf{K}}_{j,\mathrm{test}}$ and $\overline{\mathbf{K}}_{n,j,\mathrm{test}}$ are $N' \times N$ matrices containing the column means and the overall mean of $\mathbf{K}$, respectively. Notice that, unlike Equation 7.1, $\overline{\mathbf{K}}_n$ is

[ii] The preprocessing strategies suitable for three-way batch data arrays will be discussed in Section 7.4.

[iii] When VWU is applied for the analysis of batch data, one observation of the final unfolded matrices represents a single time instant of a specific process run. In such cases, the calculation of its D-statistic value was executed by considering its predicted score, centred by the mean of the training ones corresponding to the same sampling point, and the respective instantaneous score covariance matrix.

substituted by the term $\overline{\mathbf{K}}_{n',\text{test}}$ ($N' \times N$), containing the row means of $\mathbf{K}_{\text{test}}$; iv) Project $\mathbf{K}_{\text{c,test}}$ onto the *in-control* K-PCA model space as:

$$\mathbf{T}^{\mathbf{K}}_{\text{test}} = \mathbf{K}_{\text{c,test}}\mathbf{P}^{\mathbf{K}} \tag{7.6}$$

where $\mathbf{T}^{\mathbf{K}}_{\text{test}}$ ($N' \times A$) is the predicted score matrix associated to $\mathbf{K}_{\text{test}}$. Then, calculate the residual test matrix, $\mathbf{E}^{\mathbf{K}}_{\text{test}}$ ($N' \times N$), as:

$$\mathbf{E}^{\mathbf{K}}_{\text{test}} = \mathbf{K}_{\text{c,test}} - \mathbf{T}^{\mathbf{K}}_{\text{test}}\mathbf{P}^{\mathbf{K}^{\text{T}}} \tag{7.7}$$

v) Calculate the D-statistic and the Squared Prediction Error (SPE) for every observation of $\mathbf{K}_{\text{test}}$ as follows:

$$D^{\mathbf{K}}_{n',\text{test}} = \mathbf{t}^{\mathbf{K}^{\text{T}}}_{n',\text{test}}\mathbf{S}^{\mathbf{K}^{-1}}\mathbf{t}^{\mathbf{K}}_{n',\text{test}} \tag{7.8}$$

$$\text{SPE}^{\mathbf{K}}_{n',\text{test}} = \sum_{n=1}^{N} e^{\mathbf{K}^2}_{n',n,\text{test}} \tag{7.9}$$

where $\mathbf{t}^{\mathbf{K}^{\text{T}}}_{n',\text{test}}$ ($1 \times A$) is the n'-th row of $\mathbf{T}^{\mathbf{K}}_{\text{test}}$ [iii], while $e^{\mathbf{K}}_{n',n,\text{test}}$ represents the $n' \times n$-th element of $\mathbf{E}^{\mathbf{K}}_{\text{test}}$; vi) If a batch is detected as faulty, create a $V \times J$ pseudo-sample matrix, $\mathbf{V}_{j,\text{NOC}}$, for each one of the J original variables as in Equation 4.3 taking into account its minimum and its maximum values in the whole training matrix, $\mathbf{X}$; vii) Build another set of $\mathbf{V}_{j,\text{fault}}$ pseudo-sample matrices ($V' \times J$) considering, in this case, the minimum and the maximum values the J original variables assume in the concerned *out-of-control* run[iv]; viii) Apply to each couple of pseudo-sample matrices the same kernel transformation as for the training data in order to obtain a pair of pseudo-sample kernel matrices, $\mathbf{K}_{\mathbf{V}_{j,\text{NOC}}}$ ($V \times N$) and $\mathbf{K}_{\mathbf{V}_{j,\text{fault}}}$ ($V' \times N$). By this operation, the dissimilarity/distance values between the N training observations and both the V pseudo-samples in $\mathbf{V}_{j,\text{NOC}}$ and the V' pseudo-samples in $\mathbf{V}_{j,\text{fault}}$ are calculated, which allows $\mathbf{K}_{\mathbf{V}_{j,\text{NOC}}}$ and $\mathbf{K}_{\mathbf{V}_{j,\text{fault}}}$, after appropriate preprocessing, to be projected onto the K-PCA model space defined by the loadings matrix $\mathbf{P}^{\mathbf{K}}$; ix) Double-centre every $\mathbf{K}_{\mathbf{V}_{j,\text{NOC}}}$ and every $\mathbf{K}_{\mathbf{V}_{j,\text{fault}}}$ so that:

$$\mathbf{K}_{\mathbf{V}_{j,\text{NOC}},\text{c}} = \mathbf{K}_{\mathbf{V}_{j,\text{NOC}}} - \overline{\mathbf{K}}_{j,\mathbf{V}_{j,\text{NOC}}} - \overline{\mathbf{K}}_{v,\mathbf{V}_{j,\text{NOC}}} + \overline{\mathbf{K}}_{n,j,\mathbf{V}_{j,\text{NOC}}} \tag{7.10}$$

$$\mathbf{K}_{\mathbf{V}_{j,\text{fault}},\text{c}} = \mathbf{K}_{\mathbf{V}_{j,\text{fault}}} - \overline{\mathbf{K}}_{j,\mathbf{V}_{j,\text{fault}}} - \overline{\mathbf{K}}_{v',\mathbf{V}_{j,\text{fault}}} + \overline{\mathbf{K}}_{n,j,\mathbf{V}_{j,\text{fault}}} \tag{7.11}$$

where $\overline{\mathbf{K}}_{j,\mathbf{V}_{j,\text{NOC}}}$ ($V \times N$) and $\overline{\mathbf{K}}_{j,\mathbf{V}_{j,\text{fault}}}$ ($V' \times N$) carry the column means of $\mathbf{K}$, $\overline{\mathbf{K}}_{n,j,\mathbf{V}_{j,\text{NOC}}}$ ($V \times N$) and $\overline{\mathbf{K}}_{n,j,\mathbf{V}_{j,\text{fault}}}$ ($V' \times N$) include the overall mean of $\mathbf{K}$, while the v-th row of $\overline{\mathbf{K}}_{v,\mathbf{V}_{j,\text{NOC}}}$ ($V \times N$) and the v'-th row of $\overline{\mathbf{K}}_{v',\mathbf{V}_{j,\text{fault}}}$ ($V' \times N$) contain the mean of the v-th row of $\mathbf{K}_{\mathbf{V}_{j,\text{NOC}}}$ and the mean of the v'-th row of

[iv] Special circumstances in which this operation needs to be slightly adjusted will be further discussed in the following sections.

$\mathbf{K}_{\mathbf{V}_{j,\text{fault}}}$, respectively; x) Project the two groups of double-centred pseudo-sample kernel matrices onto the K-PCA model space, as follows:

$$\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}}} = \mathbf{K}_{\mathbf{V}_{j,\text{NOC}},\text{c}}\mathbf{P}^{\mathbf{K}} \tag{7.12}$$

$$\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,\text{fault}}} = \mathbf{K}_{\mathbf{V}_{j,\text{fault}},\text{c}}\mathbf{P}^{\mathbf{K}} \tag{7.13}$$

where $\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}}}$ $(V \times A)$ and $\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,\text{fault}}}$ $(V' \times A)$ represent the pseudo-sample K-PCA predicted score matrices associated to $\mathbf{K}_{\mathbf{V}_{j,\text{NOC}}}$ and $\mathbf{K}_{\mathbf{V}_{j,\text{fault}}}$, respectively. Two different pseudo-sample trajectories will be then constructed per each original variable, which can be considered as representations in the latent space of the K-PCA model of its real variability range in the NOC batches and in the process run detected as faulty; xi) Calculate for every couple of pseudo-sample trajectories related to the same j-th variable the so-called *Discriminant Distance* (DD_j) as:

$$DD_j = \sqrt{\sum_{a=1}^{A}\left(\frac{\widetilde{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{fault}},a} - \widetilde{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}},a}}{s_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}},a}}}\right)^2} \tag{7.14}$$

which will be used as an index of the j-th variable contribution to the fault. $\widetilde{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{fault}},a}$ and $\widetilde{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}},a}$ are the median values of the column vectors of $\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,\text{fault}}}$ and $\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}}}$, respectively, related to the a-th component of the model. $s_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}},a}}$ is the standard deviation of the column vector of $\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}}}$ related to the a-th component of the model. This term will permit to distinguish specific situations in which the absolute difference between the two median values, $\widetilde{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{fault}},a}$ and $\widetilde{t}^{\mathbf{K}}_{\mathbf{V}_{j,\text{NOC}},a}$, might not be statistically significant considering the variability range of the pseudo-sample predicted scores associated to the NOC observations; xii) Represent the DD_j values in a bar plot for all the J original variables, obtaining a pseudo-sample based contribution plot (named *Discriminant Distance* or *DD plot* from now on).

7.3 Datasets

Three different datasets will be analysed here. The first and the third correspond to those previously described in Section 6.3 and related to the simulated and the drying process, respectively. The second one results from a real 4-stage batch chemical process and consists of 22 measured variables registered at 1-minute intervals for 36 runs of a single product grade [64]. Such variables mainly include pressures, temperatures and flow rates. Normal batch duration is between 398 and 460 minutes.

7.4 Results and discussion

The three datasets will be resorted to in order to assess and compare the performance of classical PCA to that of K-PCA coupled to pseudo-sample projection in a monitoring scenario. VWU and BWU will be tested on the simulated[v] and the real chemical data. They will be combined with the two most common preprocessing approaches applied in batch process chemometrics: Variable Centring and Scaling (VCS) and Trajectory Centring and Scaling (TCS). This will permit to evaluate how VCS and TCS affect the quality of the final monitoring schemes. VCS mean-centres and scales to unit variance each r-th process variable. TCS consists of mean-centring and scaling to unit variance each r-th process variable at each t-th sampling point, allowing their variation around their average time trajectories to be subsequently modelled. Specifically, the comparison will be carried out considering: i) Variable Centring and Scaling-VWU (VCS-VWU), ii) Trajectory Centring and Scaling-VWU (TCS-VWU) and iii) Trajectory Centring and Scaling-BWU (TCS-BWU).
On the other hand, the third dataset will help the assessment of how bilinear and kernel methods perform when LFE is concerned.
In any case study, the number of principal components (PCs) calculated for building the PCA- and K-PCA-based monitoring schemes was determined by cross-validation[vi]. The cross-validatory approach was extended in order to take into account also a certain set of possible kernel transformations of the original training data. The final combination of number of PCs and kernel function was selected as that guaranteeing the best compromise between degree of non-linearity of the model and reconstruction error. Although such a criterion might not be the most suitable in a similar context [65], in all the applications under study, it permitted to obtain very satisfying outcomes in terms of fault detection and diagnosis accuracy.

7.4.1 Simulated dataset - Variability increase detection case study

The first approach tested on the simulated dataset was VCS-VWU. After preprocessing and unfolding the three-way arrays, 10 NOC process runs were randomly selected for building the PCA and K-PCA *in-control* models. Regarding the latter, a second-order polynomial function was chosen for the kernel transformation. The comparison of the two monitoring schemes was then carried out after having adjusted the control limits of the resulting SPE and D-statistic control charts so

[v] The simulated batches will be used to mimic two different scenarios, in which either an increase or a decrease in the process variable variability has to be detected and diagnosed.

[vi] In case VWU was exploited for the analysis of the batch data, at each iteration, the bunch of observations associated to a specific run was removed from the training set. On the other hand, one single row was iteratively left out when the unfolding step was performed by BWU or LFE.

that the Overall Type I (OTI) risk value, the false alarm rate, was approximately equal to the corresponding imposed significance level (ISL) α (5% and 1% in this specific case)[vii]. The expression for the OTI risk value is:

$$OTI = 100\frac{nf}{B_{NOC}T} \tag{7.15}$$

where nf denotes the number of sampling points detected as faulty, B_{NOC} represents the number of considered NOC batches and T is the total number of sampling points per batch. The control limit adjustment was assessed after the projection of the 5 NOC batches, left out of the training set, onto the model space (see Table 7.1). The fault detection power of the different control charts resulting from

Table 7.1 Simulated dataset/variability increase (VCS-VWU): Overall Type I (OTI) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes. The table lists also the number of principal components (PCs) and the goodness of fit (R^2) of the two different *in-control* models

	PCs	R^2	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	2	0.705	6.4%	1.6%	4.8%	1.6%
K-PCA (second-order polynomial)	2	0.743	6.4%	0.8%	4.0%	0.8%

classical PCA and K-PCA was evaluated according to the Overall Type II ($OTII$) risk value, calculated as:

$$OTII = 100\frac{nnf}{B_{faulty}T} \tag{7.16}$$

where nnf represents the number of non-signalled faulty sampling points and B_{faulty} is the number of faulty batches, projected onto the *in-control* model. A good performance would lead to a $OTII$ value close to 0. The results, obtained using the 15 faulty test runs, are reported in Table 7.2. The K-PCA-based SPE

Table 7.2 Simulated dataset/variability increase (VCS-VWU): Overall Type II ($OTII$) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes

	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	79.5%	89.1%	0.3%	5.6%
K-PCA (second-order polynomial)	0.3%	1.1%	0.0%	0.0%

[vii] Notice that the significance level imposed for the control limits of the SPE and D-statistic charts may vary from case to case. This is due to the fact that their adjustment is assessed by using an external test set, which may have different size depending on the original data array under study. The same applies for the confidence limits of every pseudo-sample based contribution plot displayed from now on, calculated by a jackknife procedure.

control chart shows $OTII$ values approximately equal to zero, much lower than the ones resulting from the classical PCA model. On the other hand, the D-statistic control charts seem to be characterised by a similar good accuracy in terms of fault detection power[viii]. As an example, the SPE and the D-statistic control charts obtained by both classical PCA and K-PCA for the first faulty batch contained in the test set are represented in Figure 7.1.

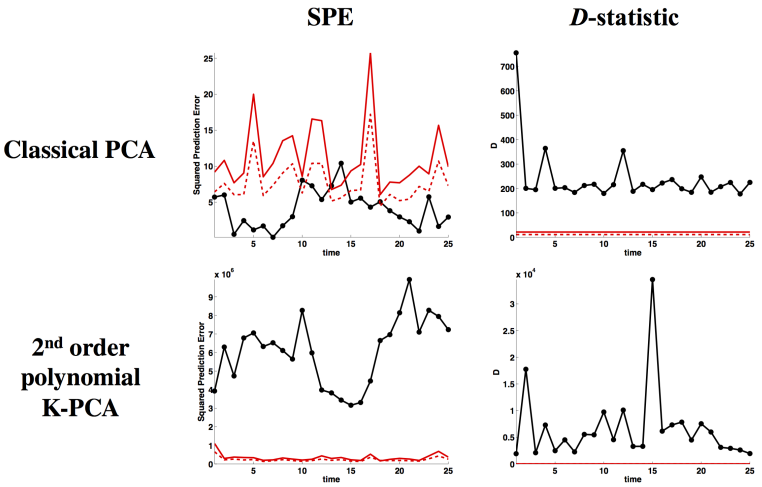

Figure 7.1 Simulated dataset/variability increase (VCS-VWU): SPE and D-statistic control charts obtained by both classical PCA and K-PCA for the first faulty test batch. The dotted and the solid red lines represent the 95% and the 99% control limits, respectively

Once a specific fault has been detected by one or both of the considered control charts, it is fundamental to diagnose it and verify which variables are the most affected by the *out-of-control* signal. In general, this is done by the so-called contribution plots[ix]. In Figure 7.2, the classical SPE and D-statistic contribution plots, related to the first sampling point of the first faulty test batch found to be beyond the 95% control limit in the corresponding PCA-based control charts (time instant #10 and #1, respectively), are displayed. The SPE contribution plot for time instant #10 (see Figure 7.2a) points out an issue implicating variables x_8, x_9 and x_{10}. Unfortunately, the conclusions one would reach looking at it would be completely wrong. In fact, these three variables do not show any difference in their evolution between NOC and faulty batches, as highlighted in Figure 6.2.

[viii] For the sake of precision, if *out-of-control* signals are returned for more than the half of the entire evolution of each faulty batch, the fault detection capability of the respective control chart is considered to be acceptable.

[ix] Such tools are commonly exploited to determine the most contributing variables to the SPE and the D-statistic, which supposedly correspond to those most responsible for the failure. A comprehensive survey on contribution plots can be found in [66].

Therefore, the fault diagnosis would be completely mistaken. On the other hand, in the D-statistic contribution plot (see Figure 7.2b), variables x_8, x_9 and x_{10} are correctly identified as having a low contribution to the fault at time instant #1. However, its interpretation is not straightforward: all the variables from x_1 to x_7 were simulated so that their variance in the faulty batches was approximately twice the variance in the NOC process runs. Thus, such large differences in their contributions were not expected and are not coherent with the nature of the simulated dataset. Furthermore, the SPE and D-statistic contribution plots related to all the other sampling times were found to have consistent profiles as the ones displayed in Figure 7.2 (results not shown). Regarding the kernel-based approach,

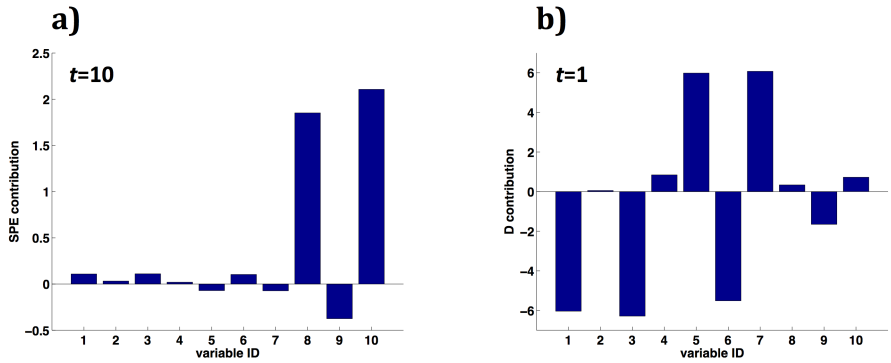

Figure 7.2 Simulated dataset/variability increase (VCS-VWU): classical PCA - a) SPE and b) D-statistic contribution plots related to the first sampling point of the first faulty test batch found to be beyond the 95% control limit in the corresponding PCA-based control charts (time instant #10 and #1, respectively)

the fault diagnosis was enabled by resorting to the strategy described in Section 7.2. Figure 7.3 shows the DD plot, related to the first faulty test batch. Notice that this is an overall contribution plot, taking into account the whole evolution of each variable in both the faulty and the NOC process runs under study and not focusing on an individual value it assumes at a certain time instant. However, the proposed pseudo-sample approach can be easily adapted to diagnose faults, occurring during specific time intervals of the batch under study. For instance, in this case, instantaneous pseudo-sample based contribution plots might have also been constructed[x], but, it was not necessary because, for this particular process run, both SPE and D-statistic were found beyond the 99% control limits at all

[x]Specifically by i) computing per each of the J considered variables one single pseudo-sample using the value it assumes when a specific *out-of-control* signal is detected; ii) transforming this pseudo-sample by applying the same kernel function as for the training data; iii) double-centring the resulting pseudo-sample kernel vector and projecting it onto the K-PCA model space, as detailed in Section 7.2; iv) substituting the resulting score value to the median value $\widetilde{t}^{\mathbf{K}}_{\mathbf{V}_{j,Fault},a}$ in Equation 7.14.

the sampling points (see Figure 7.1). However, all the possible instantaneous DD plots showed very similar outcomes as that in Figure 7.3 (results not shown). For

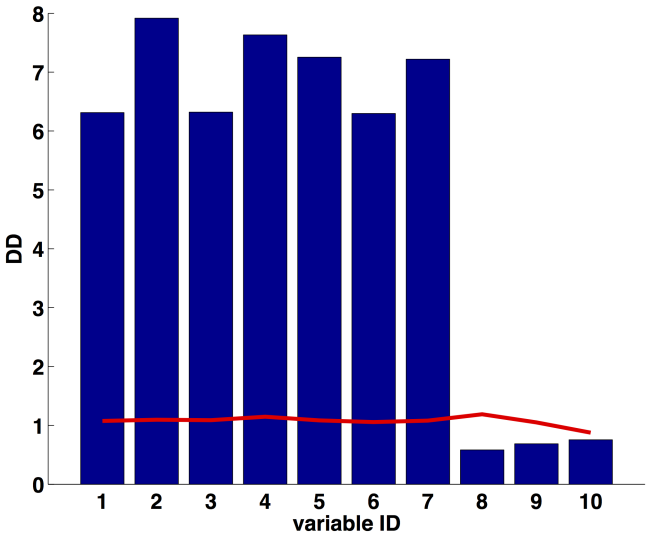

Figure 7.3 Simulated dataset/variability increase (VCS-VWU): K-PCA - Overall DD plot related to the first faulty test batch. The solid red line represents the 90% confidence limit calculated for the DD values of every variable by a jackknife-based procedure

the sake of clarity, the original pseudo-sample trajectories are represented in Figure 7.4. In order to ease the interpretation of Figure 7.3 and quickly identify the process variables that are different in the faulty batch compared to the NOC runs, 90% confidence limits were calculated for the different DD values using the jackknife procedure, proposed in [67]. As expected, variables x_8, x_9 and x_{10} have no statistically significant contributions to the fault (values below the respective confidence limits). Furthermore, in accordance with the way the data were simulated, similar contributions were found for variables from x_1 to x_7.
Exploiting the features of the pseudo-sample trajectories represented in Figure 7.4, a modification of the original DD plot can also be developed to selectively focus on changes in the variance of the process variables under study, as in this specific case. This variant is named *Pseudo-sample Trajectory Variance Ratio plot* (VR plot) and the contribution of each variable to the fault is evaluated according to the VR_m index, calculated as:

$$VR_j = \sum_{a=1}^{A} \frac{max(\{s^2_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,NOC},a}}\}, \{s^2_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,fault},a}}\})}{min(\{s^2_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,NOC},a}}\}, \{s^2_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,fault},a}}\})} \tag{7.17}$$

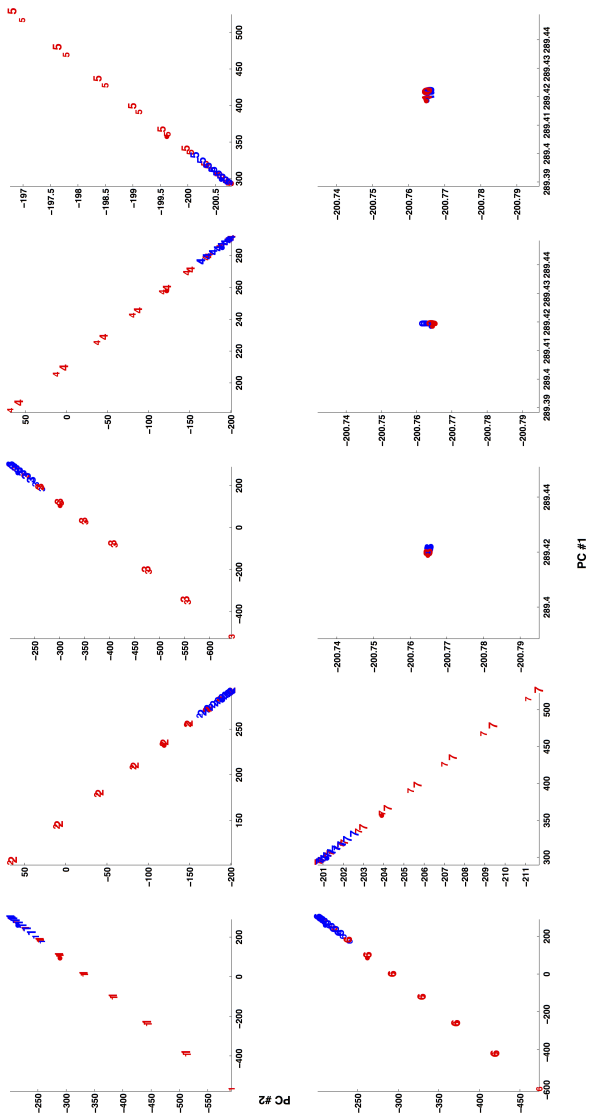

Figure 7.4 Simulated dataset/variability increase (VCS-VWU): K-PCA - Original pseudo-sample trajectories obtained by the procedure described in Section 7.2. The scores represented as blue numbers are calculated starting from the minimum and the maximum values each preprocessed variable assumes in all the training NOC batches. The scores represented as red numbers are calculated starting from the minimum and the maximum values each preprocessed variable assumes in the considered *out-of-control* test batch. Their graphical comparison permits to evaluate whether the original variables assume values either inside or outside their *in-control* variability range, during the occurrence of a fault. PC stands for *principal component*

where $s^2_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,NOC},a}}$ and $s^2_{\mathbf{t}^{\mathbf{K}}_{\mathbf{V}_{j,fault},a}}$ correspond to the variance of the column vectors of $\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,NOC}}$ and $\mathbf{T}^{\mathbf{K}}_{\mathbf{V}_{j,fault}}$, respectively, related to the a-th component of the model. In Figure 7.5 the overall VR plot related to the first faulty test batch is displayed. Also in this case, a completely correct fault diagnosis was obtained.

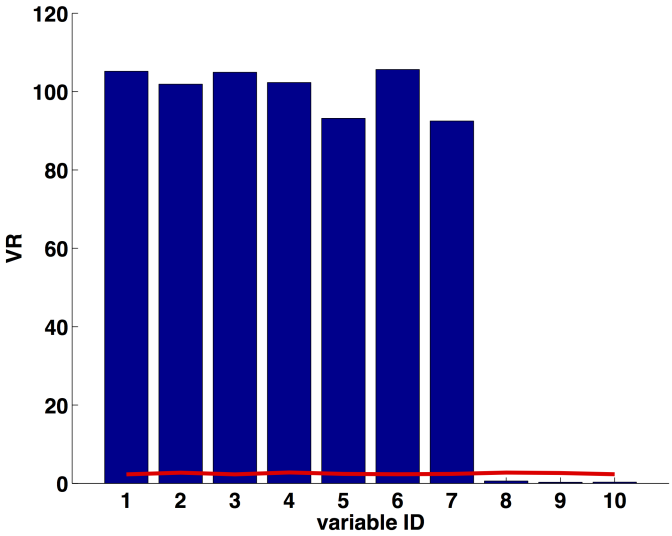

Figure 7.5 Simulated dataset/variability increase (VWU-VCS): K-PCA - Overall VR plot related to the first faulty test batch. The solid red line represents the 90% confidence limit calculated for the VR values of every variable by a jackknife-based procedure

The rationale behind the definition of the VR index is associated to the frequent need in process monitoring to distinguish whether an *out-of-control* signal is generated by shifts in the average level of specific measured variables or by changes in their variability. In fact, although the DD plot has proven to be an effective tool for detecting the latter deviations in case they represent the only differences between *in-control* and faulty batches and a second-order polynomial function is found to be the optimal for the data under study, in more complex scenarios, when a different kernel transformation is performed, coupling it with the VR plot may constitute a feasible and valid option for unveiling both types of variation.
The same modelling strategy was followed when TCS-VWU and TCS-BWU were used to preprocess and unfold the original simulated three-way arrays. The results are summarised in Figures 7.6-7.10 and Tables 7.3-7.6. Regarding TCS-VWU, a similar good accuracy was found for both the PCA- and the K-PCA-based SPE and D-statistic control charts. Nevertheless, the interpretation of the classical SPE and D-statistic contribution plots is not obvious, which drastically jeopardises the fault diagnosis. On the contrary, both the DD and the VR plots permit to cor-

Table 7.3 Simulated dataset/variability increase (TCS-VWU): Overall Type I (OTI) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes. The table lists also the number of principal components (PCs) and the goodness of fit (R^2) of the two different *in-control* models

	PCs	R^2	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	2	0.631	6.4%	1.6%	4.8%	1.6%
K-PCA (second-order polynomial)	2	0.729	6.4%	1.6%	5.6%	1.6%

Table 7.4 Simulated dataset/variability increase (TCS-VWU): Overall Type II ($OTII$) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes

	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	6.9%	13.6%	0.0%	0.0%
K-PCA (second-order polynomial)	0.0%	0.0%	0.0%	0.0%

Table 7.5 Simulated dataset/variability increase (TCS-BWU): Type I (TI) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes. The table lists also the number of principal components (PCs) and the goodness of fit (R^2) of the two different *in-control* models

	PCs	R^2	$\text{SPE}_{\text{ISL=20\%}}$	$D_{\text{ISL=20\%}}$
Classical PCA	2	0.746	20.0%	20.0%
K-PCA (Gaussian, σ=89.1)	2	0.976	20.0%	20.0%

Table 7.6 Simulated dataset/variability increase (TCS-BWU): Type II (TII) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes

	$\text{SPE}_{\text{ISL=20\%}}$	$D_{\text{ISL=20\%}}$
Classical PCA	0.0%	0.0%
K-PCA (Gaussian, σ=89.1)	0.0%	0.0%

rectly identify the process variables showing a different evolution with respect to an *in-control* situation.

Concerning TCS-BWU, as each row of the final unfolded data arrays contains all the information registered for one batch, the comparison was carried out assessing the Type I (TI) and Type II (TII) risk values. They stand for the percentage of NOC process runs detected as abnormal and the percentage of faulty ones detected as NOC, respectively. Owing to the limited number of analysed batches, in this

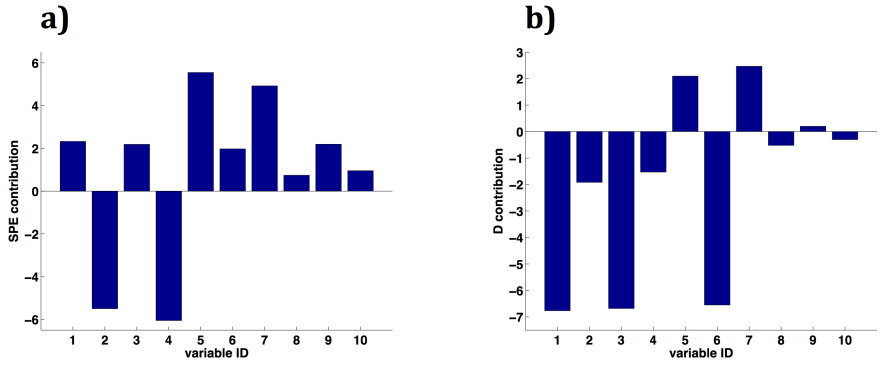

Figure 7.6 Simulated dataset/variability increase (TCS-VWU): classical PCA - a) SPE and b) D-statistic contribution plots related to the first sampling point of the first faulty test batch. The displayed profiles are consistent with those observed for all the other *out-of-control* signals

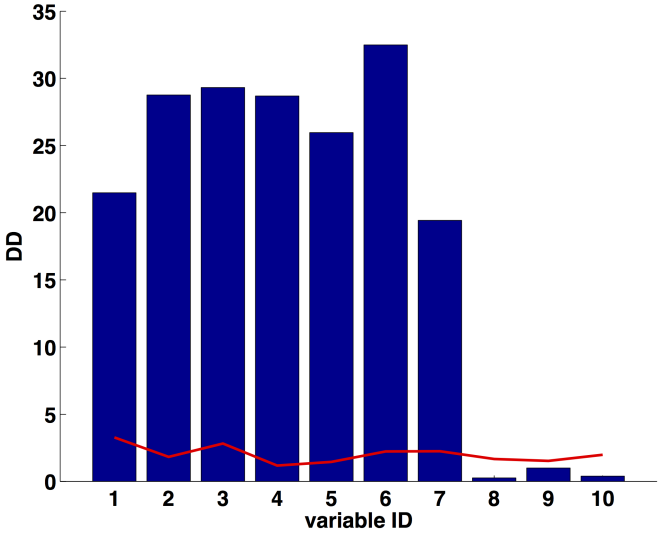

Figure 7.7 Simulated dataset/variability increase (TCS-VWU): K-PCA - Overall DD plot related to the first faulty test batch. The solid red line represents the 90% confidence limit calculated for the DD values of every variable by a jackknife-based procedure

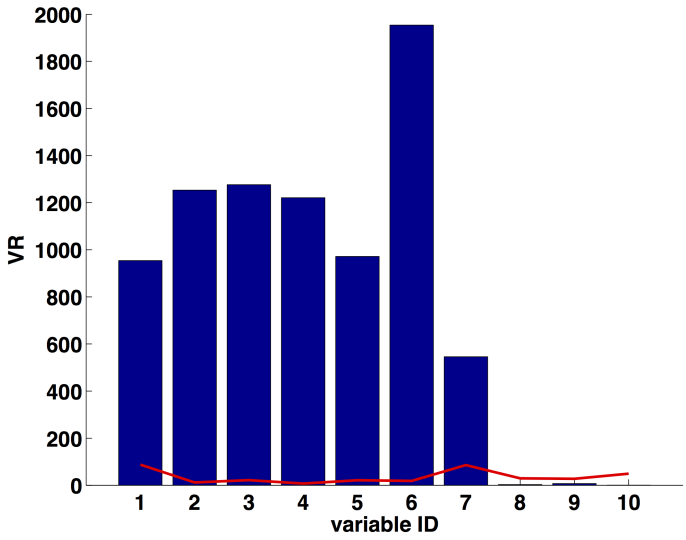

Figure 7.8 Simulated dataset/variability increase (TCS-VWU): K-PCA - Overall VR plot related to the first faulty test batch. The solid red line represents the 90% confidence limit calculated for the VR values of every variable by a jackknife-based procedure

case, the control limits of the SPE and D-statistic charts were adjusted so that the TI risk value was equal to an α of 20%. No significant differences in terms of fault detection power were found in either the SPE or the D-statistic control charts obtained by classical PCA and K-PCA (with a Gaussian data transformation and a σ parameter value of 89.1). Also the fault diagnosis was correctly addressed by both the studied approaches. Here, three aspects have to be carefully taken into account:

- since every row of the final training and test matrices contains information associated to the entire evolution of a particular process run, a single SPE and a single D-statistic value is calculated for each one of them. For the same reason, the procedure to construct the DD plots employed for diagnostic purposes has to be redefined as specified before. Notice that this point also applies when LFE is resorted to for building batch monitoring schemes;
- applying BWU to the original three-way data structures permits to compare the considered faulty process run with the training NOC ones time instant-to-time instant and, then, compute a contribution or a DD value per each sampling point of the evolution of every variable;

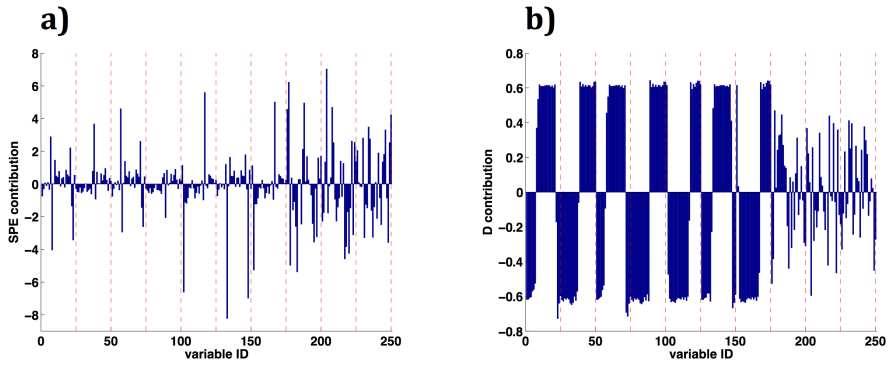

Figure 7.9 Simulated dataset/variability increase (TCS-BWU): classical PCA - a) SPE and b) D-statistic contribution plots related to the first faulty test batch. The vertical dotted red lines mark the separation between the contributions at the different sampling times of two consecutive process variables

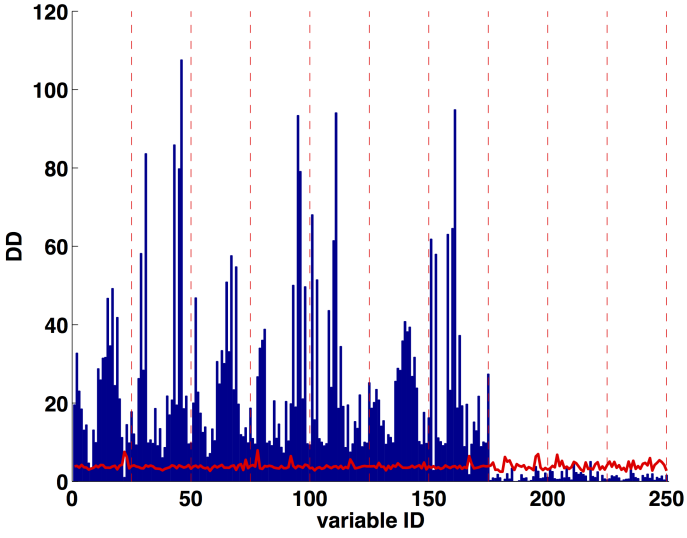

Figure 7.10 Simulated dataset/variability increase (TCS-BWU): K-PCA - Overall DD plot related to the first faulty test batch. The solid red line represents the 90% confidence limit calculated for the DD values of every variable at the different time instants by a jackknife-based procedure. The vertical dotted red lines mark the separation between the DD values at the different sampling times of two consecutive process variables

- as in this case the pseudo-sample based strategy takes into account the variability range of every variable at a specific time instant of the evolution of all the training batches and the single value that the same variable assumes at the same time instant in a specific run detected as faulty, the VR indices cannot be calculated and the respective pseudo-sample based contribution plot cannot be constructed.

The similarity between the performance of classical PCA and K-PCA results from coupling TCS and BWU. In fact, their combination permits to remove the strongly non-linear dynamic trend of the evolution of the considered variables from the data and, at the same time, model their variation about their average trajectories at the single sampling points, reducing the process analysis to a stationary problem [55]. This is also proven by the fact that a Gaussian function with a relatively high σ parameter was selected after a previous step of cross-validation as the optimal one for dealing with such data (N.B. the higher the σ parameter, the more linear the kernel transformation).

7.4.2 Simulated dataset - Variability decrease detection case study

A similar comparison was carried out inverting the training and the test sets of the previous simulated case study in order to verify how classical PCA and K-PCA perform when a decrease in the variability of the process variables has to be detected and diagnosed. In particular, the *in-control* models were built on 10 out of the 15 batches characterised by a higher variance of the measured variables (randomly chosen), the adjustment of the resulting control chart limits was assessed after the projection of the 5 batches, left out of the training set, and the fault detection and diagnosis power of the final monitoring schemes was evaluated as described before using the remaining 15 process runs. The results are displayed in Figures 7.11-7.15 and Tables 7.7-7.12.

VCS-VWU did not lead to satisfactory performance with either PCA or K-PCA. This is a consequence of the fact that mean-centring and scaling the complete variable trajectories of the test batches (showing a lower standard deviation) by the larger standard deviation of those of the training runs modified their profiles so that they showed very slight fluctuations around zero. For this reason, the scores associated to every sampling time of the evolution of a new batch were concentrated around the origin of the latent space, which, combined with the large variability of the training process runs, made the detection of any fault impossible. After TCS-VWU, the D-statistic control charts resulting from both the second-order polynomial K-PCA and classical PCA models were found to be characterised by a similar fault detection accuracy. Nevertheless, only the DD and VR plots permitted to correctly identify the process variables showing a decrease in their variability in the test batches, while the classical SPE and D-

Table 7.7 Simulated dataset/variability decrease (VCS-VWU): Overall Type I (OTI) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes. The table lists also the number of principal components (PCs) and the goodness of fit (R^2) of the two different *in-control* models

	PCs	R^2	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	2	0.703	4.8%	1.6%	4.8%	1.6%
K-PCA (second-order polynomial)	2	0.697	4.8%	2.4%	6.4%	2.4%

Table 7.8 Simulated dataset/variability decrease (VCS-VWU): Overall Type II ($OTII$) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes

	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	96.5%	98.9%	51.7%	84.0%
K-PCA (second-order polynomial)	100%	100%	90.1%	100%

Table 7.9 Simulated dataset/variability decrease (TCS-VWU): Overall Type I (OTI) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes. The table lists also the number of principal components (PCs) and the goodness of fit (R^2) of the two different *in-control* models

	PCs	R^2	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	2	0.638	4.8%	1.6%	4.8%	3.2%
K-PCA (second-order polynomial)	3	0.812	5.6%	2.4%	4.8%	1.6%

Table 7.10 Simulated dataset/variability decrease (TCS-VWU): Overall Type II ($OTII$) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes

	$\text{SPE}_{\text{ISL=5\%}}$	$\text{SPE}_{\text{ISL=1\%}}$	$D_{\text{ISL=5\%}}$	$D_{\text{ISL=1\%}}$
Classical PCA	85.9%	91.7%	0.7%	24.5%
K-PCA (second-order polynomial)	56.3%	76.5%	0.0%	0.0%

statistic contribution plots were not able to recognise them and suffer from lack of interpretability. Finally, the two monitoring schemes constructed after TCS-BWU were compared. In this case, the D-statistic control charts resulting from a Gaussian K-PCA (σ=19.3) and a classical PCA model proved to have a high accuracy in terms of fault detection power and both the classical contribution and the DD plots enabled a correct fault diagnosis.

Table 7.11 Simulated dataset/variability decrease (TCS-BWU): Type I (TI) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes. The table lists also the number of principal components (PCs) and the goodness of fit (R^2) of the two different *in-control* models

	PCs	R^2	$\text{SPE}_{\text{ISL=20\%}}$	$D_{\text{ISL=20\%}}$
Classical PCA	2	0.737	20.0%	20.0%
K-PCA (Gaussian, σ=19.3)	3	0.943	20.0%	20.0%

Table 7.12 Simulated dataset/variability decrease (TCS-BWU): Type II (TII) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes

	$\text{SPE}_{\text{ISL=20\%}}$	$D_{\text{ISL=20\%}}$
Classical PCA	60.0%	0.0%
K-PCA (Gaussian, σ=19.3)	33.3%	0.0%

A summary of the outcomes obtained in each of the described simulated case study is displayed in Table 7.13.

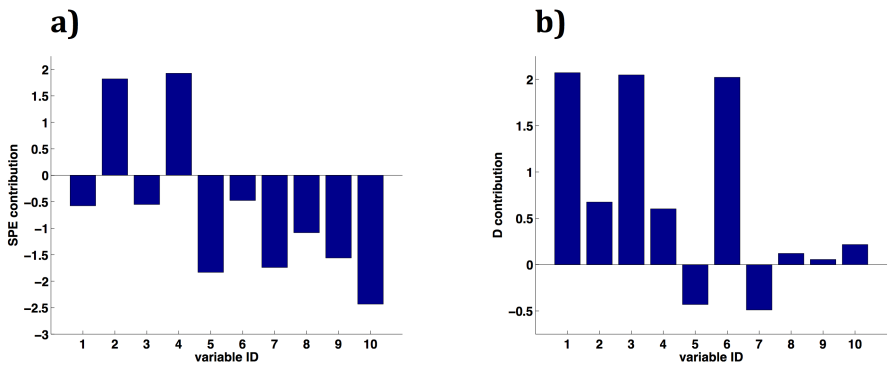

Figure 7.11 Simulated dataset/variability decrease (TCS-VWU): Classical PCA - a) SPE and b) D-statistic contribution plots related to the first sampling point of the tenth faulty test batch. The displayed profiles are consistent with those observed for all the other *out-of-control* signals

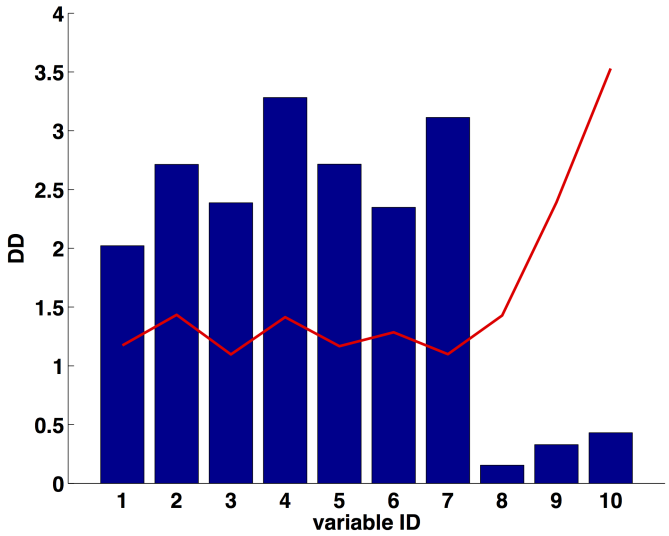

Figure 7.12 Simulated dataset/variability decrease (TCS-VWU): K-PCA - Overall DD plot related to the tenth faulty test batch. The solid red line represents the 90% confidence limit calculated for the DD values of every variable by a jackknife-based procedure

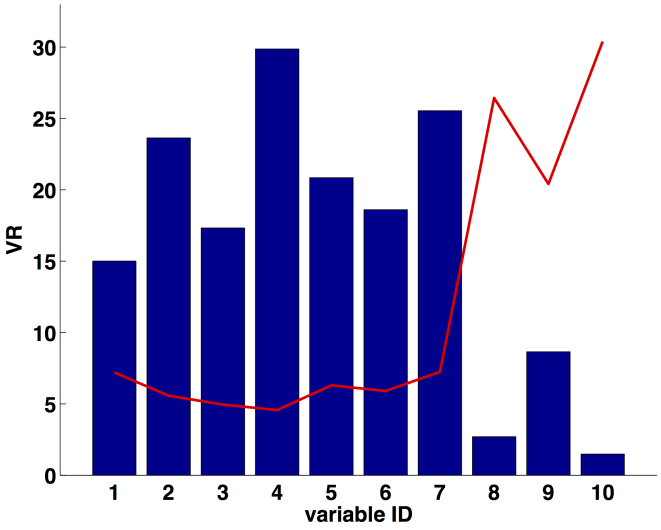

Figure 7.13 Simulated dataset/variability decrease (TCS-VWU): K-PCA - Overall VR plot related to the tenth faulty test batch. The solid red line represents the 90% confidence limit calculated for the VR values of every variable by a jackknife-based procedure

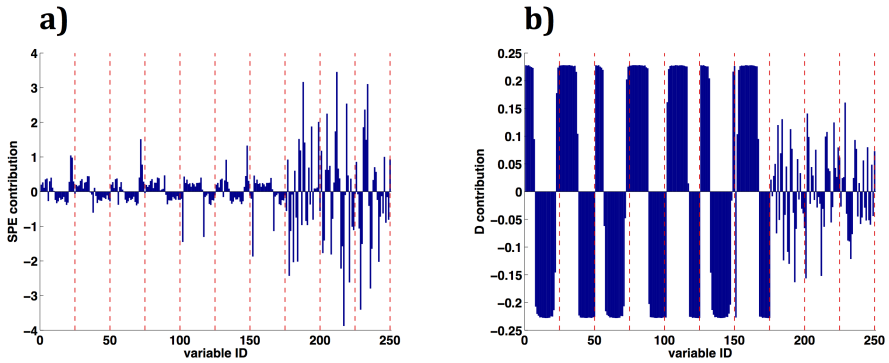

Figure 7.14 Simulated dataset/variability decrease (TCS-BWU): Classical PCA - a) SPE and b) D-statistic contribution plots related to the first faulty test batch. The vertical dotted red lines mark the separation between the contributions at the different sampling times of two consecutive process variables

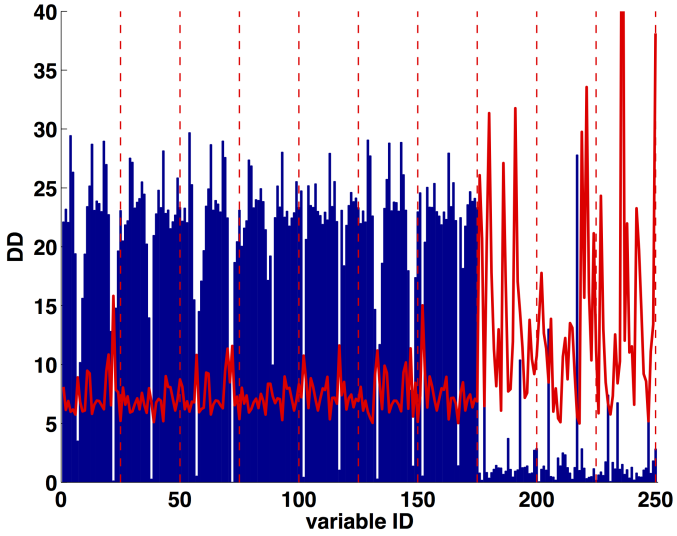

Figure 7.15 Simulated dataset/variability decrease (TCS-BWU): K-PCA - Overall DD plot related to the first faulty test batch. The solid red line represents the 90% confidence limit calculated for the DD values of every variable at the different time instants by a jackknife-based procedure. The vertical dotted red lines mark the separation between the DD values at the different sampling times of two consecutive process variables

Table 7.13 Simulated dataset: summary of the results obtained in the different case studies. The table shows which of the classical PCA- or K-PCA-based control charts and contribution plots were able to detect and diagnose the considered faults, respectively.

		Variability increase		Variability decrease	
		Fault detection	Fault diagnosis	Fault detection	Fault diagnosis
VCS-VWU	PCA	D	NO	NO	NO
	K-PCA	D, SPE	DD, VR	NO	NO
TCS-VWU	PCA	D, SPE	NO	D	NO
	K-PCA	D, SPE	DD, VR	D	DD, VR
TCS-BWU	PCA	D, SPE	D	D	D
	K-PCA	D, SPE	DD	D	DD

7.4.3 Chemical process dataset

In order to build a first K-PCA monitoring scheme on the chemical process dataset, VCS-VWU was applied[xi]. A training set, containing observations associated to 20 NOC batches, a first test set, including 11 NOC process runs, and a second test set, consisting of 5 batches manufactured with a deviating steam supply were obtained by unfolding the original three-way arrays. For increasing the quality of the final monitoring scheme, a multi-stage modelling strategy was resorted to: the training set was further divided into 4 sub-matrices, each one related to a different process phase, on which, 4 cross-validated K-PCA models were built. As highlighted in Table 7.14, except for stage #1, a Gaussian kernel function was selected for the original data transformation. After performing K-PCA on the training data and

Table 7.14 Chemical process dataset (VCS-VWU): parameters of the K-PCA *in-control* models built for the different process stages

	Stage #1	Stage #2	Stage #3	Stage #4
Function Type	3-rd order polynomial	Gaussian	Gaussian	Gaussian
CV-σ	-	23.8	15.0	12.8
PCs	3	2	2	2
R^2	0.852	0.850	0.798	0.775

assessing the adjustment of the limits of the resulting control charts using the 11 NOC batches of the first test set, the observations related to the faulty process runs were transformed in the same way as for the corresponding training set and projected onto the corresponding model space. As an example, the case of the fifth faulty test batch is reported. Figure 7.16 shows its respective SPE and the D-statistic control charts. In every process stage, the SPE values are always

[xi] Prior to the data modelling, variable trajectory synchronisation was performed by stage-wise Dynamic Time Warping [68] to guarantee all the process runs had the same evolution pace.

