Linköping studies in science and technology. Dissertations, No. 1162

Multivariate Exploration and Processing of Sensor Data – applications with multidimensional sensor systems

Henrik Petersson

Linköping University
INSTITUTE OF TECHNOLOGY

Department of Physics, Chemistry and Biology
Linköpings universitet, SE-581 83 Linköping, Sweden

Linköping 2008

During the course of the research underlying this thesis, Henrik Petersson was enrolled in Forum Scientium, a multidisciplinary doctoral programme at Linköping University, Sweden.

Multivariate Exploration and Processing of Sensor Data

ISBN 978-91-7393-841-9
ISSN 0345-7524

Printed by LiU-Tryck, Linköping, Sweden 2008

To my Family

Abstract

A sensor is a device that transforms a physical, chemical, or biological stimulus into a readable signal. The integral part that sensors make in modern technology is considerable and many are those trying to take the development of sensor technology further. Sensor systems are becoming more and more complex and may contain a wide range of different sensors, where each may deliver a multitude of signals.

Although the data generated by modern sensor systems contain lots of information, the information may not be clearly visible. Appropriate handling of data becomes crucial to reveal what is sought, but unfortunately, that process is not always straightforward and there are many aspects to consider. Therefore, analysis of multidimensional sensor data has become a science.

The topic of this thesis is signal processing of multidimensional sensordata. Surveys are given on methods to explore data and to use the data to quantify or classify samples. It is also discussed how to avoid the rise of artifacts and how to compensate for sensor deficiencies. Special interest is put on methods being practically applicable to chemical gas sensors. The merits and limitations of chemical sensors are discussed and it is argued that multivariate data analysis plays an important role using such sensors.

The contribution made to the public by this thesis is primarily on techniques dealing with difficulties related to the operation of sensors in applications. In the *second paper*, a method is suggested that aims at suppressing the negative effects caused by unwanted sensor-to-sensor differences. If such differences are not suppressed sufficiently, systems where sensors occasionally must be replaced may degrade and lose performance. The strong-point of the suggested method is its relative ease of use considering large-scale production of sensor components and when integrating sensors into mass-market products. The *third paper* presents a method that facilitates and speeds up the process of assembling an array of sensors that is optimal for a particular application. The method combines multivariate data analysis with the 'Scanning Light Pulse Technique'. In the *first and fourth papers*, the problem of source separation is studied. In two separate applications, one using gas sensors for combustion control and one using acoustic sensors for ground surveillance, it has been identified that the current sensors outputs mixtures of both interesting- and interfering signals. By different means, the two papers applies and evaluates methods to extract the relevant information under such circumstances.

■

Populärvetenskaplig sammanfattning

En sensor är en komponent som överför en fysikalisk, kemisk, eller biologisk storhet eller kvalitet till en utläsbar signal. Sensorer utgör idag en viktig del i flertalet högteknologiska produkter och sensorforskning är ett aktivt område.

Komplexiteten på sensorbaserade system ökar och det blir möjligt att registrera allt fler olika typer av mätsignaler. Mätsignalerna är inte alltid direkt tydbara, varvid signalbehandling blir ett väsentligt verktyg för att vaska fram den viktiga information som sökes. Signalbehandling av sensorsignaler är dessvärre inte en okomplicerad procedur och det finns många aspekter att beakta. Av denna anledning har signalbehandling och analys av sensorsignaler utvecklats till ett eget forskningsområde.

Denna avhandling avhandlar metoder för att analysera komplexa multidimensionella sensorsignaler. En introduktion ges till metoder för att, utifrån mätningar, klassificera och kvantifiera egenskaper hos mätobjekt. En överblick ges av de effekter som kan uppstå på grund av imperfektioner hos sensorerna och en diskussion föres kring metoder för att undvika eller lindra de problem som dessa imperfektioner kan ge uppkomst till. Speciell vikt lägges vid sådana metoder som medför en direkt applicerbarhet och nytta för system av kemiska sensorer.

I avhandlingen ingår fyra artiklar, som vart och en belyser hur de metoder som beskrivits kan användas i praktiska situationer.