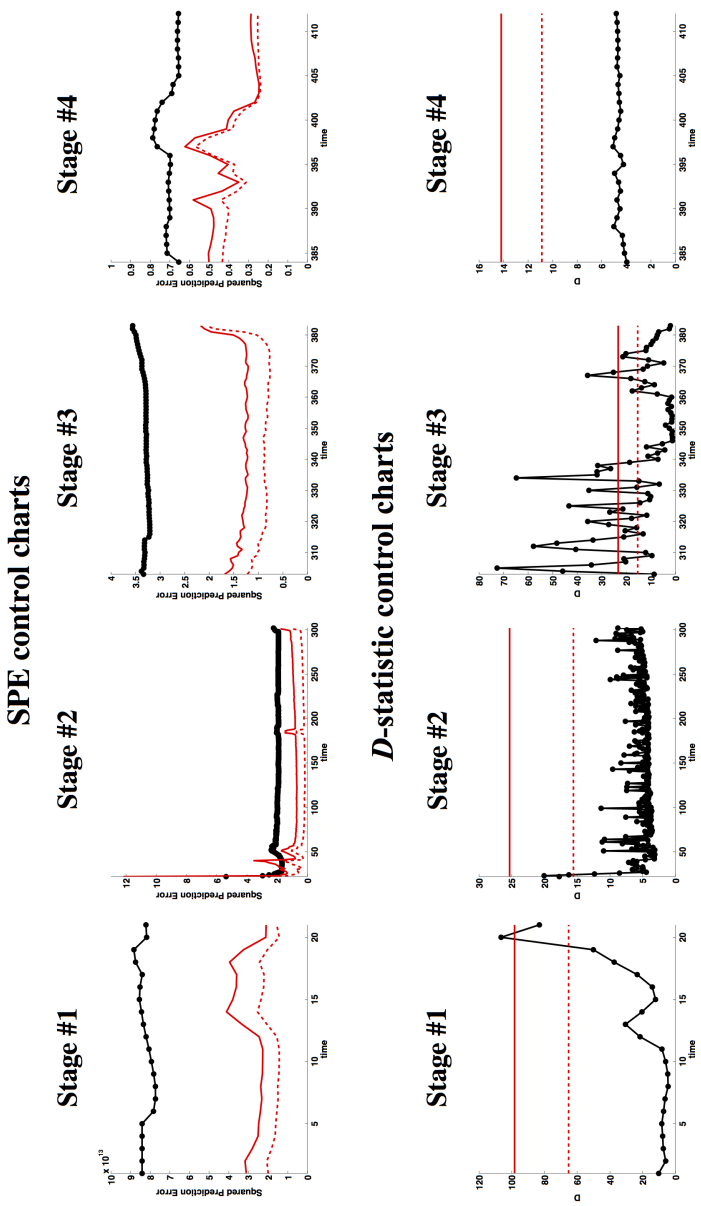

Figure 7.16 Chemical process dataset (VCS-VWU): K-PCA - SPE and D-statistic control charts related to every stage of the evolution of the fifth faulty batch contained in the second test set. The dotted and the solid red lines represent the 95% and the 99% control limits, respectively

beyond the 95% control limit and, at most of the time samples, beyond the 99% control limit, which evidently highlights an issue affecting the evolution of this process run all over its duration. To investigate the nature of this problem and pinpoint which variables evolve differently with respect to an *in-control* situation, the DD plots were exploited also in this case. In Figure 7.17, those related to every process stage and associated to the fifth faulty test batch are displayed. They clearly point out an issue which affects variable x_{22} during all the evolution of the batch. This exactly corresponds to the steam pressure, which is known to be the main cause generating the deviating behaviour of all the process runs included in the second test set. Furthermore, regarding stage #3, a further problem is highlighted. In this case, the variable pinpointed as having the highest contribution to the fault is x_4. At a first glance, the fault diagnosis might be considered correct, since the main problem associated to the steam pressure is properly identified. Nevertheless, by inspecting more carefully the original variable trajectories, it was possible to realise that the contribution of variable x_1 in stage #2 was much lower than expected (see Figure 7.18). This variable contribution masking effect is due to the combination between VCS-VWU and the pseudo-sample projection strategy

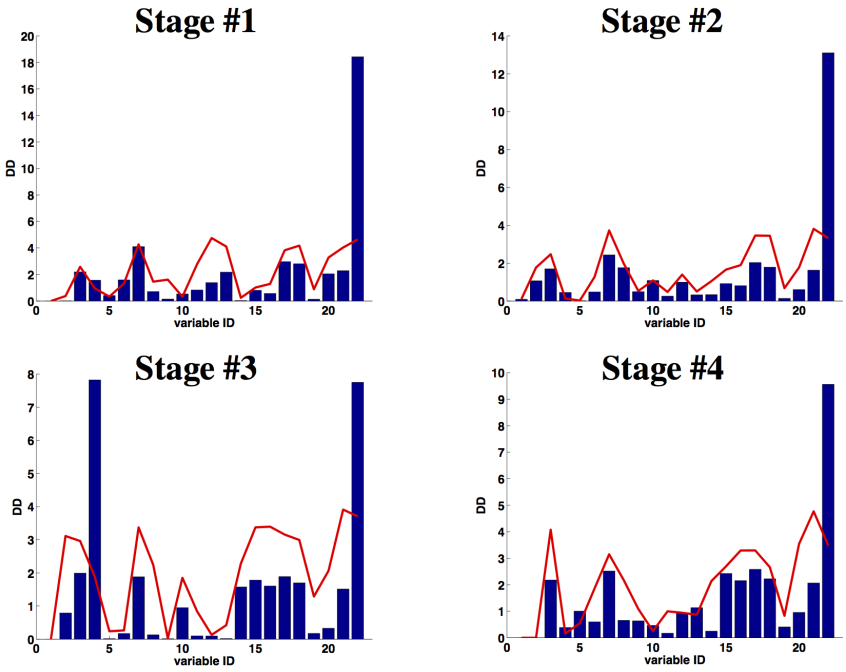

Figure 7.17 Chemical process dataset (VCS-VWU): K-PCA - Overall stage-DD plots related to the fifth faulty test batch. The solid red lines represent the 95% confidence limit calculated for the DD values of every variable by a jackknife-based procedure

adopted for fault diagnosis in this chapter. In fact, the reader should remind that the pseudo-sample matrices are filled in with values ranging from the minimum to the maximum of a certain column of the original preprocessed data related to the *in-control* process runs and to the batch detected as faulty. When VCS-VWU is applied, each of these columns contains the complete evolution of a single variable, which preserves its non-linear dynamic trend. If the minimum and the maximum of the variable under study in the two cases roughly correspond, the resulting pseudo-sample trajectories would cover approximately the same variability range. This would generate a really small difference between the medians of the pseudo-sample score distributions and, thus, a really small contribution for this variable, even if its evolution in the faulty batch is different compared to the *in-control* process runs. That is exactly what happens with variable x_1 in stage #2 for the fifth batch of the second test set, as highlighted in Figure 7.18. Similar results were obtained in terms of fault detection accuracy for TCS-VWU, while the resulting *DD* plots enabled a slightly clearer, but not completely satisfying fault diagnosis (results not shown). The cross-validated values of the σ parameters of the kernel functions were found to be all between 50 and 75 for the different process stages, which means a more linear kernel transformation is needed for adequately modelling this dataset if TCS-VWU is used.

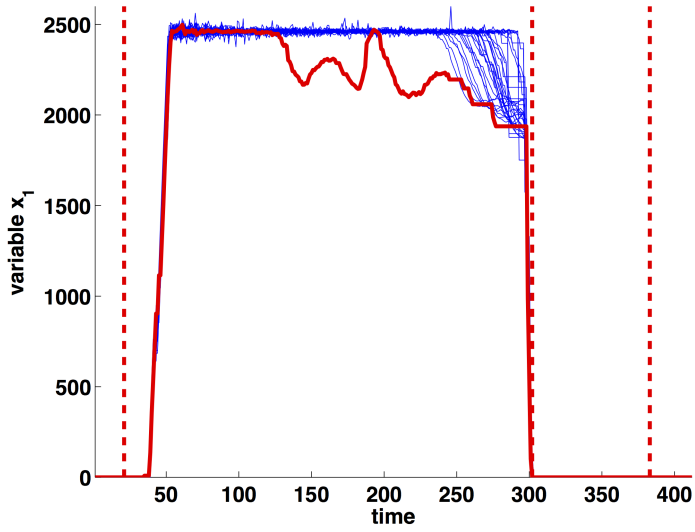

Figure 7.18 Chemical process dataset: time evolution of the variable x_1 in the NOC batches (blue thin lines) and in the fifth faulty process run contained in the second test set (red thick line). The vertical dotted red lines mark the end point of every process stage

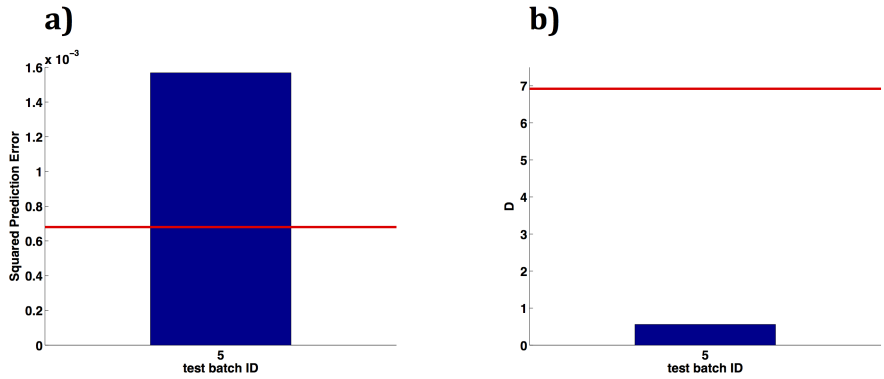

Figure 7.19 Chemical process dataset (TCS-BWU): K-PCA - a) SPE and b) D-statistic control charts related to the fifth faulty batch contained in the second test set. The solid red lines represent their respective 90.9% control limit

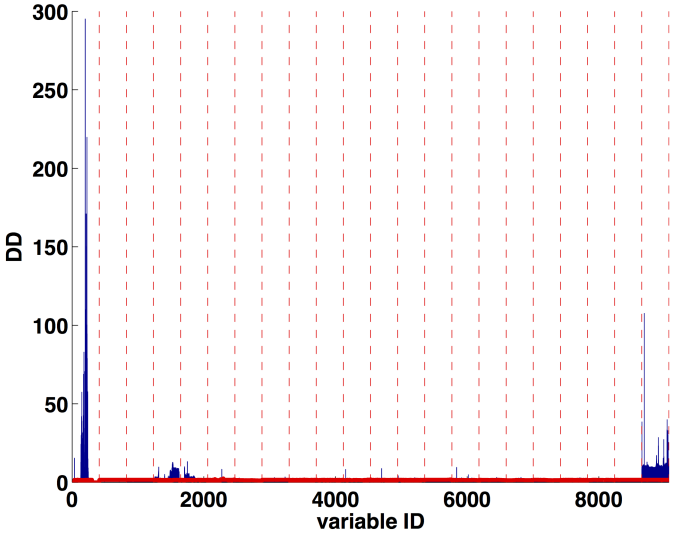

Figure 7.20 Chemical process dataset (TCS-BWU): K-PCA - Overall DD plot related to the fifth faulty test batch. The solid red line represents the 95% confidence limit calculated for the DD values of every variable by a jackknife-based procedure. The vertical dotted red lines mark the separation between the DD values at the different sampling times of two consecutive process variables

A simple way to definitely solve the problem of the inconsistency of the variable contributions to the fault of the previous *DD* plots is to switch to TCS-BWU. Here, a single K-PCA model was built on the whole training set, because no significant differences were detected when a multi-stage strategy was applied. A Gaussian function was selected to transform the unfolded data. The optimised σ parameter was fixed at a value of approximately 700, which lead to an approximately linear kernel transformation. This corroborates what was stated before about the coupling of TCS and BWU. Figure 7.19 indicates that the fifth faulty test batch is correctly identified as an outlier by the SPE control chart. In this case, a more accurate fault diagnosis is enabled, as proven by the *DD* plot related to the fifth batch of the second test set, displayed in Figure 7.20 where also the contribution of variable x_1 is correctly highlighted. Furthermore, it seems to be much higher than the other ones, in contrast with what was observed when VWU was applied for the analysis of these data. The time evolution of the original variables, found to have a high contribution to the fault (x_1, x_4, x_5 and x_{22}), in the training NOC

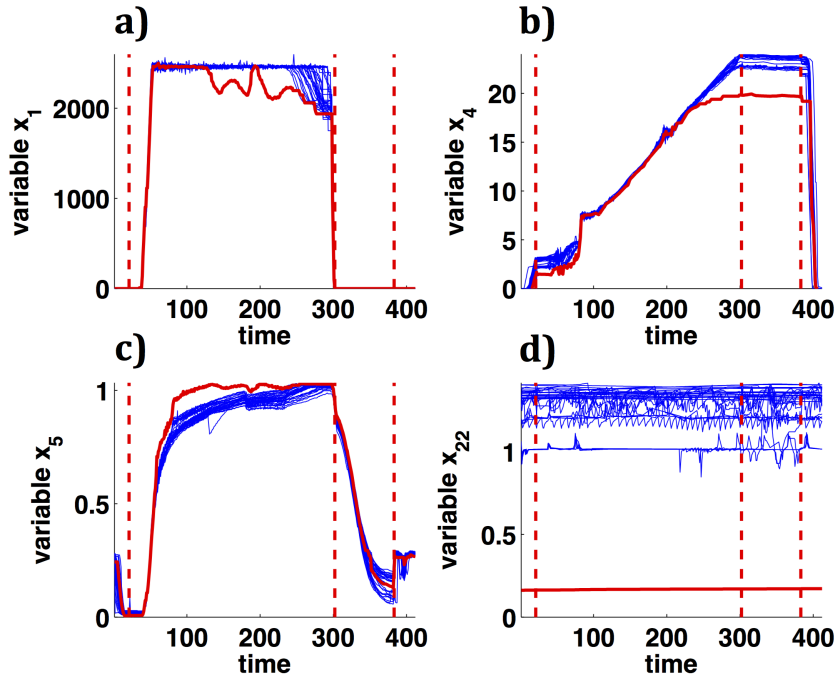

Figure 7.21 Chemical process dataset: time evolution of a) variable x_1, b) variable x_4, c) variable x_5, d) variable x_{22}, identified as having a high contribution to the fault, in the NOC batches (blue thin lines) and in the fifth faulty test process run (red thick line). The vertical dotted red lines mark the end point of every process stage

batches and the faulty test process run under study is represented in Figure 7.21. The analysis of the chemical process dataset by means of classical PCA also resulted in outcomes, which were coherent with those obtained when K-PCA was resorted to (results not shown). This may be associated to the fact that not so strong non-linear relationships were present in the original data. However, the quality of the final monitoring schemes was found to be strictly dependent on the number of PCs, which could represent an additional issue to be solved for a correct fault detection and diagnosis.

7.4.4 Pharmaceutical batch drying process dataset

The pharmaceutical batch drying process dataset represents a specific case study, in which LFE is used for monitoring batch processes[xii]. Here, the *in-control* models were built on 17 process runs, which evolved under NOC, while the remaining 9 were used to assess the adjustment of the limits of the resulting SPE and D-statistic control charts according to the TI risk values, as done when TCS-BWU was applied previously in this chapter. Owing to the low number of analysed batches, the imposed significance level α was set at a value of 11.1. The fault detection power of the control charts constructed by both PCA and K-PCA was evaluated in terms of TII values, calculated after the projection of the 23 faulty test runs onto the respective model space. The results are listed in Table 7.15 and 7.16. In this case, the use of a third-order polynomial kernel transformation

Table 7.15 Pharmaceutical batch drying process dataset (LFE): Type I (TI) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes. The table lists also the number of principal components (PCs) and the goodness of fit (R^2) of the two different *in-control* models

	PCs	R^2	$\text{SPE}_{\text{ISL=11.1\%}}$	$D_{\text{ISL=11.1\%}}$
Classical PCA	2	0.790	11.1%	11.1%
K-PCA (third-order polynomial)	2	0.830	11.1%	11.1%

clearly improved the accuracy of the SPE control chart, leading to a higher number of test batches correctly detected as faulty (TII=17.4%, α=11.1%) than when resorting to classical PCA (TII=30.4%, α=11.1%), while similar TII values were found for the two different D-statistic control charts. In order to evaluate the fault diagnosis ability of both the monitoring schemes, the classical SPE contribution plot and the DD plot (constructed, for the reason previously mentioned, according to the adapted procedure described in Section 7.4.1) related to the first faulty test batch are displayed in Figure 7.22. The D-statistic contribution plot is not shown,

[xii] As such data were provided beforehand in a two-way array, the unfolding step was not required in this specific case and they were straight analysed in this form after standard autoscaling.

Table 7.16 Pharmaceutical batch drying process dataset (LFE): Type II (TII) risk values for the SPE and the D-statistic control charts resulting from both the classical PCA- and the K-PCA-based monitoring schemes

	$\text{SPE}_{\text{ISL}=11.1\%}$	$D_{\text{ISL}=11.1\%}$
Classical PCA	30.4%	69.2%
K-PCA (third-order polynomial)	17.4%	65.2%

as this specific process run did not exhibit a value for this index beyond the control limit. The fault diagnosis was correctly addresses by both the approaches under study. The variables pinpointed as having the highest contribution to the fault were x_2 (dryer temperature) and x_8 (batch duration). In [57], these were found to be the most critical parameters affecting the quality of the final product, together with the level of the solvent collector tank (variable x_1), which showed a normal behaviour in this particular batch.

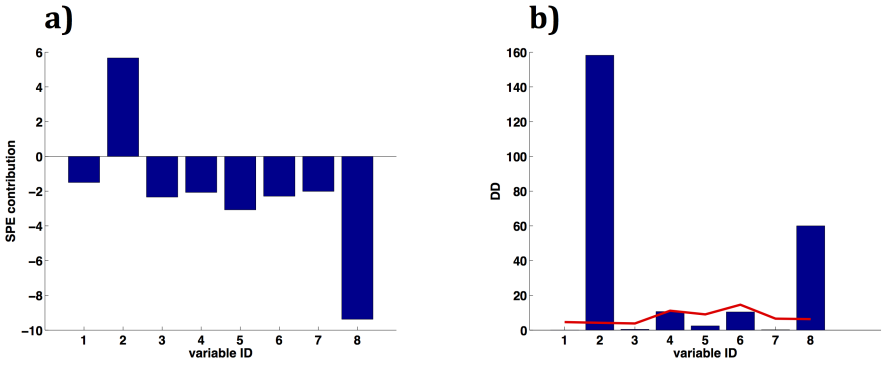

Figure 7.22 Pharmaceutical batch drying process dataset (LFE): a) SPE contribution plot (classical PCA) and b) DD plot (K-PCA) related to first faulty test batch. The solid red line represents the 94.1% confidence limit calculated for the DD values of every variable by a jackknife-based procedure

7.5 Conclusions

In this chapter a novel approach for fault detection and diagnosis, based on the combination of K-PCA and pseudo-sample projection, was proposed to deal with complex batch process datasets, usually difficult to handle if affected by strong non-linearities. In particular, an innovative tool, the so-called pseudo-sample based contribution plot (in the form of both DD and VR plot) was developed to overcome

the main drawback of kernel-based methods, that is, their hard interpretability (which does not permit to evaluate the importance of the original variables in the final models).
Two points were confirmed from the displayed outcomes:

- when strongly non-linear data structures have to be dealt with, non-linear kernel-based techniques show clear advantages compared to standard bilinear ones. In fact, in most of the simulated case studies, even if both the K-PCA- and classical PCA-based monitoring schemes were found to be characterised by a similar fault detection power, in terms of fault diagnosis, classical contribution plots suffered from lack of interpretability and did not lead to identify the process variables evolving differently with respect to an *in-control* situation. On the other hand, the pseudo-sample based contribution plots always yielded a completely correct diagnosis of the detected faults;

- even if no severe non-linear relationships affect the original analysed data, it is still possible to resort to kernel-based methods and pseudo-sample projection, obtaining very similar results to classical approaches. In fact, in all the circumstances in which PCA performed well, the cross-validation procedure, by which the best kernel transformation and its adjustable parameters were chosen, selected an approximately linear function (in general Gaussian with a relatively high value of σ), guaranteeing an equally correct fault detection and diagnosis.

In addition, the chemical process example highlighted that one of the two proposed pseudo-sample based contribution plots, the DD plot, enables a more correct and accurate identification of the measured variables responsible for the failure if it is used in combination with TCS-BWU and a single DD index is calculated for every sampling time of the evolution of every *out-of-control* batch.

Chapter 8

K-PLS and pseudo-sample trajectories for mixture data analysis

This chapter explores the potential of K-PLS coupled to pseudo-sample trajectories for the analysis of data proceeding from mixture designs of experiments.

Part of the content of this chapter has been included in:

1. Vitale, R.[i], Palací-López, D.[i], Kerkenaar, H., Postma, G., Buydens, L. & Ferrer, A. Kernel-Partial Least Squares regression coupled to pseudo-sample trajectories for the analysis of mixture designs of experiments. *Submitted*.

[i]These authors had equal contributions

8.1 Introduction

A wide range of products currently used in daily life result from processing blends of two or more ingredients. Hence, the physicochemical properties of these products mainly depend on the raw materials being mixed and on the proportions in which they are added. Alloys, as well as drugs and foodstuffs, are just some of the numerous examples where this applies, and their manufacturing can be considered a so-called *mixture problem* [69]. Traditionally, mixture problems are defined as those in which i) the proportions x_i of the I different constituents[ii] are related to the aforementioned properties, ii) these proportions are of at least as much relevance as their absolute quantities, and iii) their sum must be a fixed value (usually 1 or 100%):

$$\sum_{i=1}^{I} x_i = 1 \tag{8.1}$$

where $0 \leq x_i \leq 1$. This perfect collinearity restriction makes it impossible to modify the composition of any one of the ingredients independently from the rest. This implies that classical polynomial fitting by traditional methods (like Ordinary or Generalised Least Squares - OLS/GLS [70, 71]) is unfeasible as they assume the regressors to be linearly independent. On the other hand, alternative approaches e.g. the Scheffé models or their reparametrisation, the Cox models, can be exploited in these circumstances. However, both of them show several limitations: the former lack a constant term, cannot handle possible additional constraints, and their interpretation is non-intuitive; the latter always require a specific component blend to be set as reference, and their coefficients cannot be directly estimated by OLS/GLS [72, 73]. In order to solve such issues, PLS regression-based techniques can be resorted to [52, 74–76]: they have proven to guarantee satisfactory performance even when highly restricted mixture spaces have been dealt with and allow variables of different nature (e.g. component proportions and physicochemical properties as well as production process conditions) to be fused and simultaneously analysed. Nevertheless, if the mixture data under study are affected by strong non-linear relationships (which is rather common in e.g. industrial scenarios), applying classical PLS (even taking into account additional interaction and/or higher-degree terms) may not constitute an appropriate modelling strategy since it assumes their underlying structure is linear [7]. A good alternative may be represented by the combination of K-PLS regression and pseudo-sample trajectories, but the described way of defining the different pseudo-sample matrices (see Section 4.1.2) is not suitable when mixture problems are concerned, because it violates the constraint in Equation 8.1 (i.e. it is impossible to vary the composition of any one of the constituents independently from the rest), and then needs to be slightly adapted.

[ii] The mathematical indices and sub-indices were here modified with respect to the previous chapters in order to keep coherence with the historical scientific literature on mixture designs of experiments.

The main aim of this chapter is to evaluate the potential of such a combination in this particular field of interest, by comparing it with well-established methodologies, i.e. Scheffé model fitting by means of OLS and Cox model fitting by means of PLS. Both simulated and real case studies will be investigated.

8.2 Methods

As PLS and K-PLS regression have already been presented in Sections 2.4 and 4.1.1, only the basic principles of the Scheffé and Cox models and the extension of the pseudo-sample projection approach for mixture data handling will now be illustrated.

8.2.1 Scheffé and Cox models

Applying the constraint in Equation 8.1, the linear (first-order) and quadratic (second-order) Scheffé canonical polynomials can be expressed as:

$$\text{Linear model:} \qquad y = \sum_{i=1}^{I} \beta_i x_i + \epsilon \tag{8.2}$$

$$\text{Quadratic model:} \quad y = \sum_{i=1}^{I} \beta_i x_i + \sum_{i=1}^{I-1} \sum_{j=i+1}^{I} \beta_{i,j} x_i x_j + \epsilon \tag{8.3}$$

being y the value of the response property to be predicted, β_i the first-order model coefficient related to the i-th constituent of the mixture, $\beta_{i,j}$ the model coefficient for the interaction between the i-th and the j-th ingredient and ϵ an error term. In other words, β_i corresponds to the expected value of y for the hypothetical *pure mixture* composed by only the i-th constituent, while $\beta_{i,j}$ measures the synergism (or the antagonism) between the i-th and the j-th ingredient.
Although Scheffé polynomials can be fitted by conventional OLS, the interpretation of their parameters is not straightforward. For this reason, they are commonly reformulated into their equivalent Cox models:

$$\text{Linear model:} \qquad y = \alpha_0 + \sum_{i=1}^{I} \alpha_i x_i + \epsilon \tag{8.4}$$

$$\text{s.t.} \sum_{i=1}^{I} \alpha_i s_i = 0 \tag{8.5}$$

$$\text{Quadratic model:} \quad y = \alpha_0 + \sum_{i=1}^{I} \alpha_i x_i + \sum_{i=1}^{I-1} \sum_{j=i+1}^{I} \alpha_{i,j} x_i x_j + \sum_{i=1}^{I} \alpha_{i,i} x_i^2 + \epsilon \tag{8.6}$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{I} \alpha_i s_i = 0 \\ \sum_{j=1}^{I} c_{i,j} \alpha_{i,j} s_j = 0 \qquad \forall i \in \{1,\ 2,\ \ldots,\ I\} \end{cases} \tag{8.7}$$

where s_i is the proportion of the i-th ingredient in a specific mixture set as reference *a priori*; α_0 connotes the zero-order term of the polynomial; α_i and $\alpha_{i,i}$ denote the first-order and second-order model coefficients related to the i-th constituent of the mixture, respectively; $\alpha_{i,j}$ is the model coefficient for the interaction between the i-th and the j-th ingredient; and $c_{i,j} = \frac{1}{2}$ if $i \neq j$ or $c_{i,j} = 1$ if $i = j$. Here, α_0 represents the expected value of y for the reference mixture, α_i equals the difference between the expected value of y for the pure mixture composed by only the i-th constituent and the expected value of y for the reference mixture, and both $\alpha_{i,i}$ and $\alpha_{i,j}$ contribute to the response function curvature as for classical polynomials.
It can be easily demonstrated (see Section 14.1.3) that the Scheffé model coefficients can be derived from the Cox model ones as:

$$\text{Linear model:} \quad \beta_i = \alpha_0 + \alpha_i \tag{8.8}$$

$$\text{Quadratic model:} \quad \beta_i = \alpha_0 + \alpha_i + \alpha_{i,i} \tag{8.9}$$

$$\beta_{i,j} = \alpha_{i,j} - \alpha_{i,i} - \alpha_{j,j} \tag{8.10}$$

On the contrary, if the Scheffé model parameters are given, the corresponding Cox model ones can be calculated by solving the linear equation system encompassing either Equations 8.8 and 8.5 or Equations 8.9, 8.10 and 8.7 [69].
Cox canonical polynomials are probably the most intuitive approaches for mixture problem solving. However, the computation of their coefficients cannot be carried out by OLS/GLS. PLS regression can instead be utilised to this end by augmenting the predictor array $\mathbf{X}$ $(N \times I)$[iii] with interaction and/or higher-than-first-order terms to take into account their corresponding effect on the properties of interest.

8.2.2 Pseudo-sample trajectories for mixture data

In order to account for the mixture constraint, the pseudo-samples matrices $\mathbf{V}_i$ should be structured so that the V values in their i-th column range from the minimum to the maximum proportion of the i-th ingredient and all the elements of each one of their rows sum up to 1. E.g. if a ternary mixture problem is faced, a hypothetical $\mathbf{V}_1$ may have such an aspect:

$$\mathbf{V}_1 = \begin{bmatrix} 0, & 0.5, & 0.5 \\ 0.2, & 0.4, & 0.4 \\ 0.4, & 0.3, & 0.3 \\ 0.6, & 0.2, & 0.2 \\ 0.8, & 0.1, & 0.1 \\ 1, & 0, & 0 \end{bmatrix} \tag{8.11}$$

[iii] $\mathbf{X}$ contains the proportions of the I ingredients for the N sampled blends.

More generically:

$$\mathbf{V}_i = \begin{bmatrix} \frac{1-v_{1,i}}{I-1}, & \frac{1-v_{1,i}}{I-1}, & \dots, & \min(\mathbf{x}_i), & \frac{1-v_{1,i}}{I-1}, & \dots, & \frac{1-v_{1,i}}{I-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1-v_{v,i}}{I-1}, & \frac{1-v_{v,i}}{I-1}, & \dots, & \vdots & \frac{1-v_{v,i}}{I-1}, & \dots, & \frac{1-v_{v,i}}{I-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1-v_{V,i}}{I-1}, & \frac{1-v_{V,i}}{I-1}, & \dots, & \max(\mathbf{x}_i), & \frac{1-v_{V,i}}{I-1}, & \dots, & \frac{1-v_{V,i}}{I-1} \end{bmatrix} \tag{8.12}$$

where $\mathbf{x}_i$ is the i-th column vector of $\mathbf{X}$ and $v_{v,i}$ refers to the $v \times i$ entry of $\mathbf{V}_i$. As shown in Figure 8.1, this would mean spanning in the design space (a triangle in this case) the direction connecting the vertex associated to the pure blend composed by only the i-th constituent ([1, 0, 0], if $i = 1$) and the midpoint of its opposite side ([0, 0.5, 0.5], if $i = 1$)[iv]. As will be highlighted in Sections 8.4.3 and

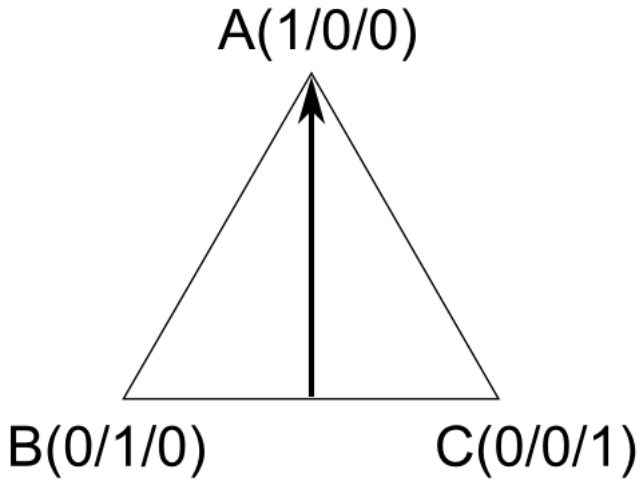

Figure 8.1 Graphical representation of the direction spanned by the pseudo-sample trajectory associated to the constituent A in a generic ternary mixture design space. Notice that the three vertices correspond to the three pure mixtures composed by only the ingredient A, B or C, respectively.

8.4.4, the representation of the corresponding pseudo-sample trajectories yields the so-called trace plot, traditionally used in Cox model analysis to get an idea of the linear and non-linear effects on the property of interest generated by the change in the proportion of every i-th ingredient. However, as most of them are

[iv] Equation 8.12 is not valid if the design space is not a simplex or if it is a simplex but the ingredient proportions do not vary from 0 to 1. Anyway, it is straightforward to extend the described methodology to handle such situations.

confounded with those due to the simultaneous variation of the proportion of the other constituents, a precise identification of the individual Scheffé polynomial coefficients cannot be achieved.

8.2.3 Pseudo-sample-based response surfaces

Alternatively, by using a combination of multiple pseudo-sample trajectories and graphing them in a contour plot, the response surface for the full mixture design space can be retrieved. To this end, every pseudo-sample matrix has to be constructed by i) fixing the proportions of all but two constituents, ii) increasing the proportion of one of these two constituents, and iii) decreasing the proportion of the other accordingly (for keeping the sum in Equation 8.1 constant and equal to 1). Such a procedure is iterated for different values of the fixed proportions of the rest of the ingredients. Graphically, this implies moving over the design space in a particular direction, as displayed in Figure 8.2. It is important to notice that a

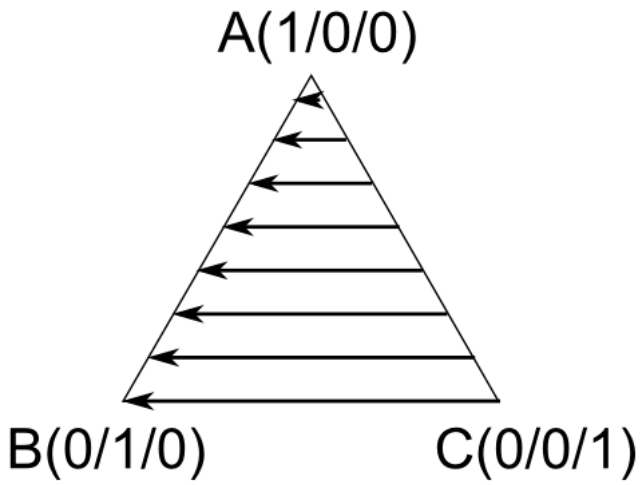

Figure 8.2 Graphical representation of the direction spanned by the pseudo-sample trajectories exploited for retrieving the response surface for a generic ternary mixture design space. Notice that the higher the number of such trajectories, the higher the resolution of the final plot. In this specific case, in every single pseudo-sample matrix, the proportion of A is fixed, while that of both B and C varies

measure of the Scheffé model coefficients for the first-order effects of the ingredients B and C and for their interaction can be derived from the trajectory covering the BC side of the triangle, as will be illustrated in Section 8.4.1[v].

[v] Clearly, this is also valid for the trajectories covering the AB and the AC side of the triangle, not represented in Figure 8.2.

8.3 Datasets

One simulated and three real datasets from mixture designs of experiments will be object of this study.

8.3.1 Simulated data

66 artificial samples (with no replicates) of a ternary mixture homogeneously distributed inside a simplex and a single response variable were simulated according to the following second-order Scheffé model:

$$\begin{gathered} y = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{1,2} x_1 x_2 + \beta_{1,3} x_1 x_3 + \beta_{2,3} x_2 x_3 \\ \beta_1 = 1.89,\ \beta_2 = -1.33,\ \beta_3 = 0.67,\ \beta_{1,2} = -2.89,\ \beta_{1,3} = 0.54,\ \beta_{2,3} = -1.33 \\ x_i \in \{0,\ 1\} \qquad \text{s.t} \sum_{i=1}^{3} x_i = 1 \end{gathered} \tag{8.13}$$

whose reformulation as a Cox model for a reference mixture where $s_1 = s_2 = s_3 = \frac{1}{3}$ can be written as (see Section 8.2.1):

$$\begin{gathered} y = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 + \alpha_{1,2} x_1 x_2 + \alpha_{1,3} x_1 x_3 + \alpha_{2,3} x_2 x_3 + \\ + \alpha_{1,1} x_1^2 + \alpha_{2,2} x_2^2 + \alpha_{3,3} x_3^2 \\ \alpha_0 = 0,\ \alpha_1 = 2,\ \alpha_2 = -2.67,\ \alpha_3 = 0.67,\ \alpha_{1,2} = -1.67,\ \alpha_{1,3} = 0.44, \\ \alpha_{2,3} = 0,\ \alpha_{1,1} = -0.11,\ \alpha_{2,2} = 1.33,\ \alpha_{3,3} = 0 \\ x_i \in \{0,\ 1\} \qquad \text{s.t} \begin{cases} \sum_{i=1}^{3} x_i = 1 \\ \sum_{i=1}^{3} \alpha_i s_i = 0 \\ \sum_{j=1}^{3} c_{i,j} \alpha_{i,j} s_j = 0 \\ s_1 = s_2 = s_3 = \frac{1}{3} \end{cases} \end{gathered} \tag{8.14}$$

According to Equation 8.14, the first constituent is characterised by a positive first-order and a small negative second-order term. Conversely, the second one features a negative first-order and a positive second-order term. The third ingredient exhibits a small positive first-order and no second-order term. Positive interaction terms were generated for both $x_1 x_2$ and $x_1 x_3$, while no interaction was assumed to involve x_2 and x_3. No noise was added after the data simulation.

8.3.2 Tablet data

This dataset was first described in [73]. 10 pharmaceutical tablets resulting from distinct blends of cellulose, lactose and phosphate were prepared to assess the influence of these substances on the release time of the active ingredient of the final manufactured drug. No replicates were performed.

8.3.3 Bubbles data

The bubbles data relate to an experiment also reported in [73]. Different proportions of two dish-washing liquids (DWL1 and DWL2), water and glycerol were combined to produce 24 soap mixtures (21 unique samples and 3 replicates) and determine which composition would have yielded the longest bubble lifetime.

8.3.4 Colorant data

This dataset was described in [77]. 49 blends (46 unique samples and 3 replicates) of different proportions of white (C_{w}), black (C_{b}), violet (C_{v}) and magenta (C_{b}) paints were manufactured to optimise the values of three specific colour responses: lightness (L^*), red-green tone (a^*) and yellow-blue tone (b^*).

8.3.5 Photographic paper data

Different proportions of three generic constituents, A, B and C were mixed to produce a certain type of photographic paper. 13 distinct blend samples were prepared according to an augmented simplex-centroid design of experiments for evaluating the effect of A, B and C on the electrical resistivity of the final product, and possibly minimise it.

8.4 Results

The simulated data were used to highlight how the coefficients of the polynomials in Equations 8.13 and 8.14 can be derived from the pseudo-samples trajectories yielded by a K-PLS model. The other four datasets were exploited for addressing an exploratory comparison among Scheffé polynomial fitting by means of OLS, Cox polynomial fitting by means of PLS and K-PLS[vi].

8.4.1 Simulated data

This section will be focused on demonstrating how the pseudo-sample trajectories can be resorted to for recovering the coefficients of the Scheffé model in Equation 8.13, which the generation scheme outlined in Section 8.3.1 is based on. Figure 8.3 shows the shape of the trajectories spanning the three sides of the ternary mixture space of the simulated dataset. The three lines reproduce the evolution of the values of the response variable (predicted by means of a 3-latent variable second-order K-PLS model) while moving from a vertex (pure blend) to another vertex of a triangle like that in Figure 8.2. Remind that every β_i ($\forall i \in \{1,\ 3\}$) measures the

[vi] No data augmentation is needed for K-PLS.

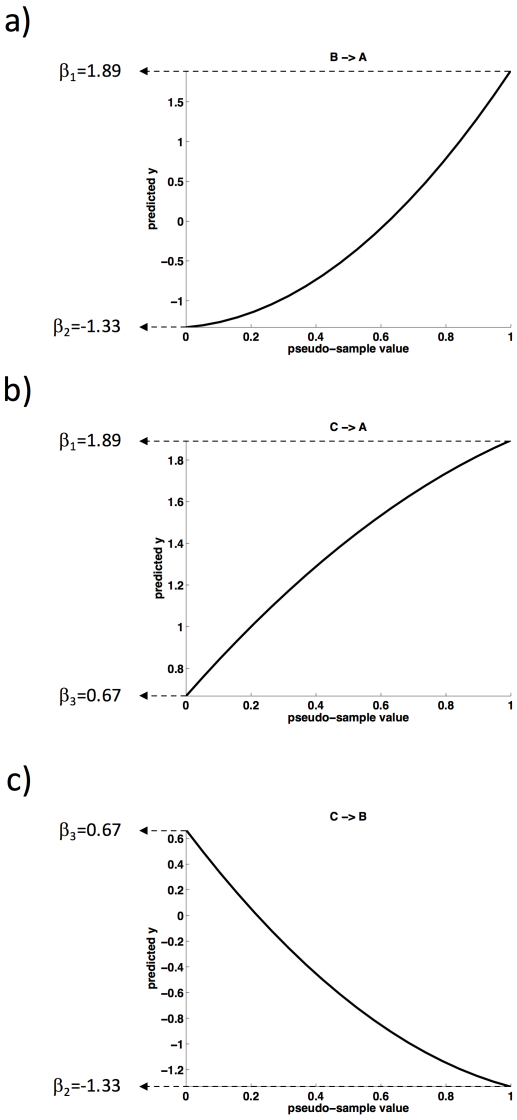

Figure 8.3 Simulated data: pseudo-sample trajectories representing the evolution of the predicted response while moving from a) the pure mixture composed by only the B constituent to the pure mixture composed by only the A constituent, b) the pure mixture composed by only the C constituent to the pure mixture composed by only the A constituent, and c) the pure mixture composed by only the C constituent to the pure mixture composed by only the B constituent

expected y for the pure mixture composed by only the i-th constituent. Therefore, each one of such parameters should match the predicted response at one of the two extremes of the respective pseudo-sample trajectories. As indicated in Figure 8.3, since the data at hand are noiseless, an exact match was here observed for β_1, β_2 and β_3. Analogously, the coefficients for the interaction terms x_1x_2, x_1x_3 and x_2x_3 can be computed as:

$$\begin{aligned}
\beta_{1,2} &= \frac{\hat{y}_{0.5,0.5,0} - 0.5\beta_1 - 0.5\beta_2}{0.25} = \frac{-0.44 - 0.5(1.89) - 0.5(-1.33)}{0.25} = -2.89 \\
\beta_{1,3} &= \frac{\hat{y}_{0.5,0,0.5} - 0.5\beta_1 - 0.5\beta_3}{0.25} = \frac{1.42 - 0.5(1.89) - 0.5(0.67)}{0.25} = 0.54 \\
\beta_{2,3} &= \frac{\hat{y}_{0,0.5,0.5} - 0.5\beta_2 - 0.5\beta_3}{0.25} = \frac{-0.44 - 0.5(-1.33) - 0.5(0.67)}{0.25} = -1.33
\end{aligned} \tag{8.15}$$

where $\hat{y}_{0.5,0.5,0}$, $\hat{y}_{0.5,0,0.5}$ and $\hat{y}_{0,0.5,0.5}$ denote the estimated y-value for the binary blends with composition $x_1 = x_2 = 0.5$, $x_1 = x_3 = 0.5$, $x_2 = x_3 = 0.5$, respectively (the mid-points of the three trajectories in Figures 8.3a, 8.3b and 8.3c).

8.4.2 Tablet data

Second-order Scheffé, Cox and K-PLS models were fitted for the analysis of the tablet dataset[vii]. The number of extracted PLS and K-PLS latent variables was selected by leave-one-out cross-validation[viii]. As Tables 8.1 and 8.2 point out, the three modelling strategies returned comparable performance indices and regression coefficients, respectively. Figure 8.4 displays their corresponding response surface

Table 8.1 Tablet data: R^2, Q^2 and Root Mean Square Error in Cross-Validation (RMSECV) values resulting from second-order Scheffé model fitting by means of OLS, second-order Cox model fitting by means of PLS, and second-order K-PLS

	# LV	R^2	Q^2	RMSECV
Scheffé second-order model (OLS)	-	0.98	0.84	38.86
Cox second-order model (PLS)	5	0.99	0.83	39.69
K-PLS second-order model	5	0.98	0.83	38.86

plots (almost identical). They also enabled a similar interpretation of the effects of the single constituents on the active ingredient release time. High contents of phosphate, moderate contents of cellulose and low contents of lactose clearly led to high values of such property of interest. More concretely, binary mixtures composed by roughly $\frac{2}{3}$ of phosphate and $\frac{1}{3}$ of cellulose are expected to exhibit the

[vii] The use of second-order models was originally suggested in [73]

[viii] Notice that when extreme observations are left out of the original data, every model is forced to predict responses for mixtures which are outside the calibration experimental domain (extrapolation). However, as this is the case for all the approaches under study, a fair comparison is still guaranteed.

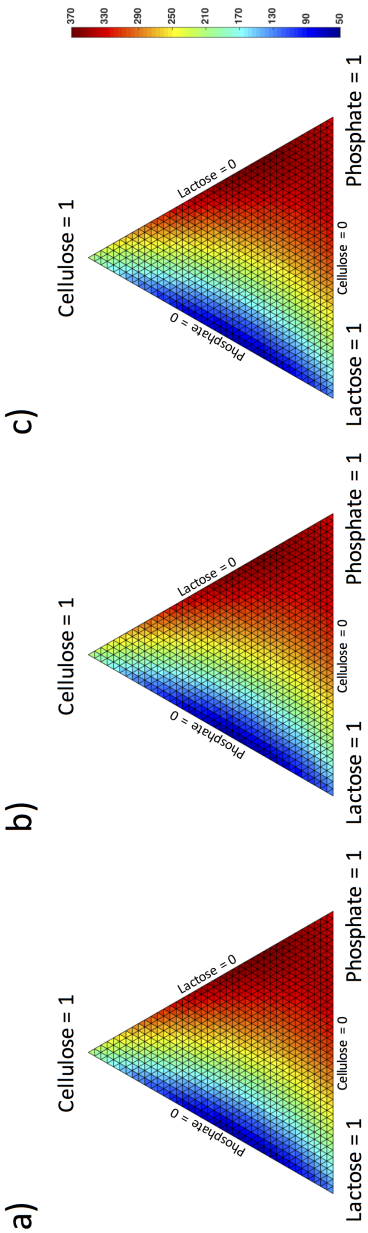

Figure 8.4 Tablet data: response surface plots resulting from a) second-order Scheffé model fitting by means of OLS, b) second-order Cox model fitting by means of PLS, and c) the combination of second-order K-PLS and pseudo-sample trajectories

Table 8.2 Tablet data: Scheffé model coefficients estimated by Scheffé polynomial fitting by means of OLS, Cox polynomial fitting by means of PLS and K-PLS

Model coefficient	Scheffé model fitting by OLS	Cox model fitting by PLS	K-PLS
β_1	198.16	198.10	198.16
β_2	114.06	111.94	114.06
β_3	328.97	326.21	328.97
$\beta_{1,2}$	-403.26	-404.98	-403.26
$\beta_{1,3}$	350.56	347.54	350.56
$\beta_{2,3}$	330.37	323.14	330.37

longest release time. Short release times are instead yielded by blends consisting of e.g. $\frac{2}{3}$ of lactose and $\frac{1}{3}$ of cellulose. Thus, it is quite reasonable to assume the presence of a positive contribution for the interaction phosphate/cellulose and a negative contribution for the interaction lactose/cellulose. As illustrated in Section 8.4.1, one can look at the pseudo-sample trajectories spanning the sides of the triangle in Figure 8.4c for an approximate determination of the Scheffé model first-order and interaction parameters.

8.4.3 Bubbles data

As well as for the previous example, the second-order Scheffé, Cox and K-PLS models adjusted for the bubbles dataset rendered very close R^2, Q^2 and RMSECV values (see Tables 8.3 and 8.4)[vii]. Since this particular mixture problem

Table 8.3 Bubbles data: R^2, Q^2 and RMSECV values resulting from second-order Scheffé model fitting by means of OLS, second-order Cox model fitting by means of PLS, and second-order K-PLS

	# LV	R^2	Q^2	RMSECV
Scheffé second-order model (OLS)	-	0.94	0.81	0.042
Cox second-order model (PLS)	9	0.94	0.81	0.042
K-PLS second-order model	9	0.94	0.81	0.042

embraces up to four constituents, the proportion of one of them has to be fixed to allow the response surfaces to be graphed as in Section 8.4.2. Given that glycerol presented a much more positive effect on the bubble lifetime and a much higher cost than any other ingredient, as also suggested in [73], its relative amount was set at 0.4. The results (virtually indistinguishable) are represented in Figure 8.5. Figure 8.6 shows instead the corresponding trace plots. As one can easily see, although the effect of DWL2 on the response of interest seems to be more positive than that of DWL1 and water, the interaction of these latter is crucial for guaranteeing high bubble lifetimes (i.e. more equilibrated blends of DWL1, DWL2 and

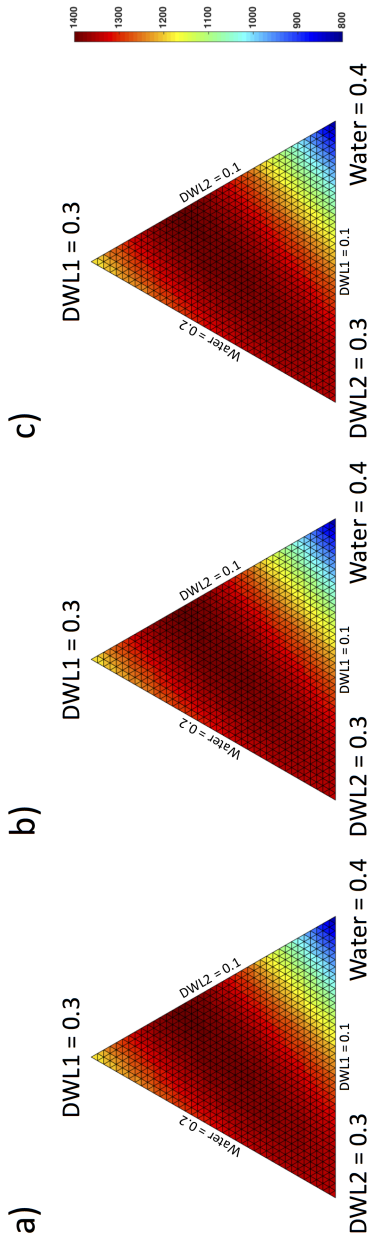

Figure 8.5 Bubbles data: response surface plots resulting from a) second-order Scheffé model fitting by means of OLS, b) second-order Cox model fitting by means of PLS, and c) the combination of second-order K-PLS and pseudo-sample trajectories

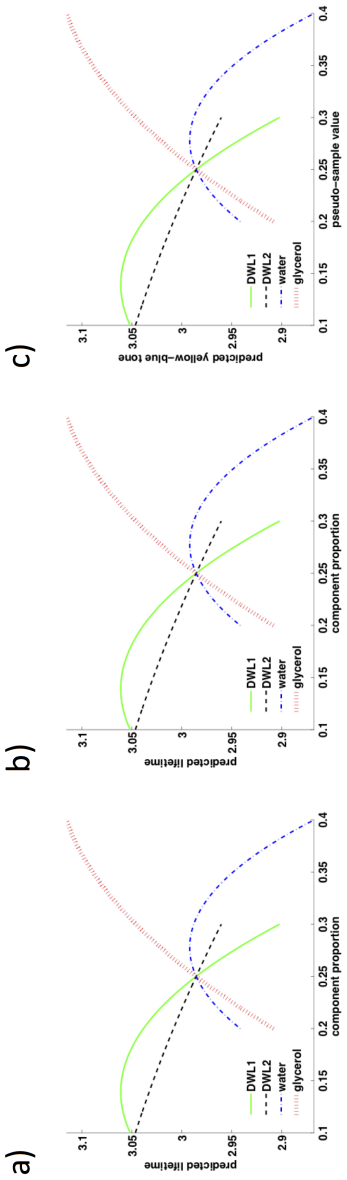

Figure 8.6 Bubbles data: trace plots representing the evolution of the predicted lifetime while varying the proportion of the 4 ingredients of the blend (DWL1, DWL2, water and glycerol) and resulting from a) second-order Scheffé model fitting by means of OLS, b) second-order Cox model fitting by means of PLS, and c) the combination of second-order K-PLS and pseudo-sample trajectories. Here, $s_1 = s_2 = s_3 = s_4 = \frac{1}{4}$

Table 8.4 Bubbles data: Scheffé model coefficients estimated by Scheffé polynomial fitting by means of OLS, Cox polynomial fitting by means of PLS and K-PLS

Model coefficient	Scheffé model fitting by OLS	Cox model fitting by PLS	K-PLS
β_1	-1.49	-1.49	-1.49
β_2	2.35	2.35	2.35
β_3	-1.35	-1.35	-1.35
β_4	2.11	2.11	2.11
$\beta_{1,2}$	2.08	2.08	2.08
$\beta_{1,3}$	14.55	14.55	14.55
$\beta_{1,4}$	7.51	7.51	7.51
$\beta_{2,3}$	6.70	6.70	6.70
$\beta_{2,4}$	2.63	2.63	2.63
$\beta_{3,4}$	7.82	7.82	7.82

water would feature more durable bubbles).
The pseudo-sample trajectories spanning the sides of the triangle in Figure 8.5c cannot be directly resorted to for the estimation of the related Scheffé model coefficients in this situation owing to the fact that the design space of the bubbles data is just a portion of a whole tetrahedron, and then they do not reflect the evolution of the predicted response while moving from a pure mixture to another. On the other hand, if these trajectories are constructed so that they exactly overlap the entire edges of this hypothetical tetrahedron, the methodology proposed in Section 8.4.1 for the retrieval of the first-order and binary interaction parameters is still valid (assuming that any effect involving the two constituents of the concerned binary mixture do not vary outside the actual data space) [69].
Until now, it has been proven that combining K-PLS and pseudo-sample trajectories might permit to achieve outcomes in strict concordance with those provided by Scheffé model fitting through OLS and Cox model fitting through PLS when the investigated mixture data are not affected by really strong non-linear relationships (i.e. second-order relationships). But what if adjusting more complex models is required to account for more severe non-linearities? In the next sections two examples in which this is actually the case will be presented.

8.4.4 Colorant data

When the colorant dataset was dealt with, second-, third- and fourth-order Scheffé, Cox and K-PLS models were fitted in order to additionally assess the effect of their complexity on the final outcomes[ix]. Table 8.5 lists their main performance indices.
It can be said that different approaches usually required a different complexity to achieve the minimum RMSECV, but, overall, their performance was found to be rather similar also in this case.

[ix] By analogy with OLS, single PLS and K-PLS models were calibrated for every response variable.

Table 8.5 Colorant data: R^2, Q^2 and RMSECV values for the three concerned properties of interest (L^*, a^* and b^*) resulting from second/third/fourth-order Scheffé polynomial fitting by means of OLS (red), second/third/fourth-order Cox polynomial fitting by means of PLS (cyan), and second/third/fourth-order K-PLS (green). The best models in terms of RMSECV are highlighted in bold

	L^*				a^*				b^*			
	# LVs	R^2	Q^2	RMSECV	LVs	R^2	Q^2	RMSECV	LVs	R^2	Q^2	RMSECV
Scheffé second-order model (OLS)	-	0.97	0.92	6.29	**-**	**0.96**	**0.93**	**3.74**	**-**	**0.92**	**0.87**	**4.27**
Cox second-order model (PLS)	5	0.97	0.92	6.27	5	0.96	0.93	3.71	5	0.92	0.87	4.24
K-PLS second-order model	3	0.97	0.95	4.94	6	0.96	0.93	3.72	**6**	**0.92**	**0.86**	**4.27**
Scheffé third-order model (OLS)	**-**	**0.99**	**0.99**	**1.35**	-	0.98	0.85	5.62	-	0.95	0.50	8.39
Cox third-order model (PLS)	**12**	**0.99**	**0.99**	**1.33**	8	0.98	0.88	4.93	3	0.84	0.74	6.12
K-PLS third-order model	**14**	**0.99**	**0.99**	**1.25**	**9**	**0.98**	**0.94**	**3.23**	10	0.95	0.86	4.41
Scheffé fourth-order model (OLS)	-	0.99	0.90	7.15	-	0.99	0.76	7.01	-	0.99	0.21	10.56
Cox fourth-order model (PLS)	12	0.99	0.96	4.61	**12**	**0.99**	**0.95**	**3.35**	**13**	**0.99**	**0.92**	**3.45**
K-PLS fourth-order model	6	0.98	0.97	3.54	9	0.98	0.93	3.77	10	0.95	0.82	5.01

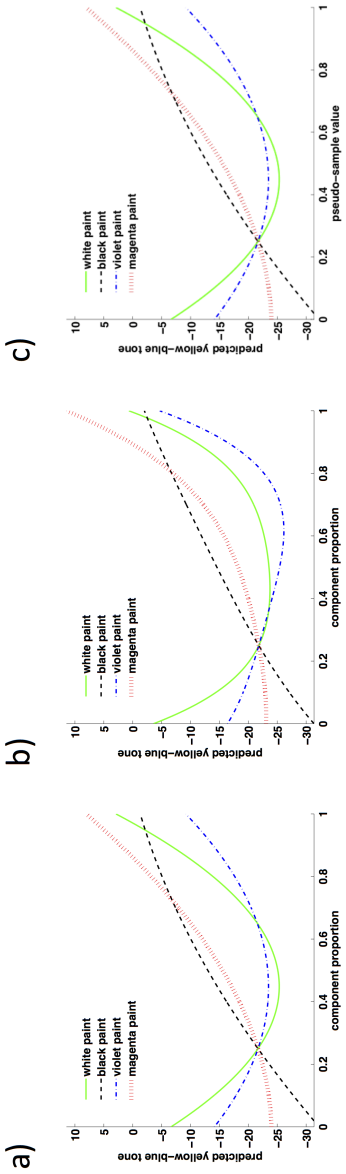

Figure 8.7 Colorant data: trace plots representing the evolution of the predicted yellow-blue tone (b^*) while varying the proportion of the 4 ingredients of the blend (white, black, violet and magenta paints) and resulting from a) second-order Scheffé model fitting by means of OLS, b) fourth-order Cox model fitting by means of PLS, and c) the combination of second-order K-PLS and pseudo-sample trajectories. Here, $s_1 = s_2 = s_3 = s_4 = \frac{1}{4}$

For the sake of interpretation, as an illustration, the trace plots resulting from the best Scheffé, Cox and K-PLS models built for the prediction of the yellow-blue tone (b^*) are displayed in Figure 8.7. They are almost in perfect agreement and only slight variations with respect to the outcomes obtained by Alman and Pfeifer in [77] were observed (the same goes for those derived for both L^* and a^* - not shown). Concretely, all the constituents exhibited a positive effect on b^*[x].

8.4.5 Photographic paper data

Second-order cross-validated Scheffé and Cox models and a cross-validated fourth-order K-PLS model were adjusted for the photographic paper dataset[xi]. As one can easily see, K-PLS clearly outmatched, in this circumstance, the other two strategies in terms of Q^2 and RMSECV (see Table 8.6). The three models led to rather similar response surfaces (see Figure 8.8) even though, owing to the different complexity degree of the K-PLS one, certain dissimilarities are observable from Figure 8.9. Nevertheless, a common explanation of how the distinct ingredients affect the values of the y-variable can be given: the ideal photographic paper (minimum resistivity) should be manufactured by blending relatively low quantities of A and B and a relatively high quantity of C[x].

8.5 Conclusions

In this chapter a novel approach for the analysis of data originating from mixture designs of experiments and based on the combination of K-PLS and pseudo-sample trajectories was proposed. Two interesting points arose from the discussed examples, corroborating the conclusions drawn in the previous chapters:

Table 8.6 Photographic paper data: R^2, Q^2 and RMSECV values resulting from second-order Scheffé model fitting by means of OLS, second-order Cox model fitting by means of PLS, and fourth-order K-PLS

	# LV	R^2	Q^2	RMSECV
Scheffé second-order model (OLS)	-	0.94	0.42	0.84
Cox second-order model (PLS)	3	0.94	0.66	0.64
K-PLS fourth-order model	4	0.95	0.81	0.46

[x] As for the colorant and the photographic paper dataset the best outcomes were returned by Scheffé, Cox and K-PLS models with diverse complexity, the comparison of their regression coefficients was skipped.

[xi] The third-order Scheffé and the third- and fourth-order Cox models returned negative Q^2 values. Moreover, due to the low number of mixture samples concerned, not enough degrees of freedom were available for the estimation of the coefficients of the fourth-order Scheffé polynomial.

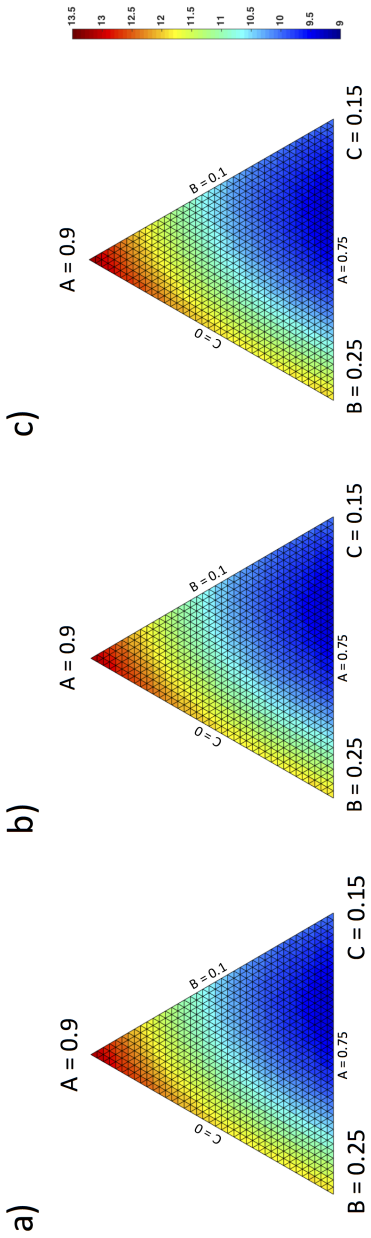

Figure 8.8 Photographic paper data: response surface plots resulting from a) second-order Scheffé model fitting by means of OLS, b) second-order Cox model fitting by means of PLS, and c) the combination of fourth-order K-PLS and pseudo-sample trajectories

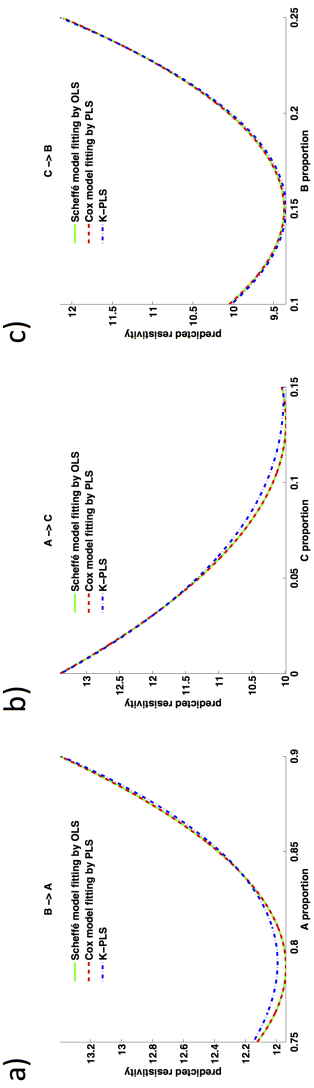

Figure 8.9 Photographic paper data: evolution of the resistivity values predicted through Scheffé model fitting by means of OLS (green solid lines), Cox model fitting by means of PLS (dashed red lines), and the combination of K-PLS and pseudo-sample trajectories (blue dashed-dotted lines) while moving from a) the pure mixture composed by only the B constituent to the pure mixture composed by only the A constituent, b) the pure mixture composed by only the A constituent to the pure mixture composed by only the C constituent, and c) the pure mixture composed by only the C constituent to the pure mixture composed by only the B constituent

- if the considered mixture data were not affected by severe non-linearities and/or featured a sufficiently high number of observations, it was possible to resort to K-PLS and pseudo-sample trajectories, obtaining very similar results to classical Scheffé model fitting by means of OLS and Cox model fitting by means of PLS;
- on the contrary, when more complex and relatively small data structures had to be analysed, K-PLS proved to be a valid alternative to both the aforementioned methodologies, and the pseudo-sample trajectories enabled a reliable interpretation of the influence of changing the proportion of the different ingredients of the blend on its properties of interest.

Furthermore, based on the definition of the Scheffé model coefficients, a procedure for recovering them from the trend of such pseudo-sample trajectories was derived and validated via a simulated case study.

Part III

On the selection of the number of factors in PCA by permutation testing

Chapter 9

A novel permutation test-based approach for PCA component selection

In this chapter an extensive guideline on how to accomplish the selection of PCA components by permutation testing is provided through the description of a novel and efficient algorithm developed to this end.

Part of the content of this chapter has been included in:

1. Vitale, R., Westerhuis, J., Næs, T., Smilde, A., de Noord, O. & Ferrer, A. Selecting the number of factors in Principal Component Analysis by permutation testing - Numerical and practical aspects. *J. Chemometr.*, In press.

9.1 Introduction

In Section 12.1.2 it was highlighted that a very critical point when deriving a PCA decomposition of a certain dataset is how to properly select the number of principal components to compute, A. First of all one should notice that, as stated in [65, 78, 79], this assessment connotes an ill-posed problem if formulated without taking into account for which objective PCA is resorted to. In [79] Camacho and Ferrer differentiated three different application scenarios: i) when the interest is on the *observable* or *original* variables; ii) when the interest is on the *principal components*; iii) when the interest is on the distributions of the principal components and residuals. i) refers to situations in which the dimensionality of the PCA subspace has to be determined so that the model-based reconstruction of the original variables is the most accurate possible (e.g. for compression or missing value imputation). ii) mainly relates to data exploration, which normally implies the extraction of all the principal components that can be safely interpreted because they are sufficiently different from noise. iii) basically concerns statistical process monitoring, where the distributions of the principal components and residuals, calculated from a set of data collected under Normal Operating Conditions (NOC), are utilised to assess whether such NOC are maintained over time or a fault is occurring. Here, the main focus will be on ii). However, in all these circumstances, if no *a priori* knowledge about the investigated systems is available, A has to be empirically retrieved.

9.1.1 Strategies for principal component selection

During the last decades many approaches for principal component selection have been developed, which can be classified in three distinct categories: *ad hoc* rules, statistical tests and computational criteria [80]. *Ad hoc* rules (like Kaiser's eigenvalue greater than 1 rule [81], Velicer's minimum average partial rule [82] and Cattell's scree test [83]) and statistical tests (like Bartlett's chi-square test [84] and Tracy-Widom statistic-based test [85]) generally show particular drawbacks: the former often constitute case-specific strategies, not easily generalisable for handling data structures of various nature, while the latter are based on distributional assumptions, which are rarely met in modern analytical contexts (e.g. the Tracy-Widom statistic-based test requires all the measured variables to be independently and identically normally or at least symmetrically distributed). On the other hand, computational criteria are completely data-driven and distribution-free. Therefore, they can be regarded as feasible options when *ad hoc* rules and statistical tests cannot be applied (for instance when the considered datasets do not fulfill particular mathematical properties), even if they might sometimes lead to an excessive time and memory consumption.
Computational criteria encompass both cross-validation and permutation test-based techniques (like Horn's parallel analysis [86] and a recent data dimension-

ality detection methodology proposed by Dray in [87]). Although cross-validation is probably the most widespread principal component selection approach, its application is not recommended when the objective of the study is discriminating relevant from noisy factors [79]. In fact, as it permits to determine the dimensionality of the PCA subspace by minimising the prediction error between the initial data and their PCA estimates, it is evidently better-suited for the applications covered by the aforementioned scenario i). On the contrary, when the systematic and non-systematic sources of variation in the data have to be differentiated and/or stable loadings and residuals distributions are desired, the focus moves from the original variables to the principal components. In similar contingencies the employment of cross-validation may not be adequate: procedures aimed at determining their statistical significance or at minimising the Overall Type II risk when process monitoring is concerned may therefore be required.
Permutation test-based techniques rely on the comparison of some attributes of the analysed data matrices with those of arrays characterised by uncorrelated variables. These attributes are conventionally: i) the singular values or the eigenvalues (the square of the singular values) or ii) functions of the eigenvalues (e.g. the difference or the ratio between consecutive eigenvalues). Based on this, it is clear that these methods directly concentrate on the identification of structured interpretable components and might represent appropriate alternatives to cross-validation when PCA is exploited for such an exploratory purpose [87–93].
For all these reasons, in this chapter, an extensive guideline on how to select the number of PCA factors by permutation tests is provided. Concretely, a novel algorithm designed to this end and originated from the preliminary proposal outlined in [94] is presented. It will be compared to other permutation test-based strategies (i.e. Horn's parallel analysis and Dray's approach, described in Appendices 14.2.1 and 14.2.2, respectively), which will permit to point out some of their limitations that were only partly spotted before in the scientific literature. Solutions for overcoming these limitations will be additionally reported. The theoretical and practical aspects of this algorithm will be examined for a better understanding of its pros over its primary version and possible adjustments for optimising the efficiency of its computational procedure (in terms of time and memory consumption) will also be discussed.

9.2 Methodology

A new algorithm for PCA component selection by permutation testing is here introduced. Let $\mathbf{X}$ be a centred data matrix of N rows and J columns with rank $Q = \min\{N-1, J\}$. This novel computational procedure rests on the estimation of the statistical significance of the eigenvalues of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ (λ) by their comparison with those resulting from covariance structures generated by permuting the order of the entries of $\mathbf{X}$ within each one of its columns independently. It comprises the following 10 steps grouped in three consecutive phases (see also Figure 9.1):

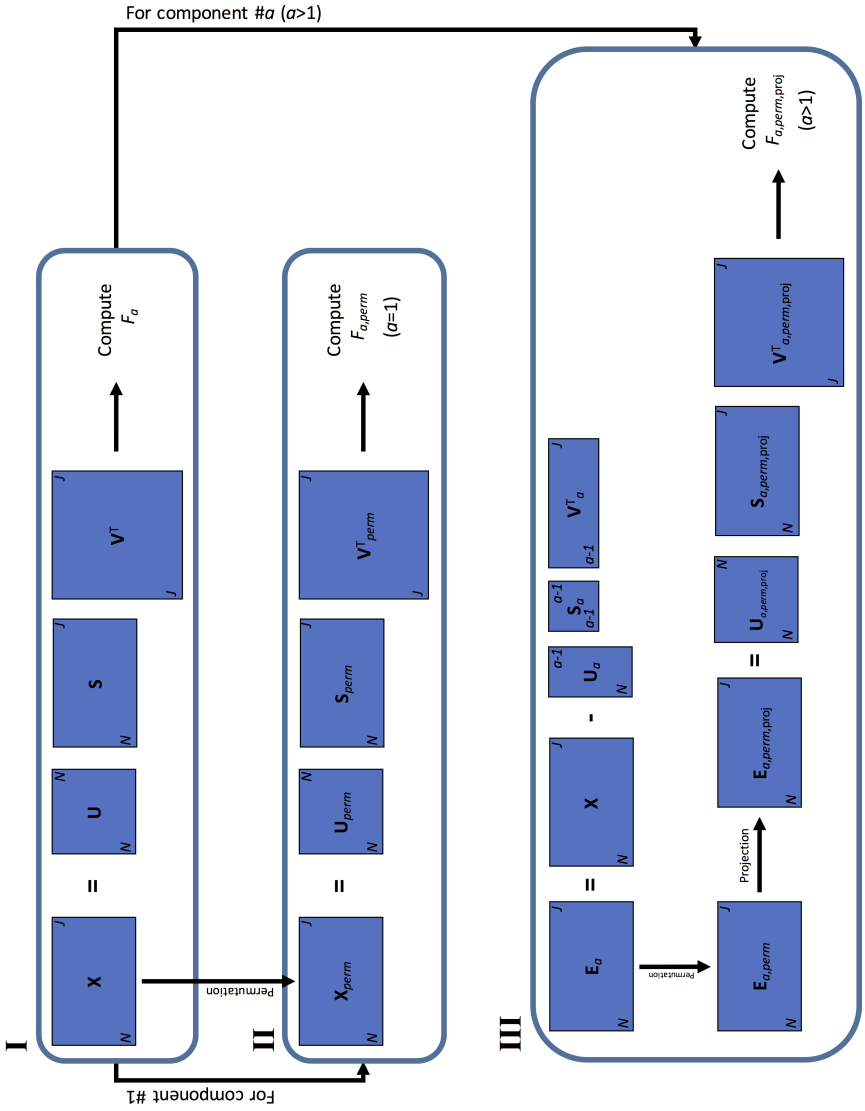

Figure 9.1 Schematic representation of the permutation test-based algorithmic procedure proposed in this chapter. A PCA component is considered statistically significant if its respective F_a value is higher than e.g. the 99$^{\text{th}}$ percentile of the associated *null*-distribution

- Phase I - Singular Value Decomposition (SVD) of $\mathbf{X}$:

  1. Perform full-rank SVD on $\mathbf{X}$:

     $$\mathbf{X} = \mathbf{USV}^{\mathrm{T}} = \mathbf{TP}^{\mathrm{T}} \tag{9.1}$$

     where $\mathbf{U}$ $(N \times N)$ and $\mathbf{V}$ $(J \times J)$ contain the left and right singular vectors of $\mathbf{X}$, respectively, and $\mathbf{S}$ $(N \times J)$ is a rectangular diagonal array whose non-zero diagonal elements are its singular values $(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_Q})$;

  2. Compute for each a-th calculated component the ratio:

     $$F_a = \frac{\lambda_a}{\sum_{q=a}^{Q} \lambda_q} \tag{9.2}$$

     where λ_a corresponds to the a-th eigenvalue obtained after the decomposition of $\mathbf{X}$.
     F_a is used for testing the statistical significance of the single factors. It equals the ratio between the amount of variation explained by the a-th component and the total amount of variation captured by the last $Q - (a - 1)$ components;

- Phase II - Test for the first component:

  3. For $a = 1$, randomly and independently permute the order of the entries within every column of $\mathbf{X}$ constructing a new matrix $\mathbf{X}_{perm}$, featuring uncorrelated variables;

  4. Apply full-rank SVD to $\mathbf{X}_{perm}$ and calculate the ratio:

     $$F_{1,perm} = \frac{\lambda_{1,perm}}{\sum_{q=1}^{Q} \lambda_{q,perm}} \tag{9.3}$$

     where $\lambda_{1,perm}$ denotes the first eigenvalue obtained after the decomposition of $\mathbf{X}_{perm}$. Note that the sum of squares of $\mathbf{X}$ and $\mathbf{X}_{perm}$ is exactly the same, despite the permutations;

  5. Iterate[i] step 3 and 4 to generate a *null*-distribution for $F_{1,perm}$. If F_1 is found to be higher than e.g. the 99^{th} percentile[i] of the *null*-distribution of $F_{1,perm}$, the first component is considered statistically significant.

[i] Both the total number of iterations and the confidence level are user-defined parameters and should be selected depending on the signal-to-noise ratio of the data under study and the degree of accuracy required for the determination of the number of their underlying principal components.

- Phase III - Test for the a-th component ($a > 1$):

  6. For $a > 1$, calculate the residual matrix:

$$\mathbf{E}_a = \mathbf{X} - \sum_{q=1}^{a-1} \mathbf{u}_q \sqrt{\lambda_q} \mathbf{v}_q^{\mathrm{T}} = \mathbf{X} - \sum_{q=1}^{a-1} \mathbf{t}_q \mathbf{p}_q^{\mathrm{T}} \tag{9.4}$$

  where $\mathbf{u}_q$, $\mathbf{v}_q$, $\mathbf{t}_q$ and $\mathbf{p}_q$ are the q-th column vectors of $\mathbf{U}$, $\mathbf{V}$, $\mathbf{T}$ and $\mathbf{P}$ (see Equation 9.1), respectively[ii]. Note that after each deflation round $\mathbf{E}_a$ has rank $Q - (a - 1)$;

  7. Randomly and independently permute the order of the entries within each column of $\mathbf{E}_a$ constructing a new matrix $\mathbf{E}_{a,perm}$. Unlike $\mathbf{E}_a$, $\mathbf{E}_{a,perm}$ has rank Q, but their total sums of squares are the same;

  8. Calculate the projection of $\mathbf{E}_{a,perm}$ on a subspace of dimensionality $Q-(a-1)$, $\mathbf{E}_{a,perm,\mathrm{proj}}$. The way to carry out this projection represents the main novelty of this study and will be discussed in the next section;

  9. Perform full-rank SVD on $\mathbf{E}_{a,perm,\mathrm{proj}}$ and retain the ratio:

$$F_{a,perm,\mathrm{proj}} = \frac{\lambda_{1,perm,\mathrm{proj}}}{\sum_{q=1}^{Q-(a-1)} \lambda_{q,perm,\mathrm{proj}}} \tag{9.5}$$

  where $\lambda_{1,perm,\mathrm{proj}}$ is the first eigenvalue obtained after the decomposition of $\mathbf{E}_{a,perm,\mathrm{proj}}$;

  10. Iterate[i] step 6, 7 and 8 to generate a *null*-distribution for $F_{a,perm,\mathrm{proj}}$. If F_a is found to be higher than e.g. the 99^{th} percentile[i] of the associated *null*-distribution, the a-th component is considered statistically significant.

Computations are stopped as soon as the first non-significant component is detected.

9.3 Theoretical and practical aspects of the algorithm

Four particular aspects, which constitute the core of the algorithm, are now elucidated from both a theoretical and practical perspective: i) Why are the data permuted column-wise? ii) Why does $\mathbf{X}$ need to be sequentially deflated? iii) Is the projection of $\mathbf{E}_{a,perm}$ necessary? iv) What is the rationale behind the relative index F_a?

[ii] According to this notation a hypothetical $\mathbf{E}_1$ would correspond to $\mathbf{X}$.

For the sake of a comprehensive assessment of the specific implications of how the calculations are performed, all the tests reported in this section were run for all the extractable components, thus not resorting to the aforementioned stopping criterion.

9.3.1 Permutations

In both Phases II and III of the computational procedure, $\mathbf{X}$ and $\mathbf{E}_a$ are permuted so that the order of the entries within each one of their columns is independently randomised. This breaks their underlying covariance structure, whereas the mean value and the standard deviation of the measured variables are maintained. Variances are then preserved, but the intrinsic mutual relationships among descriptors are cluttered.

9.3.2 Deflation

When SVD is applied on a dataset whose element order was permuted such that all correlation among the measured variables is lost, its total variation will be more or less uniformly distributed across all its extractable factors[iii]. This can lead to overlooking the actual statistical significance of some eigenvalues of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ if they do not account for a substantially high amount of variation, which is exactly what happens with Horn's parallel analysis [95], as will be shown in Section 9.4.1. Testing this significance with consecutive deflation steps allows this limitation to be overcome. This is illustrated in the following example.
Say for instance that $\mathbf{X}$ has rank 10 and contains 80% of systematic variation in 3 components (60%, 15% and 5%, respectively) and 20% of noise with a total sum of squares of 100. Therefore, $\lambda_1 = 60$, $\lambda_2 = 15$ and $\lambda_3 = 5$. The remaining 20% of the variation of $\mathbf{X}$ is roughly uniformly distributed over the other 7 eigenvalues ($\lambda_q \simeq 2.8 \ \forall q = 4, \ldots, 10$). Permuting $\mathbf{X}$ permits to generate a new matrix $\mathbf{X}_{perm}$ with the same sum of squares. However, as no correlation among the measured variables is left, the whole amount of variation is distributed across the whole set of 10 eigenvalues ($\lambda_{q,perm} \simeq 10 \ \forall q = 1, \ldots, 10$). As shown in Figure 9.2, comparing λ_3 with its *null*-distribution will not lead to detect it as statistically significant (λ_3 is clearly smaller than the 99^{th} percentile of the corresponding *null*-distribution). On the other hand, if the first two components are used to deflate $\mathbf{X}$, the resulting $\mathbf{E}_3$ matrix will be characterised by a sum of squares of 25 and a rank of 8. The first eigenvalue of $\mathbf{E}_{3,perm}$ will be then around $\frac{25}{10} \simeq 2.5$ (shuffling $\mathbf{E}_3$ makes $\mathbf{E}_{3,perm}$ have again rank 10). In this case, the third factor of $\mathbf{X}$ will be correctly identified as statistically significant (λ_3 is now larger than the 99^{th} percentile of the corresponding *null*-distribution). But, what happens with λ_8, λ_9

[iii] Theoretically, the eigenvalues associated to the single factors should be identical provided that the sum of squares of all the variables is the same. However, chance correlations generate their typical smooth descending trend observable in e.g. Figure 9.2.

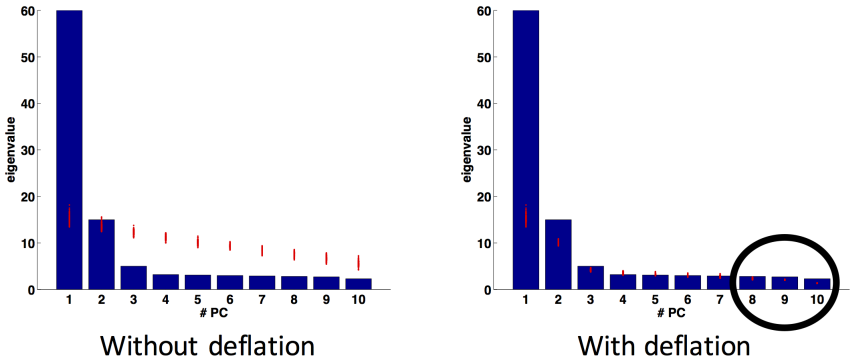

Figure 9.2 Eigenvalues (blue bars) of a simulated centred data matrix (100 × 10) containing 80% of systematic variation in 3 components (60%, 15% and 5%, respectively) and 20% of noise with a total sum of squares of 100, and their empirically estimated *null*-distributions (red dots) obtained by either not deflating (left) or deflating (right) the original array during the execution of the permutation test. Each red dot corresponds to the eigenvalue obtained for the respective component after one of the 300 performed permutation rounds

and λ_{10} (see black circle in Figure 9.2)? Are they statistically significant even if λ_4 to λ_7 are not? The root cause of this strange inconsistency will be clarified in the next section.

9.3.3 Projection

As a consequence of the permutations, $\mathbf{E}_a$ and $\mathbf{E}_{a,perm}$ have always different rank - $Q-(a-1)$ and Q, respectively, after each step of deflation for $a > 1$ - but equal sum of squares. However, this sum of squares is distributed over $Q-(a-1)$ non-zero eigenvalues in the former array and over Q non-zero eigenvalues in the latter one. Hence, the expected values of λ_a will be on average higher than those of $\lambda_{a,perm}$ as a increases. The projection of $\mathbf{E}_{a,perm}$ on a hyperplane of dimensionality $Q-(a-1)$ can correct for this effect. In the approach proposed in [94] this projection is executed both row-wise and column-wise as:

$$\mathbf{E}_{a,perm,\text{proj}} = (\mathbf{I}_N - \sum_{q=1}^{a-1} \mathbf{u}_q \mathbf{u}_q^{\mathrm{T}}) \mathbf{E}_{a,perm} (\mathbf{I}_J - \sum_{q=1}^{a-1} \mathbf{v}_q \mathbf{v}_q^{\mathrm{T}}) \tag{9.6}$$

where $\mathbf{I}_N$ is an identity matrix of dimensions $N \times N$ and $\mathbf{I}_J$ is an identity matrix of dimensions $J \times J$. Equation 7 (referred to as P1 from now on) guarantees that both the row and column space of $\mathbf{E}_{a,perm,\text{proj}}$ are identical to those of the original residuals, $\mathbf{E}_a$. Therefore, in [94] it was regarded as the most natural and intuitive way to project $\mathbf{E}_{a,perm}$. P1 proved to be a feasible option when small

sets of data were dealt with, but if N and/or J are/is very large, the calculation of the inner-product arrays $\sum_{q=1}^{a-1} \mathbf{u}_q \mathbf{u}_q^{\mathrm{T}}$ $(N \times N)$ and/or $\sum_{q=1}^{a-1} \mathbf{v}_q \mathbf{v}_q^{\mathrm{T}}$ $(J \times J)$ can be rather expensive in computational terms. An alternative strategy (referred to as P2[iv]) could be projecting $\mathbf{E}_{a,perm}$ onto the hyperplane orthogonal to the first $a-1$ components of $\mathbf{X}$ using their either left or right singular vectors:

$$\mathbf{E}_{a,perm,\mathrm{proj}} = (\mathbf{I}_N - \sum_{q=1}^{a-1} \mathbf{u}_q \mathbf{u}_q^{\mathrm{T}}) \mathbf{E}_{a,perm} \tag{9.7}$$

P2 would be significantly faster from a computational point of view, but would allow only the row or the column spaces of $\mathbf{E}_{a,perm,\mathrm{proj}}$ and $\mathbf{E}_a$ to be the same. P1 and P2 lead to different $\mathbf{E}_{a,perm,\mathrm{proj}}$ matrices (a single row-wise or column-wise projection as in P2 reduces the sum of squares of $\mathbf{E}_a$ less than a double projection as in P1), which nevertheless share the same rank, Q. This is a sufficient condition to compare them in this particular application scenario and check whether one of the two may ensure a higher testing power in a controlled situation, i.e. in the absence of significant components (that is exploring the *null*-hypothesis, H_0, of the concerned permutation test). Specifically, for each projection approach:

1. 300 matrices of size 51×200 containing random values drawn from the standard normal distribution were simulated;

2. after preprocessing, the algorithm reported in Section 9.2 was run on each one of these matrices to determine the statistical significance of all their 50 extractable components. 300 permutation rounds per matrix were performed;

3. once every test was completed, a single p-value per component was derived as the ratio between the number of $F_{1,perm}$ (if $a = 1$) or $F_{a,perm,\mathrm{proj}}$ (if $a > 1$) found to be higher than the corresponding F_a and the total number of permutations;

4. the p-values associated to each component were finally averaged over the 300 simulated matrices.

Figure 9.3 displays the outcomes of this assessment (left and central subplots). Mind that, as H_0 is true (none of the principal components is statistically significant), mean p-values of approximately 0.5 are expected for all the factors. Higher p-values would point out a lack of sensitivity in the principal component selection. Lower p-values would imply the test is prone to identify noisy components as statistically significant. Here, both P1 and P2 exhibited a very similar perfor-

[iv] Formulas 9.7 and 9.8 relate to the case in which $N < J$ and define a row-wise projection. If $N > J$, a column-wise projection can be performed resorting to the column vectors of $\mathbf{V}/\mathbf{V}_{\mathbf{E}_{a,perm}}$ instead of those of $\mathbf{U}/\mathbf{U}_{\mathbf{E}_{a,perm}}$, thus preventing excessive time and memory consumption.

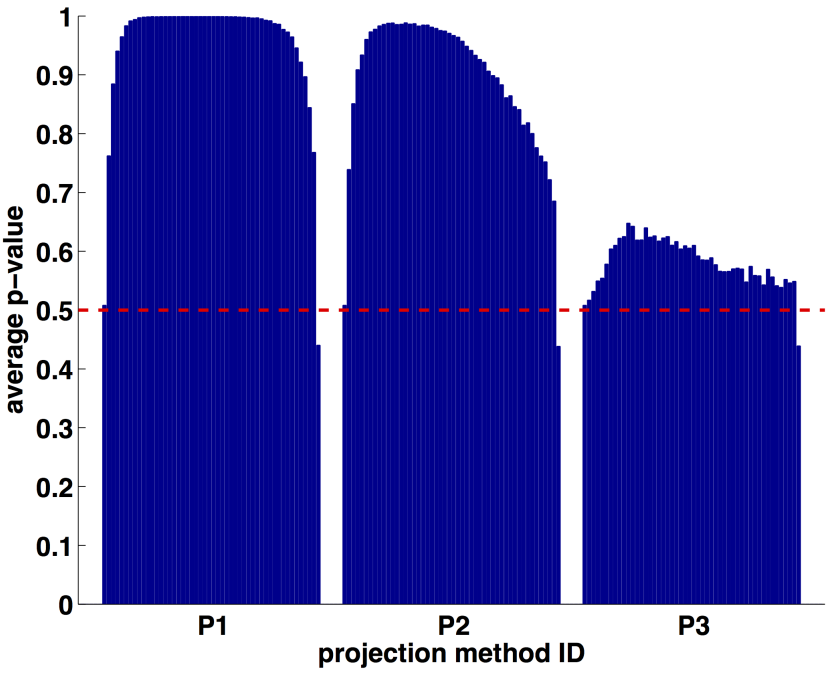

Figure 9.3 Mean p-values obtained for the 50 components extractable, after preprocessing, from 300 simulated random matrices of dimensions 51×200 by exploiting the 3 different projection strategies under comparison. Each bar quantifies the ratio, averaged over the 300 simulations, between the number of $F_{1,perm}$ (if $a = 1$) or $F_{a,perm,\text{proj}}$ (if $a > 1$) found to be higher than the corresponding F_a and the total number of permutation rounds (300 per matrix). The horizontal red dotted line is drawn at $p-\text{value} = 0.5$

mance, generally leading to overestimated p-values. Being F_a invariant no matter how the projection is attained, this discrepancy uniquely depends on the fact that the $F_{a,perm,\text{proj}}$ values (for $a > 1$) of the various empirical *null*-distributions are larger than expected when quantified by P1 and P2. This is also confirmed by Figure 9.4 (left and central subplots). Such an overestimation is probably a consequence of the fact that $\mathbf{E}_{a,perm}$ is projected onto the subspace spanned by $\mathbf{E}_a$ that still describes part of the systematic structure of the handled data (this structure is spurious but anyway present also in random matrices)[v], which might generate chance covariance in $\mathbf{E}_{a,perm,\text{proj}}$ and then a substantial increase in the corresponding $F_{a,perm,\text{proj}}$ ratios. This should be even more evident when structured data

[v]Say one draws a shape (a principal component) in the sand (the original data space) and removes (deflates) all the grains inside its borders. The remainder of the sand (the residual space) is a *negative* of the shape and thus keeps memory of its structure.

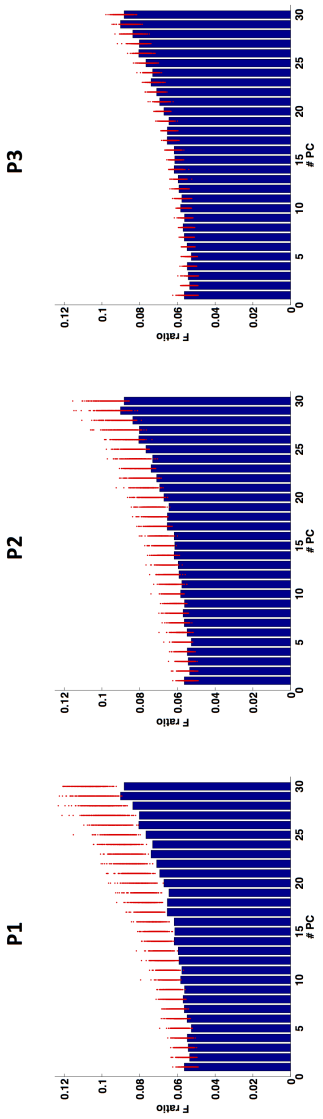

Figure 9.4 F_a ratios (blue bars) and related empirical *null*-distributions (red dots) associated to the first 30 components of a simulated 51×200 random matrix and resulting from P1, P2 and P3. Each red dot corresponds to the $F_{1,perm}$ (if $a = 1$) or the $F_{a,perm,\mathrm{proj}}$ (if $a > 1$) estimate obtained for the respective factor after one of the 300 performed permutation rounds

are dealt with. In order to overcome this issue, P3 is proposed. By P3, $\mathbf{E}_{a,perm}$ is projected onto the hyperplane orthogonal to its first $a-1$ components[iv]:

$$\mathbf{E}_{a,perm,\text{proj}} = (\mathbf{I}_N - \sum_{q=1}^{a-1} \mathbf{u}_{q,\mathbf{E}_{a,perm}} \mathbf{u}_{q,\mathbf{E}_{a,perm}}^{\mathrm{T}}) \mathbf{E}_{a,perm} \tag{9.8}$$

where $\mathbf{E}_{a,perm} = \mathbf{U}_{\mathbf{E}_{a,perm}} \mathbf{S}_{\mathbf{E}_{a,perm}} \mathbf{V}_{\mathbf{E}_{a,perm}}^{\mathrm{T}}$, and being $\mathbf{u}_{q,\mathbf{E}_{a,perm}}$ the q-th column vector of $\mathbf{U}_{\mathbf{E}_{a,perm}}$. At each permutation round, $\mathbf{E}_{a,perm}$ is subjected to SVD and a new subspace is estimated for the projection from random residuals to limit the effect of the chance covariance induced by P1 and P2. As no notable differences were observed between the performance of P1 and P2, P3 was implemented so that the projection is carried out either row-wise or column-wise[vi]. It is also worth noting that P3 yields the largest loss of data variation. Figures 9.3 and 9.4 also show the results of the previous study obtained when P3 was used (right subplots): as expected it returned p-values only slightly higher than 0.5 and clearly lower than those resulting from P1 and P2. In the light of all that, only P1 and P3 will be employed for comparison in the further case studies illustrated in this chapter.
The projection of $\mathbf{E}_{a,perm}$ represents the main advantage of this novel algorithm over Dray's approach [87], which progressively deflates $\mathbf{X}$ as new factors are extracted, but does not take into account the change of the rank of the matrices of the permuted residuals. This commonly generates very inconsistent outcomes: as a increases, the *null*-distributions associated to the single components are gradually more underestimated and the method is continuously more prone to detect noisy factors as statistically significant (see Section 9.4.1 for further details). This issue was originally solved by making the sequential computational procedure stop just after the identification of the first non-significant factor. However, for the properties of the statistic used for the testing procedure, Dray's method generally recognises less significant components than expected (see Appendix 14.2.2).

9.3.4 The rationale behind F_a

The projection of $\mathbf{E}_{a,perm}$ yields a decrease in its sum of squares. Then, the eigenvalues of $\mathbf{E}_a$ and $\mathbf{E}_{a,perm,\text{proj}}$ are not directly commensurable. As a solution to this issue, the statistical significance of each specific factor is tested through a relative measure, i.e. the ratio between the respective eigenvalue and its sum with all the smaller ones. In fact, since such a projection modifies at the same time and more or less uniformly the whole set of eigenvalues of $\mathbf{E}_{a,perm}$, this ratio is negligibly affected by the aforementioned decrease in its total sum of squares.

[vi] Nevertheless, P3 allows both the row- and the column-spaces of $\mathbf{E}_{a,perm}$ and $\mathbf{E}_{a,perm,\text{proj}}$ to be the same no matter if the projection is performed either row-wise or column-wise.

9.4 Performance of the algorithm

9.4.1 Synthetic datasets

Four synthetic matrices were exploited to verify whether the number of their underlying components (known *a priori*) could be correctly retrieved by the developed methodology. The data generation design was first detailed in [79]: 4, 8, 12 and 15 principal components, simulated independently at random and following a normal distribution with zero mean and unit variance, were respectively exploited to calculate a certain amount of observed variables according to the equations listed in Table 9.1. All the final arrays, featuring 100 objects, are examples of different correlation structures (from industrial process-like to spectral-like) and were contaminated with measurement noise of diverse magnitude to get an idea about the robustness of the implemented approach. Table 9.2 shows how many factors were retained at each noise level for the 4 datasets (for both the P1- and P3-based algorithms). The displayed values represent the median and the range of the number of selected components estimated over 300 simulation replicates. Clearly, P3 generally enabled an accurate identification of the number of significant components. Nevertheless, from a certain noise level on, depending on the nature of the considered covariance structure, the procedure more often tended to be less sensitive, but this is reasonable considering that noise covers successively more the less predominant factors and prevents them to be correctly pointed out as statistically significant.
On the other hand, regarding the last data matrix, the general overestimation of the number of components was not unexpected. In fact, as stressed in [79], in this particular circumstance and even for very small noise percentages, a notable portion of the variation of the last two factors gets lost in the residuals. Thus, it is difficult to conclude if the actual data dimensionality is either equal to or higher than 15.
Concerning P1, it commonly gave rise to a more conservative selection. In fact, as also evidenced by the comparison reported in Section 9.3.3, it allows only the major principal components to be appropriately recognised.
Finally, the outcomes resulting from the application of Horn's parallel analysis and Dray's method to the 4 synthetic datasets (displayed in Figures 9.5 and 9.6, respectively) corroborate what was stated before about their respective limitations: the former always overlooked some components of the original data structures just because they did not account for a high enough amount of variation, while the latter was usually prone to detect their last factors as statistically significant[vii]. Furthermore, in all four cases, even if the first non-significant component was used as stopping rule, Dray's method would have selected a too low number of fac-

[vii] Figures 9.5 and 9.6 are simply illustrative examples related to a single simulation replicate (noise level: 5%). However, the performance of the two techniques was found to be consistent regardless of both noise percentage and data generation repetition.

Table 9.1 Generation scheme of the 4 synthetic datasets. x identifies the observed variables, while pc denotes the principal components exploited for their simulation

Dataset #ID	Number of principal components	Number of original variables	Data generation scheme
1	4	10	$x_i = \sqrt{\frac{i}{5}}pc_1 + \sqrt{\frac{1-i}{5}}pc_2 \ \ \forall i \in 1,\dots,5$ $x_i = \sqrt{0.5}pc_1 + \sqrt{\frac{i}{10-0.5}}pc_2 + \sqrt{\frac{1-i}{10}}pc_3 \ \ \forall i \in 6,\dots,9$ $x_{10} = \frac{\sqrt{0.01}pc_1 + \sqrt{0.01}pc_2 + \sqrt{0.01}pc_3 + pc_4}{\sqrt{1.03}}$
2	8	10	$x_i = \sqrt{0.5}pc_k + \sqrt{0.5}pc_l \ \ \forall i \in 1,\dots,6 \ \ \forall k \neq l \in 1,\dots,4$ $x_i = \sqrt{0.5}pc_k + \sqrt{0.5}pc_l \ \ \forall i \in 7,8,9 \ \ \forall k \neq l \in 5,6,7$ $x_{10} = pc_8$
3	12	27	$x_i = pc_i \ \ \forall i \in 1,\dots,12$ $x_i = \sqrt{0.5}pc_k + \sqrt{0.5}pc_l \ \ \forall i \in 13,\dots,27 \ \ \forall k \neq l \in 1,\dots,6$
4	15	50	$x_i = \sqrt{0.5}pc_k + \sqrt{0.5}pc_l \ \ \forall i \in 1,\dots,45 \ \ \forall k \neq l \in 1,\dots,10$ $x_{46} = pc_{11}$ $x_{47} = pc_{12}$ $x_{48} = \sqrt{0.5}pc_{11} + \sqrt{0.5}pc_{13}$ $x_{49} = \sqrt{0.5}pc_{12} + \sqrt{0.5}pc_{14}$ $x_{50} = pc_{15}$

Table 9.2 Median value and range of the number of components retained at each noise level for the 4 synthetic datasets (estimated over 300 simulation replicates for both the P1- and P3-based algorithms). Their real number of factors is reported in the second column. Bold characters point out a correctly addressed assessment

Noise level*	Real number of components/Number of original variables	Number of estimated components	
		P1	P3
5%	4/10	1 {1 − 4}	**4 {4 − 4}**
10%	4/10	1 {1 − 4}	**4 {4 − 4}**
15%	4/10	1 {1 − 4}	**4 {1 − 4}**
20%	4/10	1 {1 − 4}	**4 {1 − 5}**
25%	4/10	1 {1 − 4}	**4 {1 − 5}**
50%	4/10	1 {1 − 4}	**4 {1 − 4}**
75%	4/10	1 {1 − 3}	2 {1 − 4}
100%	4/10	1 {1 − 3}	1 {1 − 4}
5%	8/10	2 {2 − 6}	**8 {3 − 9}**
10%	8/10	2 {1 − 6}	7 {1 − 8}
15%	8/10	2 {1 − 6}	6 {2 − 8}
20%	8/10	2 {1 − 5}	5 {2 − 8}
25%	8/10	2 {1 − 4}	5 {1 − 8}
50%	8/10	2 {1 − 3}	3 {1 − 7}
75%	8/10	2 {1 − 3}	2 {1 − 6}
100%	8/10	1 {0 − 3}	2 {0 − 6}
5%	12/27	6 {6 − 7}	**12 {5 − 12}**
10%	12/27	6 {6 − 7}	**12 {5 − 12}**
15%	12/27	6 {6 − 7}	**12 {5 − 12}**
20%	12/27	6 {5 − 7}	**12 {5 − 12}**
25%	12/27	6 {5 − 6}	**12 {5 − 12}**
50%	12/27	6 {4 − 6}	**12 {5 − 13}**
75%	12/27	5 {4 − 6}	**12 {5 − 13}**
100%	12/27	5 {2 − 6}	10 {4 − 14}
5%	15/50	12 {10 − 16}	16 {15 − 19}
10%	15/50	12 {10 − 15}	16 {15 − 19}
15%	15/50	12 {10 − 15}	16 {15 − 20}
20%	15/50	12 {10 − 15}	16 {14 − 22}
25%	15/50	12 {10 − 15}	17 {14 − 21}
50%	15/50	10 {9 − 12}	17 {13 − 22}
75%	15/50	10 {8 − 12}	16 {12 − 23}
100%	15/50	10 {7 − 12}	16 {11 − 22}

*__N.B.__ The percentage refers to the variation of the noise-free data.