■

List of Papers

Papers included in thesis

I **Initial studies on the possibility to use chemical sensors to monitor and control boilers**
Henrik Petersson, Martin Holmberg
Sensors and Actuators B, volumes 111–112, 2005, pages 487–493

The respondent took part in the planning and execution of the experimental work. With support from his supervisor, the respondent developed, applied and evaluated methods for data analysis. The respondent prepared, with additional input from his supervisor, a manuscript for publication in a scientific journal.

II **Calibration Transfer Procedures Based on Sensor Models**
Henrik Petersson, Martin Holmberg
submitted manuscript

The respondent took part in the planning of the experimental work. With support from his supervisor, the respondent developed, applied and evaluated methods for data analysis. The respondent prepared, with additional input from his supervisor, a manuscript for publication in a scientific journal.

III **Sensor Array Optimization using Variable Selection and Scanning Light Pulse Technique**
Henrik Petersson, Roger Klingvall, Martin Holmberg
submitted manuscript

The respondent took part in planning of the experimental work. With support from his supervisor, the respondent developed, applied and evaluated methods for data analysis. In co-operation with co-authors, the respondent prepared a manuscript for publication in a scientific journal.

IV **Classification of Vehicles in a Multi-Object Scenario using Acoustic Sensor Arrays**
Henrik Petersson, Andris Lauberts, Martin Holmberg
submitted manuscript

With support from his supervisor, the respondent developed, applied and evaluated methods for data analysis. In co-operation with co-authors, the respondent prepared a manuscript for publication in a scientific journal.

Related publications

a **The characteristics and utility of SiC-FE gas sensors for control of combustion in domestic heating systems [MISFET sensors]**
M. Andersson, H. Petersson, N. Padban, J. Larfeldt, M. Holmberg, A.L. Spetz
Proceedings of the IEEE Sensors, volume 3, 2004, pages 1157–1160

b **Gas sensor arrays for combustion control**
M. Andersson, H. Wingbrant, H. Petersson, L. Unéus, H. Svenningstorp, M. Löfdahl, M. Holmberg and A.L. Spetz
in Encyclopedia of Sensors, eds., C. A. Grimes and E. C. Dickey, American Scientific Publishers, Stevenson Ranch, Ca, USA, volume 4, 2006, pages 139–154

c **Simultaneous estimation of soot and diesel contamination in engine oil using electrochemical impedance spectroscopy**
C. Ulrich, H. Petersson, H. Sundgren, F. Björefors, C. Krantz-Rülcker
Sensors and Actuators B, volume 127, 2007, pages 613–618

Preface

Fortunately, the work presented in this thesis is not the result of my efforts only. I have had the pleasure to recieve support from many people and I owe them a lot of gratitude.

First of all, I would like to direct my gratitude to Professor Martin Holmberg. Your talent to be a supportive friend at the same time as being a respectful supervisor has always made me feel inspired and confident.

I recognize my collegues at the department of applied physics and the center of excelence S-SENCE as most kindfull. Admittedly, there was times when work felt less inspiring, but the reason was never due to the atmosphere among us collegues.

I have recieved much support from the graduate school Forum Scientium. From its other participants I was constantly reminded that my situation was not unique and that both difficult and joyful times could be shared with others. The course director, Stefan Klintström, is acknowledged for taking interest in my research and my personal develoment and for being helpful with many administrative challanges.

There are a number of persons that has been practically and scientifically involved in my work. These persons are, in order of appearance, Ingemar Lundström, Mats Eriksson, Roger Klingvall, Anita Lloyd-Spetz, Mike Andersson, David Lindgren, Andris Lauberts, Per Holmberg, Tom Artursson, Christian Ulrich, John Olsson, Per Mårtensson. Thank you all for the fruitful cooperation.

Till familj och vänner. Ni har gjort tappra försök i att intressera er för min forskning, men framförallt har ni visat intresse för mig som person och försäkrat er om att jag finner glädje i det jag gör. Det stöd ni ger mig är unikt och kan inte ersättas. Tack!

Mest av allt vill jag tacka Anne och Erik för att ni får mitt sinne att leva ovan och bortom vardagens små futiliteter.

■

Contents

1
Introduction

The topic of this thesis is *signal processing* – how to visualize, explore and extract information from signals and collections of data. Signal processing is a wide science applicable to many different problems and applications. This thesis emphasizes methods versatile for the processing and exploration of signals generated by sensor systems.