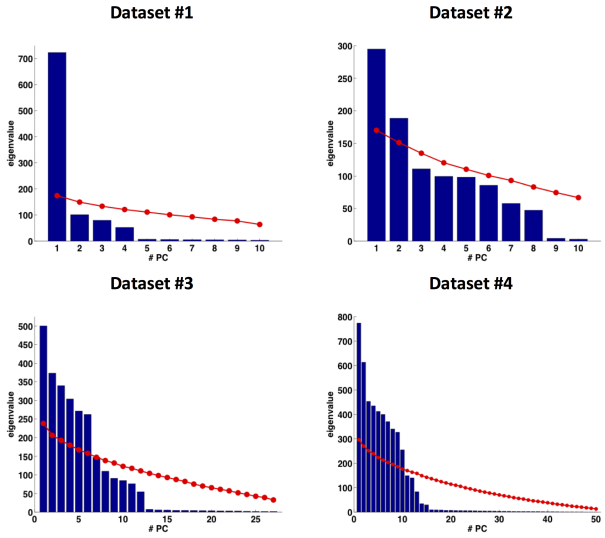

Figure 9.5 Results of the application of Horn's parallel analysis to the 4 synthetic datasets. The blue bars indicate the eigenvalues of the covariance matrices associated to the arrays under study, while the red dots correspond to the 99$^{\text{th}}$ percentiles of their respective *null*-distributions generated after 300 permutations

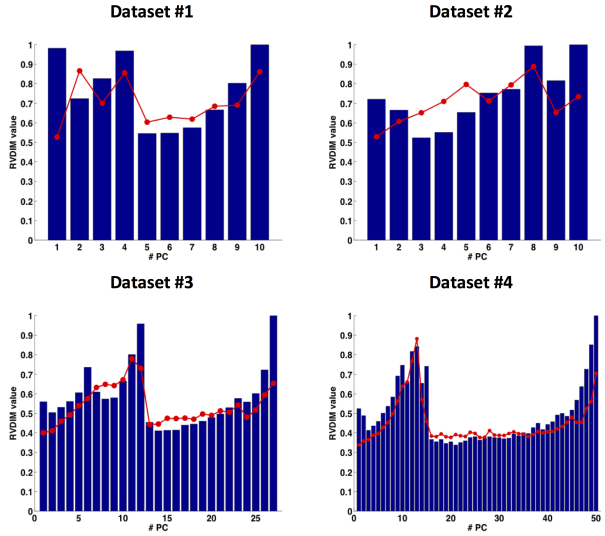

Figure 9.6 Results of the application of Dray's method to the 4 synthetic datasets. The blue bars indicate the $RVDIM_a$ values used for the testing procedure (see Appendix 14.2.2 for further details) and associated to the single components of the original matrices under study, while the red dots correspond to the 99$^{\text{th}}$ percentiles of their respective *null*-distributions generated after 300 permutations

tors. In this sense, it can be said that here the proposed permutation test-based procedure (encompassing the P3 step) outperformed both approaches.

9.4.2 Real case studies

The developed algorithm was also applied to 10 real datasets from distinct research fields, from archaeology to food preference. The purpose was to illustrate the different results for the concerned approaches, not to discuss them in great detail. For some of these datasets, based on the findings described in the original publications, a putative number of underlying components could be identified. The outcomes for both P1 and P3 are reported in Figure 9.7, while those for Horn's parallel analysis and Dray's method are graphed in Figures 9.8 and 9.9 (see Table 9.3 for a comprehensive summary). Notice that i) all the methods were run on the auto-scaled data matrices with a 99% confidence level and performing 300 permutation rounds; ii) Dray's computational procedure was stopped just after the detection of the first non-significant factor. When information was available on the actual dimensionality of the data, P3 always permitted to retrieve the correct number of components. On the other hand, Horn's parallel analysis and Dray's method generally led to more conservative selections. In 4 out of 10 situations, P1 returned similar results as P3, probably because the significant components were large enough to limit the effect related to the different projection procedures pointed out in Sections 9.3.3 and 9.4.1. However, that is not valid for the other real case studies where P1 yielded an underestimated number of factors with respect to P3 (see e.g. the performance of the 2 methodologies when the juice sensory array was handled)[viii]. In the light of that and although the actual dimensionality of some of the matrices taken into account was not known, the P3-based permutation test seemed to enable a more appropriate identification of how many principal components to extract in the different scenarios. It is true that sometimes more conservative selections may be safer especially when factors accounting for small amounts of data variation are detected as statistically significant (in this sense, the eigenvalues of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$, used for the Horn's parallel analysis testing procedure, can be helpful to additionally evaluate this aspect). But rather often phenomena of interest are just captured by such small components, and, thus, a tool being able to systematically unveil them can definitely be of use for many disparate applications.

[viii] For all the datasets, the 99^{th} percentile front resulting from P1 was found to be higher (as expected) than that obtained when P3 was concerned.

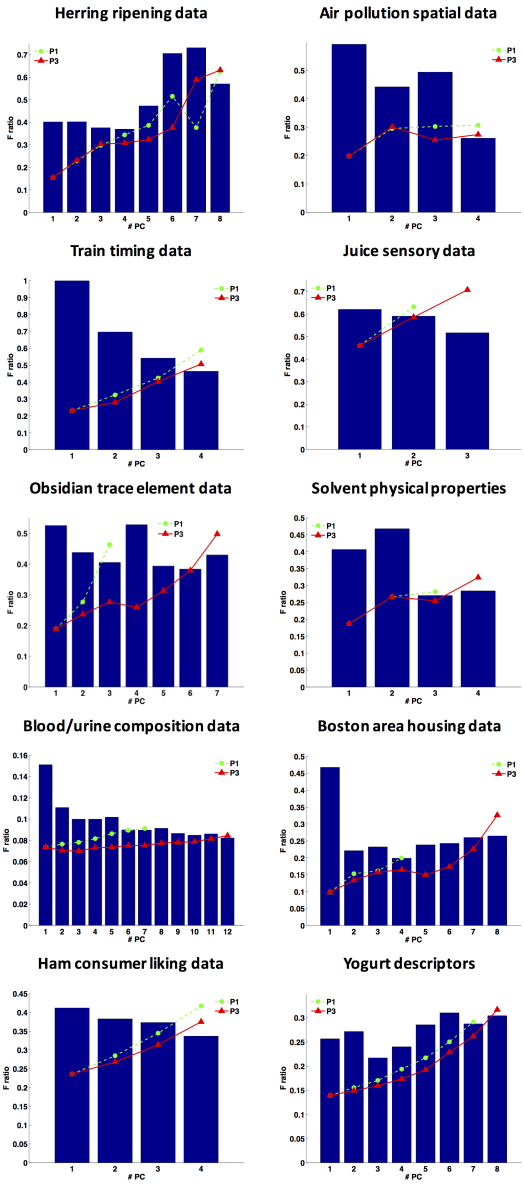

Figure 9.7 Results of the application of the P1- and P3-based permutation tests to the 10 real datasets. The blue bars indicate the F_a ratios used for the testing procedure and associated to the single components of the original matrices under study, while the green dots (for P1) and the red triangles (for P3) correspond to the 99^{th} percentiles of their respective *null*-distributions generated after 300 permutations

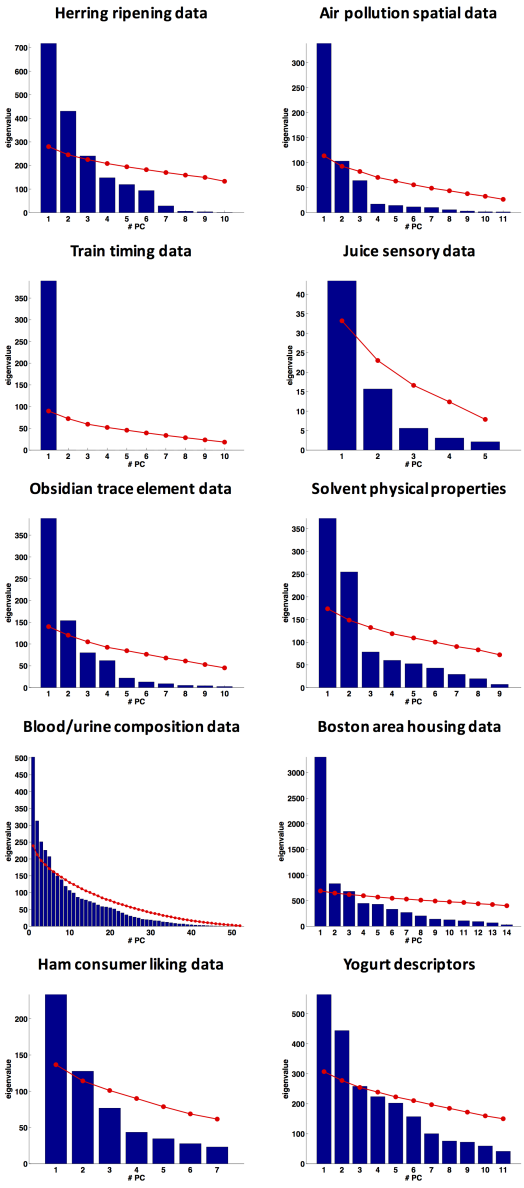

Figure 9.8 Results of the application of Horn's parallel analysis to the 10 real datasets. The blue bars indicate the eigenvalues of the covariance matrices associated to the arrays under study, while the red dots correspond to the 99^{th} percentiles of their respective *null*-distributions generated after 300 permutations

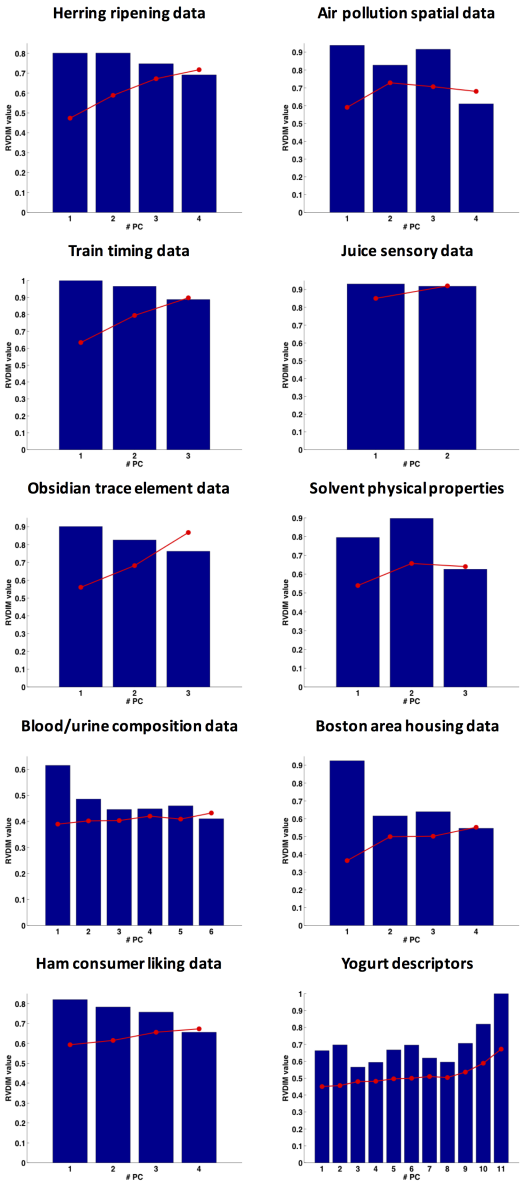

Figure 9.9 Results of the application of Dray's method to the 10 real datasets. The blue bars indicate the $RVDIM_a$ values used for the testing procedure (see Appendix 14.2.2 for further details) and associated to the single components of the original matrices under study, while the red dots correspond to the 99^{th} percentiles of their respective *null*-distributions generated after 300 permutations

Table 9.3 Number of statistically significant components estimated by Horn's parallel analysis, Dray's method and both the P1- and P3-based permutation tests for the 10 real datasets. Their putative number of factors (if known) is reported in the fourth column. Bold characters point out a correctly addressed assessment

Dataset	Size	Reference	Putative number of components	Number of estimated components			
				Horn's parallel analysis	Dray's method	P1	P3
Herring ripening data	180×10	[96]	7	3	3	**7**	**7**
Air pollution spatial data	53×11	[97]	3	2	**3**	**3**	**3**
Train timing data	40×10	[97]	3	1	2	**3**	**3**
Juice sensory data	6×14	[98]	2	1	1	1	**2**
Obsidian trace element data	75×10	[99]	-	2	2	2	6
Solvent physical properties	103×9	[100]	-	2	2	2	3
Blood/urine composition data	65×52	[101, 102]	-	6	5	6	11
Boston area housing data	506×14	[103]	-	3	3	3	7
Ham consumer liking data	8×81	[104]	-	2	3	3	3
Yogurt descriptors	12×200	[105]	-	3	11	6	7

9.5 Conclusions

In this chapter, an extensive guideline on how to accomplish the selection of PCA components by permutation testing was provided through the description of a novel and efficient algorithm. Its most relevant theoretical and practical aspects were discussed and clarified, namely the way the considered covariance structures are randomised, the importance of sequentially deflating the original matrix once every factor is computed, the necessity of a relative measure, the F_a ratio, to estimate their statistical significance and the need of a projection after each permutation round. This also permitted to mathematically formalise all the single numerical operations required when trying to quantify the number of factors underlying particular sets of data in this fashion. Furthermore, the application of the proposed method to both simulated and real case studies highlighted that it can constitute a feasible and valid alternative to classical permutation test-based approaches such as Horn's parallel analysis and Dray's method, which exhibit specific limitations mainly related to their intrinsic mathematical procedures.

Part IV

On modelling common and distinctive sources of variability in multi-set data analysis

Chapter 10

Some considerations on two-block common and distinctive component analysis

In this chapter several practical aspects of modelling common and distinctive components underlying two different sets of data are discussed.

10.1 Introduction

In many research and practical domains, it has recently become quite frequent to exploit multiple analytical platforms to comprehensively study the same system of interest [106]. For example, it is rather common nowadays to monitor a chemical process by at the same time i) collecting samples from the reaction medium during its progress and subjecting them to specific laboratory tests, ii) following the time evolution of so-called *engineering* variables, like temperatures, pressures or flow rates, measured by particular sensors installed inside the reactor, and iii) recording spectroscopic data in-line through advanced fibre optic-based instrumentations. In such cases, an intriguing and challenging task would be to distinguish the common and distinctive sources of variability (*factors* or *components*) underlying, for instance, two sets of data produced by the application of the different characterisation techniques. By the term *common* one refers to events or phenomena affecting all the concerned data blocks and by the term *distinctive* those affecting only one of them (or few of them if more data blocks are coped with) [107, 108]. But in which contingencies such data blocks can be, say, *concatenated* and analysed in this fashion? Generally when they share either the same number of rows/objects or the same number of columns/variables [109]. Imagine e.g. temperature, pressure and flow rate values and spectral data are simultaneously collected to monitor an industrial process. The two related datasets can be merged object-wise into a single matrix featuring a number of rows equal to the number of time points at which the measurements are executed and a number of columns equal to the sum of the number of the registered engineering and spectroscopic variables. Here, an *object-wise linked* data structure is concerned (see Figure 10.1).

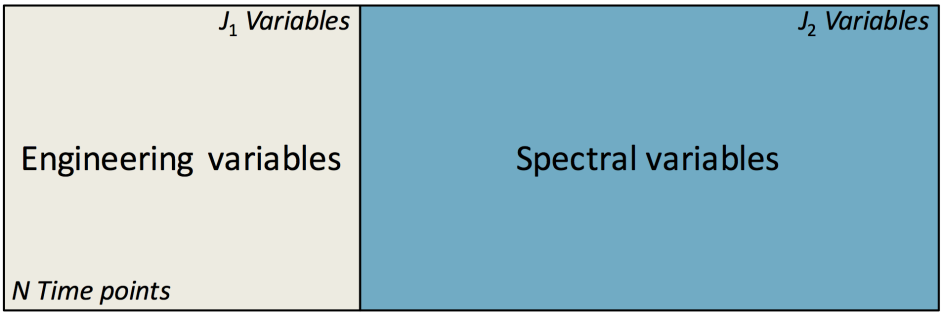

Figure 10.1 Example of object-wise linked data structure

Conversely, if the same engineering variables are resorted to for characterising the evolution of the same industrial process run in two different reacting units, the two resulting data blocks can be concatenated variable-wise, generating a unique array with a number of rows equal to the sum of the number of time points at which the measurements are performed and a number of columns equal to the number of such

engineering variables. Hence, a *variable-wise linked* data structure is obtained (see Figure 10.2). In the last decades, many dimensionality reduction methods have

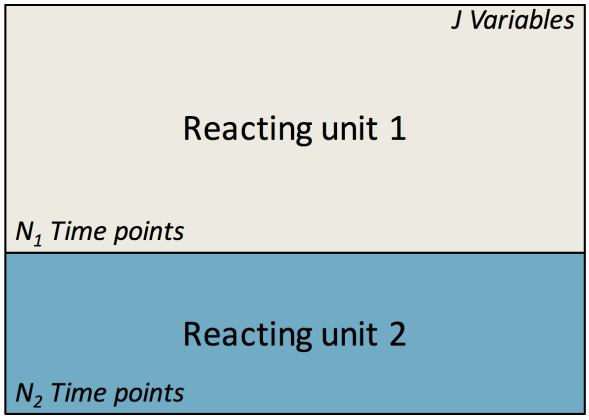

Figure 10.2 Example of variable-wise linked data structure

been proposed to model common and distinctive components when dealing with multi-set data analysis problems. Table 10.1 lists some of the most commonly used of these approaches, recently compared in [110]. These techniques can be classified

Table 10.1 List of some of the most commonly used dimensionality reduction methods for common and distinctive component modelling. The techniques are differentiated according to their capability of handling object-wise or variable-wise linked data structures and retrieving distinctive components affecting the variability of the data blocks under study. SCA, DISCO-SCA, GSVD, CCA, and O2PLS stand for *Simultaneous Component Analysis*, *DIStinctive and COmmon Simultaneous Component Analysis*, *Generalised Singular Value Decomposition*, *Canonical Correlation Analysis*, and *2-block Orthogonal Projections to Latent Structures*, respectively

	SCA	DISCO-SCA	Adapted GSVD	ECO-POWER	CCA	O2PLS
Common components	✓	✓	✓	✓	✓	✓
Distinctive components	✗	✓	✓	✗	✗	✓
Object-wise linked data	✓	✓	✓	✓	✓	✓
Variable-wise linked data	✓	✓	✓	✗	✗	✗

according to their capability of handling object-wise or variable-wise linked data structures and retrieving the distinctive components affecting the variability of the considered data blocks. However, they all share the same limitation: none of them encompasses preliminary computational steps aimed at identifying the number of such common and distinctive components, which in most cases can jeopardise the stability, and therefore the interpretation of the final results. Alternatively, authors usually deem a factor as common if it accounts for similar fractions of the total variation of each dataset under study, or as distinctive if not [110]. In the

present chapter, first, the inadequacy of this criterion will be highlighted through a comparison among the methodologies mentioned in Table 10.1. Then, a novel strategy to systematically detect the number of common and distinctive components when two blocks of measurements are dealt with will be described. Its power will be assessed in simulated and real examples. Finally, a recently developed algorithmic procedure, Joint-Y PLS (JYPLS), will be tested as a possible option for modelling common and distinctive sources of variation, strictly related to specific responses or properties of interest of the investigated objects/samples.

10.2 Methods

Let $\mathbf{X}_k$ be the k-th of multiple data matrices with dimensions $N \times J_k$ in the object-wise linked data case and $N_k \times J$ in the variable-wise linked data case. Assume from now on that each $\mathbf{X}_k$ is initially auto-scaled[i] and afterwards scaled to equal sum-of-squares[ii].

10.2.1 Simultaneous Component Analysis (SCA)

Simultaneous Component Analysis (SCA) [113, 114] is an extension of PCA suitable for the analysis of multi-set data. Concretely, PCA is applied to the concatenated $\mathbf{X}_k$ arrays, which can then be represented by the same mathematical structure (either scores or loadings) as:

$$\text{Object-wise linked data:} \quad \mathbf{X}_k = \mathbf{T}\mathbf{P}_k^{\mathrm{T}} + \mathbf{E}_k \tag{10.1}$$

$$\text{Variable-wise linked data:} \quad \mathbf{X}_k = \mathbf{T}_k\mathbf{P}^{\mathrm{T}} + \mathbf{E}_k \tag{10.2}$$

From Equations 10.1 and 10.2 it is clear that depending on whether object-wise linked data or variable-wise linked data are concerned, all the single blocks share the same component scores ($\mathbf{T}$, $N \times A$), while featuring individual loadings ($\mathbf{P}_k$, $J_k \times A$) or *vice versa* ($\mathbf{T}_k$, $N_k \times A$, and $\mathbf{P}$, $J \times A$)[iii].
The SCA decomposition is attained by solving the objective functions in Equations 10.3 and 10.4:

$$\text{Object-wise linked data:} \quad \min_{\mathbf{T},\mathbf{P}_k} \sum_k \|\mathbf{X}_k - \mathbf{T}\mathbf{P}_k^{\mathrm{T}}\|^2 \quad \text{s.t. } \mathbf{T}^{\mathrm{T}}\mathbf{T} = \mathbf{I} \tag{10.3}$$

$$\text{Variable-wise linked data:} \quad \min_{\mathbf{T}_k,\mathbf{P}} \sum_k \|\mathbf{X}_k - \mathbf{T}_k\mathbf{P}^{\mathrm{T}}\|^2 \quad \text{s.t. } \sum_k \mathbf{T}_k^{\mathrm{T}}\mathbf{T}_k = \mathbf{I} \tag{10.4}$$

where $\mathbf{I}$ ($A \times A$) is an identity matrix.

[i] Again, auto-scaling here corresponds to centering and scaling to unit variance the concerned J_k or J variables.

[ii] This will allow all the measured variables to have equal weight and prevent potential bias due to differences in e.g. the size of the various $\mathbf{X}_k$ [111, 112].

[iii] A denotes the number of extracted principal components.

10.2.2 DIStinctive and COmmon Simultaneous Component Analysis (DISCO-SCA)

DIStinctive and COmmon Simultaneous Component Analysis (DISCO-SCA) [115, 116] rotates the SCA components to a target distinctive and common structure according to the following objective functions (for the two-block case):

$$\text{Object-wise linked data:} \quad \min_{\mathbf{B}} \|\mathbf{W} \circ (\mathbf{P}_{\text{target}} - [\mathbf{P}_1^{\text{T}}\ \mathbf{P}_2^{\text{T}}]^{\text{T}}\mathbf{B})\|^2 \quad \text{s.t. } \mathbf{B}^{\text{T}}\mathbf{B} = \mathbf{I} = \mathbf{B}\mathbf{B}^{\text{T}} \tag{10.5}$$

$$\text{Variable-wise linked data:} \quad \min_{\mathbf{B}} \|\mathbf{W} \circ (\mathbf{T}_{\text{target}} - [\mathbf{T}_1^{\text{T}}\ \mathbf{T}_2^{\text{T}}]^{\text{T}}\mathbf{B})\|^2 \quad \text{s.t. } \mathbf{B}^{\text{T}}\mathbf{B} = \mathbf{I} = \mathbf{B}\mathbf{B}^{\text{T}} \tag{10.6}$$

being $\mathbf{W}$ ($J_1 + J_2 \times A$ for object-wise linked data and $N_1 + N_2 \times A$ for variable-wise linked data) a selectivity matrix containing ones in the entries matching the elements of $(\mathbf{P}_{\text{target}} - [\mathbf{P}_1^{\text{T}}\ \mathbf{P}_2^{\text{T}}]^{\text{T}}\mathbf{B})$ or $(\mathbf{T}_{\text{target}} - [\mathbf{T}_1^{\text{T}}\ \mathbf{T}_2^{\text{T}}]^{\text{T}}\mathbf{B})$ across which the minimisation has to be carried out, $\mathbf{P}_{\text{target}}$ $(J_1 + J_2 \times A)$/$\mathbf{T}_{\text{target}}$ $(N_1 + N_2 \times A)$ a predefined array of loadings/scores with zeros in the positions corresponding to the dataset that specific components do not underlie and arbitrary values elsewhere, $\mathbf{B}$ $(A \times A)$ the adjustable rotation matrix, and denoting $\circ$ the element-wise (Hadamard) product. As an example, if $J_1 = 3$, $J_2 = 4$, $A = 3$ and

$$\mathbf{P}_{\text{target}} = \begin{bmatrix} c & 0 & c \\ c & 0 & c \\ c & 0 & c \\ 0 & c & c \\ 0 & c & c \\ 0 & c & c \\ 0 & c & c \end{bmatrix} \tag{10.7}$$

with $c \neq 0$, then the first calculated factor will be distinctive for $\mathbf{X}_1$, the second distinctive for $\mathbf{X}_2$, and the third common and shared by both of them. Once found the optimal $\mathbf{B}$, either the SCA scores or the SCA loadings are counter-rotated accordingly, which yields the final DISCO-SCA solution.

10.2.3 Adapted Generalised Singular Value Decomposition (Adapted GSVD)

Adapted Generalised Singular Value Decomposition (Adapted GSVD) [117, 118] is a popular dimensionality reduction method in computational biology. It models the data under study as:

$$\text{Object-wise linked data:} \quad \mathbf{X}_k = \mathbf{T}\mathbf{D}_k\mathbf{V}_k^{\text{T}} + \mathbf{E}_k = \mathbf{T}\mathbf{P}_k^{\text{T}} + \mathbf{E}_k \quad \text{s.t. } \mathbf{V}_k^{\text{T}}\mathbf{V}_k = \mathbf{I} \tag{10.8}$$

$$\text{Variable-wise linked data:} \quad \mathbf{X}_k = \mathbf{U}_k\mathbf{D}_k\mathbf{P}^{\text{T}} + \mathbf{E}_k = \mathbf{T}_k\mathbf{P}^{\text{T}} + \mathbf{E}_k \quad \text{s.t. } \mathbf{U}_k^{\text{T}}\mathbf{U}_k = \mathbf{I} \tag{10.9}$$

with $\mathbf{V}_k$ and $\mathbf{U}_k$ of size $J_k \times A$ and $N_k \times A$, respectively, and $\mathbf{D}_k$ $(A \times A)$ diagonal. Its algorithm encompasses two consecutive steps: first, SVD is applied to the

concatenated $\mathbf{X}_k$ blocks; afterwards, SVD is again performed on each block-specific part of the resulting right (in the object-wise linked data case) or left (in the variable-wise linked data case) singular vector matrix.

10.2.4 ECO-POWER

ECO-POWER [119] is a variant of SCA which can be exploited for the analysis of object-wise linked data and that solves the objective function in Equation 10.10 (for the two-block case):

$$\text{Object-wise linked data: } \max_{\mathbf{W}} \sum_a R^2_{1,a} R^2_{2,a} \quad \text{s.t. } \mathbf{T}^{\mathrm{T}}\mathbf{T} = \mathbf{I} \tag{10.10}$$

denoting $R^2_{k,a}$ the proportion of variance of the k-th block explained by the a-th extracted component, and being $\mathbf{T} = [\mathbf{X_1}\ \mathbf{X}_2]\mathbf{W}$ and $\mathbf{W}$ of dimensions $(J_1 + J_2) \times A$[iv]. Given $\mathbf{T}$ $(N \times A)$, $J_k \times A$ arrays of individual loadings can be then retrieved as:

$$\mathbf{P}_1 = \mathbf{X}_1^{\mathrm{T}}\mathbf{T}^{\mathrm{T}} \tag{10.11}$$

$$\mathbf{P}_2 = \mathbf{X}_2^{\mathrm{T}}\mathbf{T}^{\mathrm{T}} \tag{10.12}$$

The ECO-POWER optimisation problem can be solved by an iterative majorisation procedure. Furthermore, maximising $\sum_a R^2_{1,a} R^2_{2,a}$ guarantees that factors accounting for a similar amount of variance in both datasets are obtained.

10.2.5 Canonical Correlation Analysis (CCA)

Canonical Correlation Analysis (CCA) [120] is a technique suitable for handling pairs of object-wise linked datasets. Here, the $\mathbf{X}_k$ blocks $(k = 1, 2)$ are modelled as:

$$\text{Object-wise linked data: } \mathbf{X}_k = \mathbf{X}_k\mathbf{W}_k\mathbf{P}_k^{\mathrm{T}} + \mathbf{E}_k = \mathbf{T}_k\mathbf{P}_k^{\mathrm{T}} \tag{10.13}$$

where $\mathbf{W}_k$ $(J_k \times A)$ is a matrix containing the so-called canonical weights, $\mathbf{T}_k$ $(N \times A)$ represents the canonical variate array, while the loadings $\mathbf{P}_k$ $(J_k \times A)$ are obtained by regressing $\mathbf{X}_k$ on $\mathbf{T} = \mathbf{X}_k\mathbf{W}_k$.
CCA solves the following objective function:

$$\text{Object-wise linked data: } \max_{\mathbf{W}_1, \mathbf{W}_2} \mathrm{tr}(\mathbf{W}_1^{\mathrm{T}}\mathbf{X}_1^{\mathrm{T}}\mathrm{X}_2\mathrm{W}_2) \quad \text{s.t. } N^{-1}\mathbf{T}_1^{\mathrm{T}}\mathbf{T}_1 = \mathbf{I} = N^{-1}\mathbf{T}_2^{\mathrm{T}}\mathbf{T}_2 \tag{10.14}$$

Thus, $\mathbf{W}_1$ and $\mathbf{W}_2$ result from the maximisation of the sum of the correlations between the A couples of canonical variates. Since the variance of the different $\mathbf{X}_k$ explained by such canonical variates is not taken into account in Equation 10.14,

[iv] This implies that the component scores in $\mathbf{T}$ lie in the space spanned by $[\mathbf{X}_1\ \mathbf{X}_2]$.

they might be poor descriptors of the original data [121]. In order to overcome this limitation, which can generate certain instability in the final outcomes[v], one may apply PCA block-wise prior to CCA (as in Principal Component Regression - PCR, see Section 14.3.2) or use regularisation [122–124]. The extension of the CCA algorithm for coping with more than two datasets is known as Generalised Canonical Correlation Analysis (GCCA) [124].

10.2.6 2-block Orthogonal Projections to Latent Structures (O2PLS)

The 2-block Orthogonal Projections to Latent Structures (O2PLS) [125] decomposition of two object-wise linked data blocks can be written as:

$$\text{Object-wise linked data: } \mathbf{X}_k = \mathbf{T}_{k,\text{c}}\mathbf{P}_{k,\text{c}}^{\text{T}} + \mathbf{T}_{k,\text{d}}\mathbf{P}_{k,\text{d}}^{\text{T}} + \mathbf{E}_k \quad \text{for } k = 1, 2 \quad (10.15)$$

being $\mathbf{T}_{k,\text{c}}$ ($N \times A_c$) and $\mathbf{P}_{k,\text{c}}$ ($J_k \times A_c$) the common component scores and loadings arrays, and $\mathbf{T}_{k,\text{d}}$ ($N \times A_d$) and $\mathbf{P}_{k,\text{d}}$ ($J_k \times A_d$) the distinctive component scores and loadings arrays, respectively[vi].
For the sake of comparison, the first implementation of O2PLS will be resorted to. It is based on a three-step computational procedure:

1. the common variation shared by both $\mathbf{X}_1$ and $\mathbf{X}_2$ is preliminarily extracted by performing SVD on the between-block covariance matrix $\mathbf{X}_2^{\text{T}}\mathbf{X}_1$, which gives:
$$\mathbf{X}_2^{\text{T}}\mathbf{X}_1 = \mathbf{U}\mathbf{S}\mathbf{V}^{\text{T}} + \mathbf{E} \quad (10.16)$$

2. the first distinctive factors underlying $\mathbf{X}_1$ and $\mathbf{X}_2$ are then retrieved solving:
$$\max_{\mathbf{w}_{1,\text{d}}} \mathbf{w}_{1,\text{d}}^{\text{T}}\mathbf{E}_1^{\text{T}}\mathbf{X}_1\mathbf{V}\mathbf{V}^{\text{T}}\mathbf{X}_1^{\text{T}}\mathbf{E}_1\mathbf{w}_{1,\text{d}} \quad (10.17)$$
$$\max_{\mathbf{w}_{2,\text{d}}} \mathbf{w}_{2,\text{d}}^{\text{T}}\mathbf{E}_2^{\text{T}}\mathbf{X}_2\mathbf{V}\mathbf{V}^{\text{T}}\mathbf{X}_2^{\text{T}}\mathbf{E}_2\mathbf{w}_{2,\text{d}} \quad (10.18)$$
with $\mathbf{w}_{1,\text{d}}$ of dimensions $J_1 \times 1$, $\mathbf{w}_{2,\text{d}}$ of dimensions $J_2 \times 1$, $\mathbf{E}_1 = \mathbf{X}_1 - \mathbf{X}_1\mathbf{V}\mathbf{V}^{\text{T}}$, and $\mathbf{E}_2 = \mathbf{X}_2 - \mathbf{X}_2\mathbf{U}\mathbf{U}^{\text{T}}$;

3. once estimated all the distinctive components by consecutively deflating $\mathbf{X}_1$ and $\mathbf{X}_2$, the profiles of the common ones are recovered individually by exploiting the MAXDIFF criterion [126]:
$$\max_{\mathbf{w}_{1,\text{c}},\mathbf{w}_{2,\text{c}}} \text{tr}(\mathbf{w}_{1,\text{c}}^{\text{T}}\mathbf{X}_{1,\text{res}}^{\text{T}}\mathbf{X}_{1,\text{res}}\mathbf{w}_{2,\text{c}}) \quad \text{s.t. } \mathbf{W}_{k,\text{c}}^{\text{T}}\mathbf{W}_{k,\text{c}} = \mathbf{I} \quad \text{for } k = 1, 2 \quad (10.19)$$

[v] E.g. when $N < J_k$ for some k Equation 10.14 leads to an undetermined system of equations.
[vi] A_c and A_d denote the number of common and distinctive O2PLS components, which is not necessarily the same.

where the columns of $\mathbf{W}_{k,\mathrm{c}}$ $(J_k \times A_\mathrm{c})$ correspond to the single J_k-dimensional vectors $\mathbf{w}_{k,\mathrm{c}}$, $\mathbf{X}_{1,\mathrm{res}} = \mathbf{X}_1 - \mathbf{X}_1\mathbf{V}\mathbf{V}^\mathrm{T} - \mathbf{T}_{1,\mathrm{d}}\mathbf{P}_{1,\mathrm{d}}^\mathrm{T}$ and $\mathbf{X}_{2,\mathrm{res}} = \mathbf{X}_2 - \mathbf{X}_2\mathbf{U}\mathbf{U}^\mathrm{T} - \mathbf{T}_{2,\mathrm{d}}\mathbf{P}_{2,\mathrm{d}}^\mathrm{T}$. As for CCA, regressing $\mathbf{X}_k$ on the scores $\mathbf{T}_{k,\mathrm{d}} = \mathbf{X}_k\mathbf{W}_{k,\mathrm{d}}$ yields the loadings $\mathbf{P}_{k,\mathrm{d}}$[vii].

The extension of O2PLS for dealing with more than two datasets is known as n-block Orthogonal Projections to Latent Structures (OnPLS) [127].

10.3 Datasets

Three datasets will be object of this study.

10.3.1 Gene expression data

Gene expression data, including 54715 probe sets, were collected by micro-array analysis of peripheral blood mononuclear cells of 51 individuals vaccinated against influenza [128]. 24 of them were treated with Trivalent Inactivated Influenza Vaccine (TIIV), while the remaining 27 with Live Attenuated Influenza Vaccine (LAIV). Data at baseline (day 0), and at day 3 and day 7 after vaccination were registered. An object-wise linked structure of size $24 \times (54715 \times 2)$ was then built by merging the two gene expression matrices related to the 24 subjects dosed with TIIV, the first pertaining to the measurements recorded at three and the second to those recorded at seven days after vaccination. On the other hand, a variable-wise linked array of dimensions $(24 + 27) \times 54715$ was constructed combining the gene expression matrix associated to the 24 individuals treated by TIIV and the gene expression matrix associated to the 27 individuals treated by LAIV, both resulting from the micro-array analysis performed at day 3 after vaccination.
Prior to the modelling phase, each block was baseline-corrected (by subtracting the day 0 data), auto-scaled, and subsequently scaled to equal sum-of-squares. For every subject, the plasma hemagglutination-inhibition antibody response (titer) was also quantified 28 days after vaccination.

10.3.2 Simulated pseudo-spectral data

Two datasets, both 70×96-sized, were generated such that they share two pseudo-spectral components, while featuring one distinctive pseudo-spectral factor each. The term *pseudo-spectral* refers to the fact that the four components used for the data simulation are orthogonal and not necessarily non-negative. The blocks were centred[viii], scaled to equal sum-of-squares, and concatenated object-wise.

[vii] The columns of $\mathbf{W}_{k,\mathrm{d}}$ $(J_k \times A_\mathrm{d})$ correspond to the single J_k-dimensional vectors $\mathbf{w}_{k,\mathrm{d}}$.

[viii] The variance of all the spectral variables within each dataset was found to be rather similar. Therefore, they were not scaled before further processing.

10.3.3 Industrial batch process data

20 engineering variables (mainly temperatures, pressures and flow rates) were recorder over time during the evolution of 77 batches. The data were first synchronised by a recently proposed algorithm, Multisynchro [129], to guarantee all the 77 process runs had the same evolution pace, and afterwards unfolded batch-wise. The final two-way array was thereafter split into two different blocks, namely $\mathbf{X}_1$ and $\mathbf{X}_2$. $\mathbf{X}_1$ contained information associated to runs which evolved under NOC. Conversely, the batches in $\mathbf{X}_2$ led to products with a gradually lower and fluctuating quality. $\mathbf{X}_1$ and $\mathbf{X}_2$ were then auto-scaled, scaled to equal sum-of-squares, and, as the column dimension is common between them, linked variable-wise.
A similar data structure was available for a second set of process runs (22 from the NOC time period and 14 from the later time period), manufactured in another reacting unit and for which the same engineering variables were monitored.

10.4 Results and discussion

This section will try to answer the following questions: i) Is the variation accounted for by every component a suitable criterion to determine whether it is common or distinctive? ii) If not, are there alternative strategies to correctly detect their number? iii) What if common and distinctive factors have to be related to specific responses or properties of interest? As evident from Section 10.2, none of the described approaches is able to fulfil this last option.

10.4.1 Is the variation accounted for reliable?

In Section 10.1, it was stressed that, when handling techniques - like SCA or DISCO-SCA - to cope with multiple datasets, most authors usually consider a component as common if it accounts for similar amounts of the total variation of each one of them, or as distinctive otherwise. This permits to bypass the main limitation of many of such methods which do not encompass primary algorithmic steps for the tentative identification of the exact number of common and distinctive factors to be calculated. But, is that really dependable?
The Variation Accounted For (VAF) by the a-th component in the k-th data block can be expressed as:

$$\text{Object-wise linked data:} \quad VAF_{k,a} = \frac{1 - \|\mathbf{X}_k - \mathbf{t}_a \mathbf{p}_{k,a}\|^2}{\|\mathbf{X}_k\|^2} \tag{10.20}$$

$$\text{Variable-wise linked data:} \quad VAF_{k,a} = \frac{1 - \|\mathbf{X}_k - \mathbf{t}_{k,a} \mathbf{p}_a\|^2}{\|\mathbf{X}_k\|^2} \tag{10.21}$$

denoting $\mathbf{t}$ and $\mathbf{p}$ the scores and loadings computed by the particular methodology under study. Table 10.2 shows the VAF indices of all the factors extracted by

Table 10.2 Gene expression data - Object-wise case: VAF indices of all the components estimated by SCA, DISCO-SCA, Adapted GSVD, ECO-POWER, CCA and O2PLS. The bottom part of the table includes the Pearson's correlation coefficients of every pair of corresponding block-specific loadings vectors. Notice that the selection of the DISCO-SCA solution was attained by rotating the 5 SCA factors towards every possible target and minimising their deviation from it. As an example to interpret the notation, $C1_2$ refers to the first component loadings vector calculated for the second dataset under study. Red, blue and green colours connotes components identified as common, distinctive for the Day 3 block, and distinctive for the Day 7 block, respectively

	SCA*		DISCO-SCA*		Adapted GSVD		ECO-POWER		CCA		O2PLS	
	Day 3	Day 7	Day 3	Day 7	Day 3	Day 7	Day 3	Day 7	Day 3	Day 7	Day 3	Day 7
C1	0.0950	0.1036	0.0950	0.1036	0.0875	0.0500	0.0628	0.0610	0.0959	0.1045	0.0870	0.0871
C2	0.0743	0.0681	0.0743	0.0681	0.0858	0.0644	0.0678	0.0660	0.0694	0.0718	0.0689	0.0751
C3	0.0730	0.0609	0.0730	0.0609	0.0614	0.0705	0.0723	0.0692	0.0724	0.0614	0.0626	0.0595
C4	0.0622	0.0614	0.0622	0.0614	0.0685	0.0940	0.0964	0.1020	0.0624	0.0583	0.0565	0.0546
C5	0.0492	0.0627	0.0492	0.0627	0.0505	0.0778	0.0550	0.0580	0.0491	0.0603	0.0794	0.0681
$C1_1/C1_2$	0.7955		0.7955		0.6181		0.5177		0.7700		0.6819	
$C2_1/C2_2$	0.6427		0.6427		0.6400		0.5651		-0.6857		0.6829	
$C3_1/C3_2$	0.5546		0.5546		0.6243		0.6038		-0.5754		0.5608	
$C4_1/C4_2$	0.5267		0.5267		0.7127		0.7979		0.5383		0.6234	
$C5_1/C5_2$	0.5568		0.5568		0.5257		0.5949		0.5645		-0.1994	

*Both SCA and DISCO-SCA led to the same component model.
**By O2PLS, individual distinctive components are separately retrieved for every concerned data block. For this reason, the background colour pattern of their corresponding cells is slightly different with respect to that of the rest of the entries of the table.

SCA, DISCO-SCA, Adapted GSVD, ECO-POWER, CCA and O2PLS for the gene expression data in the object-wise case. Based on the difference between the two VAF values associated to the single components, a rough assessment of their status was carried out (relatively large/low differences were regarded as typical of distinctive/common components - see e.g. C1 for Adapted GSVD and C1 for ECO-POWER, respectively). As a way of validating this assessment, the Pearson's correlation coefficient of every pair of corresponding block-specific loadings vectors was also estimated[ix]. Model complexity was chosen to be the same as in [110]. Under the assumption that the more correlated the loadings vectors, the more similar the phenomena the concerned factors explain, two interesting points arise from the displayed results:

- even if for most of the compared approaches the components deemed as common seem to be characterised by a higher correlation between blocks than those recognised as distinctive, setting a univocal criterion to discriminate them according to the VAF is not straightforward;
- consequently, no consensus can be easily achieved among techniques. For instance, the first Adapted GSVD factor, labelled as distinctive, exhibits a rather high difference between its individual VAF values, but its between-block Pearson's correlation coefficient is clearly larger than that of the first ECO-POWER factor, classified as common.

In addition, one could also think of a situation where the same phenomenon might contribute differently to the global variability of the various datasets, and where the component(s) capturing it might therefore account for different proportions of such a variability. All these aspects point out that exploiting the VAF as a decision rule to distinguish common and distinctive variation may be problematic.

10.4.2 Determining the number of common and distinctive components: a novel strategy

Suppose the information associated to the common and distinctive components of two data blocks sharing the row dimension, say $\mathbf{X}_1$ ($N \times J_1$) and $\mathbf{X}_2$ ($N \times J_2$), has to be recovered. Assuming again that the common components are those showing the highest correlation between blocks (either positive or negative), their number can be approximately identified by CCA. Specifically, by combining its principles

[ix]The determination of the Pearson's correlation coefficient of every pair of loadings vectors is possible and meaningful because the two analysed data blocks feature not only the same column dimension, but also the exactly same measured variables. On the contrary, as distinct groups of subjects were dosed with either TIIV or LAIV, such a calculation is unfeasible when linking them variable-wise. For this reason and because not all the considered approaches can be directly applied to datasets concatenated in this manner, the study reported here was conducted only in the object-wise case.

to those of permutation testing. However, as highlighted in Section 10.2.5, CCA suffers from severe drawbacks when e.g. the number of rows of $\mathbf{X}_1$ and $\mathbf{X}_2$ is lower than the number of their columns. In order to bypass such an issue, CCA can be directly run on the left singular vectors obtained from a preliminary SVD decomposition of the two datasets. But to this end, the effective rank of these two matrices has to be precisely determined to prevent noisy factors from being involved in its computational procedure, which might lead to detect spurious correlations between $\mathbf{X}_1$ and $\mathbf{X}_2$. The permutation testing-based algorithm illustrated in Part III will be resorted to for addressing this task. Once the number of common components has been assessed, their profiles can be retrieved by one of the techniques described in Section 10.2. Notice that here the focus will not be on how to model such components, but primarily on how to quantify their number. Nevertheless, to get a very general idea of the intrinsic information they carry, a simple and fast pseudo-O2PLS method, by which these common factors are extracted by performing SVD on the between-block covariance matrix $\mathbf{X}_1^{\mathrm{T}}\mathbf{X}_2$, will be employed. The distinctive components can be subsequently explored by applying PCA separately on $\mathbf{X}_1$ and $\mathbf{X}_2$ after deflating the common ones and estimating their number by the same approach used for the initial effective rank determination of the single data blocks.
In summary, the proposed strategy comprises six different steps:

1. the total number of factors underlying $\mathbf{X}_1$ and $\mathbf{X}_2$ (A_1 and A_2) is calculated by permutation testing (see Part III);

2. $\mathbf{X}_1$ and $\mathbf{X}_2$ are decomposed by SVD and their first A_1 and A_2 left singular vectors are retained, respectively;

3. the left singular vector matrices ($\mathbf{U}_1$, $N \times A_1$, and $\mathbf{U}_2$, $N \times A_2$) are then subjected to CCA. The statistical significance of the resulting canonical correlations is evaluated through a permutation test carried out randomising iteratively the order of the entire rows of either $\mathbf{U}_1$ or $\mathbf{U}_2$ and recomputing the CCA solution. Canonical correlations larger than the 99^{th} percentile of their null-distributions are considered statistically significant. The number of statistically significant canonical correlations (A_{c}) is set as the final number of common components between $\mathbf{X}_1$ and $\mathbf{X}_2$;

4. the A_{c} common factors are modelled as:

$$\mathbf{X}_1^{\mathrm{T}}\mathbf{X}_2 = \mathbf{U}_{\mathrm{c}}\mathbf{S}_{\mathrm{c}}\mathbf{V}_{\mathrm{c}}^{\mathrm{T}} + \mathbf{E}_c \tag{10.22}$$

   with $\mathbf{U}_{\mathrm{c}}$ of dimensions $J_1 \times A_{\mathrm{c}}$, $\mathbf{S}_{\mathrm{c}}$ of dimensions $A_{\mathrm{c}} \times A_{\mathrm{c}}$, $\mathbf{V}_{\mathrm{c}}$ of dimensions $J_2 \times A_{\mathrm{c}}$, and $\mathbf{E}_{\mathrm{c}}$ of dimensions $J_1 \times J_2$;

5. the common components are deflated from $\mathbf{X}_1$ and $\mathbf{X}_2$ as:

$$\mathbf{X}_{1,\mathrm{d}} = \mathbf{X}_1 - \mathbf{X}_1\mathbf{U}_\mathrm{c}\mathbf{U}_\mathrm{c}^\mathrm{T} \tag{10.23}$$

$$\mathbf{X}_{2,\mathrm{d}} = \mathbf{X}_2 - \mathbf{X}_2\mathbf{V}_\mathrm{c}\mathbf{V}_\mathrm{c}^\mathrm{T} \tag{10.24}$$

6. SVD is finally used to retrieve the distinctive factors of $\mathbf{X}_1$ and $\mathbf{X}_2$:

$$\mathbf{X}_{1,\mathrm{d}} = \mathbf{U}_{1,\mathrm{d}}\mathbf{S}_{1,\mathrm{d}}\mathbf{V}_{1,\mathrm{d}}^\mathrm{T} + \mathbf{E}_{1,\mathrm{d}} \tag{10.25}$$

$$\mathbf{X}_{2,\mathrm{d}} = \mathbf{U}_{2,\mathrm{d}}\mathbf{S}_{2,\mathrm{d}}\mathbf{V}_{2,\mathrm{d}}^\mathrm{T} + \mathbf{E}_{2,\mathrm{d}} \tag{10.26}$$

being $\mathbf{U}_{1,\mathrm{d}}$ $N \times A_{1,\mathrm{d}}$-sized, $\mathbf{S}_{1,\mathrm{d}}$ $A_{1,\mathrm{d}} \times A_{1,\mathrm{d}}$-sized, $\mathbf{V}_{1,\mathrm{d}}$ $J_1 \times A_{1,\mathrm{d}}$-sized, $\mathbf{E}_{1,\mathrm{d}}$ $N \times J_1$-sized, $\mathbf{U}_{2,\mathrm{d}}$ $N \times A_{2,\mathrm{d}}$-sized, $\mathbf{S}_{2,\mathrm{d}}$ $A_{2,\mathrm{d}} \times A_{2,\mathrm{d}}$-sized, $\mathbf{V}_{2,\mathrm{d}}$ $J_2 \times A_{2,\mathrm{d}}$-sized, and $\mathbf{E}_{2,\mathrm{d}}$ $N \times J_2$-sized[x].