A sensor is a device that transforms a physical, chemical or biological stimulus into a readable signal. As an example, the thermometer is a relatively simple sensor used to read the temperature. The lambda-sond of a modern automobile is a more advanced sensor, integrated in the exhaust system to read the fuel-to-air ratio. Today, sensors make an integral part in modern technology and the list of existing sensor technologies can be extended in length.

This thesis will briefly introduce some general properties that, to various extent, are common to all sensors and link these properties to their impact on the work of analyzing sensor data. A few sensor types will be described due to their presence in the works included in the thesis. Readers seeking expertise knowledge in sensors and sensor science will be able to find more comprehensive information elsewhere.

To indicate the core concept of the thesis, a parallel will now be made to the human sense of taste. It must be made clear that the parallel is *not* made to plant an idea that the thesis is related to the development of an artificial tongue. The parallel is made since the sense of taste is a complex sensory system we all are aware of.

The human tongue can be divided into five separate areas, each with a different sensitivity to taste. The areas sense *bitterness*, *saltiness*, *sourness*, *sweetness* and *umami*(richness), respectively. Now, think upon each area as if it was a sensor, then each sensor is non-selective and has the ability to get stimulated by many different molecules. There are many species that e. g. makes the sourness sensor signal for sourness. The sourness signal alone, however, does not define what is known as taste. It is our brain's ability to analyze the joint signal pattern from the different taste sensors [1] that results in our full perception of taste and makes us able to differentiate between flavors.

Likewise our perception of taste is the result of a joint information processing of the signals provided by each of the many taste receptors, many technical systems can improve in functionality by incorporating procedures for joint processing of sensor signals. This thesis puts special interest in such procedures.

1.1 Outline of the Thesis

The thesis gives a survey on methods for exploring data and for classifying or quantifying samples from information contained within sensor signals. It will be discussed how to avoid the rise of artifacts and how to counteract for potential defects in sensor systems. Special interest is put on methods being practically applicable to chemical gas sensors. Merits and limitations of chemical sensors are discussed and it is explained why multivariate data analysis is of particular importance using such sensors.

The next chapter will introduce the reader to various aspects of sensing and a few sensor technologies which are relevant for this thesis will be described. Chapter 3 introduces to the area of multivariate data processing. Techniques specialized for the problems of classification, regression and source separation will be presented in chapter 4, chapter 5 and chapter 6 respectively. Chapter 7 will discuss how to counteract for drift and differences between sensors. Chapter 8 will conclude the thesis and give a summary of the work conducted by the respondent.

■

[1] Here, it is disregarded that also our olfactory system plays an important role in the perception of taste.

2
Sensors

An impressive amount of different sensor types has been exploited and to give a collective view of the entire field must be a difficult task. Certainly, this thesis will leave more to wish for readers primarily interested in sensor science. This chapter serves to highlight the merits and limitations that sensors might have and which make the processing of sensor data interesting. The chapter also serves as an introduction to sensor technologies appearing throughout the thesis.

2.1 The definition of a sensor

A sensor is a device that transforms a physical, chemical, or biological stimulus into a readable signal. Mostly, the readable signal falls in the electrical domain, while the domain in which the stimulus is generated varies, see TABLE 2.1. A *sensing mechanism* must be exploited to get a stimulus from a certain domain. Different sensing mechanisms present their own sets of merits and limitations which results in different problems to consider while analyzing data.

Many devices fit into the definition of a sensor above and there is room for confusion. An engineer who needs a sensor for integration into the on-board diagnostic system of a car engine to control exhausts does not want a delicate piece of equipment taking up half of the engine compartment. In that case, a small, robust, reasonably accurate, and inexpensive device is what is needed. In other cases, prime accuracy is a major concern while complexity, cost, ease-of-handling etc might be of less importance. In the mindset of this thesis, a *sensor* is needed in the first example while the latter example rather calls for an *instrument*.

The differentiation between sensors and instruments is not necessarily an academic trifle, but there might be practical differences in how to analyze the generated data. Assume there is a certain “cost” of inconvenience related to making a

TABLE 2.1: A list of examples on possible domains in which stimuli can be generated.