The developed methodology will be tested in both the pseudo-spectral and the industrial case study[xi].

Simulated pseudo-spectral data

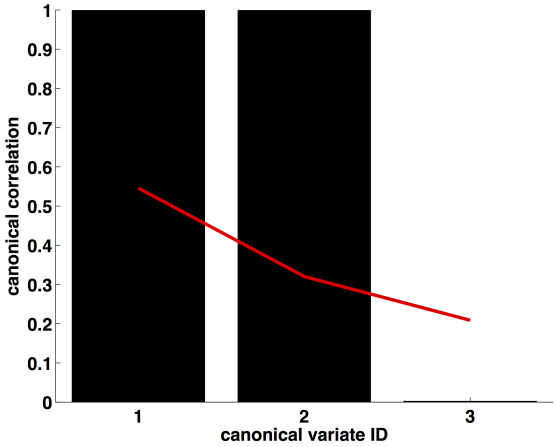

Figure 10.3 Simulated pseudo-spectral data: detection of the number of common components. The black bars represent the observed canonical correlations, while the red line connects the 99$^{\mathrm{th}}$ percentiles of their respective *null*-distributions

Three components were found to underlie each pseudo-spectral data block, but, as displayed in Figure 10.3, only two out of three canonical variates exhibited

[x] $A_{1,\mathrm{d}}$ and $A_{2,\mathrm{d}}$ can be estimated as in step 1.

[xi] Here, in contrast to the gene expression data, it is easier to validate the obtained outcomes based on *a priori* knowledge and previous findings.

statistically significant canonical correlations. The two common factors shared by the concerned datasets were then correctly identified. Their profiles, together with those of the two remaining distinctive components[xii], were recovered by the aforementioned pseudo-O2PLS approach. A good agreement between such profiles and the loadings originally utilised for the data generation is observed (see Figure 10.4).

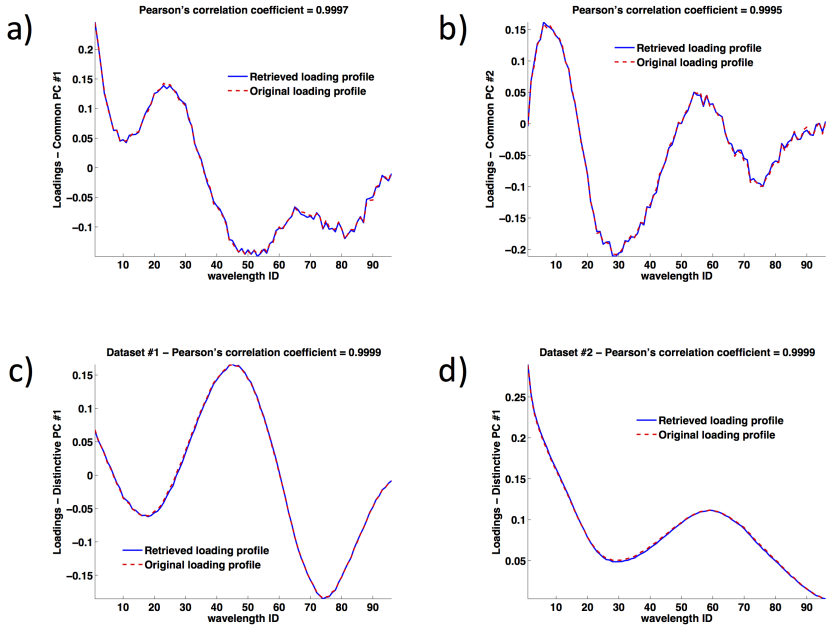

Figure 10.4 Simulated pseudo-spectral data: retrieved (blue solid lines) *vs* original (red dashed lines) loadings profiles. a) and b) refer to the first and the second common component shared by the two datasets, respectively, c) to the first distinctive factor of the first data block, and d) to the first distinctive factor of the second data block

Industrial batch process data

To test its feasibility for a real-world application, this novel strategy was also applied to the industrial data collected in the first of the two monitored reacting units. In this case, assuming that the distinctive variation of the second set of batches is mainly dependent on a possible deviation from NOC, the idea is to i) determine the number of common components shared by the two data blocks and supposedly accounting for the *in-control* variability of the process, ii) filter

[xii] Their statistical significance was assessed executing again the permutation-based effective rank estimation test.

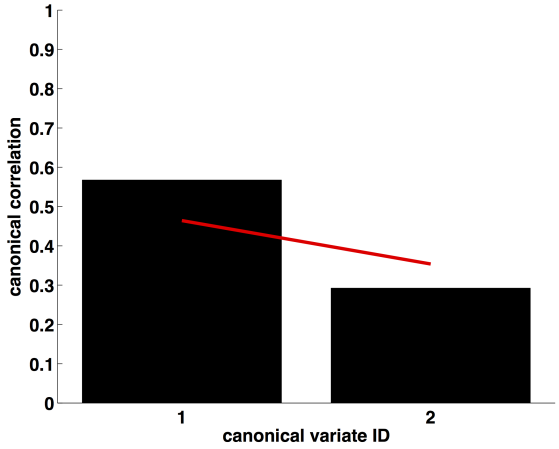

Figure 10.5 Industrial batch process data - Reacting unit #1: detection of the number of common components. The black bars represent the observed canonical correlations, while the red line connects the 99^{th} percentiles of their respective *null*-distributions

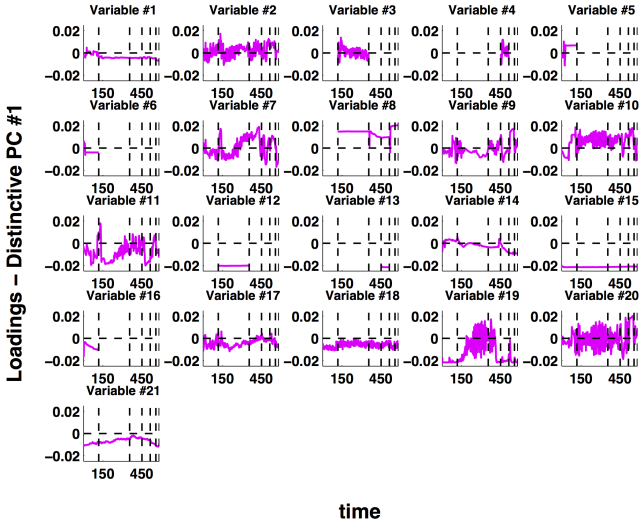

Figure 10.6 Industrial batch process data - Reacting unit #1: time profiles of the loadings of the first distinctive component of the second time period batch data block for the 21 measured variables. The vertical dashed lines separate the 6 stages of the industrial process. As not all the variables were active in these stages, part of such profiles is missing

them from the second dataset and iii) explore it after its deflation trying to unveil the root causes of such a deviation. As here the common dimension between the two data blocks is the column one, the analysis was carried out by transposing them after preprocessing. 2 and 5 of their factors were detected as statistically significant, respectively, but the presence of a single common component was highlighted by the CCA-based permutation test (see Figure 10.5). Since the attention is focused on the distinctive variability of the second dataset, Figure 10.6 shows the time evolution of the loadings of its first distinctive component (found to be statistically significant after executing again the permutation-based effective rank estimation algorithm) for the 21 measured variables. Among those presenting a consistent non-zero temporal trend and then a consistent contribution to this component, x_1, x_{12}, x_{13} and x_{14} generally exhibited both a higher variability and a higher average level in the later runs than in the *in-control* batches (see Figure 10.7). They were also pointed out as those most affected by a possible ongoing

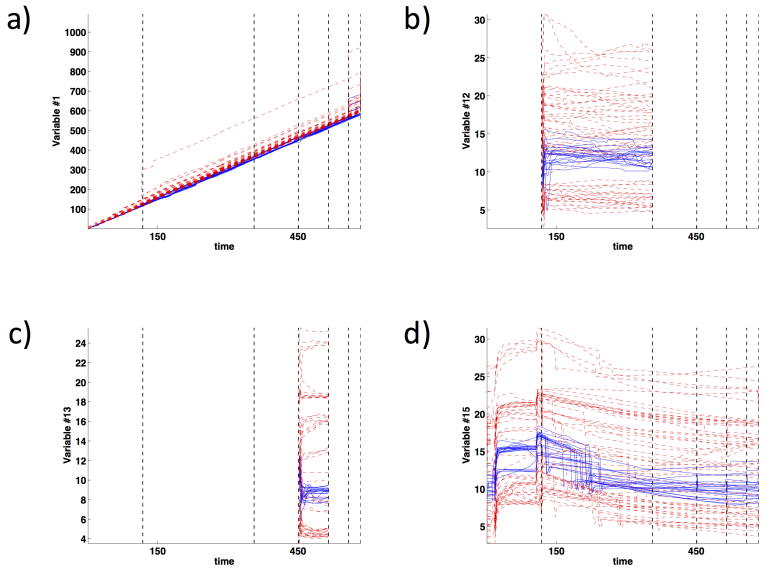

Figure 10.7 Industrial batch process data - Reacting unit #1: original time trajectories of a) x_1, b) x_{12}, c) x_{13} and d) x_{14} for the NOC (blue solid lines) and the faulty (red dashed lines) runs. The vertical dashed lines separate the 6 stages of the industrial process. As these variables were not active in every stage, part of their time trajectories is missing

failure in a preliminary investigation of such data.
But, what if more than two data blocks have to be dealt with? Imagine one wants to merge together the two sets of data recorded in the first reactor and the two sets of data recorded in the second reactor. Here, apart from evaluating the nature of the common and distinctive components within every manufacturing unit, it would

be also interesting, for example, to assess whether something is shared by the *out-of-control* variation related to each unit, plausibly captured by the distinctive factors of the faulty batch data blocks. Hence, these latter were subjected to the global procedure described at the beginning of this section, after the unit-specific common variability was removed, as Figure 10.8 indicates. Only a single com-

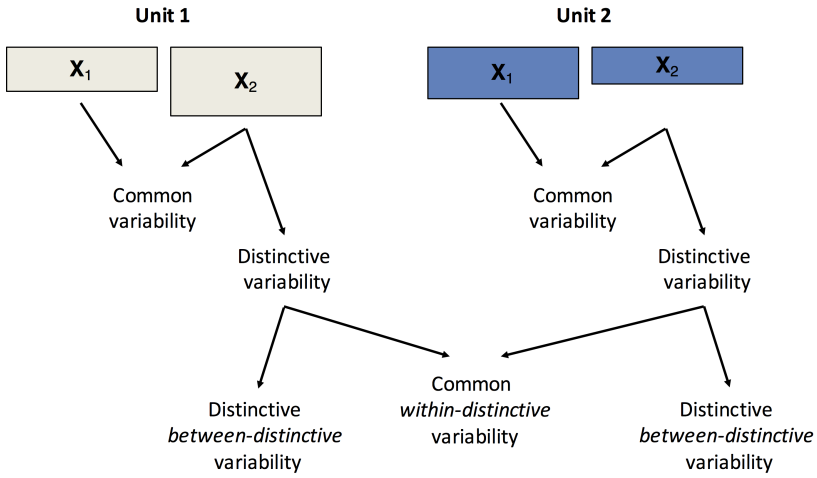

Figure 10.8 Hierarchical extension of the proposed modelling strategy. For the sake of clarity, the $\mathbf{X}_1$ and $\mathbf{X}_2$ matrices contain the evolution of the NOC and faulty batches manufactured in the two different reacting units, respectively

mon *within-distinctive* component was isolated by running again the CCA-based permutation test. As expected, owing to the fact that the common component retrieval is attained by a pseudo-O2PLS modelling technique, its loadings profiles (see Figures 10.9a and 10.9b) are very similar between units but not identical. They however highlight that x_1 (at least starting from the second process stage) features the most consistent contribution to this factor over time. Therefore, this variable could constitute the common problem affecting both the reacting units (see also Figure 10.10)[xiii]. For the sake of completeness, the loadings profiles of the first distinctive *between-distinctive* component (statistically significant) are also represented in Figure 10.11. For several patterns of variables (e.g. x_8, x_{12}, x_{13}, x_{14}, x_{15} and x_{16} for the first reactor, and x_8, x_{16}, x_{17}, x_{18} and x_{21} for the second reactor) a consistent non-zero temporal trend is observed. These variables might have been affected by a specific abnormal event occurring in the respective unit[xiv]. It is also important to notice that x_1 is characterised by practically

[xiii] x_1 in fact generally exhibited a higher level in the faulty runs manufactured in both the reacting units from the second process stage on.

[xiv] The original variable trajectories are not graphed because the outcomes refer to data blocks which were deflated before being analysed.

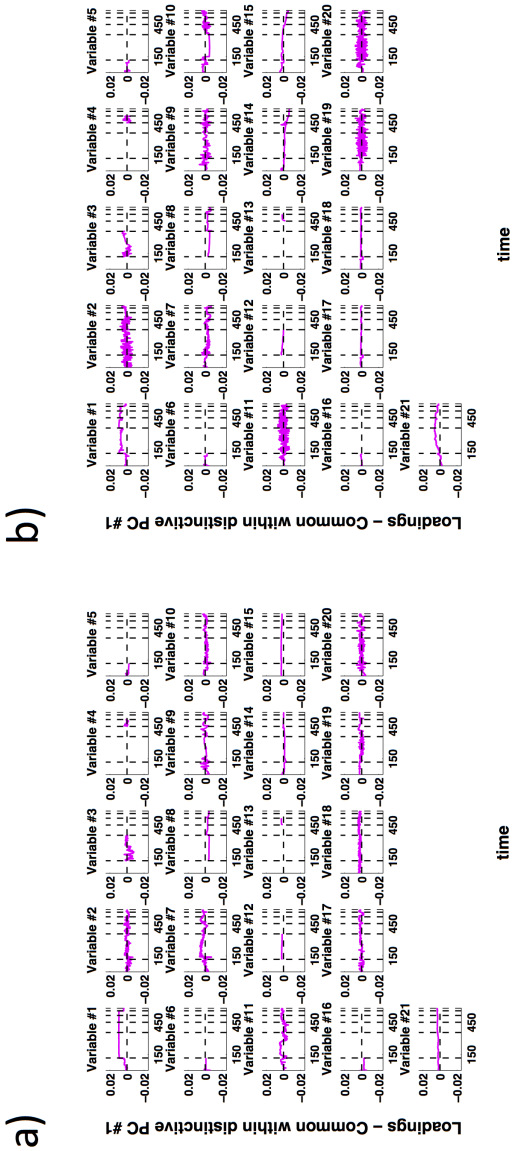

Figure 10.9 Industrial batch process data - a) Reacting unit #1 *vs* b) reacting unit #2: time profiles of the loadings of the first common *within-distinctive* component for the 21 measured variables. The vertical dashed lines separate the 6 stages of the industrial process. As not all the variables were active in these stages, part of such profiles is missing

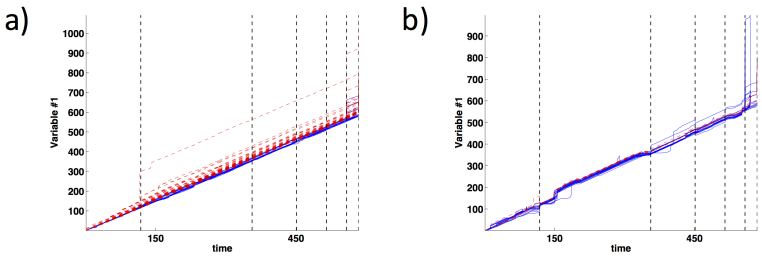

Figure 10.10 Industrial batch process data - a) Reacting unit #1 *vs* b) reacting unit #2: original time trajectories of x_1 for the NOC (blue solid lines) and the faulty (red dashed lines) runs. The vertical dashed lines separate the 6 stages of the industrial process

zero and non-consistent loadings from the second process stage on, which is in good agreement with what stated before for the unique common *within-distinctive* factor.

10.4.3 Modelling common and distinctive components in regression scenarios

Until now, the main focus of the discussion has been on identifying the common and distinctive phenomena affecting the intrinsic variability of two or more blocks of data. But, what if the relations between them and particular responses or properties of interest of the samples/subjects under study are to be modelled? In general, when coping with this type of problems, common and distinctive factors are first retrieved by e.g. one of the approaches compared in Section 10.4.1, and, afterwards, these properties of interest are regressed onto the resulting component scores. However, similarly to what happens with Principal Component Regression (PCR - see Section 14.3.2), it is not always guaranteed that such common and distinctive factors, extracted from the concerned datasets in the first of these two steps, are the most correlated to the registered responses. A novel algorithm named Joint-Y PLS (JYPLS) can help in overcoming this issue.

Joint-Y Partial Least Squares regression (JYPLS)

JYPLS [130] is a Non-linear Iterative PArtial Least Squares (NIPALS) algorithm variant (refer also to Section 14.3.1), initially developed for modelling the latent variable structure shared by two or more sets of data (say $\mathbf{X}$s) via a PLS-based regression against their corresponding responses (say $\mathbf{Y}$s). When only two different couples of data blocks are dealt with, namely $\mathbf{X}_1$-$\mathbf{Y}_1$ and $\mathbf{X}_2$-$\mathbf{Y}_2$, the mathematical

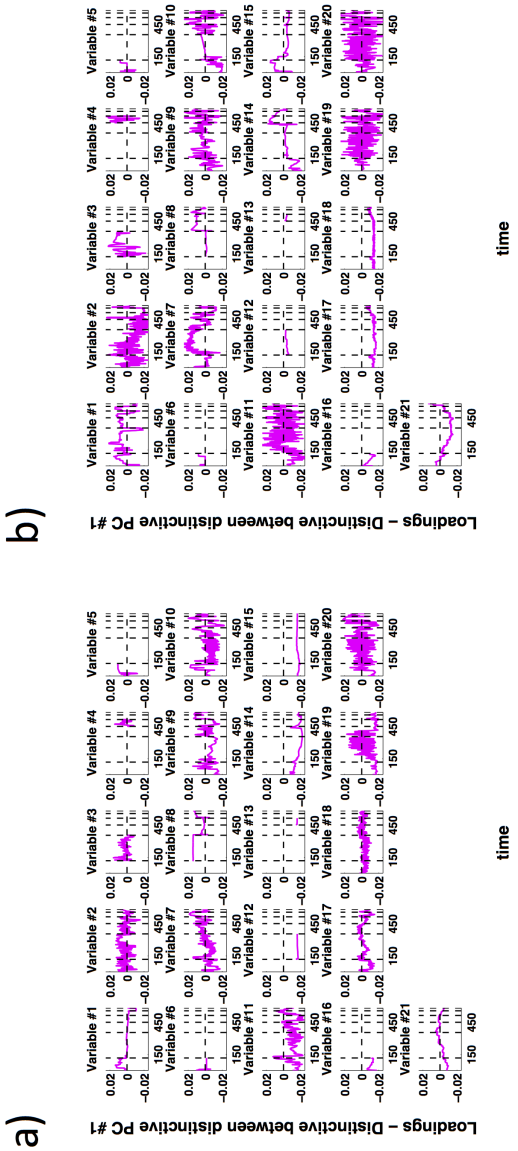

Figure 10.11 Industrial batch process data - a) Reacting unit #1 *vs* b) reacting unit #2: time profiles of the loadings of the first distinctive *between-distinctive* component for the 21 measured variables. The vertical dashed lines separate the 6 stages of the industrial process. As not all the variables were active in these stages, part of such profiles is missing

formulation of the JYPLS model is given by:

$$\mathbf{Y}_{\mathrm{J}} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{T}_1 \\ \mathbf{T}_2 \end{bmatrix} \mathbf{Q}_{\mathrm{J}}^{\mathrm{T}} + \mathbf{E}_{\mathbf{Y}_{\mathrm{J}}} \tag{10.27}$$

$$\mathbf{X}_1 = \mathbf{T}_1 \mathbf{P}_1^{\mathrm{T}} + \mathbf{E}_{\mathbf{X}_1} \tag{10.28}$$

$$\mathbf{X}_2 = \mathbf{T}_2 \mathbf{P}_2^{\mathrm{T}} + \mathbf{E}_{\mathbf{X}_2} \tag{10.29}$$

$$\mathbf{T}_1 = \mathbf{X}_1 \mathbf{W}_1^{*} \tag{10.30}$$

$$\mathbf{T}_2 = \mathbf{X}_2 \mathbf{W}_2^{*} \tag{10.31}$$

where $\mathbf{T}_1/\mathbf{T}_2$, $\mathbf{P}_1/\mathbf{P}_2$ and $\mathbf{W}_1^{*}/\mathbf{W}_2^{*}$ are the JYPLS scores, loadings and weighing matrices related to $\mathbf{X}_1/\mathbf{X}_2$, respectively. The originality of this approach concerns the fact that only one single set of loadings, $\mathbf{Q}_{\mathrm{J}}$, is derived for both $\mathbf{Y}_1$ and $\mathbf{Y}_2$, which defines a combined plane mapped by the $\mathbf{Y}_{\mathrm{J}}$ joint array (see Equation 10.27). The key consequence of that will be illustrated in the next subsection. Notice that the only requirement to run JYPLS is that $\mathbf{Y}_1$ and $\mathbf{Y}_2$ share the same variables. No restriction is imposed on the dimensions of the $\mathbf{X}$-blocks, which can be then concatenated object-wise, variable-wise or both object-wise and variable-wise (when the number of handled datasets is larger than 2).

Gene expression data

To get some insights into its potential for the kind of applications described in this chapter, JYPLS was applied to the gene expression data and compared to SCA, DISCO-SCA, Adapted GSVD, ECO-POWER, CCA, and O2PLS in both the object-wise and the variable-wise case. Tables 10.3 and 10.4 list the Pearson's correlation coefficients estimated between every single component scores vector resulting from the application of all the aforementioned techniques and the plasma hemagglutination-inhibition antibody titer. As one can easily see, JYPLS permits to retrieve factors exhibiting i) a gradually lower degree of correlation with the immunological response as they are sequentially extracted and ii) similar Pearson's correlation coefficients with it between datasets, which may be considered an evidence of the fact that such an algorithmic procedure can model the shared covariation of the $\mathbf{X}$-blocks with respect to a definite number of y-variables. However, classifying a JYPLS component as common or distinctive based on these outcomes is not straightforward and depends on how the concepts of *commonness* and *distinctiveness* are conceived in a regression scenario like this. Should a factor be deemed common if it accounts for the same events or phenomena regardless of how they influence the variability of the $\mathbf{Y}$-matrices? Or should it be deemed common if it carries information about important common sources of covariation between predictors and properties of interest? These questions still constitute an open debate in the field, and clearly regard not only JYPLS, but the largest part of the plethora of existing multi-set data analysis methods.

Table 10.3 Gene expression data - Object-wise case: Pearson's correlation coefficients estimated between every single component scores vector resulting from the application of SCA, DISCO-SCA, Adapted GSVD, ECO-POWER, CCA, O2PLS and JYPLS and the plasma hemagglutination-inhibition antibody titer. As an example to interpret the notation, C1 refers to the first extracted factor. Model complexity was chosen as to be the same as in [110]. Notice that the same solution is attained by SCA and DISCO-SCA

	SCA*	DISCO-SCA*	Adapted GSVD*	ECO-POWER*	CCA		O2PLS		JYPLS	
					Day 3	Day 7	Day 3	Day 7	Day 3	Day 7
C1	-0.3951	-0.3951	0.4044	0.1498	0.3758	0.4175	0.0518	-0.1372	0.9018	0.8684
C2	-0.4492	-0.4492	0.3526	0.1624	0.4294	-0.3112	0.3376	0.2856	0.4003	0.4612
C3	0.2917	0.2917	0.1231	0.4773	0.3272	-0.3188	0.1068	0.1378	0.1438	0.1709
C4	0.0990	0.0990	0.3378	-0.3746	0.0947	0.1278	0.1114	0.1937	0.0727	0.0586
C5	0.1181	0.1181	-0.3081	-0.2266	0.1070	0.1513	0.5339	-0.0082	0.0225	0.0203

*SCA, DISCO-SCA, Adapted GSVD and ECO-POWER led to a single component scores matrix which is shared by the two different data blocks.

Table 10.4 Gene expression data - Variable-wise case: Pearson's correlation coefficients estimated between every single component scores vector resulting from the application of SCA, DISCO-SCA, Adapted GSVD and JYPLS and the plasma hemagglutination-inhibition antibody titer. As an example to interpret the notation, C1 refers to the first extracted factor. Model complexity was chosen as to be the same as in [110]. Notice ECO-POWER, CCA and O2PLS were not included in such a comparison as they cannot deal with variable-wise linked data structures

	SCA		DISCO-SCA		Adapted GSVD		JYPLS	
	TIIV	LAIV	TIIV	LAIV	TIIV	LAIV	TIIV	LAIV
C1	0.2249	0.0490	-0.2021	0.0645	0.5934	-0.1865	0.9018	0.9441
C2	-0.0056	0.0290	-0.0150	0.0261	0.1846	-0.0733	0.4003	0.2966
C3	0.6363	-0.0696	-0.6209	0.1997	0.3198	-0.2856	0.1438	0.1334
C4	0.1604	-0.2258	-0.3344	-0.1023	-0.2448	-0.1703	0.0727	0.0505
C5	0.4080	-0.3717	0.1138	0.0717	0.0899	-0.3676	0.0225	0.0176
C6	-0.0800	-0.2790	-0.2516	0.4880	-0.2018	-0.0643	0.0088	0.0061

10.5 Conclusions

In this chapter, the inadequacy of the VAF index as a criterion for the selection of the common and distinctive components underlying two different sets of measurements was demonstrated (at least in an object-wise case) by a comprehensive comparison among SCA, DISCO-SCA, Adapted GSVD, ECO-POWER, CCA and O2PLS. The principal limitation affecting most of these methods, i.e. the absence of a univocal procedure to determine the number of factors shared by the data blocks under study, was tentatively overcome by combining the principles of permutation testing and CCA. Although preliminary, the displayed results appear to be encouraging, especially in the light of the conclusions drawn after the application of the pseudo-O2PLS methodology to the simulated and real industrial data. In addition, JYPLS was found to represent a feasible and flexible approach for modelling common sources of variability strictly related to specific responses of interest, applicable also in more complex scenarios where the various handled datasets may not share the same row and/or column space.

Chapter 11

Calibration transfer between near-infrared spectrometers

In this chapter two novel methods to perform calibration transfer between near-infrared (NIR) spectrometers are proposed and compared with classical well-established techniques such as Maximum Likelihood Principal Component Analysis (MLPCA) and Piecewise Direct Standardisation (PDS) in two real case studies.

Part of the content of this chapter has been included in:

1. Folch-Fortuny, A.[i], Vitale, R.[i], de Noord, O. & Ferrer, A. Calibration transfer between NIR spectrometers: new proposals and a comparative study. *J. Chemometr* **31**, e2874 (2017).

[i]These authors had equal contributions

11.1 Introduction

A scenario in which modelling the common variability shared by multiple datasets could be useful is when existing calibration models are to be applied to spectral measurements recorded by a new instrument and/or in different environmental conditions. In such circumstances, in fact, classical tools like PLS may suffer from severe practical limitations because even very similar spectrometers generally exhibit variations in their responses, that may seriously jeopardise this so-called *calibration transfer*. Several methods have been proposed to overcome this issue and avoid at the same time an expensive and time-consuming full recalibration, using the newly acquired spectra. One of these approaches consists in updating the calibration model by merging measurements collected by both the first and the second spectrometer. However, that is commonly effective only when the two sets of spectral profiles are rather similar [131].
For the sake of simplicity, suppose now that a certain number of samples has been analysed by the primary instrument and a subgroup of these samples characterised also by the secondary one[ii]. Among all the other strategies proposed in the scientific literature for dealing with such a scenario, Piecewise Direct Standardisation (PDS) [132] has been unanimously pointed out as *a reference for novel techniques* because of its local and multivariate nature [131, 133–135]. PDS basically transforms the spectra recorded by the secondary instrument so that its spectral response matches the one of the primary instrument. This allows any calibration model, built on the data resulting from the primary spectrometer, to be used for the analysis of those acquired by the secondary apparatus.
From a slightly different perspective, the transfer of a calibration model from a spectrometer to another can be looked at as a missing data imputation problem. In this circumstance, all the information contained in the available primary and secondary spectra can be exploited to entirely reconstruct the profiles associated to those samples that were not analysed by the secondary instrument. These profiles can be then utilised for fitting an improved predictive model, suitable for the assessment of future incoming recordings. Maximum Likelihood Principal Component Analysis (MLPCA) [136] has been the first computational methodology to be applied for solving the calibration transfer issue in this fashion.
In this chapter, two innovative strategies to perform calibration transfer based on Trimmed Scores Regression (TSR) [137] and, for its particular algorithmic structure (delineated in Section 10.4.3), JYPLS [130] are proposed. Specifically, their performance will be assessed and compared to that of MLPCA and PDS in two real case studies, in which the same set of samples was characterised by two different near-infrared (NIR) spectrometers.

[ii] Alternatively, two distinct or partially distinct sets of samples may be analysed by the two spectrometers. However, this contingency will not be contemplated here.

11.2 Methods

Let $\mathbf{X}_\mathrm{a}$ ($N_\mathrm{a} \times J_\mathrm{a}$) and $\mathbf{X}_\mathrm{b}$ ($N_\mathrm{b} \times J_\mathrm{b}$) be the matrices containing the spectral profiles collected by the primary and the secondary spectrometer, respectively. Mind that here the N_b samples characterised by the secondary instrument were also analysed by the primary one.

11.2.1 Piecewise Direct Standardisation (PDS)

PDS executes a series of local linear transformations of the spectra collected by the secondary instrument to subsequently allow the calibration model built for the primary spectrometer to be exploited for prediction purposes. Specifically, at each j-th wavelength, the absorbance values registered by the primary instrument ($\mathbf{x}_{\mathrm{a},j}$) are related by Principal Component Regression (PCR, see Section 14.3.2) to a specific spectral window of the profiles of the same samples collected by the secondary spectrometer ($\mathbf{X}_{\mathrm{b},j}$):

$$\mathbf{x}_{\mathrm{a},j} = \mathbf{1}b_j + \mathbf{X}_{\mathrm{b},j}\mathbf{f}_j \tag{11.1}$$

where $\mathbf{1}$ represents a vector of ones of appropriate dimensions. Incoming secondary instrument data are then adjusted through the estimated standardisation parameters, $\mathbf{f}_j$ and b_j. Here, PDS was applied so that all the principal components, whose eigenvalue (divided by the first one) was found to be larger than 0.0001, were included in each local regression model. On the other hand, the spectral window width was automatically optimised within the modelling procedure (see Section 11.3).

11.2.2 Maximum Likelihood Principal Component Analysis (MLPCA)

The adaptation of the MLPCA algorithm to model building with missing data is an iterative procedure based on an imputation alternately carried out by rows and columns. It has been proven [138–140] that for both PCA model building [137] (i.e. when a PCA model has to be fitted on incomplete data) and model exploitation [141] (i.e. when a PCA model is fitted on complete data and exploited to predict the scores of new incomplete observations) the MLPCA object-wise imputation step is equivalent to performing a projection to the model plane.
Let $\mathbf{X}$ be a matrix of dimensions $N \times J$. When data are missing in its n-th row, $\mathbf{x}_n^\mathrm{T}$, $\mathbf{X}$ can be rearranged so that such missing values are located in its last, say R, columns. Thus,

$$\mathbf{x}_n^\mathrm{T} = [\mathbf{x}_n^{*\mathrm{T}}\ \mathbf{x}_n^{\#\mathrm{T}}] \tag{11.2}$$

and

$$\mathbf{X} = [\mathbf{X}^*\ \mathbf{X}^\#] \tag{11.3}$$

with $^*/^\#$ connoting the available/missing entries in $\mathbf{x}_n^{\mathrm{T}}$, respectively.
Based on the following partition of the Singular Value Decomposition (SVD) of $\mathbf{X}$:

$$[\mathbf{X}^* \ \mathbf{X}^\#] = \mathbf{UD}[\mathbf{P}^{*\mathrm{T}} \ \mathbf{P}^{\#\mathrm{T}}] + \mathbf{E} \tag{11.4}$$

the MLPCA algorithm imputes $\mathbf{x}_n^{\#\mathrm{T}}$ as:

$$\hat{\mathbf{x}}_n^\# = \mathbf{P}^\#(\mathbf{P}^{*\mathrm{T}}\mathbf{P}^*)^{-1}\mathbf{P}^{*\mathrm{T}}\mathbf{x}_n^* \tag{11.5}$$

Concerning the missing data in the j-th column of $\mathbf{X}$, $\mathbf{x}_j$, the data partition is performed according to the available and missing observations of the corresponding variable. This way:

$$\begin{bmatrix}\mathbf{X}^* \\ \mathbf{X}^\#\end{bmatrix} = \begin{bmatrix}\mathbf{U}^* \\ \mathbf{U}^\#\end{bmatrix}\mathbf{DP}^{\mathrm{T}} + \mathbf{E} \tag{11.6}$$

and:

$$\hat{\mathbf{x}}_j^\# = \mathbf{U}^\#(\mathbf{U}^{*\mathrm{T}}\mathbf{U}^*)^{-1}\mathbf{U}^{*\mathrm{T}}\mathbf{x}_j^* \tag{11.7}$$

The imputation is iteratively executed until the reconstruction of the available values stabilises.
The MLPCA algorithm is comprehensively detailed in [136]. Besides, a thorough assessment of the use of MLPCA for missing data imputation is provided in [140].
To transfer a calibration model using this methodology, the complete set of N_a primary instrument spectra, $\mathbf{X}_\mathrm{a}$, has to be concatenated with the N_b spectra collected by the secondary spectrometer, $\mathbf{X}_\mathrm{b}$. An augmented data matrix $\mathbf{X}_\mathrm{ab}$ ($N_\mathrm{a} \times (J_\mathrm{a} + J_\mathrm{b})$) is then constructed, where the unrecorded secondary instrument profiles are missing (see Figure 11.1, *Imputation* box). In other words, if the sample associated to the n-th row of $\mathbf{X}_\mathrm{ab}$ has not been analysed by the secondary spectrometer, the partition in Equation 11.2 applies: $\mathbf{x}_n^{*\mathrm{T}}$ and $\mathbf{x}_n^{\#\mathrm{T}}$ denote its available primary and its missing secondary instrument spectrum, respectively.
$\mathbf{X}_\mathrm{ab}$ is finally subjected to MLPCA.

11.2.3 Trimmed Scores Regression (TSR)

TSR is an iterative missing data imputation method, originally proposed for PCA model exploitation [142, 143]. Afterwards, it was adapted to the more general framework of PCA model building in the presence of missing data [137].
TSR imputes the missing values in a dataset by carrying out a regression using the scores of its available entries. Considering the partition of $\mathbf{X}$ in Equation 11.3 and its decomposition in Equation 11.4, the missing elements in $\mathbf{x}_n^{\mathrm{T}}$ are estimated as:

$$\hat{\mathbf{x}}_n^\# = \mathbf{S}^{\#*}\mathbf{P}^*(\mathbf{P}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{P}^*)^{-1}\mathbf{P}^{*\mathrm{T}}\mathbf{x}_n^* \tag{11.8}$$

where $\mathbf{S}^{**} = \frac{\mathbf{X}^{*\mathrm{T}}\mathbf{X}^*}{N-1}$ and $\mathbf{S}^{\#*} = \frac{\mathbf{X}^{\#\mathrm{T}}\mathbf{X}^*}{N-1}$.
The imputation is iteratively executed until the reconstruction of the missing values stabilises.

A complete survey on TSR can be found in [137]. A Graphical User Interface for TSR-based missing data imputation, the Missing Data Imputation Toolbox for MATLAB [144], is also available at `http://mseg.webs.upv.es/Software_e.html`.
Calibration transfer by TSR is achieved in the same way as for MLPCA, that is building the augmented array $\mathbf{X}_{\mathrm{ab}}$ and submitting it to the computational procedure described before (see Figure 11.1, *Imputation* box).

11.2.4 JYPLS-based approaches

Two possible JYPLS-based computational strategies were implemented, namely JYPLS-noinv and JYPLS-inv.

- JYPLS-noinv - Let $\mathbf{Y}_{\mathrm{a}}$ and $\mathbf{Y}_{\mathrm{b}}$ be the matrices including the measured response variables associated to the samples analysed by the primary and the secondary spectrometer, respectively[iii]. Once a JYPLS model is built as in Equations 10.27-10.31 (see Figure 11.1, *Model transfer* box), the same responses for new samples characterised by the secondary instrument, $\mathbf{Y}_{\mathrm{b,new}}$, can be predicted from their spectral profiles, $\mathbf{X}_{\mathrm{b,new}}$, as (see Figure 11.1, *External validation II* box):

$$\mathbf{Y}_{\mathrm{b,new}} = \mathbf{X}_{\mathrm{b,new}}\mathbf{W}^{*}_{\mathrm{b}}\mathbf{Q}^{\mathrm{T}}_{\mathrm{J}} \tag{11.9}$$

- JYPLS-inv - On the other hand, as for TSR, spectra unrecorded by the secondary instrument can be reconstructed, provided they are associated to samples analysed by the primary one and whose response values ($\mathbf{Y}_{\mathrm{b,unrecorded}}$) are then present in $\mathbf{Y}_{\mathrm{a}}$, by the following inversion (see Figure 11.1, *Model inversion* box):

$$\mathbf{X}_{\mathrm{b,unrecorded}} = \mathbf{Y}_{\mathrm{b,unrecorded}}(\mathbf{Q}_{\mathrm{J}}\mathbf{Q}^{\mathrm{T}}_{\mathrm{J}})^{\dagger}\mathbf{Q}_{\mathrm{J}}\mathbf{P}^{\mathrm{T}}_{\mathrm{b}} \tag{11.10}$$

  where † denotes the Moore-Penrose pseudo-inverse [145]. Such imputed spectra, fused to $\mathbf{X}_{\mathrm{b}}$, are then exploited for fitting an improved PLS predictive model (see Figure 11.1, *Model calibration* box), suitable for the assessment of future incoming data (see Figure 11.1, *External validation I* box).

[iii] Here the rows of $\mathbf{Y}_{\mathrm{b}}$ are also contained in $\mathbf{Y}_{\mathrm{a}}$, as they relate to samples analysed by both the secondary and the primary instrument. This is, however, not a necessary requirement to apply JYPLS.

11.3 Modelling procedure

The comparative study among PDS, MLPCA, TSR and JYPLS was carried out according to a 5-step procedure (see Figure 11.1):

1. both the primary and secondary instrument data blocks were randomly split into calibration (2 thirds of the original spectra) and validation (1 third of the original spectra) sets (see Figure 11.1, *Data split* box). 20 split rounds were conducted to prevent spurious results from being yielded;

2. secondary instrument calibration subsets of increasing size were generated to determine the minimum number of measurements needed to be collected for accomplishing an accurate calibration transfer. The samples belonging to each one of these subsets were selected by the Kennard-Stone (KS) algorithm [146], probably the most popular computational procedure for data-representative object identification [147, 148] (see Figure 11.1, *Sample selection* box)[iv];

3. the four methods under study were then applied in the following fashion:

   - when TSR, MLPCA and JYPLS-inv, which are missing data imputation-based approaches (see Section 11.2), were handled, the secondary instrument calibration spectra left out of each subset were consecutively reconstructed as described before (see Figure 11.1, *Imputation*, *Model transfer* and *Model inversion* boxes). They were then merged with those belonging to the respective calibration subset to fit a new PLS regression model (see Figure 11.1, *Model calibration* box);

   - by JYPLS-noinv, predictive JYPLS models were constructed fusing both the primary spectrometer calibration set and the different secondary spectrometer calibration subsets as detailed in Section 11.2.4 (see Figure 11.1, *Model transfer* box);

   - the PDS standardisation was executed relating the secondary instrument calibration subsets of spectra to their corresponding profiles registered by the primary spectrometer (see Figure 11.1, *Parameter fitting* and *Standardisation* boxes). Notice that the properties of interest of new samples are thereafter predicted from their corrected spectra by a PLS regression model built on the whole primary instrument calibration set (see Figure 11.1, *External validation III* box).

   For the various strategies, the parameters to be optimised (dimensionality of the imputation model, dimensionality of the regression model, PDS spectral

[iv] Here, KS was run on the scores of a PLS model resulting from the primary spectrometer calibration data.

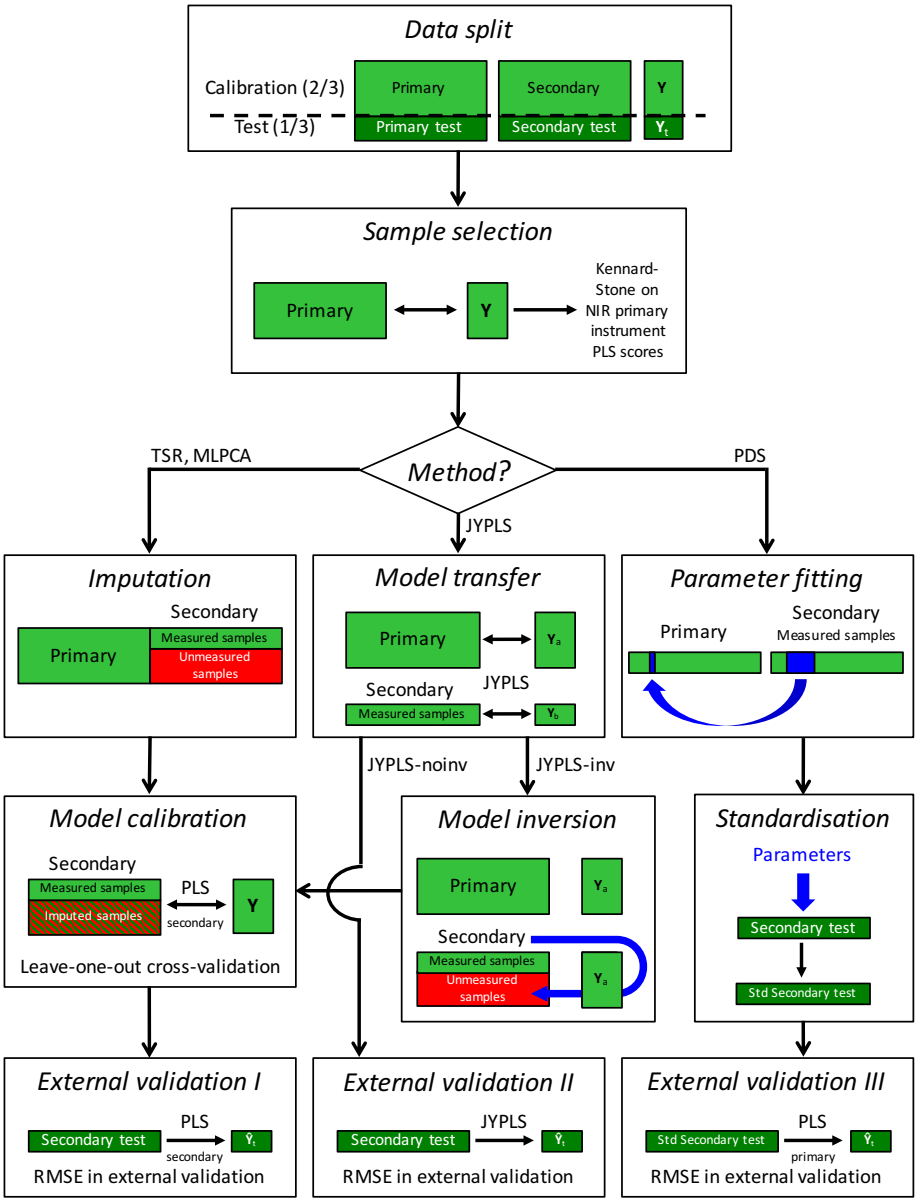

Figure 11.1 Flow-chart of the comparative study. *Std* stands for *standardised*. ^ refers to predicted values. Notice that part of the whole secondary instrument calibration set is assumed to be unmeasured when addressing the calibration transfer

window width) were adjusted in order to minimise the average Root Mean Square Error in Cross-Validation (RMSECV), defined as:

$$\text{RMSECV} = \frac{\sum_{k=1}^{K} \sqrt{\frac{\sum_{n=1}^{N}(y_{n,k}-\hat{y}_{n,k})^2}{N}}}{K} \tag{11.11}$$

where $y_{n,k}$ represents the actual value of the k-th response variable associated to the n-th calibration sample and $\hat{y}_{n,k}$ is its final prediction[v].

4. The performance of PDS, MLPCA, TSR and JYPLS was finally assessed according to the average Root Mean Square Error in Prediction (RMSEP):

$$\text{RMSEP} = \frac{\sum_{k=1}^{K} \sqrt{\frac{\sum_{n'=1}^{N'}(y_{n',k}-\hat{y}_{n',k})^2}{N'}}}{K} \tag{11.12}$$

where $y_{n',k}$ represents the actual value of the k-th response variable associated to the n'-th validation sample and $\hat{y}_{n',k}$ is its final prediction, while N' equals the total number of spectra included in the validation set[v] (see Figure 11.1, *External validation I*, *External validation II* and *External validation III* boxes);

5. Statistically significant differences among the considered approaches were finally evaluated via a mixed-effect ANOVA, taking into account four factors: calibration transfer technique, size of the secondary instrument calibration subset and their interaction (fixed-effect factors) as well as split round (random-effect factor, nested to the size of the secondary instrument calibration subset). If any of their effects was found to be statistically significant, the 95% Least Significance Difference (LSD) intervals were calculated to assess which methods were different from the others.

11.4 Datasets

The first dataset analysed here contains 60 spectra measured on 30 pseudo-gasoline samples within 800 and 1600 nm (401 scanned wavelengths, 30 spectra per instrument). Heptane, iso-octane, toluene, xylene and decane concentration are the properties of interest to be predicted.
The second dataset relates to 80 corn samples, whose spectral profiles were registered within 1100 and 2498 nm (700 scanned wavelengths for a total number of 160 spectra, 80 per each spectrometer). The response variables are moisture, oil, protein and starch content.
Both of them have been widely used to compare calibration transfer methods [149–

[v]The reported RMSECV and RMSEP values concern auto-scaled response variables owing to the differences in their original units of measurements.

151]. The gasoline dataset is included in the PLS_Toolbox for MATLAB [101], the corn dataset can be downloaded from `http://www.eigenvector.com/data`. Prior to any analysis the gasoline and the corn spectra were simply mean-centred, as also proposed in [151], while the respective response variables were auto-scaled.

11.5 Results

11.5.1 Gasoline dataset

For each spectrometer, 20 pseudo-gasoline samples were assigned to the calibration set and the remaining 10 to the validation set. 15 secondary instrument calibration subsets, containing from 5 to 19 spectral profiles, were generated.

Missing data imputation

As TSR, JYPLS-inv and MLPCA rely on a preliminary missing data imputation step, it is worth assessing the accuracy of the reconstruction of the unmeasured spectra, since they will be then resorted to for building the final predictive PLS model.
Figure 11.2 permits to compare original and imputed profiles for one of the 20 split rounds. Their correlation and χ^2 distance are represented in Figures 11.2A, 11.2D, 11.2G and 11.2B, 11.2E, 11.2H, respectively. Each line refers to the best model selected for one specific secondary instrument calibration subset. High correlations (larger than 0.9999) and low χ^2 distance values (smaller than 0.001) were yielded by TSR and JYPLS-inv, while several issues appeared when dealing with MLPCA. First, it often suffered from convergence problems (as already pointed out by Feudale *et al.* [131]), which dramatically slowed the computational procedure down. Consequently, the reconstructed spectra were found to be substantially different from their actual profiles (see Figure 11.2I). For these reasons, MLPCA was not taken into account in the final study.

Comparative study

Figure 11.3A allows the performance of the different considered calibration transfer techniques to be examined. Each point in the plot represents the average RMSEP value, derived from the 10-sample external validation set, over the 20 split rounds (for 5- to 19-sample secondary instrument calibration subsets). As expected, for all the approaches, the higher the size of the secondary instrument calibration subset, the lower the RMSEP.
As the effect of all the factors included in the ANOVA model was found to be statistically significant (p-value< 0.05), the 95% LSD intervals were calculated to

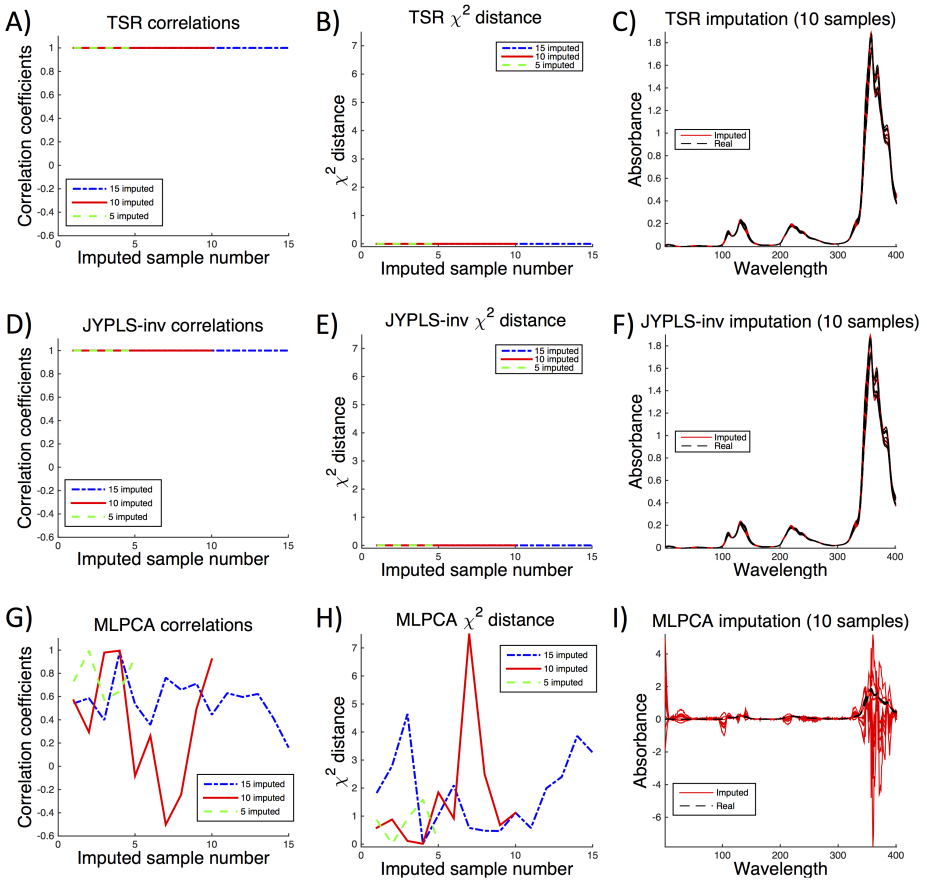

Figure 11.2 Gasoline dataset - A), D) and G) show the correlation coefficients between the original spectra and those imputed by TSR, JYPLS-inv and MLPCA, respectively. B), E) and H) represent their corresponding χ^2 distance values. The dotted-dashed blue lines refer to the case in which the secondary instrument calibration subset was constituted by 5 samples and 15 spectra were imputed. The solid red lines refer to the case in which the secondary instrument calibration subset was constituted by 10 samples and 10 spectra were imputed. The dashed green lines refer to the case in which the secondary instrument calibration subset was constituted by 15 samples and 5 spectra were imputed. C), F) and I) display the original (dashed black lines) and reconstructed (solid red lines) profiles in the second of these three cases

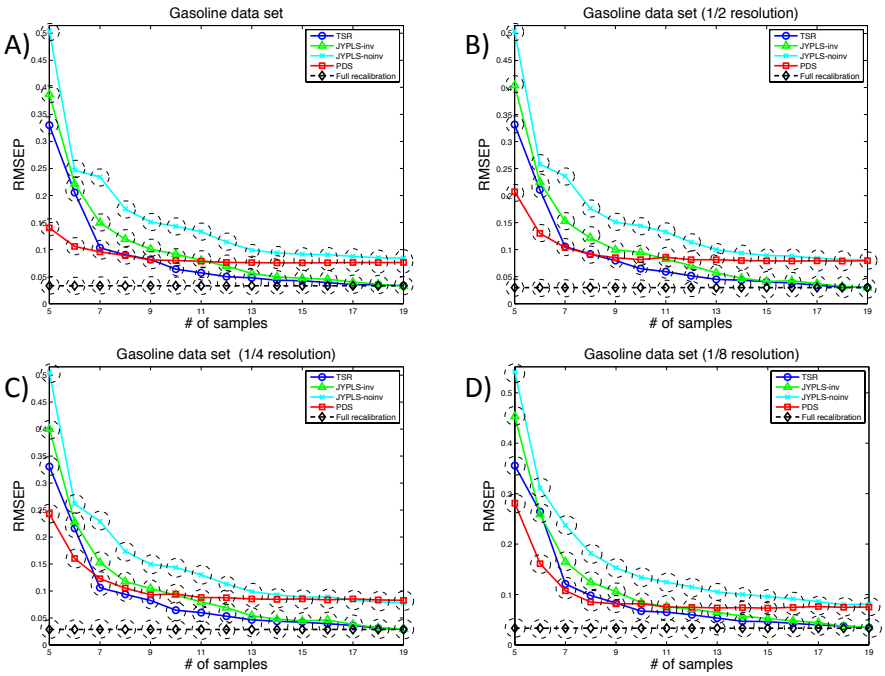

Figure 11.3 Gasoline dataset - RMSEP values obtained for the different sizes of the secondary spectrometer calibration subset with A) the same spectral resolution for both instruments, B) $\frac{1}{2}$, C) $\frac{1}{4}$ and D) $\frac{1}{8}$ of the primary instrument spectral resolution for the secondary spectrometer. Dashed ellipses mark the statistically significant differences among groups of methods ($p-$value < 0.05)

point out existing differences among methods. For the sake of an easy visualisation, dashed-line ellipses are used in Figure 11.3A to highlight them. Specifically, methods embraced by the same ellipse show no statistical difference. On the other hand, methods embraced by different ellipses are statistically different.
Clearly, PDS guaranteed the lowest RMSEP when the secondary instrument calibration subset consisted of 5/6 samples. No statistically significant differences were detected between PDS and TSR for a 7-sample secondary instrument calibration subset and between PDS and JYPLS-inv when a 10-sample secondary instrument calibration subset was concerned. From 10 samples onwards, the RMSEP stabilised around 0.09 for PDS, but it continuously decreased for TSR and JYPLS-inv, reaching values of approximately 0.05-0.06 (for 12-13 to 19 samples). The straight line in Figure 11.3A indicates the RMSEP value obtained when a full recalibration was performed, i.e. when the whole set of 20 secondary instrument calibration samples was used to build a new predictive model. Although it cannot be directly compared to the outcomes resulting from PDS, TSR, JYPLS-inv and

JYPLS-noinv, it simplifies the determination of the number of spectra needed to be collected by the secondary spectrometer for generating no statistically significant differences with respect to full recalibration. TSR required 12, while JYPLS-inv 13. On the other hand, PDS and JYPLS-noinv always showed a statistically worse performance than full recalibration.

Instruments with different resolutions

A common situation faced by practitioners in industrial environments is transferring calibration models between instruments with diverse spectral resolution. This problem has already been addressed in [152], where the authors propose a novel PLS-based approach resulting in similar outcomes as PDS.
Figures 11.3B-11.3D show the results of the whole analysis, conducted gradually reducing the spectral resolution of the secondary instrument. The performance of all the methods was basically the same as in the full resolution case described in Section 11.5.1. However, for PDS, a slight continuous decrease in the quality of the calibration transfer can be noticed.

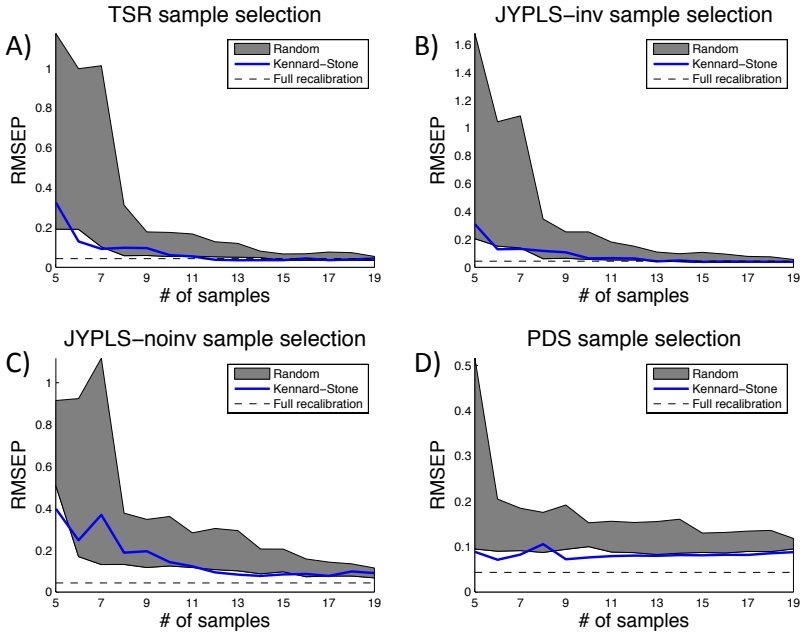

Figure 11.4 Gasoline dataset - Effect of the Kennard-Stone algorithm-based sample selection on the performance of the calibration transfer methods under study

Sample selection effect

The effect of the secondary spectrometer calibration subset sample selection is here assessed. 10 random picks were performed for one particular split round and the final RMSEP values were then compared to those obtained by preliminarily running KS. It is clear from Figure 11.4 that KS generally returned a lower RMSEP, very close to that achievable through a full recalibration. It then enabled a better calibration transfer plausibly due to the fact that it permits to choose a subset of samples, which is statistically representative of the experimental domain of the spectral data collected by the primary instrument. This is not necessarily the case when such a selection is carried out at random.

11.5.2 Corn dataset

For each spectrometer, 54 corn samples were assigned to the calibration set and the remaining 26 to the validation set. 10 secondary instrument calibration subsets, containing from 5 to 50 spectral profiles (5-spectra intervals), were generated.

Missing data imputation

Figure 11.5 allows original and imputed corn sample spectral profiles to be compared for one of the 20 split rounds. TSR preserved its reconstruction ability and MLPCA suffered from the same problems observed for the gasoline dataset. Regarding JYPLS-inv, the correlation coefficients/χ^2 distance values were rather high/low, but the imputed spectra showed less variability than the real ones (see e.g. Figure 11.5F). This happened because the large difference in the offset of these latter is scarcely related to the properties to be predicted. As the imputation here involves the joint-$\mathbf{Y}$ loadings matrix, $\mathbf{Q}_{\mathrm{J}}$, such a difference is not transferred to the reconstructed spectra (see Equation 10.27). Thus, one can think of JYPLS-inv as filtering spectral variations, which are uninteresting from a predictive point of view.

Comparative study

Existing differences among methods were investigated as in the previous case study (also here the effect of all the ANOVA factors was found to be statistically significant). Figure 11.6A displays the results of the comparative study conducted on the corn dataset. Again, PDS showed a better performance for small secondary instrument calibration subsets (5-10 samples). For 20-25 samples, there were no statistical differences among PDS, TSR, JYPLS-inv and JYPLS-noinv. Finally, from 30 samples onwards, the proposed approaches outperformed PDS, similarly

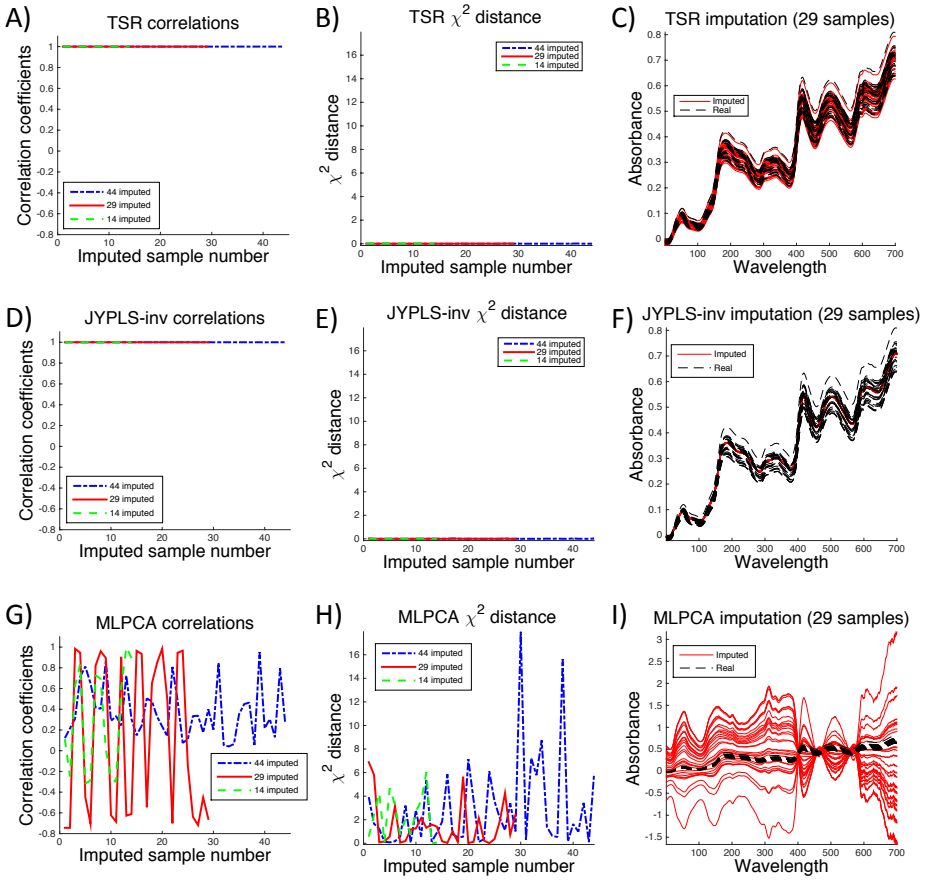

Figure 11.5 Corn dataset - A), D) and G) show the correlation coefficients between the original spectra and those imputed by TSR, JYPLS-inv and MLPCA, respectively. B), E) and H) represent their corresponding χ^2 distance values. The dotted-dashed blue lines refer to the case in which the secondary instrument calibration subset was constituted by 10 samples and 44 spectra were imputed. The solid red lines refer to the case in which the secondary instrument calibration subset was constituted by 25 samples and 29 spectra were imputed. The dashed green lines refer to the case in which the secondary instrument calibration subset was constituted by 40 samples and 14 spectra were imputed. C), F) and I) display the original (dashed black lines) and reconstructed (solid red lines) profiles in the second of these three cases

as for the gasoline dataset. From 40 samples onwards, TSR, JYPLS-inv and JYPLS-noinv exhibited no significant differences with respect to full recalibration.

Instruments with different resolutions

In this case, the reduction of the spectral resolution of the secondary instrument strongly affected the quality of the PDS-based calibration transfer. In fact, when it was decreased to $\frac{1}{8}$, even for small secondary calibration subsets, the performance of PDS was found to be statistically worse than that of the other compared approaches. On the other hand, TSR, JYPLS-inv and JYPLS-noinv proved to be quite robust towards such a change (see Figures 11.6B-11.6D).

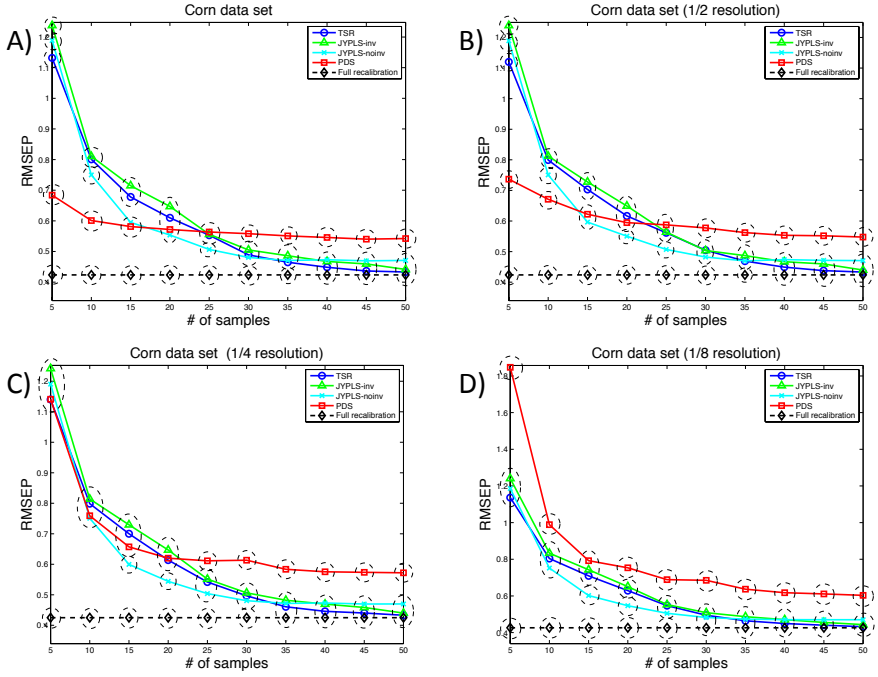

Figure 11.6 Corn dataset - RMSEP values obtained for the different sizes of the secondary spectrometer calibration subset with A) the same spectral resolution for both instruments, B) $\frac{1}{2}$, C) $\frac{1}{4}$ and D) $\frac{1}{8}$ of the primary instrument spectral resolution for the secondary spectrometer. Dashed ellipses mark the statistically significant differences among groups of methods ($p-$value < 0.05)

Sample selection effect

The effect of the secondary spectrometer calibration subset sample selection can be evaluated by looking at Figure 11.7. Here, especially when the number of spectra included in such a subset was not particularly high, some random picks permitted to obtain better results in terms of RMSEP. However, when it increased, the KS-based selection enabled more accurate predictions than random ordering.

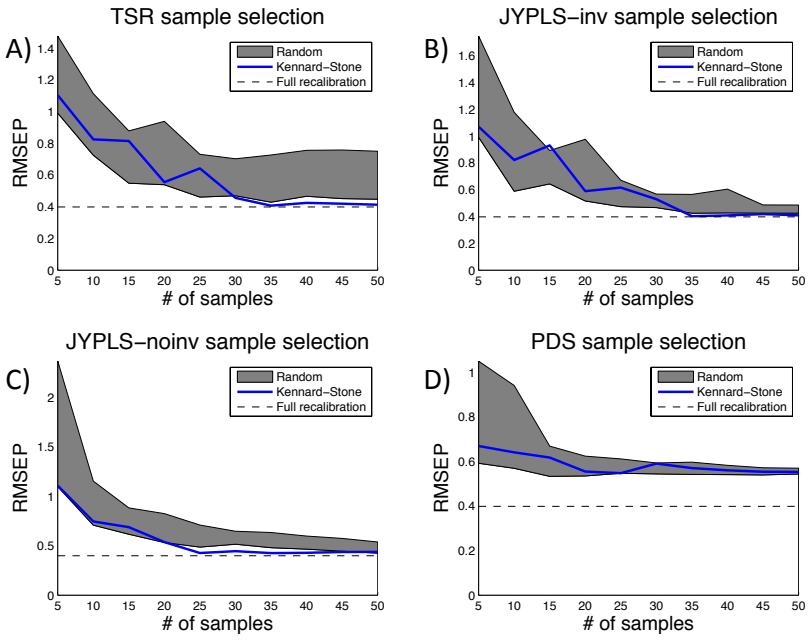

Figure 11.7 Corn dataset - Effect of the Kennard-Stone algorithm-based sample selection on the performance of the calibration transfer methods under study

11.6 Discussion

When carrying out a calibration transfer with a very small secondary instrument calibration subset (around 5-10 samples), PDS showed better (or equal) results than TSR and JYPLS-inv, but its performance was always worse than for a full recalibration. Nevertheless, when the size of the secondary instrument calibration subset was enlarged, TSR and JYPLS-inv clearly outmatched PDS, achieving a similar error rate as for full recalibration. No evident conclusions can be drawn regarding the performance of JYPLS-noinv, as the quality of its outcomes changed

depending on the analysed dataset. Still, it was found to be, in general, as reliable as TSR and JYPLS-inv when the corn dataset was dealt with, but statistically worse in the gasoline case study.
The number of spectra to be collected by the secondary spectrometer for a precise calibration transfer was also assessed. TSR and JYPLS-inv yielded very close results to full recalibration even if only about the 60% of the available spectra were included in the corresponding calibration subset. Conversely, PDS never reached such a degree of accuracy.
PDS was strongly affected by the reduction of the spectral resolution of the secondary instrument when the corn dataset was concerned, while TSR, JYPLS-inv and JYPLS-noinv seemed not to suffer from the same issue.
In terms of unmeasured spectra reconstruction, TSR resulted in the best performance. In contrast, JYPLS-inv acted as a sort of filter removing the variations in the spectra not related to the properties to be predicted, and consequently producing deviations from their original shape.
Moreover, it is worth saying that both JYPLS-inv and JYPLS-noinv are the unique strategies, which could be resorted to when distinct or partially distinct sets of samples[vi] are analysed by the two spectrometers.
Finally, it was shown that selecting the samples of the secondary instrument calibration subset using KS generally permitted to achieve better outcomes, regardless the exploited calibration transfer technique.

11.7 Conclusions

Two novel methods to perform calibration transfer between NIR spectrometers, based on TSR and JYPLS, respectively, were here proposed. They outmatched PDS and guaranteed a very similar performance to that resulting from a full recalibration when only about the 60% of the spectra collected by the secondary instrument was available. Both the approaches also showed a sufficient robustness towards the reduction of its spectral resolution. In addition, TSR allowed unmeasured spectra to be accurately imputed, while the inversion of the JYPLS models yielded reconstructed spectral profiles filtered of all the variation not of interest from a predictive point of view.

[vi] Provided that the values of their properties of interest are known.

Part V

On the on-the fly processing and modelling of continuous high-dimensional data streams

Chapter 12

The On-The-Fly Processing tool

In this chapter a novel software system for rational handling of multi-channel measurements streaming in real time, the On-The-Fly Processing (OTFP) tool, is presented.

Part of the content of this chapter has been included in:

1. Vitale, R., Zhyrova, A., Fortuna, J., de Noord, O., Ferrer, A. & Martens, H. On-The-Fly Processing of continuous high-dimensional data streams. *Chemometr. Intell. Lab.* **161**, 118-129 (2017).

12.1 Introduction

As mentioned in Section 2.1, a massive *data tsunami*, due to the fast development and dissemination of novel measurement technologies, is currently taking place in many fields of applied science. In spite of that:

- the human ability to grasp content of interest from data remains fairly constant, and data simplification is therefore desirable for interpretative purposes. Here, one possible solution could be the removal of irrelevant variables among the whole set of collected ones. Nevertheless, for most applications their identification is not straightforward which makes such a simplification risky and complicated;
- despite Moore's first law [153], which predicts a continuous exponential increase for both computer processing speed and storage capacity along time, it is estimated that in the near future they will not be sufficient for coping with this ongoing *data explosion*. For instance, *Internet of Things* threatens to flood both communication channels and the users' cognitive capacity with overwhelming torrents of repetitive, more or less redundant data;
- traditional computing systems are generally not capable of performing analytics on constantly streaming data, typical of today's world of multimedia communication [154].

In a scenario like this, if it were possible to simultaneously compress and model high-dimensional measurement series as they flow from e.g. an analytical platform and without significant loss of useful information content, their storage, transfer, retrieval, visualisation and interpretation would be radically eased. The *On-The-Fly Processing* (OTFP) tool is here proposed to achieve this goal.

12.1.1 Data compression strategies

Data compression plays a central role in telecommunications and many other scientific and technological branches of interest [155]. According to the nature and features of the algorithmic procedure through which it is performed, it can be defined as either *lossless* or *lossy*. Lossless methods utilise statistical distribution properties and simple patterns in the data for compression, converting the inputs into compressed bit series[i].
Lossy compression techniques - e.g. the various dedicated versions of JPEG and MPEG methods used for digital image, video and sound compression - approximate the main, perceptible variations in the input data by local *ad hoc* patterns,

[i] Most of the lossless compression approaches, such as standard file *zipping*, recode the original input by using shorter bit sequences for *probable* (e.g. often encountered) data and larger ones for *improbable* (e.g. rare) data.

filtering out less perceptible variation types and noise. Lossy approaches are commonly much more efficient (in terms of compression rate) than lossless ones, like *algebraic* zipping, but allow the original input to be only roughly restored. Moreover, when set to compress too much, they not only cause loss of valid information (resulting in e.g. image blurring or loss of high-frequency sound), but can also introduce undesired decoding artefacts (e.g. visible block effects or audible errors). Whether lossless or lossy compression methods are used, the compressed data are represented by *per se* meaningless streams that cannot be directly used for quantitative calculations, mathematical modelling or graphical representation.
The novelty of the OTFP is represented by the fact that a hitherto under-utilised source of redundancy (the intercorrelation usually evolving in multi-channel data streams) is mathematically modelled to prevent significant loss of useful systematic information carried by the original measurements. Based on the model's automatically estimated parameters the data stream may be interpreted and utilised for prediction, forecasting and fault detection in the compressed state. The idea behind this strategy was recently outlined in [156]. Here, more algorithmic details will be given and its applicability to different types of high-dimensional data streams demonstrated.
Conceptually, the OTFP system may be motivated by the following thought experiment: assume that a space probe should be constructed and sent out to explore - for the first time - the unknown geological properties of the hidden back side of a remote planet, using a multi-wavelength camera. Prior to the launch, scarce knowledge about this planet is available to design the ideal instrument, and after the probe has landed, it is too late to change anything. Which wavelength should be chosen, and how should the imaging data be transmitted back to Earth? Some individual wavelengths distinguishing between already known, earthly rock types might be included. But possible geological *surprises* should also be taken into account. Therefore, it is decided to equip the probe camera with a wide spectral range detector, capable of measuring e.g. 1000 different wavelength channels. However, the limited communication bandwidth then becomes a problem: the probe cannot transmit all those measurements for every point in time and space. What would be the best way to send spectral data back to Earth? Perhaps, could that be automatically settled on-the-fly by the space probe's computer itself, based on what its camera measures? The on-board computer could be programmed to discover, compress and transmit the essence of all the recorded images, in a continuous learning-and-communicating process that never sends the same information twice. But how to quantify this compact spectral essence comprehensively? To understand the unknown geological landscape, a reliable approximation of the spectral profile of every pixel in every image, with as many spectral and spatial details and as few artefacts as possible, is needed. A lossless multivariate spectral preprocessing followed by a continuously developing bilinear compression/classification model could deliver a compact summary of the sequence of hyperspectral image data, which would yield maximal insight here on Earth from the limited quan-

tity of received data. The first three application examples described below will illustrate this, albeit in more mundane settings.

12.1.2 Subspace compression

The OTFP is based on evolving bilinear subspace modelling. The software automatically detects systematic patterns of covariation in the data, say $\mathbf{X}$ $(N \times J)$, and use these to model the data mathematically. For their intrinsic properties, subspace projection and dimensionality reduction techniques based on bilinear models, e.g. PCA, constitute one of the possible ways to compress and approximate a certain set of data, removing simultaneously both statistical redundancy and uninformative noise. In fact, recalling Section 2.2.1, by such techniques it is possible to reduce the J-dimensional space of the original descriptors to an A-dimensional subspace, onto which all the N objects under study can be projected and represented as new points. As generally $A < J$, this projection can be regarded as a compression operation, whose efficiency is related to the ratio $\frac{A}{J}$. However, choosing the A components of this subspace is probably one of the most critical point when deriving the PCA approximation of a set of data. Some of these A dimensions may sometimes be defined according to prior knowledge of the investigated system. For instance, the number of known chemical constituents of mixtures characterised by spectroscopic methods might be resorted to for this purpose. However, in cases like this, also more or less *unexpected* constituents and/or physical phenomena may affect the performed measurements, generating new patterns of variation and thus new subspace dimensions which need to be retained for a proper data approximation and interpretation. Therefore, at least to a certain extent, the identification of the new basis vectors associated to these unforeseen sources of variability has to be carried out through a preliminary exploratory analysis of the available empirical records.
If a continuous data stream is dealt with and N rapidly grows over time, correctly determining new possible subspace dimensions is even more complex: new, unexpected patterns of covariation may spring up in the information flow. Therefore, in such situations, it becomes crucial to automatically recognise when the set of initial basis vectors needs to be reestimated and extended, and to address this task in a statistically valid and computationally efficient way.

12.1.3 PCA as a multivariate series expansion of the underlying data generation mechanism

The bilinear PCA model can be thought of as a Taylor expansion of the function $\boldsymbol{f}$ defining how the measurement descriptors are jointly related to their common structure [157–159]. For example, for each of the J aforementioned input channels, one can envision a local linear approximation of the underlying (unknown) causal phenomena driving their evolution. Mathematical summary modelling of such J

local approximations (achieved by PCA or related methods e.g. PLS, Independent Component Analysis - ICA - or non-linear versions of these) can detect and display their main patterns of covariation. This can unveil the underlying causalities of the data generation mechanism.

12.1.4 Algorithms for PCA decomposition

As remarked in Section 2.2.1, the PCA approximation of a certain dataset can be efficiently attained by a variety of algorithms, among which the most widespread and popular one is certainly Singular Value Decomposition (SVD) [4]. However, if N is very high, standard SVD may be very demanding in terms of both CPU load and memory requirements. In the last few years, several variants of classical SVD have been proposed for performing PCA on very large matrices without entirely keeping them in the computer memory (*out-of-core*) [160–164]. *Out-of-core* PCA can be carried out by different procedures e.g.:

- a $J \times 1$ cumulative sum vector and a $J \times J$ cross-product matrix may be accumulated over time, combined and used for eigen-analysis of the covariance in $\mathbf{X}$, which yields the PCA loadings. That is appropriate for parallelisation but then the scores for the past time or space samples are lost;
- if also J is very high (e.g. thousands of wavelengths in an hyperspectral camera monitoring a certain scene or process), the $J \times J$ covariance matrix cannot be easily handled. Evolving moving-window/recursive PCA approaches may then be used instead, working on the most recent subset of observations. But that gives problems when comparing past and present records, e.g. in graphical scores plots.

The OTFP tool is actually proposed in the attempt of overcoming all these limitations. The purpose of the OTFP is to identify systematic trends and patterns in high-dimensional data flows, compress these and display them graphically, while automatically detecting outliers - key points to be addressed when continuous quantitative data streams are dealt with [165]. Based on what detailed before, it rather represents an extension of classical bilinear PCA, specifically developed for processing multi-channel records as soon as they are collected. It extracts patterns of covariation between the input variables by comparing previous and new observations and thereby identifying and modelling new variation phenomena, without needing large amounts of data or parameters to be retained in memory. Given for instance a continuously growing stream of high-dimensional data, the OTFP modelling system gradually develops a minimal bilinear summary model of the input data stream. For each point in space and/or time, already established components are quantified as spatiotemporal scores by projection of their multi-channel loadings. Moreover, new, unmodelled patterns of covariation are automatically

detected, refined and quantified in terms of additional spatiotemporal scores and multi-channel loadings, then appended to the OTFP model. Hence, unlike bilinear moving-window solutions, this dynamic model extension is executed so that the system preserves the quantitative connection between all past and present records. Yet it does not need to retain all past inputs or bilinear scores in memory - for long-lasting processes the memory usage would grow prohibitively high. Besides, the OTFP system does not require to hold and update a huge $J \times J$ covariance matrix - for many applications that would also be of a prohibitive size. Instead, it repeatedly stores the necessary scores and loadings, avoiding an excessive memory consumption during the process.

12.2 System overview

The present OTFP algorithm (schematically outlined in Figure 12.1) is characterised by three fundamental aspects: i) its self-learning and ii) adaptive nature and iii) its stabilising modelling principles. It allows massive amounts of data collected along time to be compressed and modelled with minimal loss of significant information content. The algorithm is initiated with the preliminary choice of a typical input vector, $\mathbf{m}$, and the best guess of which weights to give to the different input channels for balancing their variances, $\mathbf{c}$. Furthermore, a set of predefined component loadings, $\mathbf{P}$, derived for instance from an initial exploratory investigation of the system under study and representing systematic variation patterns expected to affect the incoming data stream, may or may not be supplied. Then, various system parameters such as the desired modelling fidelity (e.g. the fraction of data variance the OTFP model has to capture, also known as amount of *explained* data variance) need to be specified. As the multi-channel data starts to flow it may deliver a more or less continuous stream of individual J-dimensional input records, e.g. a set of measurements collected by the same set of J sensors during the evolution of an industrial process. Alternatively, it may deliver a sequence of input data blocks, each containing N_g records ($g = 1, 2, \ldots, G$) and the same set of J channels, e.g. N_g spectral profiles, constituted of absorbance values measured at J wavelengths and associated to individual pixels of an hyperspectral image. Such records are then treated by the following procedure:

1. The J-dimensional data are (optionally) submitted to a lossless preprocessing, linearising the responses and balancing the variable variances to ease the subsequent bilinear modelling. This step is domain-specific and the way it is executed has to be set *a priori*. For this reason, the best pretreatment strategy should always be selected based on both the nature of the handled instrumental equipment and technical knowledge;

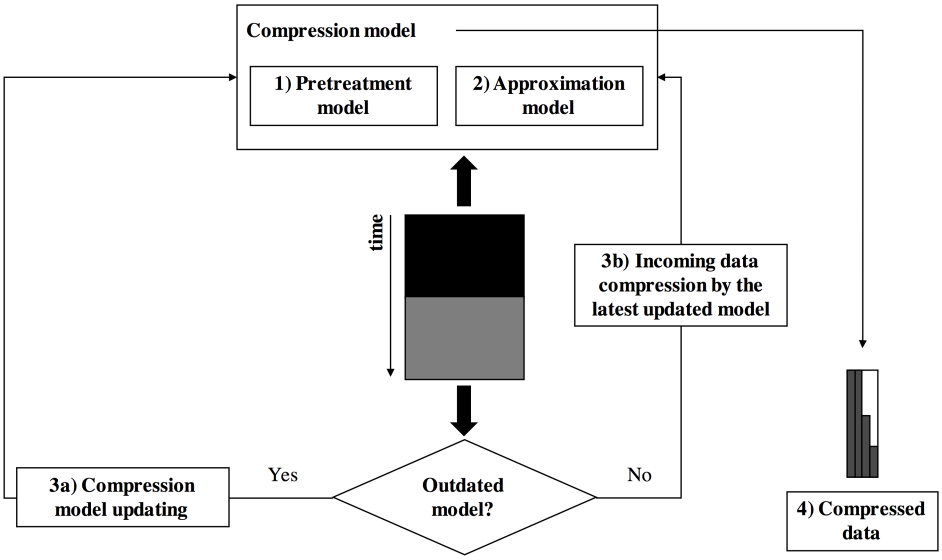

Figure 12.1 Schematic representation of the OTFP algorithm: a first set of data (black block) is input to 1) a pretreatment and 2) a PCA-based dimensionality reduction stage. As new measurements are recorded (grey block), they can be either 3a) exploited for the reparametrisation of the compression model, if it is found to be outdated, or 3b) just approximated by its latest version. 4) Bilinear approximation loadings and preprocessing parameters are saved by keeping track of how they have been initially defined and/or changed during model updating. The time series of bilinear approximation scores are more or less continuously stored and deleted from memory to subsequently process new input data

2. The preprocessed data are projected onto the subspace defined by the bilinear loadings, $\mathbf{P}$, already established at this point in time, to estimate the scores for the respective components;

3. The residuals left in the data after the projection on known components are input to a bilinear (here PCA-like) modelling stage to detect new unknown components and isolate outliers. If new components are found, they are quantified in terms of new scores and loadings. Thus, the statistically redundant J original variables are replaced by a smaller number (A) of principal components (PCs). The number of such components determines the degree of fidelity initially specified by the user. The algorithm automatically learns to identify and quantify all the systematic types of covariation in the data stream as it flows, while most of the random measurement errors and individual or irrelevant outliers are removed, provided the latter do not constitute a new pattern of variation. This compressed representation is suitable

for graphical interpretation and quantitative use, and from it the pretreated data can be reconstructed at any time;

4. At regular intervals, the OTFP model may be refined and reorthogonalised in a linear updating stage;

5. The pretreatment information associated to the different blocks is stored as output together with the approximation model scores and loadings.

As specified before, the OTFP algorithm detects all the systematic types of covariation in the data stream - be it from the flow of observed objects (expected information) or from the measuring process itself (unexpected information, anyway needed for reliable interpretation and quantitative use of the data). Phenomena considered irrelevant during preprocessing, as well as individual outliers discovered by the OTFP algorithm, are noted and then excluded from the self-modelling process. So is much of the random, independent measurement error, since it does not represent a systematic pattern of covariation.
At any time, the systematic part of the data stream can be reconstructed from the data model, e.g. for visualisation. But this reconstruction is not mandatory; the compressed data model parameters, representing the known and/or unknown types of systematic phenomena in the data stream, are themselves suitable for efficient storage and transmission, human graphical interpretation and applied quantitative usage.
The aforementioned steps will now be described.

12.2.1 Input

The *ever-lasting* raw data stream, $\mathbf{X}$, divided into a sequence of blocks, $\mathbf{X}_g$ ($N_g \times J$, $g = 1, 2, \ldots, G$), is submitted to the optional preprocessing stage, which includes a linearisation and a signal-conditioning step, and then to the OTFP self-modelling. The number of observations encompassed by these blocks can be freely set by the user. Unless the preprocessing parameters and the $\mathbf{m}$ and $\mathbf{c}$ vectors are established *a priori*, the start of the modelling process (i.e. for $\mathbf{X}_1$) requires sufficient observations to enable a precise and relevant initialisation of them.

Linearisation

The linearisation of the input data in $\mathbf{X}_g$ is domain-specific. For instance, non-linearities in light spectroscopy data may be reduced by transformation of the recorded light intensity, I, at each wavelength, first to transmittance, $T = \frac{I}{I_0}$, where I_0 represent the blank signal, and then to absorbance, $A = \log \frac{1}{T}$, to better conform to Beer's law of linear chemical responses.
Another aspect of the linearisation is to convert non-additive variation types (e.g.

multiplicative light scattering in absorbance spectra, motions in RGB or hyperspectral videos, *etc.*) into additive signal contributions or preprocessing parameters. For instance, multi-channel pretreatments such as Standard Normal Variate (SNV) [166], Multiplicative Scatter Correction (MSC) [167, 168] and Extended Multiplicative Signal Correction (EMSC) [169] can reparametrise multiplicative effects. Two-domain IDLE modelling [156] can convert confusing motion effects into nicely additive motion flow fields. Domain transforms, like Fast Fourier Transform (FFT) and wavelet analysis can change data locally from time to frequency domain. Such a more or less lossless, model-based preprocessing may produce additional parameters, which may be highly informative and must be stored for later data reconstruction.

Weighing the variables for better signal conditioning

In general, for an optimal data approximation, the J originally measured descriptors in $\mathbf{X}_g$ are approximately centred, e.g. by subtraction of their mean values estimated from the data flowed up to the current step. They may then be weighed to ensure a better balance among their variances so that:

$$\mathbf{X}_{g,p} = (\mathbf{X}_g - \mathbf{1}\mathbf{m}^{\mathrm{T}}) \circ \mathbf{1}\mathbf{c}^{\mathrm{T}} \tag{12.1}$$

where $\mathbf{1}$ ($N_g \times 1$) is a vector of ones, $\mathbf{m}$ ($J \times 1$) and $\mathbf{c}$ ($J \times 1$) contain the model centre and the input weighing factors (these weighing factors could e.g. be defined as the inverse of the standard deviation values of the J recorded variables at the current step), respectively, while $\circ$ identifies the element-wise (Hadamard) product. The same pretreatment is applied to all consecutive data blocks until $\mathbf{m}$ and/or $\mathbf{c}$ are readjusted as part of the model updating operation (see below).

12.2.2 Fit to already established model subspace

The linearised, centred and weighed records in $\mathbf{X}_{g,p}$ are now projected onto the already established loadings $\mathbf{P}$ (if they exist at the current step), according to the linear structure model:

$$\mathbf{X}_{g,p} = \mathbf{T}_{g,p}\mathbf{P}^{\mathrm{T}} + \mathbf{E}_{g,p} \tag{12.2}$$

Clearly, the frequency at which such a projection step is carried out depends on the number of observations in $\mathbf{X}_{g,p}$, that is, as said, a user-defined parameter[ii].

[ii] In the case studies described in Section 12.4, the projection frequency was found to affect only the computational time of the algorithmic procedure (as it increases, the number of data blocks the OTFP has to consecutively handle becomes larger) but not its final outcomes.

12.2.3 Bilinear model expansion

In the present implementation of the OTFP, once calculated, the residual vectors in $\mathbf{E}_{g,p}$ are examined: if they are deemed small enough to be considered uninteresting noise, the respective original records are simply discarded and their scores gathered in $\mathbf{T}_{g,p}$. If this is not the case such residual vectors are introduced into a temporary repository to check whether they represent a new systematic trend in the data stream or not. At regular intervals or when its size or variance exceeds a specific user-defined threshold, this temporary repository is used for the estimation of a new set of loadings and scores. If their respective factors are found to explain a sufficiently high amount of the repository variation[iii], these new scores and loadings are appended to those of the already established PCs in $\mathbf{P}$ and $\mathbf{T}_{g,p}$, respectively. Otherwise, if leverage analysis of the new scores points out that only scattered objects have contributed to them, these are dismissed as incidental outliers, their scores are stored, and the original model is retained.
Since the size of the entire scores matrix can become very large as the information flows, the scores are saved to the local disk at regular intervals and then deleted from memory along with $\mathbf{X}_{g,p}$ and $\mathbf{E}_{g,p}$[iv].

12.2.4 Model updating

Whenever necessary (e.g. if the model is characterised by a relatively high bias), preprocessing parameters, loadings and scores for both old and new observations are readjusted to ensure PCA-like orthogonality and thus a more efficient compression of the data. For such an updating, the OTFP does not need to recall the whole array of scores stored on the local disk, but directly operates on two summary indices of such an array, which are kept in memory in place of it (namely its column-wise cumulative sum vector and its cross-product matrix). The dimensionality of the reestimated model is automatically established according to the user's desired optimisation criterion. Here, for simplicity, the percentage of data variance that has to be captured is used. This allows the original information stream to be retrieved with a predetermined reconstruction accuracy. Other criteria, e.g. the statistical significance of the eigenvalues associated to the single components [94], may additionally be exploited (see also Part III).

[iii] The scores for these new PCs are - implicitly - defined to be zero for all the previous observations.

[iv] In the case studies described in Section 12.4, the storage of the scores on the local disk proved not to constitute a limiting step for the execution of the OTFP algorithm.

12.3 Datasets

To evaluate the potential of the proposed method, 4 different sets of time series data were compressed and modelled as reported before and reconstructed afterwards:

- High-speed multi-channel monitoring of a chemical reaction: 4329 multi-channel Vis-NIR spectra were measured in-line between 400 and 1098 nm (350 wavelengths) via a NIRS 6500 spectrophotometer, equipped with a fibre-optic bundle, during several replicates of the self-oscillating Belousov-Zabhotinsky (B-Z) reaction [170]. The final matrix had dimensions 4329×350. This example is intended to illustrate a new way to handle more or less continuous, high-dimensional measurements of a complex dynamic system not yet fully understood from a scientific point of view;

- Detailed remote characterisation of a set of related, complex objects: three 245×210-sized hyperspectral NIR images of three oranges were registered within the near-infrared spectral range 898-1690 nm (247 wavelengths) by a XEVA-FPA-1.7-320 line-scanner camera (Xenics, Belgium). To enable their handling, such three-way arrays needed to be unfolded into a unique matrix, so that a single pixel spectrum was contained in each one of its rows. After background removal, its dimensions were 72365×247. This example was chosen to illustrate how non-invasive bio-spectroscopy can reveal hidden aspects of related complex biological samples;

- Airborne environmental surveillance: an hyperspectral image was recorded by a push-broom device installed on an Unmanned Aerial Vehicle (DroneSpex, Norut AS - University Centre in Svalbard - Norwegian University of Science and Technology, Norway [171]), flying over Faial (Azorean Islands, Portugal). At each accumulation step, the optical sensor collected the absorbance values at 450 wavelengths in the visible light range between 420 and 640 nm for a strip of 245 pixels. A total number of 1000 consecutive snapshots were captured, which led to a three-way array of dimensions $1000 \times 245 \times 450$. Also in this case, it was unfolded into a 245000×450 matrix. This example is intended to show how data from a modern environmental monitoring instrument, a drone, can be automatically compressed for efficient storage and transmission and interpreted in the compressed state;

- Traditional industrial process analysis: 76 engineering variables, mainly including temperatures, pressures and flow rates, were recorded at hourly intervals to follow the evolution of a continuous industrial process. The complete data structure had dimensions 14561×76. This example illustrates the application of the OTFP to records measured over time by a relatively small set of conventional sensors.

12.4 Results and discussion

The power of the OTFP approach and the quality of the initial data retrieval were assessed in all the case studies at hand according to the following indices:

- A: number of extracted PCs;
- $\mathrm{EV_{raw}}$: percentage of explained raw data variance;
- $\mathrm{EV_p}$: percentage of explained preprocessed data variance;
- RMSRE: Root Mean Square Reconstruction Error defined as $\sqrt{\frac{\sum_{n=1}^{N}\sum_{j=1}^{J}(x_{n,j}-\hat{x}_{n,j})^2}{NJ}}$, where $x_{n,j}$ is the (n,j)-th element of $\mathbf{X}$ and $\hat{x}_{n,j}$ refers to its respective reconstructed value;
- t_c: compression time expressed in seconds[v];
- CR: compression ratio[vi].

$\mathrm{EV_{raw}}$, $\mathrm{EV_p}$ and RMSRE are strictly related to the OTFP approximation accuracy degree, while A, t_c and CR can be considered measures of computational speed and efficiency.

12.4.1 High-speed multi-channel monitoring of the Belousov-Zhabotinsky reaction

Table 12.1 lists the values of the aforementioned parameters related to the Vis-NIR data compression. The initialisation measurements were centred and weighed ($\mathbf{c} = \frac{1}{\mathbf{m}+0.05}$) after baseline correction[vii]. The model centre vector, $\mathbf{m}$, was updated at regular intervals as new spectroscopic details were encountered in the process, while the variable weighing vector, $\mathbf{c}$, was kept constant for simplicity.
In order to more clearly appreciate the performance of the OTFP, 3 uncompressed and reconstructed spectra associated to different reaction stages are displayed in Figure 12.2. The full approximation model is sketched in Figure 12.3, in terms of final model mean (Figure 12.3a), chosen weighing factors (Figure 12.3b), de-weighed and scaled loadings (Figure 12.3c) and lack-of-fit residuals (Figure 12.3d). This example has shown that the OTFP automatically discovered and quantified various systematic variation patterns in the complex, ill understood B-Z reaction. At our chosen fidelity fraction (relative reconstruction error variance $< 0.01\%$, resulting

[v] t_c is computed as the time needed to compress the entire concerned dataset.

[vi] CR is computed as the ratio between the memory usage of the uncompressed and compressed (preprocessing parameters, scores and loadings matrices) data structures, both saved as double precision .mat files.

[vii] The reported results refer to the baseline-subtracted spectra for better illustration.

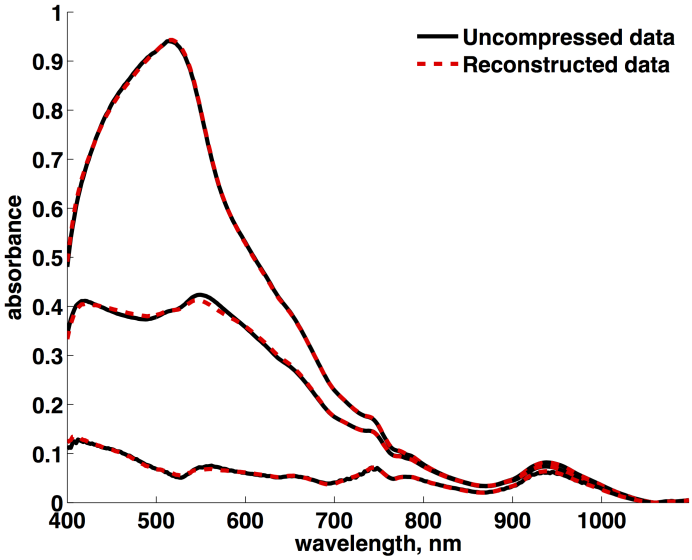

Figure 12.2 Vis-NIR light absorbance spectra from the B-Z reaction at three different points in time: input (black solid lines) and OTFP modelled and reconstructed (red dotted lines) spectra

Table 12.1 Vis-NIR light absorbance spectra from the B-Z reaction: values of the compression quality indices. The number of original measured variables is reported in the first column

J	A	$\mathrm{EV}_{\mathrm{raw}}$	EV_{p}	RMSRE	t_{c}	CR
350	10	99.93	99.61	0.0019	12.5	26.81 $(\frac{9768173 \text{ bytes}}{364370 \text{ bytes}})$

in 10 PCs), only very slight differences between the original and reconstructed profiles are detectable to the naked eye. Had we demanded higher fidelity fractions, more PCs would have been included. Conversely, had we demanded fewer PCs, that would have given higher reconstruction error variance. When submitting this high-dimensional data stream to the automatic model-based data compression, the main patterns of systematic variability in the data were automatically found and extracted. In this example, each high-dimensional spectrum was measured at a single space point only. The next example will show how an overwhelming data stream that arises when thousands of such high-dimensional spectra are measured in parallel by a hyperspectral camera can be dealt with by the OTFP.