Domain	Example of input signals
Mechanical	length, area, volume, time derivatives such as linear/angular-velocity/acceleration, mass flow, force, torque, pressure, acoustic wavelength and intensity
Thermal	temperature, specific heat, entropy, heat flow, state of matter
Electrical	voltage, current, charge, resistance, inductance, capacitance, dielectric constant, polarization, electric field, frequency, dipole moment
Magnetic	field intensity, flux density, magnetic moment, permaebility
Radiant	intensity, phase, wavelength, polarization, reflectance, transmittance, refractive index
Chemical	composition, concentration, reaction rate, pH, oxidation/reduction potential
Biological	kinetic constants, affinity, specificity, physiological responses, concentration, hormones, antigens

measurement. In applications requiring an instrument, that cost is probably not a limiting factor and additional costs can presumably also be taken while analyzing the data. Sensor applications, on the other hand, might put tougher demands on the signal processing procedures in terms of e. g. which auxiliary actions that are allowed. Going back to on-board diagnostics example above, such application would probably require that any necessary signal processing must be resource efficient, instant and require no human interaction.

2.2 Sensor utilization imply signal processing

The sensing mechanism transforms a stimulus into a readable signal, as said. The generated signal(s) must thereafter usually undergo refinements to take a useable form. These refinements are made by applying different techniques for signal processing. Typically, the reason to conducting such processing include to:

improve interpretability In its simplest form, improvement in interpretability is reached by e. g. re-scaling the sensor signals and transforming them into physically meaningful measures, such as temperature, pH etc.

alleviate for shortcomings Most sensor devices have shortcomings causing artifacts within the rendered signals. Under the right circumstances, many of these artifacts can be suppressed using appropriate signal processing techniques.

enhance information The incorporation of signal analysis and statistics makes it possible to raise alarms etc when significant deviations from normal conditions occur. While using several sensors, a joint analysis of the signals may reveal hidden *patterns* that can be extracted and correlated to important properties of the samples under investigation. Advanced signal processing can be used to enhance the information contained within sensor signals, yielding a higher level of usability.

From the glimpses given above, the respondent now states that

> "By the usage of sensors comes the necessity to, in a more or less advanced manner, process the generated signals and analyze the recorded data"

This statement can serve as a justification of the thesis.

Introductory views will be given below to some of the shortcomings sensors might have that needs to be alleviated. Those "shortcomings" will also be presented that can be exploited by signal processing techniques to effectively improve the performance of a sensor system.

2.3 Problematic shortcomings

Perfection is rare in reality. All sensors have their shortcomings rendering errors and uncertainties in data. Some shortcomings can be related back to theoretical limitations of the sensing principles, while others are related to construction- and production weaknesses.

2.3.1 Noise

Noise is associated with *randomly* appearing disturbances and errors. The term has its origin among radio engineers, experiencing ill-sounds in transmissions caused by random fluctuations in radio signals. By now, the term is generally adopted in all fields of science.

Noise can be characterized in terms of its *origin* and in terms of its *characteristics*. When analyzing sensor data, noise is already present within the recorded signals and primary interest is to explore its characteristics, to find proper techniques for counteraction, and to assure it causes a minimum of damage. A hardware designer, on the other hand, focuses on eliminating the noise's source of origin.

The characteristics of noise

The *spectral properties* of noise are interesting to explore. If the noise appears in a bounded region of the power spectrum, the construction of a *filter* is a traditional approach for alleviation. The process is complicated, though, if the sought information is located in the same frequency range as the noise. Spectral filtering is applicable only when time-continuous signal are under analysis. *White noise* is the term used to describe noise with a homogeneous distribution over the frequency range. For white noise, the momentary magnitude of the noise signal will have a

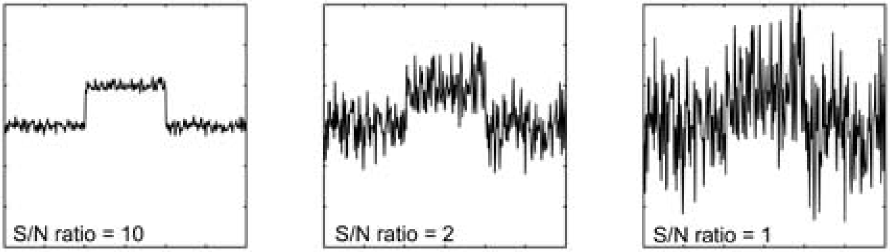


FIGURE 2.1: A square wave signal with different signal-to-noise ratios (10, 2 and 1).