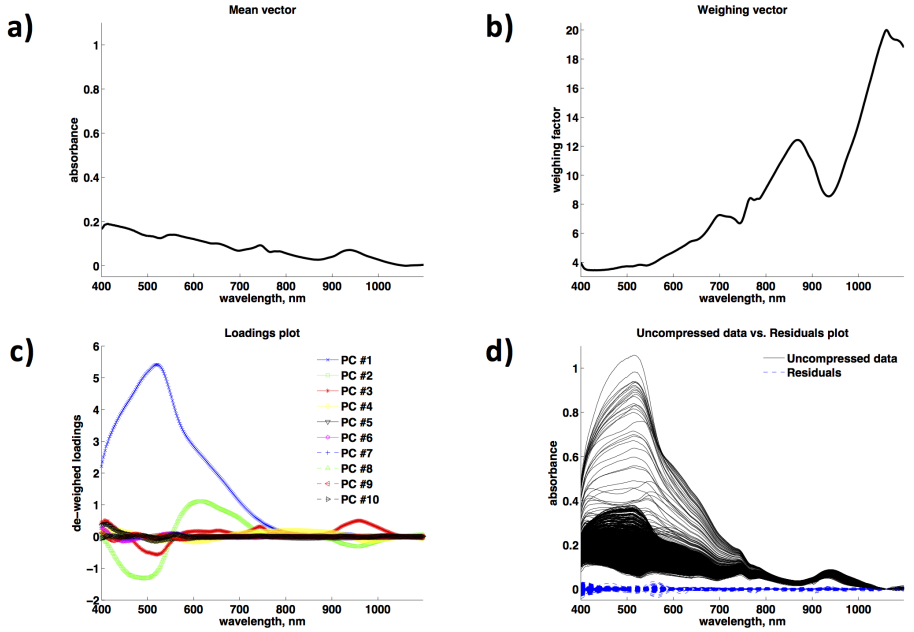

Figure 12.3 Vis-NIR light absorbance spectra from the B-Z reaction: representation of the full compression model. a) Final mean vector[viii], b) variable weighing factors (kept constant throughout the algorithmic procedure), c) loadings profiles (divided by the channel weights, **c**, and scaled by their respective singular values) and d) input absorbance spectra (black solid lines) and lack-of-fit residuals (blue dotted lines)

12.4.2 Detailed remote characterisation of orange samples

This example concerns efficient quality control of physical objects - in this case oranges. The individual pixel NIR spectra were submitted to a model-based pre-treatment, MSC, to remove the undesired light scattering effects and prevent actual chemical signals, often of lesser magnitude [172], from being overlooked. They were subsequently centred and weighed to down-scale noisy spectral regions. Here, the model centre was continuously updated, the variable weighing factors kept constant all over the processing and the MSC parameters additionally stored along with all the other retained information.
As indicated in Table 12.2, the compression of the orange hyperspectral images also yielded satisfactory outcomes. In addition to a very precise data retrieval, since noise is partly filtered out, various imperfections, probably due to instrumental problems, are apparently removed (see Figure 12.4). As an example of the added value the bilinear modelling offers unlike conventional compression meth-

[viii] The mean vector closely resembles the lower-absorbance spectral profiles, due to their high abundance in the Vis-NIR dataset.

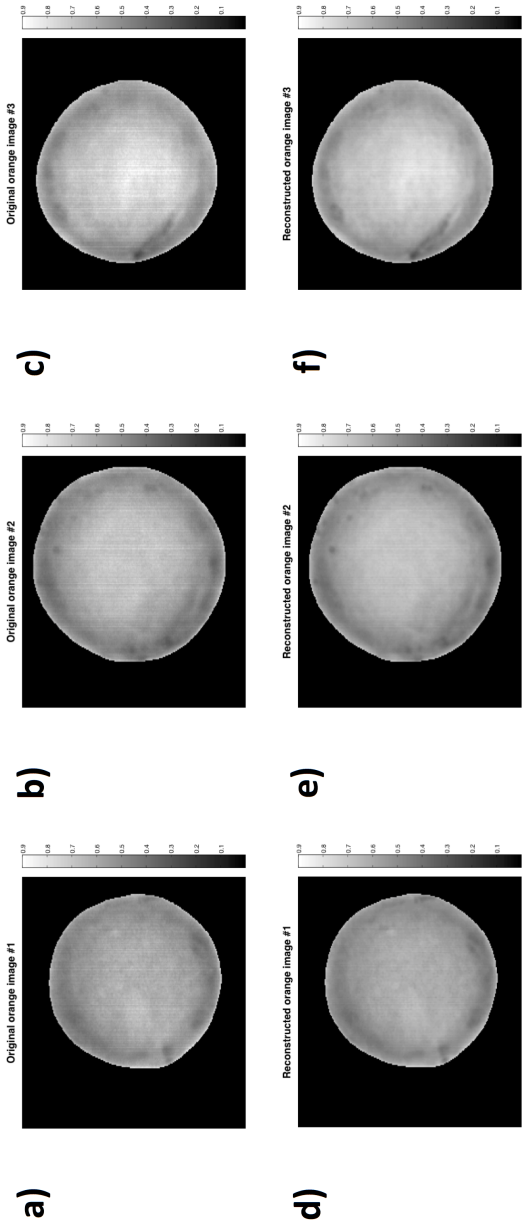

Figure 12.4 Hyperspectral NIR images: a-c) uncompressed and d-f) OTFP modelled and reconstructed grey-scale orange image #1, #2 and #3 at 1675 nm

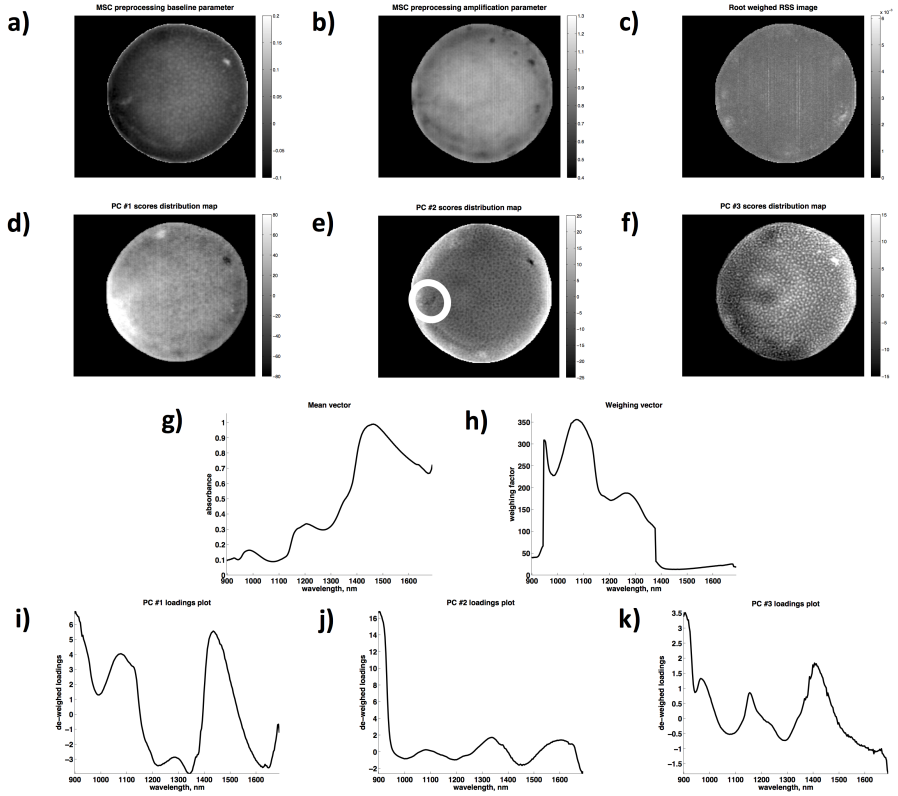

Figure 12.5 Hyperspectral NIR images - Modelling of orange image #2: a) baseline variations and b) amplification variations, estimated by MSC preprocessing and used to correct the spectra of the individual pixels, c) summary of the unmodelled residuals (root Residuals Sum-of-Squares - RSS - of the weighed wavelength channels after the extraction of 5 OTFP PCs), d) PC #1, e) PC #2 and f) PC #3 grey-scale scores distribution maps, g) final wavelength mean vector and h) wavelength weighing factors (kept constant throughout the algorithmic procedure), i) PC #1, j) PC #2 and k) PC #3 loadings profiles (divided by the channel weights, **c**, and scaled by their respective singular values). The white circle in e) highlights a particular defect on the surface of the orange sample

ods in terms of understanding and interpretability, the scores distribution maps[ix] (or scores plots) of image #2 related to the first three extracted PCs and their corresponding loadings profiles are displayed in Figure 12.5 along with the MSC preprocessing parameters used to correct the spectra of the individual pixels, the corresponding root weighed Residuals Sum-of-Squares (RSS) image (after the ex-

[ix]The darkness of the pixels is proportional to the value of their scores on the respective components.

Table 12.2 Hyperspectral NIR images: values of the compression quality indices. The number of original measured variables is reported in the first column

J	A	$\mathrm{EV}_{\mathrm{raw}}$	EV_{p}	RMSRE	t_{c}	CR
247	5	99.93	93.27	0.0096	43.8	33.29 $(\frac{129235545 \text{ bytes}}{3882254 \text{ bytes}})$

traction of five PCs), the final mean vector and the variable weighing factors resulting from the OTFP. PC #1 seems to reflect an overall lighting variation on the 3D orange. The texture of the orange peel is partly captured by PC #2, together with a particular defect located on the bottom-left area of its surface and a 3D illumination and/or penetration effect generating a gradual decrease in the scores values from the border to the centre of the sample. PC #3 seems to represent a purely textural component.
This example has shown that the self-modelling process simplified the interpretation and usage of the enormous amount of data from a hyperspectral camera recording a series of similar objects. The model parameters gave high compression as well as interesting graphical insights. The next case study will illustrate an even more overwhelming data stream from a continuously measuring hyperspectral camera installed on a flying drone.

12.4.3 Environmental surveillance by airborne hyperspectral imaging

The high compression of the hyperspectral push-broom image is proven by both Table 12.3 and Figures 12.6a and 12.6b. In this case the spectrum of each pixel at each point in time was just centred. Specifically, the model centre was continuously updated, while the variable weighing factors were set to 1 and kept constant all over the processing.
Despite the notable reduction in the memory usage, the uncompressed and reconstructed pseudo-RGB pictures, constructed by selecting only three of the available wavelength channels[x], exhibit barely perceptible discrepancies.
It is well known that while bilinear models from subspace estimation methods like PCA and the present OTFP capture the essential information in the data, the individual components are not intended to be meaningful from a physicochemical perspective for their mutual orthogonality (see Equations 2.2 and 2.3). Relaxing these orthogonality constraints and possibly adding other criteria, such as non-negativity in loadings and scores, may give more meaningful individual component plots. As an example, Figure 12.6 also includes three different component scores distribution maps and loadings profiles (Figures 12.6c, 12.6d, 12.6e, 12.6f,

[x]Around 445 nm, 535 nm and 575 nm, where the eye cones have their maximum sensitivity to blue, green and red light, respectively.

12.6g and 12.6h), obtained by a Multivariate Curve Resolution-Alternating Least Squares (MCR-ALS) [173] transformation of the global OTFP model (see Section 14.43 for additional details).
Although MCR-ALS components #1 and #3 are seemingly dominated by the sea foam pixels (whose corresponding signal was found to be saturated in a large spectral range), three distinct patterns are visibly recognisable: the field pixels in the first scores distribution map, the pixels surrounding the sea foam in the second

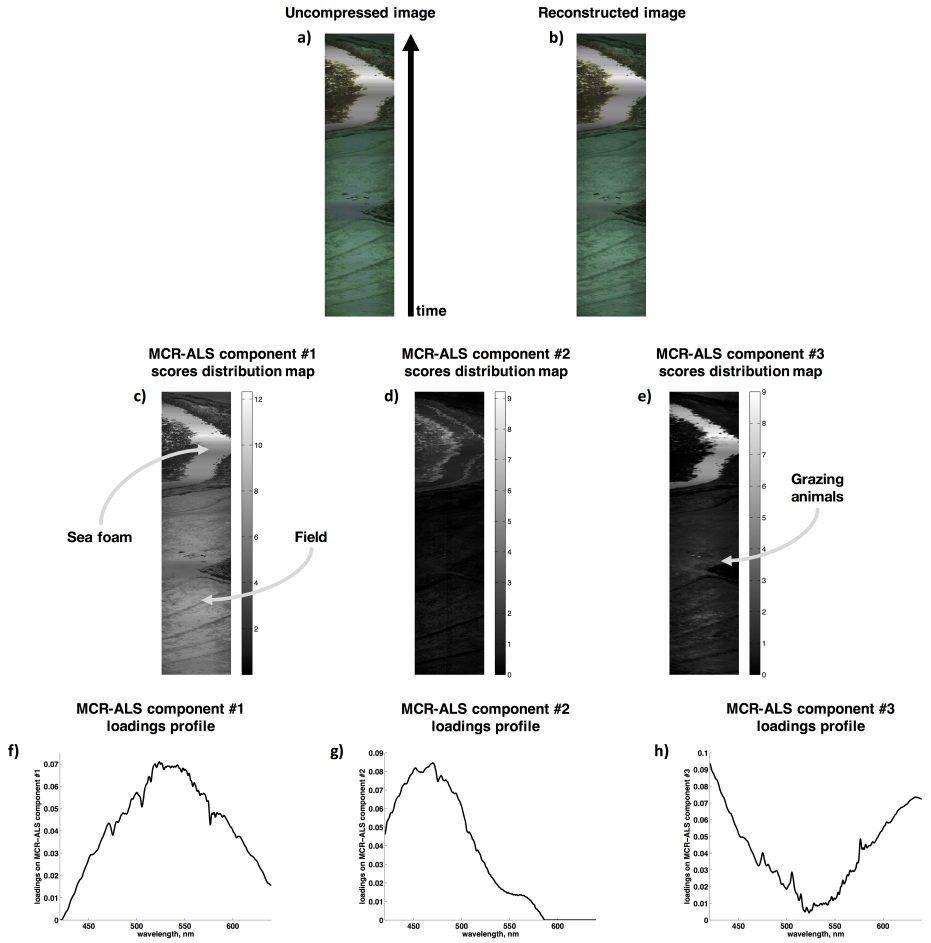

Figure 12.6 Hyperspectral image from a push-broom camera installed on a flying drone: a) uncompressed and b) OTFP modelled and reconstructed images in pseudo-RGB colours, c) MCR-ALS component #1, d) MCR-ALS component #2 and e) MCR-ALS component #3 grey-scale scores distribution maps, f) MCR-ALS component #1, g) MCR-ALS component #2 and h) MCR-ALS component #3 loadings profiles

Table 12.3 Hyperspectral image from a push-broom camera installed on a flying drone: values of the compression quality indices. The number of original measured variables is reported in the first column

J	A	$\mathrm{EV}_{\mathrm{raw}}$	EV_{p}	RMSRE	t_{c}	CR
450	3	99.82	99.02	0.015	300.2	45.02 $(\frac{241451269 \text{ bytes}}{5363455 \text{ bytes}})$

scores distribution map and those capturing several animals grazing at the centre of the image in the third scores distribution map. Therefore, *ça va sans dire*, the OTFP may be employed for preliminary image treatment before further handling or segmentation.
Independent Component Analysis (ICA) [174, 175] or Parallel Factor Analysis (PARAFAC) [176, 177] and extensions of these coupled with various pixel clustering methods also belong to the rich flora of post-processing methods that can be applied to bilinear models like those coming from the OTFP.
The first three illustrations have shown how broad data streams from multi-channel sensors can be handled by the OTFP. The last example concerns a very different kind of data - a more or less random collection of individual, single-channel sensors. Traditional process industry is often extensively equipped with similar apparatus. Often, each one of them gets its own display screen with its own alarm limits. How can the burden for the process operators be reduced as well as the number of false alarms? Perhaps by finding common patterns of covariation among the many sensors?

12.4.4 Analysis of an industrial manufacturing process

According to the quality indices reported in Table 12.4, the general performance of the OTFP when applied to this rather low-dimensional stream of industrial process data was found to be slightly worse than in the previous case studies. This is not unexpected, given the low number of variables under study and their widely varying nature, and is a consequence of the fact that their correlation structure is not so strong that just few PCs can practically summarise all their significant variation. Nevertheless, for most of them an acceptable reconstruction was achieved, as Figure 12.7 confirms.
Besides, examining both scores and loadings can provide remarkable insights into the process behaviour, particularly if meaning can be ascribed to the input records - or at least to some of them - by human expert characterisation. This is illustrated by the scores plot in Figure 12.8a. PC #1 separates two groups of observations: blue dots and red squares refer in fact to Normal Operating Conditions (NOC) and shut-down time samples, respectively. As the latter present negative projection

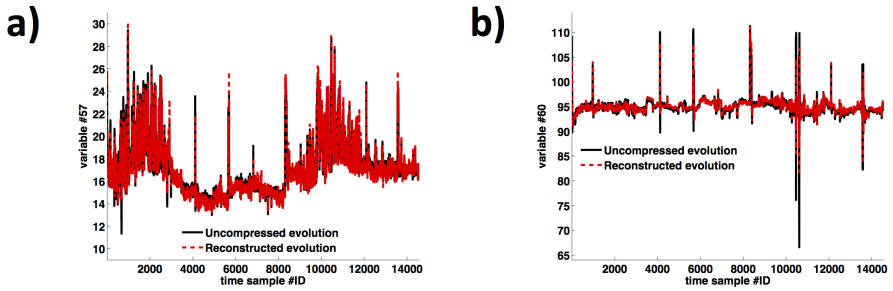

Figure 12.7 Industrial process data: uncompressed (black solid line) and OTFP modelled and reconstructed (red dotted line) temporal evolution of a) variable #57 and b) variable #60

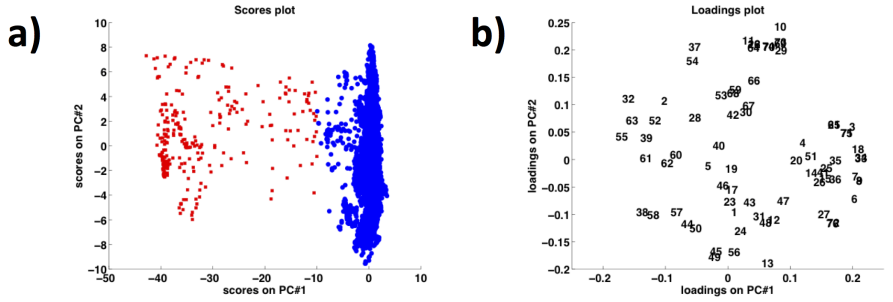

Figure 12.8 Industrial process data: a) PC #1/PC #2 scores (blue dots and red squares refer to Normal Operating Conditions - NOC - and shut-down time samples, respectively) and b) loadings plots (the numbers correspond to the #IDs of the original variables and are represented according to their respective PC #1/PC #2 loadings values)

Table 12.4 Industrial process data: values of the compression quality indices. The number of original measured variables is reported in the first column

J	A	EV_{raw}	EV_p	RMSRE	t_c	CR
76	13	99.47	81.33	0.4640[xi]	49.5	3.35 ($\frac{4895674 \text{ bytes}}{1459315 \text{ bytes}}$)

coordinates on this component, they will be characterised by lower-than-average values of all the measured variables featuring a relatively large positive PC #1 loading (which actually assumed a nearly 0-level during shut-down periods) and

[xi] As the original variables were characterised by different units of measurements, the reported RMSRE value concerns the final centred and weighed data array.

vice versa (see Figure 12.8b). On the other hand, within-cluster differences seem to be mainly spotted by PC #2.

12.5 Comparison with classical PCA

To what extent does the OTFP mimic the corresponding traditional data modelling strategy? The present implementation of the OTFP employs similar criteria in the model updating stage to those of standard PCA, so it is natural to compare both the approaches. While the OTFP needs to hold only a small part of the data in memory at a time, traditional PCA requires all the data to be held in memory at the same time, at least if both loadings and scores are to be assessed. The time span of the hyperspectral drone imaging example (Figure 12.6 and Table 12.3) was chosen short enough to allow conventional PCA to be run and its solution to be compared to the final OTFP model.
Figure 12.9 permits to appraise the performance of the two methods for the same dataset. Figure 12.9a shows that the mean spectrum used for model centring in

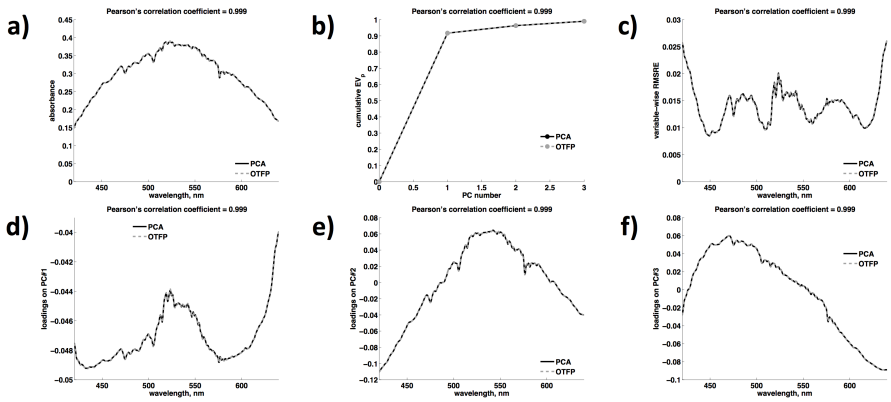

Figure 12.9 Hyperspectral image from a push-broom camera installed on a flying drone - Classical PCA (black solid lines) *vs.* OTFP (grey dotted lines): a) mean vectors, b) cumulative percentages of explained preprocessed data variance, c) lack-of-fit (root mean square error) for the individual variables after the extraction of 3 OTFP PCs (negligible if compared with the original signal magnitude) d) PC #1, e) PC #2 and f) PC #3 loadings profiles. Variable weighing factors (not shown) were set to 1 for all the spectral wavelengths and kept constant all over the OTFP

PCA is more or less identical to the model centre vector, **m**, gradually developed by the OTFP. The outcomes of the two techniques are also virtually indistinguishable if looking at the plot of the cumulative percentage of explained preprocessed data variance (Figure 12.9b) and the variable-wise RMSRE (after the extraction of three PCs, Figure 12.9c) as well as at the loadings profiles of PC #1 (Figure

12.9d), PC #2 (Figure 12.9e) and PC #3 (Figure 12.9f). The corresponding spatiotemporal OTFP scores (not displayed due to the high number of data points) were also found to be very similar to the PCA ones. Consequently, both PCA and the OTFP led to practically identical values of the diagnostic indices listed at the beginning of Section 12.4 except for t_c. The decomposition was in fact achieved faster by PCA, which had simultaneous access to all the available information. On the other hand, the OTFP had to handle it by evolving its bilinear model on-the-fly as the data flowed. Nevertheless, the comparison highlighted the OTFP can be considered a feasible alternative to standard PCA, when this latter is not applicable (e.g. when the measurements are collected in real time or the size of the analysed matrix is prohibitively large).

12.6 Discussion

The OTFP treats the incoming data one record or one batch of records at a time, and gradually develops a compact quantitative model of this data stream from the covariation patterns that it discovers. Still, Figure 12.9 illustrates the OTFP behaved quite similarly - at least for the first three components - to the corresponding conventional *global* multivariate data modelling method (in this case PCA), which analyses all the data simultaneously. Their results are almost identical even if the OTFP repeatedly has to make sense out of small chunks of data as they arrive. So it has to make many temporary decisions about what to throw away as random noise, while the global PCA has access to all the information at once. On the other hand, this is precisely the motivation behind the development of the OTFP tool, that is to always maintain an updated, compact summary of all the systematic changes, which have taken place in an otherwise overwhelming, *ever-lasting* high-dimensional data stream, with low computational or memory requirements.

The OTFP uses a multivariate data driven approximation model (here, PCA-like) as a generic Taylor expansion around a chosen set point or model centre, to summarise whatever known or unknown phenomena have caused the systematic covariation patterns in the input data stream. The OTFP data model has a linear, additive structure and therefore gives the best approximation performance when non-additive and/or non-linear effects in the input data have been corrected for in the preprocessing step. Preprocessing is then helpful for reducing response curvature and other types of non-linearities[xii]. Response linearity was improved in the first example (Figures 12.2 and 12.3) by converting the fibre-optic transflection data to absorbance values. Patterns known to be non-interesting may be removed during preprocessing, as illustrated by the simple baseline correction in the same

[xii] In case preprocessing is not of help, complex non-linearities and system heterogeneities may be handled by e.g. automatically splitting the data stream under study into two or more disjoint OTFP models (in a similar way as for the well-known static Soft Independent Modelling of Class Analogy - SIMCA - approach [178, 179]).

case study. In the second example (see Figure 12.4), unknown additive baseline variations and multiplicative amplification variations were estimated, parametrised and removed jointly by MSC. On the other hand, when the input variables are given in very different units, preprocessing should also scale them to balance their uncertainty levels - or, if that is unknown, to balance their total variances as shown for the industrial process data (Figures 12.7 and 12.8).
The OTFP components are mathematical basis vectors that characterise the data stream. When plotted in combination they give useful insights into the main patterns of data variation, as illustrated in Figure 12.8. But such orthogonal basis vectors are not meant to be interpreted individually. Therefore, the OTFP solution may be at any time readjusted for better visualisation and more causal interpretation. This was shown by the conversion of the orthogonal, PCA-like OTFP component profiles into more graphically distinct ones by requiring non-negative scores and loadings in an MCR-ALS-based post-processing.

12.7 Conclusions and perspectives

In the near future, a drastic increase in the collection and use of high-dimensional, continuous measurements is expected. Rational use of such data streams requires generic data modelling tools that not only give good predictions and classifications as the information flow evolves, but that also reveal its essential structure for human interpretation and efficient compression. In this chapter the On-The-Fly Processing (OTFP) tool for the on-the-fly gradual modelling and compression of continuous quantitative data streams was proposed. It is based on an evolving implementation of PCA that updates on-line, when necessary, both preprocessing parameters and principal component structure (whose changes and possible expansion can be optionally monitored in real time through intuitive graphical displays). It combines the advantages of three different ways of attaining PCA or PCA-like bilinear decompositions, while avoiding their disadvantages:

- repeated use of conventional PCA, each time bringing increasing amounts of data into memory for simultaneous analysis, which yields bilinear models relevant for both past and present observations, but becomes prohibitively slow and memory-demanding for *ever-lasting* data streams;
- moving-window PCA, which repeatedly merges new observations with a bilinear subspace loading summarising past ones, ensuring that the bilinear model is up-to-date and thus relevant for the latest observations at any given moment, while losing relevance for older observations;
- eigen-analysis of the cumulative $J \times J$ cross-product matrix, a simple and fast computation as long as J is not too large, which is suitable for parallelisation

and *out-of-core* estimation of the PCA loadings with relevance also for past observations, but without quantitative scores for them.

The OTFP discovers new systematic patterns of covariation in multi-channel data streams, and thereby extends its current bilinear model with new dimensions in a computationally efficient way. Over time, the observation scores are stored to disk in packets and then deleted from memory. The model is continuously updated to be as PCA-like as possible, but in such a way that past scores can always be recalled and compared to present ones.
The algorithmic procedure exhibited very satisfactory performance in terms of compression rate and time and quality of the input reconstruction, especially if measurement series underlain by strong correlation structures (e.g. Vis-NIR spectra or hyperspectral images) were dealt with. On the other hand, in the industrial process example, its power was not as prominent, probably due to a lower degree of intercorrelation in this data stream. Still, the retrieval of the temporal evolution of the original variables was reasonably precise. This could represent an important cross-road for manufacturing companies, whose modern information storage systems are commonly based on univariate calculations, not taking into account the possible interdependences among various instrumental responses, destroying their essential multivariate nature and eliminating much of their meaningful content [180]. Last but not least, the scores and loadings estimated through the PCA-based dimensionality reduction feature distinctive interpretability properties, extremely helpful for data understanding, utilisation and further exploration by complementary statistical approaches (e.g. MCR-ALS). Apparently, no available compressor guarantees such a noteworthy added value.

Part VI

Epilogue

Chapter 13

Conclusions and perspectives

13.1 Accomplishment of the objectives

Based on the results shown in the previous chapters, the following general conclusions are drawn, from which the degree of accomplishment of the different objectives established in Section 1.2 can be appraised.

13.1.1 Objective I - Exploring the potential of kernel-based methodologies for statistical process monitoring, improved fault diagnosis, translation of *out-of-control signals* to operator actions, and analysis of mixture designs of experiments

Novel algorithmic solutions exploiting the principles of both the kernel extension of PCA, PLS and PLS-DA and the so-called pseudo-sample projection were proposed and tested in four distinct scenarios: RGB image segmentation, on-/off-specification batch run discrimination, batch process monitoring, and analysis of mixture designs of experiments. Two fundamental points arose from all the case studies dealt with, no matter the nature of the handled datasets:

- when strongly non-linear relationships among variables and/or samples had to be modelled, non-linear kernel methods exhibited clear advantages compared to classical bilinear ones, in terms of classification/discrimination quality, fault detection and diagnosis capability, and prediction power and stability;
- even if no severe non-linearities affected the analysed data, K-PCA, K-PLS and K-PLSDA generally permitted to achieve very similar outcomes to their

corresponding conventional techniques, provided that an appropriate mathematical function is selected for the initial kernel transformation.

Such approaches may be particularly useful in those contexts where huge amounts of very complex information are routinely collected (e.g. in the industrial field), but Chapters 4-8 proved that they can be easily resorted to for a wider range of applications. Furthermore, new intuitive graphical tools (i.e. the DD plot, the VR plot and the pseudo-sample based surface plot) were implemented to support potential users in the complicated task of kernel model assessment and interpretation, thus facilitating and accelerating decision making and troubleshooting.

13.1.2 Objective II - Proposing rational approaches for selecting the optimal amount of information to be modelled for data exploration and understanding

Chapter 9 provides theoretical and practical insights for an improved understanding of permutation testing, mathematically formalises the numerical operations to be carried out when applying it for the determination of the number of factors in PCA by the description of an innovative computational procedure designed to this end, and illustrates *ad hoc* solutions for optimising calculation time and memory consumption. The advantages of such a computational procedure over standard methods like Horn's parallel analysis were shown in both simulated and real case studies and through a comprehensive discussion about the implications of its main algorithmic steps. This makes it reasonable to look at it as a very useful addendum to other available statistical techniques for PCA model selection. The possibility of employing the proposed strategy for effective rank identification prior to multi-set data analysis by means of Canonical Correlation Analysis was also preliminarily explored in Chapter 10. Its combination with other types of approaches (e.g. Joint and Individual Variation Explained) deserves further research.

13.1.3 Objective III - Enhancing model transfer between manufacturing units or workstations

In Chapters 10 and 11, the attention was focused on the simultaneous processing of multiple data matrices. Specifically, a comprehensive comparison of the most commonly used methodologies for retrieving the common and distinctive components underlying two datasets sharing the row- or the column-dimension was carried out. This comparison pointed out one of their main limitations: none of them encompasses primary computational steps aimed at determining, at least tentatively, the number of these common and distinctive factors. In order to overcome this issue, an innovative computational procedure coupling permutation testing and Canonical Correlation Analysis was developed. It led to satisfactory results in both a simulated and a real example. Moreover, the recently designed Joint-Y

PLS algorithm was applied as an alternative option to model the shared covariation of various predictor blocks with a definite number of response variables in a regression scenario. Even though the illustrated results are still preliminary, they can be considered a significant contribution to this topic, especially with a view to the implementation of strategies to identify the specific sources of variability to be described for building e.g. a general monitoring scheme which could be valid for more than a single manufacturing unit or workstation.
In addition, two novel techniques for calibration transfer between near-infrared spectrometers, based on Trimmed Scores Regression and Joint-Y PLS regression, respectively, were proposed. They were found to outmatch a reference approach like Piecewise Direct Standardisation in two different case studies and to be rather robust towards the reduction of the instrumental resolution. In the near future, their applicability to different types of spectroscopic measurements will also be investigated.

13.1.4 Objective IV - Implementing new computational strategies for real-time data processing

The *On-The-Fly Processing* (OTFP), an original software system for rational handling of continuous data streaming in real time, was presented in Chapter 12. This quantitative learning tool constructs reduced-rank bilinear subspace models that summarise massive series of multi-channel measurements, capturing the evolving covariation patterns among the many recorded variables over time and space. It mostly exhibited very promising performance in terms of compression rate and speed and quality of the input reconstruction, mainly due to the fact it combines the pros of three different ways of attaining a PCA-like decomposition, while avoiding their cons, i.e. repetead use of conventional PCA, moving-window PCA and eigen-analysis of the cumulative cross-product matrix.
Soon, this strategy for continuous, automatic model development based on multi-channel recordings, may become useful for processing a very wide range of data stream types. For instance, Internet of Things will result in an enormous increase of technical measurements in many fields of interest (medicine, communications *etc.*). Many of these Internet of Things sensors will be multi-channel (e.g. cameras and spectrometers). Others will be univariate, but even these will generate multi-channel data: the time series from one single, more or less continuous data source will lead to high-dimensional frequency spectra (spectrograms), after suitable domain transforms (e.g. by Fast Fourier Transform or wavelet analysis). Since the methodology here relies solely on linear algebra, it is expected to work properly also within such a more general BIG DATA context.
Furthermore, an extension of the OTFP is currently being implemented for real-time statistical control of industrial processes. Among the various aspects of its algorithm, it factors in a completely different treatment of severe outliers, which in principle need to be removed from the information flow as the focus is not

on compression anymore, but on keeping the manufacturing *in-control*. Such an adjustment together with the self-learning and adaptive nature and the delayed modelling principle of the OTFP will permit to dramatically decrease the time needed for starting up new monitoring schemes when process conditions are updated or novel production processes are run.

13.2 Future research lines

The outcomes of this Ph.D. thesis open new interesting perspectives, whose aspects may merit further attention in the near future:

- Expand the application range of kernel methods. It is true that they are already quite well-known in the scientific community, but their combination with pseudo-samples and pseudo-sample projection has only partially been exploited in fields like medicine or biotechnology, where rather complex data are generally collected;
- Adapt the K-PCA-based methodology presented in Chapter 7 for continuous and real-time batch process monitoring;
- Redefine the principles of pseudo-sample projection in the attempt of unveiling and accounting for possible interactions among measured variables;
- Investigate feasible alternatives to cross-validation for tuning the kernel function parameters depending on the specific problem at hand;
- Couple the model selection approach illustrated in Chapter 9 to multi-set data analysis methods like SCA or Joint and Individual Variation Explained;
- Design novel permutation testing-based computational solutions for the tentative identification of the dimensionality of the projection subspace in Factor Analysis;
- Extend the applicability of the CCA-based permutation test proposed in Chapter 10 to cases in which more than two data blocks have to be simultaneously dealt with;
- Test the performance of the whole computational procedure described in Section 10.4.2 under a wider variety of scenarios and assess the effect of retrieving common and distinctive component profiles by using another modelling approach instead of the aforementioned pseudo-O2PLS methodology;
- Implement a Graphical User Interface for calibration transfer between near-infrared spectrometers and make it available for academic and industrial use;

- Evaluate the potential of TSR and JYPLS for calibration transfer between distinct types of analytical platforms (mid-infrared or Raman spectrometers, nuclear magnetic resonance instrumentations *etc.*);
- Explore the capability of the OTFP tool of handling and processing proper BIG DATA streams;
- Modify the core of its algorithmic structure to enable PLS bilinear modelling of evolving multi-channel time series;
- Develop a variant of the OTFP for real-time monitoring of continuous manufacturing processes.

Chapter 14

Appendices

14.1 Annex to Part II

14.1.1 Relationship between the Euclidean distance matrix, D, and the inner product matrix, $\mathbf{XX^T}$

The Euclidean distance between two observations, $\mathbf{x}_n^{\mathrm{T}}$ and $\mathbf{x}_{n^*}^{\mathrm{T}}$, of a generic dataset, $\mathbf{X}$ $(N \times J)$, can be expressed as:

$$d_{n,n^*} = \|\mathbf{x}_n - \mathbf{x}_{n^*}\|^2 = (\mathbf{x}_n - \mathbf{x}_{n^*})^{\mathrm{T}}(\mathbf{x}_n - \mathbf{x}_{n^*}) = \mathbf{x}_n^{\mathrm{T}}\mathbf{x}_n + \mathbf{x}_{n^*}^{\mathrm{T}}\mathbf{x}_{n^*} - 2\mathbf{x}_n^{\mathrm{T}}\mathbf{x}_{n^*} \quad (14.1)$$

Let $\mathbf{F}$ $(N \times N)$ be the inner product matrix and $\mathbf{D}$ $(N \times N)$ the Euclidean distance matrix, defined as:

$$\mathbf{F} = \mathbf{X}\mathbf{X}^{\mathrm{T}} \quad (14.2)$$

$$\mathbf{D} = \mathbf{f}\mathbf{1}^{\mathrm{T}} + \mathbf{1}\mathbf{f}^{\mathrm{T}} - 2\mathbf{F} \quad (14.3)$$

where $\mathbf{f}$ $(N \times 1)$ denotes the diagonal vector of $\mathbf{F}$ and $\mathbf{1}$ $(N \times 1)$ is a vector of ones. Centring $\mathbf{X}$ such that:

$$\overline{\mathbf{X}} = \mathbf{X} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathbf{X} \quad (14.4)$$

then:

$$\begin{aligned} \overline{\mathbf{F}} = \overline{\mathbf{X}}\overline{\mathbf{X}}^{\mathrm{T}} = (\mathbf{X} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathbf{X})(\mathbf{X} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathbf{X})^{\mathrm{T}} = \\ = \mathbf{F} - \frac{1}{N}\mathbf{F}\mathbf{1}\mathbf{1}^{\mathrm{T}} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathbf{F} + \frac{1}{N^2}\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathbf{F}\mathbf{1}\mathbf{1}^{\mathrm{T}} \end{aligned} \quad (14.5)$$

If $\mathbf{D}$ is double-centred as:

$$\mathbf{B} = -\frac{1}{2}\mathbf{H}\mathbf{D}\mathbf{H}^{\mathrm{T}} \quad (14.6)$$

where $\mathbf{H}$ $(N \times N)$ connotes the operator:

$$\mathbf{H} = \mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}} \tag{14.7}$$

and $\mathbf{I}$ is an $N \times N$ identity matrix, it follows:

$$\mathbf{B} = -\frac{1}{2}\mathbf{H}(\mathbf{f}\mathbf{1}^{\mathrm{T}} + \mathbf{1}\mathbf{f}^{\mathrm{T}} - 2\mathbf{F})\mathbf{H}^{\mathrm{T}} \tag{14.8}$$

Since:

$$\mathbf{f}\mathbf{1}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}} = \mathbf{f}\mathbf{1}^{\mathrm{T}}(\mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}})^{\mathrm{T}} = \mathbf{f}\mathbf{1}^{\mathrm{T}} - \mathbf{f}(\frac{\mathbf{1}^{\mathrm{T}}\mathbf{1}}{N})\mathbf{1}^{\mathrm{T}} = 0 \tag{14.9}$$

it is verified:

$$\mathbf{H}\mathbf{f}\mathbf{1}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}} = 0 = \mathbf{H}\mathbf{1}\mathbf{f}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}} \tag{14.10}$$

Therefore:

$$\begin{aligned} \mathbf{B} = \mathbf{H}\mathbf{F}\mathbf{H}^{\mathrm{T}} = (\mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}})\mathbf{F}(\mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}})^{\mathrm{T}} = \\ = \mathbf{F} - \frac{1}{N}\mathbf{F}\mathbf{1}\mathbf{1}^{\mathrm{T}} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathbf{F} + \frac{1}{N^2}\mathbf{1}(\mathbf{1}^{\mathrm{T}}\mathbf{F}\mathbf{1})\mathbf{1}^{\mathrm{T}} = \overline{\mathbf{F}} \end{aligned} \tag{14.11}$$

that is:

$$\mathbf{B} = -\frac{1}{2}\mathbf{H}\mathbf{D}\mathbf{H}^{\mathrm{T}} = \overline{\mathbf{X}}\overline{\mathbf{X}}^{\mathrm{T}} \tag{14.12}$$

The double-centred Euclidean distance matrix, $\mathbf{B}$, equals the inner product matrix, $\overline{\mathbf{X}}\overline{\mathbf{X}}^{\mathrm{T}}$, *quod erat demonstrandum.*

14.1.2 Practical meaning of the pseudo-samples in the feature space

K-PLS/K-PLSDA models

Suppose $\mathbf{B} = \overline{\mathbf{X}}\overline{\mathbf{X}}^{\mathrm{T}}$ has been resorted to for calibrating a 1-latent variable PLS model. The scores of the N objects under study, $\mathbf{t}^{\mathbf{B}}$ $(N \times 1)$, can be written as:

$$\mathbf{t}^{\mathbf{B}} = \mathbf{B}\mathbf{w}^{*\mathbf{B}} \tag{14.13}$$

where $\mathbf{w}^{*\mathbf{B}}$ $(N \times 1)$ represents the PLS vector of weights, which, in this case, does not contain any useful information about the J original variables in $\mathbf{X}$. Substituting 14.12 in 14.13 it follows:

$$\mathbf{t}^{\mathbf{B}} = \overline{\mathbf{X}}\overline{\mathbf{X}}^{\mathrm{T}}\mathbf{w}^{*\mathbf{B}} = \overline{\mathbf{X}}(\overline{\mathbf{X}}^{\mathrm{T}}\mathbf{w}^{*\mathbf{B}}) = \overline{\mathbf{X}}\mathbf{w}^{*\prime} \tag{14.14}$$

where $\mathbf{w}^{*\prime}$ $(J \times 1)$ is now relevant for interpretation purposes. Projecting the pseudo-sample $\mathbf{g}^{\mathrm{T}} = [0,\ 0,\ \ldots,\ 1,\ 0,\ \ldots,\ 0]$ $(1 \times J)$ onto the PLSDA subspace returns:

$$t_{\mathbf{g}^{\mathrm{T}}} = \mathbf{g}^{\mathrm{T}}\overline{\mathbf{X}}^{\mathrm{T}}\mathbf{w}^{*\mathbf{B}} = \mathbf{g}^{\mathrm{T}}\mathbf{w}^{*'} = [0,\ 0,\ \ldots,\ 1,\ 0,\ \ldots,\ 0]\mathbf{w}^{*'} = w_j^{*\prime} \tag{14.15}$$

which permits to access the content of $\mathbf{w}^{*\prime}$. $t_{\mathbf{g}}$, in fact, is exactly equal to its j-th element, *quod erat demonstrandum*.

K-PCA models

Extracting the first principal component of $\mathbf{B} = \overline{\mathbf{X}}\overline{\mathbf{X}}^{\mathrm{T}}$ by SVD yields:

$$\overline{\mathbf{X}}\overline{\mathbf{X}}^{\mathrm{T}}\mathbf{v}_1 = \lambda_1\mathbf{v}_1 \tag{14.16}$$

$\mathbf{v}_1$ $(N \times 1)$ corresponds to the first normalised score vector computed by applying SVD to $\overline{\mathbf{X}}^{\mathrm{T}}\overline{\mathbf{X}}$ $(J \times J)$:

$$\begin{aligned} \overline{\mathbf{X}}^{\mathrm{T}}\overline{\mathbf{X}}\mathbf{u}_1 &= \lambda_1\mathbf{u}_1 \\ \overline{\mathbf{X}}\mathbf{u}_1 &= \mathbf{t}_1 \\ \mathbf{v}_1 = \frac{\mathbf{t}_1}{\|\mathbf{t}_1\|} = \frac{\mathbf{t}_1}{\sqrt{\lambda_1}} &= \frac{\overline{\mathbf{X}}\mathbf{u}_1}{\sqrt{\lambda}} \end{aligned} \tag{14.17}$$

where:

$$\|\mathbf{t}_1\| = \sqrt{\mathbf{t}_1^{\mathrm{T}}\mathbf{t}_1} = \sqrt{\mathbf{u}_1\overline{\mathbf{X}}^{\mathrm{T}}\overline{\mathbf{X}}\mathbf{u}_1} = \sqrt{\lambda_1\mathbf{u}_1^{\mathrm{T}}\mathbf{u}_1} = \sqrt{\lambda_1} \tag{14.18}$$

The projection of the kernel vector $\mathbf{g}^{\mathrm{T}}\overline{\mathbf{X}}^{\mathrm{T}} = [0,\ 0,\ \ldots,\ 1,\ 0,\ \ldots,\ 0]\overline{\mathbf{X}}^{\mathrm{T}}$ $(1 \times N)$ onto the space defined by $\mathbf{v}_1$ can be expressed as:

$$\begin{aligned} &\mathbf{g}^{\mathrm{T}}\overline{\mathbf{X}}^{\mathrm{T}}\mathbf{v}_1 = \mathbf{g}^{\mathrm{T}}\overline{\mathbf{X}}^{\mathrm{T}}\overline{\mathbf{X}}\frac{\mathbf{u}_1}{\sqrt{\lambda_1}} = \mathbf{g}^{\mathrm{T}}\frac{\lambda_1\mathbf{u}_1}{\sqrt{\lambda_1}} = \\ &= \sqrt{\lambda_1}\mathbf{g}^{\mathrm{T}}\mathbf{u}_1 = \sqrt{\lambda_1}[0,\ 0,\ \ldots,\ 1,\ 0,\ \ldots,\ 0]\mathbf{u}_1 = \sqrt{\lambda_1}u_{1,j} \end{aligned} \tag{14.19}$$

and, thus, carries information about the importance of the j-th variable of the original data matrix, *quod erat demonstrandum*.
As proven in [30], the properties derived here and in Section 14.1.2 also apply for all those kernel transformations generating sets of distance, which may be embedded in a Euclidean space. The Euclidean nature of the polynomial and the Gaussian kernels results from the fact they are calculated as functions of the linear one and the Euclidean distance, respectively [181].

14.1.3 Relationship between Scheffé and Cox model coefficients

Consider the following formulations of the second-order Scheffé and Cox polynomials (Equation 14.20 and 14.21, respectively):

$$y = \sum_{i=1}^{I}\beta_i x_i + \sum_{i=1}^{I-1}\sum_{j=i+1}^{I}\beta_{i,j}x_i x_j + \epsilon \tag{14.20}$$

$$y = \alpha_0 + \sum_{i=1}^{I} \alpha_i x_i + \sum_{i=1}^{I-1} \sum_{j=i+1}^{I} \alpha_{i,j} x_i x_j + \sum_{i=1}^{I} \alpha_{i,i} x_i^2 + \epsilon \tag{14.21}$$

By applying to Equation 14.21 the mixture constraint in Equation 8.1:

$$\sum_{i=1}^{I} x_i = 1 \tag{14.22}$$

and reformulating the second-order term, x_i^2, as:

$$x_i^2 = x_i(1 - \sum_{\substack{j=1 \\ j \neq i}}^{I} x_j) \tag{14.23}$$

it follows:

$$\begin{aligned} \hat{y} &= \alpha_0 \sum_{i=1}^{I} x_i + \sum_{i=1}^{I} \alpha_i x_i + \sum_{i=1}^{I} \sum_{\substack{j=1 \\ j \neq i}}^{I} \alpha_{i,j}^* x_i x_j + \sum_{i=1}^{I} \alpha_{i,i} x_i (1 - \sum_{\substack{j=1 \\ j \neq i}}^{I} x_j) \\ \hat{y} &= \sum_{i=1}^{I} (\alpha_0 + \alpha_i) x_i + \sum_{i=1}^{I} \sum_{\substack{j=1 \\ j \neq i}}^{I} \alpha_{i,j}^* x_i x_j + \sum_{i=1}^{I} \alpha_{i,i} x_i - \sum_{i=1}^{I} \sum_{\substack{j=1 \\ j \neq i}}^{I} \alpha_{i,i} x_i x_j \\ \hat{y} &= \sum_{i=1}^{I} (\alpha_0 + \alpha_i + \alpha_{i,i}) x_i + \sum_{i=1}^{I} \sum_{\substack{j=1 \\ j \neq i}}^{I} (\alpha_{i,j}^* - \alpha_{i,i}) x_i x_j \end{aligned} \tag{14.24}$$

being $\hat{y}$ the estimated value of the response property to be predicted. The notation $\alpha_{i,j}^*$ permits to explicitly differentiate the interaction terms $x_i x_j$ and $x_j x_i$. Specifically, $\alpha_{i,j}^* = \alpha_{j,i}^*$ and $\alpha_{i,j} = 2\alpha_{i,j}^* = 2\alpha_{j,i}^*$ if $i \neq j$. Rewriting Equation 14.24, it is obtained:

$$\hat{y} = \sum_{i=1}^{I} (\alpha_0 + \alpha_i + \alpha_{i,i}) x_i + \sum_{i=1}^{I-1} \sum_{j=i+1}^{I} (\alpha_{i,j} - \alpha_{i,i} - \alpha_{j,j}) x_i x_j \tag{14.25}$$

As:

$$\hat{y} = \sum_{i=1}^{I} (\alpha_0 + \alpha_i + \alpha_{i,i}) x_i + \sum_{i=1}^{I-1} \sum_{j=i+1}^{I} (\alpha_{i,j} - \alpha_{i,i} - \alpha_{j,j}) x_i x_j = \sum_{i=1}^{I} \beta_i x_i + \sum_{i=1}^{I-1} \sum_{j=i+1}^{I} \beta_{i,j} x_i x_j \tag{14.26}$$

it is then proved that:

$$\begin{aligned} \beta_i &= \alpha_0 + \alpha_i + \alpha_{i,i} \\ \beta_{i,j} &= \alpha_{i,j} - \alpha_{i,i} - \alpha_{j,j} \end{aligned} \tag{14.27}$$

In the particular case where $\alpha_{i,i} = \alpha_{i,j} = \alpha_{j,j} = 0$ (first-order polynomial), then:

$$\beta_i = \alpha_0 + \alpha_i \tag{14.28}$$

quod erat demonstrandum.

14.2 Annex to Part III

14.2.1 Horn's parallel analysis

Horn's parallel analysis [86] is a Monte Carlo-based approach, whose basic idea is to compare the eigenvalues of the covariance matrix resulting from the data array under study with their sampling distribution, obtained simulating uncorrelated variables. A factor or component is retained if its respective eigenvalue is larger than e.g. the 99^{th} percentile of its sampling distribution. Since the 70s, Horn's parallel analysis has been often considered the best available option for PCA component selection in psychometrics [182–186].

14.2.2 Dray's method

In its most efficient form [87], Dray's method encompasses the following 9 algorithmic steps grouped in three consecutive phases:

- Phase I - Singular Value Decomposition (SVD) of $\mathbf{X}$:

  1. Perform full-rank SVD on $\mathbf{X}$:

     $$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}} = \mathbf{T}\mathbf{P}^{\mathrm{T}} \tag{14.29}$$

     where $\mathbf{U}$ $(N \times N)$ and $\mathbf{V}$ $(J \times J)$ contain the left and right singular vectors of $\mathbf{X}$, respectively, and $\mathbf{S}$ $(N \times J)$ is a rectangular diagonal array whose non-zero diagonal elements are its singular values $(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_Q})$;

  2. Compute for each a-th calculated component the so-called $RVDIM$ statistic:

     $$RVDIM_a = \frac{\lambda_a}{\sqrt{\sum_{q=a}^{Q} \lambda_q^2}} \tag{14.30}$$

     where λ_a corresponds to the a-th eigenvalue obtained after the decomposition of $\mathbf{X}$. $RVDIM_a$ is used for testing the statistical significance of the single factors;

- Phase II - Test for the first component:

  3. For $a = 1$, randomly and independently permute the order of the entries within every column of $\mathbf{X}$ constructing a new matrix $\mathbf{X}_{perm}$, featuring uncorrelated variables;

4. Apply full-rank SVD to $\mathbf{X}_{perm}$ and calculate the $RVDIM$ index for the first extracted component:

$$RVDIM_{1,perm} = \frac{\lambda_{1,perm}}{\sqrt{\sum_{q=1}^{Q} \lambda_{q,perm}^2}} \tag{14.31}$$

where $\lambda_{1,perm}$ denotes the first eigenvalue obtained after the decomposition of $\mathbf{X}_{perm}$;

5. Iterate step 3 and 4 to generate a *null*-distribution for $RVDIM_{1,perm}$. If $RVDIM_1$ is found to be higher than e.g. the 99^{th} percentile of the *null*-distribution of $RVIDIM_{1,perm}$, the first component is considered statistically significant;

- Phase III - Test for the a-th component ($a > 1$):

6. For $a > 1$, calculate the residual matrix:

$$\mathbf{E}_a = \mathbf{X} - \sum_{q=1}^{a-1} \mathbf{u}_q \sqrt{\lambda_q} \mathbf{v}_q^{\mathrm{T}} = \mathbf{X} - \sum_{q=1}^{a-1} \mathbf{t}_q \mathbf{p}_q^{\mathrm{T}} \tag{14.32}$$

where $\mathbf{u}_q$, $\mathbf{v}_q$, $\mathbf{t}_q$ and $\mathbf{p}_q$ are the q-th column vectors of $\mathbf{U}$, $\mathbf{V}$, $\mathbf{T}$ and $\mathbf{P}$ (see Eq. 14.29), respectively;

7. Randomly and independently permute the order of the entries within each column of $\mathbf{E}_a$ constructing a new matrix $\mathbf{E}_{a,perm}$. As specified in Section 2, unlike $\mathbf{E}_a$, $\mathbf{E}_{a,perm}$ has rank Q;

8. Perform full-rank SVD on $\mathbf{E}_{a,perm}$ and retain:

$$RVDIM_{a,perm} = \frac{\lambda_{1,perm}}{\sqrt{\sum_{q=1}^{Q} \lambda_{q,perm}^2}} \tag{14.33}$$

where $\lambda_{1,perm}$ is the first eigenvalue obtained after the decomposition of $\mathbf{E}_{a,perm}$;

9. Iterate step 6, 7 and 8 to generate a *null*-distribution for $RVDIM_{a,perm}$. If $RVDIM_a$ is found to be higher than e.g. the 99^{th} percentile of the associated *null*-distribution, the a-th component is considered statistically significant;

The original procedure additionally includes a sequential Bonferroni correction for multiple testing to limit the increase of the Type I error and automatically stops as soon as the first non-significant factor is detected.

It is worth noting that, as detailed in [87], $RVDIM_a$ measures the similarity between the original data reconstruction $\hat{\mathbf{X}}_a = \mathbf{u}_a\sqrt{\lambda_a}\mathbf{v}_a^{\mathrm{T}}$ (where $\mathbf{u}_a/\mathbf{v}_a$ represents the a-th left/right singular vector of $\mathbf{X}$) and $\mathbf{E}_a$. The higher this similarity, the higher the content of relevant information that the a-th component carries.
For the properties of the $RVDIM$ statistic, Dray's method is generally prone to recognise fewer significant components than expected. In fact, $RVDIM_a$ and $RVDIM_{a,perm}$ (for $a = 1, \ldots, Q$) are inversely proportional to the terms $\sqrt{\sum_{q=a}^{Q}\lambda_q^2}$ and $\sqrt{\sum_{q=1}^{Q}\lambda_{q,perm}^2}$, respectively, where λ_q corresponds to the q-th eigenvalue obtained after the decomposition of $\mathbf{X}$, and $\lambda_{q,perm}$ denotes the q-th eigenvalue obtained after the decomposition of $\mathbf{X}_{perm}$ (if $a = 1$) or $\mathbf{E}_{a,perm}$ (if $a > 1$). For each a, $\sum_{q=a}^{Q}\lambda_q$ and $\sum_{q=1}^{Q}\lambda_{q,perm}$ are identical, but that is not the case for $\sum_{q=a}^{Q}\lambda_q^2$ and $\sum_{q=1}^{Q}\lambda_{q,perm}^2$ owing to the redistribution of the total variation of $\mathbf{X}$ (if $a = 1$) or $\mathbf{E}_a$ (if $a > 1$) induced by the permutation of their elements, which modifies the single values of $\lambda_{q,perm}$ with respect to those of λ_q. On average, $\sqrt{\sum_{q=1}^{Q}\lambda_{q,perm}^2}$ is lower than $\sqrt{\sum_{q=a}^{Q}\lambda_q^2}$ when components accounting for relatively large amounts of data variation are still to be deflated. Consequently, for small a, the values of $RVDIM_{a,perm}$ may be overestimated, giving rise to a too conservative selection.

14.3 Annex to Part IV

14.3.1 The JYPLS algorithm

The modified NIPALS algorithm for the JYPLS problem comprises the following 8 steps (see Figure 14.1):

1. $\mathbf{u}_1$ ($N_1 \times 1$) and $\mathbf{u}_2$ ($N_2 \times 1$) are initialised with the first column of $\mathbf{Y}_1$ ($N_1 \times M$) and $\mathbf{Y}_2$ ($N_2 \times M$), respectively;

2. $\mathbf{w}_1$ ($J_1 \times 1$) and $\mathbf{w}_2$ ($J_2 \times 1$) are yielded by regressing $\mathbf{X}_1$ ($N_1 \times J_1$) and $\mathbf{X}_2$ ($N_1 \times J_2$) onto $\mathbf{u}_1$ and $\mathbf{u}_2$:

$$\begin{aligned}\mathbf{w}_1 &= \mathbf{X}_1^{\mathrm{T}}\mathbf{u}_1(\mathbf{u}_1^{\mathrm{T}}\mathbf{u}_1)^{-1}\\ \mathbf{w}_2 &= \mathbf{X}_2^{\mathrm{T}}\mathbf{u}_2(\mathbf{u}_2^{\mathrm{T}}\mathbf{u}_2)^{-1}\end{aligned} \tag{14.34}$$

3. $\mathbf{w}_1$ and $\mathbf{w}_2$ are normalised as:

$$\left\|\begin{matrix}\mathbf{w}_1\\ \mathbf{w}_2\end{matrix}\right\| = 1 \tag{14.35}$$

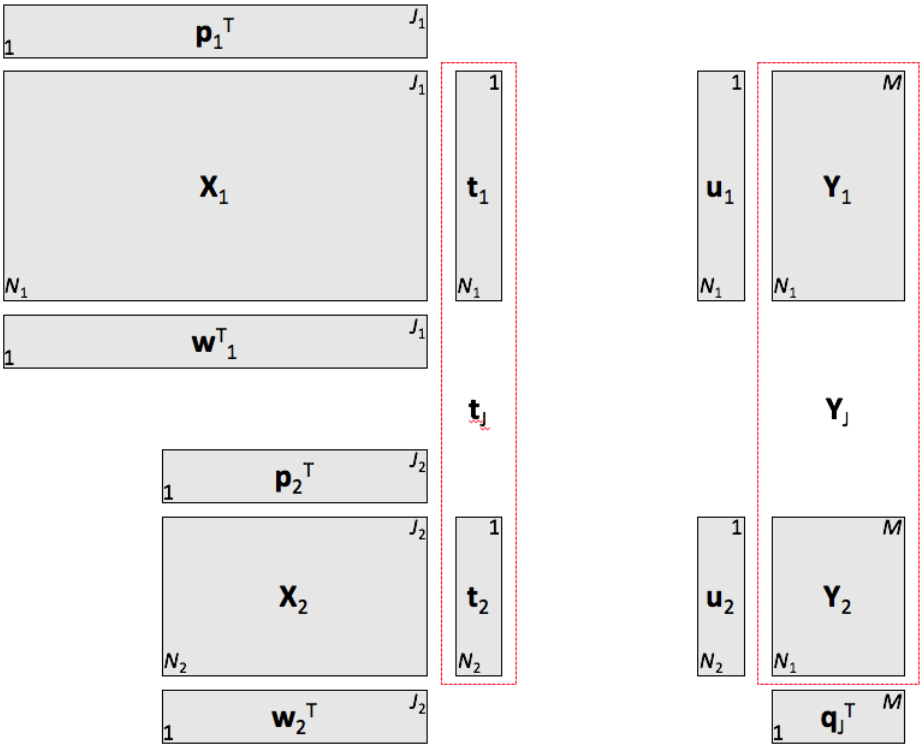

Figure 14.1 Schematic representation of a generic 1-latent variable JYPLS model

4. $\mathbf{X}_1$ and $\mathbf{X}_2$ are regressed onto $\mathbf{w}_1$ and $\mathbf{w}_2$ to obtain $\mathbf{t}_1$ $(N_1 \times 1)$ and $\mathbf{t}_2$ $(N_2 \times 1)$:

$$\begin{aligned} \mathbf{t}_1 &= \mathbf{X}_1\mathbf{w}_1(\mathbf{w}_1^{\mathrm{T}}\mathbf{w}_1)^{-1} \\ \mathbf{t}_2 &= \mathbf{X}_2\mathbf{w}_2(\mathbf{w}_2^{\mathrm{T}}\mathbf{w}_2)^{-1} \end{aligned} \tag{14.36}$$

5. $\mathbf{t}_1$ and $\mathbf{t}_2$ are concatenated into the joint vector $\mathbf{t}_{\mathrm{J}}$ of size $(N_1+N_2)\times 1$, as well as $\mathbf{Y}_1$ and $\mathbf{Y}_2$ into the joint matrix $\mathbf{Y}_{\mathrm{J}}$, which has dimensions $(N_1+N_2)\times M$;

6. The loadings $\mathbf{q}_{\mathrm{J}}$ $(M \times 1)$ are then estimated as:

$$\mathbf{q}_{\mathrm{J}} = \mathbf{Y}_{\mathrm{J}}^{\mathrm{T}}\mathbf{t}_{\mathrm{J}}(\mathbf{t}_{\mathrm{J}}^{\mathrm{T}}\mathbf{t}_{\mathrm{J}})^{-1} \tag{14.37}$$

7. $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are regressed onto $\mathbf{q}_{\mathrm{J}}$ to recompute $\mathbf{u}_1$ and $\mathbf{u}_2$:

$$\begin{aligned} \mathbf{u}_1 &= \mathbf{Y}_1\mathbf{q}_{\mathrm{J}}(\mathbf{q}_{\mathrm{J}}^{\mathrm{T}}\mathbf{q}_{\mathrm{J}})^{-1} \\ \mathbf{u}_2 &= \mathbf{Y}_2\mathbf{q}_{\mathrm{J}}(\mathbf{q}_{\mathrm{J}}^{\mathrm{T}}\mathbf{q}_{\mathrm{J}})^{-1} \end{aligned} \tag{14.38}$$

8. If convergence is checked, $\mathbf{X}_1$, $\mathbf{X}_2$, $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are deflated as:

$$\begin{aligned} \mathbf{E}_{\mathbf{X}_1} &= \mathbf{X}_1 - \mathbf{t}_1\mathbf{p}_1^{\mathrm{T}} \\ \mathbf{E}_{\mathbf{X}_2} &= \mathbf{X}_2 - \mathbf{t}_2\mathbf{p}_2^{\mathrm{T}} \\ \mathbf{E}_{\mathbf{Y}_1} &= \mathbf{Y}_1 - \mathbf{t}_1\mathbf{q}_{\mathrm{J}}^{\mathrm{T}} \\ \mathbf{E}_{\mathbf{Y}_2} &= \mathbf{Y}_2 - \mathbf{t}_2\mathbf{q}_{\mathrm{J}}^{\mathrm{T}} \end{aligned} \tag{14.39}$$

and the whole procedure is iterated for the following latent variables. $\mathbf{p}_1$ ($J_1 \times 1$) and $\mathbf{p}_2$ ($J_2 \times 1$) are calculated by regressing $\mathbf{X}_1$ and $\mathbf{X}_2$ onto $\mathbf{t}_1$ and $\mathbf{t}_2$, respectively:

$$\begin{aligned} \mathbf{p}_1 &= \mathbf{X}_1^{\mathrm{T}}\mathbf{t}_1(\mathbf{t}_1^{\mathrm{T}}\mathbf{t}_1)^{-1} \\ \mathbf{p}_2 &= \mathbf{X}_2^{\mathrm{T}}\mathbf{t}_2(\mathbf{t}_2^{\mathrm{T}}\mathbf{t}_2)^{-1} \end{aligned} \tag{14.40}$$

If the convergence criterion is not met, the updated $\mathbf{u}_1$ and $\mathbf{u}_2$ are set as new input for step 2.

14.3.2 Principal Component Regression (PCR)

The scores $\mathbf{T}$ ($N \times A$) resulting from the PCA decomposition of a particular dataset, say $\mathbf{X}$ ($N \times J$), (see Equation 2.1) can also be related to a certain number of responses - organised in e.g. $\mathbf{Y}$ ($N \times M$) - via what is commonly known as Principal Component Regression (PCR). PCR model structure can be written as:

$$\mathbf{Y} = \mathbf{TB} + \mathbf{F} = \mathbf{XPB} + \mathbf{F} \tag{14.41}$$

where $\mathbf{B}$ ($A \times M$) is the PCR coefficient matrix, which can be estimated as:

$$\mathbf{B} = (\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1}\mathbf{T}^{\mathrm{T}}\mathbf{Y} \tag{14.42}$$

$\mathbf{P}$ ($J \times A$) denotes the PCA loadings array, while $\mathbf{F}$ ($N \times M$) contains the $\mathbf{Y}$-residuals, i.e. the portion of $\mathbf{Y}$ not explained at the chosen rank, A.
Since the columns of $\mathbf{T}$ are orthogonal, the inversion of $\mathbf{T}^{\mathrm{T}}\mathbf{T}$ is not ill-conditioned as would be for $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ in a hypothetical Multiple Linear Regression (MLR) scenario. This prevents possible collinearity among the original measured variables from destabilising the calibration coefficients [75].

14.4 Annex to Part V

14.4.1 Multivariate Curve Resolution-Alternating Least Squares (MCR-ALS)

MCR-ALS is a standard soft-modelling approach (analogous to PCA) for the resolution of multi-component evolving chemical systems into individual components/contributions (not necessarily orthogonal) according to the bilinear model

in Equation 14.43:

$$\mathbf{X} = \mathbf{C}\mathbf{S}^{\mathrm{T}} + \mathbf{E} \tag{14.43}$$

The goal of this method is the decomposition of the matrix $\mathbf{X}$ $(N \times J)$ into the pure profiles $\mathbf{C}$ $(N \times A)$ and $\mathbf{S}^{\mathrm{T}}$ $(A \times J)$ associated to the data variation along its rows and columns, respectively. Also here, $\mathbf{E}$ denotes the residuals array, i.e. the portion of $\mathbf{X}$ not *explained* at the chosen rank, A.
As Equations 2.2 and 2.3 are usually not valid for $\mathbf{C}$ and $\mathbf{S}^{\mathrm{T}}$, Equation 14.43 is iteratively solved by alternately calculating $\mathbf{C}$ and $\mathbf{S}^{\mathrm{T}}$, optimally fitting the experimental data matrix $\mathbf{X}$ and minimising $\mathbf{E}$. Such an optimisation is carried out for a certain number of components and exploiting initial guesses of either $\mathbf{C}$ or $\mathbf{S}^{\mathrm{T}}$, computed e.g. based on a preliminary knowledge of the system under study, by Evolving Factor Analysis (EFA) [187], SIMPLe-to-use Interactive Self-modeling Mixture Analysis (SIMPLISMA) [188] or derived methods. Specific constraints to $\mathbf{C}$ and/or $\mathbf{S}^{\mathrm{T}}$ may be applied in order to restrict the MCR-ALS solution[i] and obtain meaningful response profiles from a physicochemical point of view.
In the case study reported in Section 12.4.3, the scores and the loadings represented in Figures 12.6c, 12.6d, 12.6e, 12.6f, 12.6g and 12.6h were reestimated by executing MCR-ALS on the OTFP reconstructed data, resorting to the final OTFP loadings as preliminary estimates of $\mathbf{S}^{\mathrm{T}}$.

[i]Since the MCR-ALS components are not necessarily orthogonal, the data decomposition is usually non-unique, i.e. different combinations of $\mathbf{C}/\mathbf{S}^{\mathrm{T}}$ may result in an identical data fit.

Bibliography

1. Buydens, L. Towards tsunami-resistant chemometrics. *Anal. Scien.* **0813,** 24–30 (2013).

2. Pearson, K. On lines and planes of closest fit to systems of points in space. *Philos. Mag.* **2,** 559–572 (1901).

3. Hotelling, H. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.* **24,** 417–441 (1933).

4. Eckart, C. & Young, G. The approximation of one matrix by another of lower rank. *Psychometrika* **1,** 211–218 (1936).

5. Wold, S. *et al. Pattern recognition: finding and using regularities in multivariate data* in *Food research and data analysis* First Edition (Elsevier Applied Science, Barking, UK, 2009), 147–188.

6. Barker, M. & Rayens, W. Partial Least Squares for discrimination. *J. Chemometr.* **17,** 166–173 (2003).

7. Cao, D. *et al.* Exploring nonlinear relationships in chemical data using kernel-based methods. *Chemometr. Intell. Lab.* **107,** 106–115 (2011).

8. Walczak, B. & Massart, D. The radial basis functions-partial least squares approach as a flexible non-linear regression technique. *Anal. Chim. Acta* **331,** 177–185 (1996).

9. Walczak, B. & Massart, D. Application of radial basis functions-partial least squares to non-linear pattern recognition problems: diagnosis of process faults. *Anal. Chim. Acta* **331,** 187–193 (1996).

10. Yacoub, F. & MacGregor, J. Product optimization and control in the latent variable space of nonlinear PLS models. *Chemometr. Intell. Lab.* **70,** 63–74 (2004).

11. Wold, S., Kettaneh-Wold, N. & Skagerberg, B. Nonlinear PLS modeling. *Chemometr. Intell. Lab.* **7,** 53–65 (1989).

12. Wold, S. Nonlinear Partial Least Squares modelling II. Spline inner relation. *Chemometr. Intell. Lab.* **14,** 71–84 (1992).

13. Frank, I. A nonlinear PLS model. *Chemometr. Intell. Lab.* **8,** 109–119 (1990).

14. Höskuldsson, A. Quadratic PLS regression. *J. Chemometr.* **6,** 307–334 (1992).

15. Berglund, A. & Wold, S. INLR, Implicit Non-linear Latent variable Regression. *J. Chemometr.* **11,** 141–156 (1997).

16. Gasteiger, J. & Zupan, J. Neural networks in chemistry. *Angew. Chem. Int. Ed. Engl.* **32,** 503–527 (1993).

17. Schölkopf, B. & Smola, A. *Learning with Kernels* First Edition (MIT Press, Cambridge, USA, 2002).

18. Li, H., Liang, Y. & Xu, Q. Support vector machines and its applications in chemistry. *Chemometr. Intell. Lab.* **95,** 188–198 (2009).

19. Williams, P. Influence of water on prediction of composition and quality factors: the aquaphotomics of low moisture agricultural materials. *J. Near Infrared Spectroscop.* **17,** 315–328 (2009).

20. Tan, C. & Li, M. Mutual information-induced interval selection combined with kernel partial least squares for near-infrared spectral calibration. *Spectrochim. Acta Pt. A-Mol. Biomol. Spectrosc.* **71,** 1266–1273 (2008).

21. Embrechts, M. & Ekins, S. Classification of metabolites with kernel-partial least squares (K-PLS). *Drug Metab. Dispos.* **35,** 325–327 (2007).

22. Arenas-Garcia, J. & Camps-Valls, G. Efficient kernel orthonormalized PLS for remote sensing applications. *IEEE Trans. Geosci. Remote Sens.* **46,** 2872–2881 (2008).

23. Struc, V. & Pavesic, N. Gabor-based kernel partial-least-squares discrimination features for face recognition. *Informatica* **20,** 115–138 (2009).

24. Sun, R. & Tsung, F. A kernel-distance-based multivariate control chart using support vector methods. *Int. J. Prod. Res.* **41,** 2975–2989 (2003).

25. Lee, J., Yoo, C., Choi, S., Vanrolleghem, P. & Lee, I. Nonlinear process monitoring using kernel principal component analysis. *Chem Eng. Sci.* **59,** 223–234 (2004).

26. Bennett, K. & Embrechts, M. *Advances in Learning Theory: Methods, Models and Applications* First Edition (IOS Press, Amsterdam, The Netherlands, 2003).

27. Kewley, R., Embrechts, M. & Breneman, C. Data strip mining for the virtual design of pharmaceuticals with neural networks. *IEEE Trans. Neural Netw.* **11,** 668–679 (2000).

28. Üstün, B., Melssen, W. & Buydens, L. Visualization and interpretation of support vector regression models. *Anal. Chim. Acta* **595,** 299–309 (2007).

29. Alcala, C. & Qin, S. Reconstruction-based contribution for process monitoring with kernel principal component analysis. *Ind. Eng. Chem. Res.* **49,** 7849–7857 (2010).

30. Gower, J. & Hardings, S. Nonlinear biplots. *Biometrika* **75,** 445–455 (1988).

31. Krooshof, P., Üstün, B., Postma, G. & Buydens, L. Visualisation and recovery of the (bio)chemical interesting variables in data analysis with support vector machine classification. *Anal. Chem.* **82,** 7000–7007 (2010).

32. Postma, G., Krooshof, P. & Buydens, L. Opening the kernel of Kernel Partial Least Squares and Support Vector Machines. *Anal. Chim. Acta* **705,** 123–134 (2011).

33. Smolinska, A. *et al.* Interpretation and visualization of nonlinear data fusion in kernel space: study on metabolomic characterization of progression of multiple sclerosis. *PLoS One* **7,** e38163 (2012).

34. Engel, J., Postma, G., van Peufflik, I., Blanchet, L. & Buydens, L. Pseudo-sample trajectories for variable interaction detection in Dissimilarity Partial Least Squares. *Chemometr. Intell. Lab.* **146,** 89–101 (2015).

35. Shapiro, L. & Stockman, G. *Computer vision* First Edition (Prentice Hall Inc., Upper Saddle River, USA, 2001).

36. Szeliski, R. *Computer vision - Algorithms and applications* First Edition (Springer-Verlag Ltd., London, UK, 2011).

37. Prats-Montalbán, J., de Juan, A. & Ferrer, A. Multivariate image analysis: a review with applications. *Chemometr. Intell. Lab* **107,** 1–23 (2011).

38. Bevilacqua, M. *et al. Classification and class-modelling* in *Chemometrics in food chemistry* First Edition (Elsevier B.V., Oxford, UK, 2013), 171–233.

39. Manning, C., Raghavan, P. & Schütze, H. *Introduction to information retrieval* First Edition (Cambridge University Press, Cambridge, UK, 2008).

40. MacQueen, J. *Some methods for classification and analysis of multivariate observations* in *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability - Volume 1* (University of California Press, Berkeley, USA, 1967), 281–297.

41. Haralick, R. Statistical and structural approaches to texture. *P. IEEE* **67,** 786–804 (1979).

42. Felzenszwalb, P. & Huttenlocher, D. Efficient graph-based image segmentation. *Int. J. Comput. Vision* **59,** 167–181 (2004).

43. Prats-Montalbán, J. & Ferrer, A. Integration of colour and textural information in multivariate image analysis: defect detection and classification issues. *J. Chemometr.* **21,** 10–23 (2007).

44. Prats-Montalbán, J. *Control estadístico de procesos mediante análisis multivariante de imágenes* PhD thesis (Departamento de Estadística e Investigación Operativa Aplicadas y Calidad, Universitat Politècnica de València, 2005).

45. López, F., Prats, J., Ferrer, A. & Valiente, J. Defect detection in random colour textures using the MIA T^2 defect maps. *Lect. Notes Comput. Sc.* **4142,** 752–763 (2006).

46. Ho, P. *Image segmentation* First Edition (In Tech d.o.o., Rijeka, Croatia, 2011).

47. Pal, N. & Pal, S. A review on image segmentation techniques. *Pattern Recogn.* **26,** 1277–1294 (1993).

48. *MATLAB R2012b (8.0.0.783)*, The MathWorks Inc., Natick, USA.

49. *R Development Core Team, R: a language and environment for statistical computing* (R Foundation for Statistical Computing, Wien, Austria, 2008).

50. Gonzalez, R. & Woods, R. *Digital image processing* Third Edition (Prentice Hall Inc., Upper Saddle River, USA, 2007).

51. Joliffe, I. *Principal Component Analysis* Second Edition (Springer-Verlag Inc., New York, USA, 2002).

52. Geladi, P. & Kowalski, B. Partial Least Squares Regression: a tutorial. *Anal. Chim. Acta* **185,** 1–17 (1986).

53. Hirschfeld, H. A connection between correlation and contingency. *Math. Proc. Cambridge* **31,** 520–524 (1935).

54. Camacho, J., Picó, J. & Ferrer, A. Bilinear modelling of batch processes. Part I: theoretical discussion. *J. Chemometr.* **22,** 299–308 (2008).

55. Wold, S., Kettaneh-Wold, N., MacGregor, J. & Dunn, K. *Batch process modeling and MSPC* in *Comprehensive Chemometrics* First Edition Vol. 2 (Elsevier B.V., Oxford, UK, 2009), 163–197.

56. Nomikos, P. & MacGregor, J. Multivariate SPC charts for monitoring batch processes. *Technometrics* **37,** 41–59 (1995).

57. García-Muñoz, S., Kourti, T., MacGregor, J., Mateos, A. & Murphy, G. Troubleshooting of an industrial batch process using multivariate methods. *Ind. Eng. Chem. Res.* **42,** 3592–3601 (2003).

58. Quintás, G. *et al.* Chemometric approaches to improve PLSDA model outcome for predicting human non-alcoholic fatty liver disease using UPLC-MS as a metabolic profiling tool. *Metabolomics* **8,** 86–98 (2012).

59. Camacho, J., Picó, J. & Ferrer, A. Bilinear modelling of batch processes. Part II: a comparison of PLS soft-sensors. *J. Chemometr.* **22,** 533–547 (2008).

60. González-Martínez, J., Camacho, J. & Ferrer, A. Bilinear modelling of batch processes. Part III: parameter stability. *J. Chemometr.* **28,** 10–27 (2014).

61. Tracy, N., Young, J. & Mason, R. Multivariate control charts for individual observations. *J. Qual. Technol.* **24,** 88–95 (1992).

62. Box, G. Some theorems on quadratic forms applied in the study of analysis of variance problems: effect of inequality of variance in one-way classification. *Ann. Math. Stat.* **25,** 290–302 (1954).

63. Jackson, J. & Mudholkar, G. Control procedures for residuals associated to Principal Component Analysis. *Technometrics* **21,** 341–349 (1979).

64. De Noord, O. Improvements to multivariate data analysis and monitoring of batch processes by multilevel methods. *J. Chemometr.* **26,** 340–344 (2012).

65. Camacho, J. & Ferrer, A. Cross-validation in PCA models with the element-wise k-fold (*ekf*) algorithm: theoretical aspects. *J. Chemometr.* **26,** 361–373 (2012).

66. Kourti, T. & MacGregor, J. Multivariate SPC methods for process and product monitoring. *J. Qual. Technol.* **28,** 409–428 (1996).

67. Westerhuis, J., Gurden, S. & Smilde, A. Generalized contribution plots in multivariate statistical process monitoring. *Chemometr. Intell. Lab.* **51,** 95–114 (2000).

68. Kassidas, A., MacGregor, J. & Taylor, P. Synchronization of batch trajectories using Dynamic Time Warping. *AIChe J.* **44,** 864–875 (1998).

69. Cornell, J. *Experiments with mixtures - Designs, models and the analysis of mixture data* Third Edition (John Wiley & Sons Inc., New York, USA, 2002).

70. Rao, C. *Linear statistical inference and its applications* Second Edition (John Wiley & Sons Inc., New York, USA, 1973).

71. Aitken, A. On least squares and linear combination of observations. *Proc. R. Soc. Edinb.* **55,** 42–48 (1936).

72. Kettaneh-Wold, N. Analysis of mixture data with Partial Least Squares. *Chemometr. Intell. Lab.* **14,** 57–69 (1992).

73. Eriksson, L., Johansson, E. & Wikström, C. Mixture design - Design generation, PLS analysis, and model usage. *Chemometr. Intell. Lab.* **43,** 1–24 (1998).

74. Martens, H. & Næs, T. *Multivariate Calibration* First Edition (John Wiley & Sons Ltd., New York, USA, 1989).

75. Wold, S., Ruhe, A., Wold, H. & Dunn III, W. The collinearity problem in linear regression. The Partial Least Squares (PLS) approach to generalized inverses. *SIAM J. Sci. Stat. Comput.* **5,** 735–743 (1984).

76. Höskuldsson, A. PLS regression methods. *J. Chemometr.* **2,** 211–228 (1988).

77. Alman, D. & Pfeifer, C. Empirical colorant mixture models. *Color Res. Appl.* **12,** 210–222 (1987).

78. Bro, R., Kjeldahl, K., Smilde, A. & Kiers, H. Cross-validation of component models: a critical look at current methods. *Anal. Bioanal. Chem.* **390,** 1241–1251 (2008).

79. Camacho, J. & Ferrer, A. Cross-validation in PCA models with the element-wise *ekf* algorithm: practical aspects. *Chemometr. Intell. Lab.* **131,** 37–50 (2014).

80. Saccenti, E. & Camacho, J. Determining the number of components in Principal Components Analysis: a comparison of statistical, crossvalidation and approximated methods. *Chemometr. Intell. Lab.* **149,** 99–116 (2015).

81. Kaiser, H. The application of electronic computers to Factor Analysis. *Educ. Psychol. Meas.* **20,** 141–151 (1960).

82. Velicer, W. Determining the number of components from the matrix of partial correlations. *Psychometrika* **41,** 321–327 (1976).

83. Cattell, R. The scree test for the number of factors. *Multivar. Behav. Res.* **1,** 245–276 (1966).

84. Bartlett, M. A note on the multiplying factors for various χ^2 approximations. *J. Roy. Stat. Soc. B Met.* **16,** 296–298 (1954).

85. Saccenti, E., Smilde, A., Westerhuis, J. & Hendriks, M. Tracy-Widom statistic for the largest eigenvalue of autoscaled real matrices. *J. Chemometr.* **25,** 644–652 (2011).

86. Horn, J. A rationale and test for the number of factors in factor analysis. *Psychometrika* **30,** 179–185 (1965).

87. Dray, S. On the number of principal components: a test of dimensionality based on measurements of similarity between matrices. *Comput. Stat. Data An.* **52,** 2228–2237 (2008).

88. Kosanovich, K., Dahl, K. & Piovoso, M. Improved process understanding using multiway Principal Component Analysis. *Ind. Eng. Chem. Res.* **35,** 138–146 (1996).

89. Camacho, J., Picó, J. & Ferrer, A. Data understanding with PCA: structural and variance information plots. *Chemometr. Intell. Lab.* **100,** 48–56 (2010).

90. Camacho, J. Missing-data theory in the context of exploratory data analysis. *Chemometr. Intell. Lab.* **103,** 8–18 (2010).

91. Camacho, J. Observation-based missing data methods for exploratory data analysis to unveil the connection between observations and variables in latent subspace models. *J. Chemometr.* **25,** 592–600 (2011).

92. Vieira, V. Permutation tests to estimate significances on Principal Components Analysis. *Comput. Ecol. Softw.* **2,** 103–123 (2012).

93. Peres-Neto, P., Jackson, D. & Somers, K. How many principal components? Stopping rules for determining the number of non-trivial axes revisited. *Comput. Stat. Data An.* **49,** 974–997 (2005).

94. Endrizzi, I., Gasperi, F., Rødbotten, M. & Næs, T. Interpretation, validation and segmentation of preference mapping models. *Food Qual. Prefer.* **32,** 198–209 (2014).

95. Saccenti, E. & Timmerman, M. Considering Horn's parallel analysis from a random matrix theory point of view. *Psychometrika* **82,** 186–209 (2017).

96. Bro, R., Nielsen, H., Stefánsson, G. & Skåra, T. A phenomenological study of ripening of salted herring. Assessing homogeneity of data from different countries and laboratories. *J. Chemometr.* **16,** 81–88 (2002).

97. Henry, R., Park, E. & Spiegelman, C. Comparing a new algorithm with the classical methods for estimating the number of factors. *Chemometr. Intell. Lab.* **48,** 91–97 (1999).

98. Rødbotten, M. *et al.* A cross-cultural study of preference for apple juice with different sugar and acid contents. *Food Qual. Prefer.* **20,** 277–284 (2009).

99. Kowalski, B., Schatzki, T. & Stross, F. Classification of archaeological artifacts by applying pattern recognition to trace element data. *Anal. Chem.* **44,** 2176–2180 (1972).

100. *OpenMV website,* `http://openmv.net/`

101. *PLS_Toolbox 7.0.2*, Eigenvector Research Inc., Manson, USA.

102. *Infometrix, Inc. website,* `https://infometrix.com/`

103. Harrison, D. & Rubinfeld, D. Hedonic housing prices and the demand for clean air. *J. Environ. Econ. Manag.* **5,** 81–102 (1978).

104. Næs, T., Brockhoff, P. & Tomic, O. *Statistics for sensory and consumer science* First Edition (John Wiley & Sons Ltd., Chichester, United Kingdom, 2010).

105. Næs, T., Berget, I., Liland, K., Ares, G. & Varela, P. Estimating and interpreting more than two consensus components in projective mapping: INDSCAL vs. Multiple Factor Analysis (MFA). *Food Qual. Prefer.* **58,** 45–60 (2017).

106. Smilde, A. *et al.* Common and distinct components in data fusion. *arXiv,* arXiv:1607.02328 (2016).

107. Kettenring, J. Canonical analysis of several sets of variables. *Biometrika* **58,** 433–460 (1971).

108. Van de Geer, J. Linear relations among k sets of variables. *Psychometrika* **49,** 79–94 (1984).

109. Tauler, R., Smilde, A. & Kowalski, B. Selectivity, local rank, three-way data analysis and ambiguity in multivariate curve resolution. *J. Chemometr.* **9,** 31–58 (1995).

110. Van Deun, K., Smilde, A., Thorrez, L., Kiers, H. & Van Mechelen, I. Identifying common and distinctive processes underlying multiset data. *Chemometr. Intell. Lab.* **129,** 40–51 (2013).

111. Timmerman, M. & Kiers, H. Four simultaneous component models for the analysis of multivariate time series from more than one subject to model intraindividual and interindividual differences. *Psychometrika* **68,** 105–121 (2003).

112. Måge, I., Mevik, B. & Næs, T. Regression models with process variables and parallel blocks of raw material measurements. *J. Chemometr.* **22,** 443–456 (2008).

113. Kiers, H. & ten Berge, J. Hierarchical relations between methods for simultaneous component analysis and a technique for rotation to a simple simultaneous structure. *Brit. J. Math. Stat. Psy.* **47,** 109–126 (1994).

114. Van Deun, K., Smilde, A., van der Werf, M., Kiers, H. & Van Mechelen, I. A structured overview of simultaneous component based data integration. *BMC Bioinformatics* **10,** 246–260 (2009).

115. Schouteden, M., Van Deun, K., Pattyn, S. & Van Mechelen, I. SCA with rotation to distinguish common and distinctive information in linked data. *Behav. Res. Methods* **45,** 822–833 (2013).

116. Van Deun, K. *et al.* DISCO-SCA and properly applied GSVD as swinging methods to find common and distinctive processes. *PLoS One* **7,** e37840 (2012).

117. Paige, C. & Saunders, M. Towards a generalized singular value decomposition. *SIAM J. Numer. Anal.* **18,** 398–405 (1981).

118. Friedland, S. A new approach to generalized singular value decomposition. *SIAM J. Matrix Anal. A.* **27,** 434–444 (2005).

119. Schouteden, M., Van Deun, K. & Van Mechelen, I. *ECO-POWER: a novel method to reveal common mechanisms underlying linked data* in *Proceedings of the 20th International Conference on Computational Statistics (COMPSTAT 2012)* (Physica-Verlag, Heidelberg, Germany, 2012), 757–768.

120. Hotelling, H. Relations between two sets of variates. *Biometrika* **28,** 321–377 (1936).

121. Kiers, H. & Smilde, A. A comparison of various methods for Multivariate Regression with highly collinear variables. *Stat. Method. Appl.* **16,** 193–228 (2007).

122. Van den Berg, R., Rubingh, C., Westerhuis, J., van der Werf, M. & Smilde, A. Metabolomics data exploration guided by prior knowledge. *Anal. Chim. Acta* **651,** 173–181 (2009).

123. Dahl, T. & Næs, T. A bridge between Tucker-1 and Carroll's generalized canonical analysis. *Comput. Stat. Data An.* **50,** 3086–3098 (2006).

124. Tenenhaus, A. & Tenenhaus, M. Regularized Generalized Canonical Correlation Analysis. *Psychometrika* **76,** 257–284 (2011).

125. Trygg, J. O2-PLS for qualitative and quantitative analysis in multivariate calibration. *J. Chemometr.* **16,** 283–293 (2002).

126. Hanafi, M. & Kiers, H. Analysis of K sets of data, with differential emphasis on agreement between and within sets. *Comput. Stat. Data An.* **51,** 1491–1508 (2006).

127. Löfsted, T. & Trygg, J. OnPLS - A novel multiblock method for the modelling of predictive and orthogonal variation. *J. Chemometr.* **25,** 441–455 (2011).

128. Nakaya, H. *et al.* Systems biology of vaccination for seasonal influenza in humans. *Nat. Immunol.* **12,** 786–795 (2011).

129. González-Martínez, J., de Noord, O. & Ferrer, A. Multisynchro: a novel approach for batch synchronization in scenarios of multiple asynchronisms. *J. Chemometr.* **28,** 462–475 (2014).

130. García Muñoz, S., MacGregor, J. & Kourti, T. Product transfer between sites using Joint-Y PLS. *Chemometr. Intell. Lab.* **79,** 101–114 (2005).

131. Feudale, R. *et al.* Transfer of multivariate calibration models: a review. *Chemometr. Intell. Lab.* **64,** 181–192 (2002).

132. Wang, Y., Veltkamp, D. & Kowalski, B. Multivariate instrument standardization. *Anal. Chem.* **63,** 2750–2756 (1991).

133. Bouveresse, E. & Massart, D. Standardisation of near-infrared spectrometric instruments: a review. *Vib. Spectrosc.* **11,** 3–15 (1996).

134. Fearn, T. Standardisation and calibration transfer for near infrared instruments: a review. *J. Near Infrared Spec.* **9,** 229–244 (2001).

135. De Noord, O. Multivariate calibration standardisation. *Chemometr. Intell. Lab.* **25,** 85–97 (1994).

136. Andrews, D. & Wentzell, P. Applications of Maximum Likelihood Principal Component Analysis: incomplete data sets and calibration transfer. *Anal. Chim. Acta* **350,** 341–352 (1997).

137. Folch-Fortuny, A., Arteaga, F. & Ferrer, A. PCA model building with missing data: new proposals and a comparative study. *Chemometr. Intell. Lab.* **146,** 77–88 (2015).

138. Nelson, P. *The treatment of missing measurements in PCA and PLS models* PhD thesis (MacMaster University, 2002).

139. Arteaga, F. *Control estadístico multivariante de procesos con datos faltantes mediante Análisis de Componentes Principales* PhD thesis (Universitat Politècnica de València, 2003).

140. Folch-Fortuny, A., Arteaga, F. & Ferrer, A. Assessment of Maximum Likelihood PCA missing data imputation. *J. Chemometr.* **30,** 386–393 (2016).

141. Nelson, P., Taylor, P. & MacGregor, J. Missing data methods in PCA and PLS: score calculations with incomplete observations. *Chemometr. Intell. Lab.* **35,** 45–65 (1996).

142. Arteaga, F. & Ferrer, A. Dealing with missing data in MSPC: several methods, different interpretations, some examples. *J. Chemometr.* **16,** 408–418 (2002).

143. Arteaga, F. & Ferrer, A. Framework for regression-based missing data imputation methods in on-line MSPC. *J. Chemometr.* **19,** 439–447 (2005).

144. Folch-Fortuny, A., Arteaga, F. & Ferrer, A. Missing Data Imputation Toolbox for MATLAB. *Chemometr. Intell. Lab.* **154,** 93–100 (2016).

145. Alves Barata, J. & Hussein, M. The Moore-Penrose pseudoinverse: a tutorial review of the theory. *Braz. J. Phys.* **42,** 146–165 (2012).

146. Kennard, R. & Stone, L. Computer aided design of experiments. *Technometrics* **11,** 137–148 (1969).

147. Daszykowski, M., Walczak, B. & Massart, D. Representative subset selection. *Anal. Chim. Acta* **468,** 91–103 (2002).

148. Vitale, R. *et al.* A rapid and non-invasive method for authenticating the origin of pistachio samples by NIR spectroscopy and chemometrics. *Chemometr. Intell. Lab.* **121,** 90–99 (2013).

149. Wise, B., Martens, H., Høy, M., Bro, R. & Brockhoff, P. *Calibration transfer by Generalized Least Squares* tech. rep. (Eigenvector Research Inc., Manson, USA, `www.eigenvector.com/Docs/GLS_Standardization.pdf`).

150. Bouveresse, E. & Massart, D. Improvement of the piecewise direct standardisation procedure for the transfer of NIR spectra for multivariate calibration. *Chemometr. Intell. Lab.* **2,** 201–213 (1996).

151. Wang, Y. & Kowalski, B. Calibration transfer and measurement stability of near-infrared spectrometers. *Appl. Spectrosc.* **46,** 764–771 (1992).

152. Fernández-Pierna, J., Vermeulen, P., Lecler, B., Baeten, V. & Dardenne, P. Calibration transfer from dispersive instruments to handheld spectrometers. *Appl. Spectrosc.* **64,** 644–647 (2010).

153. Moore, G. Cramming more components onto integrated circuits. *Electronics* **38,** 114–117 (1965).

154. Katal, A., Wazid, M. & Goudar, R. *Big Data: issues, challenges, tools and good practices* in *Sixth International Conference on Contemporary Computing (IC3), Noida* (IEEE, Piscataway, USA, 2013), 404–409.

155. Salomon, D. & Motta, G. *Handbook of Data Compression* Fifth Edition (Springer-Verlag Inc., London, UK, 2010).

156. Martens, H. Quantitative Big Data: where chemometrics can contribute. *J. Chemometr.* **29,** 563–581 (2015).

157. Wold, S. A theoretical foundation of extrathermodynamic relationships (Linear Free Energy relationships). *Chem. Scripta* **5,** 97–106 (1974).

158. Wold, S. & Sjöström, M. Chemometrics and its roots in physical organic chemistry. *Acta Chem. Scand.* **52,** 517–523 (1998).

159. Martens, H. *et al. PLS-based multivariate metamodeling of dynamic systems* in *New perspectives in Partial Least Squares and related methods* First Edition Vol. 56 (Springer-Verlag Inc., New York, USA, 2013), 3–30.

160. Balsubramani, A., Dasgupta, S. & Freund, Y. *The fast convergence of incremental PCA* in *Advances in Neural Information Processing Systems 26* (Curran Associates Inc., Red Hook, USA, 2013), 3174–3182.

161. Halko, N., Martinsson, P., Shkolnisky, Y. & Tygert, M. An algorithm for the Principal Component Analysis of large data sets. *SIAM J. Sci. Comput.* **33,** 2580–2594 (2011).

162. Kettaneh, N., Berglund, A. & Wold, S. PCA and PLS with very large data sets. *Comput. Stat. Data An.* **48,** 69–85 (2005).

163. Rabani, E. & Toledo, S. *Out-of-core SVD and QR decompositions* in *SIAM Proceedings Series* (Society for Industrial and Applied Mathematics, Philadelphia, USA, 2001).

164. Vogt, F. & Tacke, M. Fast Principal Component Analysis of large data sets. *Chemometr. Intell. Lab.* **59,** 1–18 (2001).

165. Camacho, J. Visualizing Big Data with Compressed Score Plots: approach and research challenges. *Chemometr. Intell. Lab.* **135,** 110–125 (2014).

166. Barnes, R., Dhanoa, M. & Lister, S. Standard Normal Variate transformation and de-trending of near-infrared diffuse reflectance spectra. *Appl. Spectrosc.* **43,** 772–777 (1989).

167. Martens, H., Jensen, S. & Geladi, P. *Multivariate linearity transformation for near-infrared reflectance spectrometry* in *Proceedings of the Nordic Symposium on Applied Statistics* (Stokkand Forlag Publ., Stavanger, Norway, 1983), 208–234.

168. Geladi, P., MacDougall, D. & Martens, H. Linearization and scatter-correction for near-infrared reflectance spectra of meat. *Appl. Spectrosc.* **39,** 491–500 (1985).

169. Martens, H., Nielsen, J. & Engelsen, S. Light scattering and light absorbance separated by Extended Multiplicative Signal Correction. Application to near-infrared transmission analysis of powder mixtures. *Anal. Chem.* **75,** 394–404 (2003).

170. Zaikin, A. & Zhabotinsky, A. Concentration wave propagation in two-dimensional liquid-phase self-oscillating system. *Nature* **225,** 535–537 (1970).

171. Nordkvist, K. *Ocean color retrieval using DroneSpex - A miniature imaging spectrometer* MA thesis (Luleå University of Technology, 2007).

172. Esbensen, K. *Multivariate Data Analysis - in practice* Fifth Edition (CAMO Process AS, Oslo, Norway, 2002).

173. Jaumot, J., Gargallo, R., de Juan, A. & Tauler, R. A graphical user-friendly interface for MCR-ALS: a new tool for multivariate curve resolution in MATLAB. *Chemometr. Intell. Lab.* **76,** 101–110 (2005).

174. Comon, P. Independent Component Analysis, a new concept? *Signal Process.* **36,** 287–314 (1994).

175. Hyvärinen, A., Karhunen, J. & Oja, E. *Independent Component Analysis* First Edition (John Wiley & Sons Inc., New York, USA, 2001).

176. Hitchcock, F. The expression of a tensor or a polyadic as a sum of products. *J. Math. Phys. Camb.* **6,** 164–189 (1927).

177. Bro, R. PARAFAC. Tutorial and applications. *Chemometr. Intell. Lab.* **38,** 149–171 (1997).

178. Wold, S. Pattern recognition by means of disjoint principal components models. *Pattern Recogn.* **8,** 127–139 (1976).

179. Wold, S. & Sjöström, M. *SIMCA: a method for analyzing chemical data in terms of similarity and analogy* in *Chemometrics: Theory and Application* First Edition Vol. 52 (American Chemical Society, Washington D.C., USA, 1977), 243–282.

180. Kourti, T. Application of latent variable methods to process control and multivariate statistical process control in industry. *Int. J. Adapt. Control Signal Process.* **19,** 213–246 (2005).

181. Courrier, P. Straight monotonic embedding of data sets in Euclidean space. *Neural Networks* **15,** 1185–1196 (2002).

182. Humphreys, L. & Montanelli, R. An investigation of the parallel analysis criterion for determining the number of common factors. *Multivar. Behav. Res.* **10,** 193–206 (1975).

183. Zwick, W. & Velicer, W. Comparison of five rules for determining the number of components to retain. *Psychol. Bull.* **99,** 432–442 (1986).

184. Glorfeld, L. An improvement on Horn's parallel analysis methodology for selecting the correct number of factors to retain. *Educ. Psychol. Meas.* **55,** 377–393 (1995).

185. Thompson, B. & Daniel, L. Factor analytic evidence for the construct validity of scores: a historical overview and some guidelines. *Educ. Psychol. Meas.* **56,** 197–208 (1996).

186. Ledesma, R. & Valero-Mora, P. Determining the number of factors to retain in EFA: an easy-to-use computer program for carrying out parallel analysis. *PARE* **12,** 1–11 (2007).

187. Maeder, M. Evolving Factor Analysis for the resolution of overlapping chromatographic peaks. *Anal. Chem.* **59,** 527–530 (1987).

188. Windig, W. & Guilment, J. Interactive self-modeling mixture analysis. *Anal. Chem.* **63,** 1425–1432 (1991).

189. Lock, E., Hoadley, K., Marron, J. & Nobel, A. Joint and Individual Variation Explained (JIVE) for intergrated analysis of multiple data types. *Ann. Appl. Stat.* **7,** 523–542 (2013).


manufacturing
industrial process
spectroscopy
data processing
pseudo-samples
calibration transfer
mixture designs
variables
latent variables
components
permutation
BIG DATA
projection
information
regression
data streams
exploration
images
bilinear
batch
matrix
PCA
PLS
PLSDA
control
recordings
measurements
classification
product
multivariate
segmentation
near-infrared
array
distinctive
monitoring
prediction
jackknifing
common
fault
statistics
synchronisation
correlation
chemometrics
supervised
modelling
kernel methods
high-dimensional
data analysis
diagnosis
cross-validation
contribution