Gaussian distribution and its influence can be alleviated for by means of statistical approaches. Using statistical approaches, not only time-continuous signals can be processed for noise suppressing purposes.

The *signal-to-noise ratio* (S/N) is another characteristic, defined as the power ratio between a useful signal and the noise. As a general rule, detection of a signal, by visual means, becomes difficult when the ratio gets below approximately $S/N < 2$, see FIGURE 2.1. Signal processing methods, often those that are based on a statistical analysis, can improve the detection capability finding signals also under bad S/N conditions. On the other hand, poor S/N ratios impair many statistical methods making it more difficult for them to e. g. discriminate between different types of measured specimens, see FIGURE 2.2.

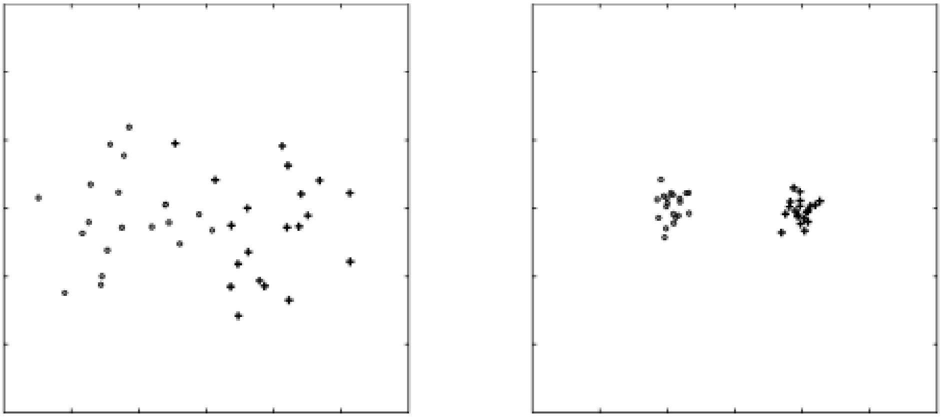
The *absolute noise level* is another measure of importance. In many setups the noise magnitude is constant regardless of the magnitude of the main signal. In those cases, the noise level directly affects the detection limit of the system.

The sources of noise

Although the source of noise play only a minor role while analyzing data, a few typical processes responsible for the rise of noise will be described below for orientational purposes (see e. g. [1] for further details).

All electronic equipments are to various extent affected by thermal noise and shot-noise. Both kinds generate white noise and occur due to microscopic effects explained within thermal physics. *Thermal noise* is caused by the thermal movement of charge carriers, such as electrons, in resistors, capacitors, electrochemical cells, and other resistive elements. The random, but periodical, movements increase with temperature and produce charge inhomogeneities in the resistive elements, generating voltage fluctuations. *Shot noise* is encountered wherever charged particles flow across junctions such as vacuum tubes, or across *pn*-interfaces in semiconductors. The transfer of individual charges occurs randomly, causing small fluctuations in the overall current and thereby the generation of noise.

$1/f$-noise and *environmental noise* are both examples of non-white noise sources. $1/f$-noise is characterized by having a magnitude inversely proportional to the frequency and its contribution often becomes significant below $\sim$ 100Hz. The source(s) of origin for $1/f$ noise is not well understood.



(a) poor S/N-ratio (= 2)

(b) high S/N-ratio (= 10)

FIGURE 2.2: Two different specimens have been measured repeatedly. To the left, signal-to-noise ratio is low and there is no pronounced statistical difference between the specimens. To the right, the two group-means are still the same as in the left figure, but the signal-to-noise ratio is high. In the right figure, there is no difficulty to claim the existence of a difference between the specimens.

Environmental noise is due to a composite of electromagnetic radiation generated by power-lines, radio, electrical motors, lightning, etc. The radiation is picked up in measurement equipments since internal conductors also function as antennas. The phenomenon is illustrated in FIGURE 2.3, where a recorded power spectrum shows both $1/f$-noise and typical environmental disturbances.

2.3.2 Drift

Drift is described as a temporal shift of sensors' response under *apparent* constant physical and chemical conditions [2]. Due to drift, the outcome of a series of experiments may vary with time, unpredictably but systematically, even though the same instrumentation and sensors are used throughout the session. For an example of drift see EXAMPLE 2.1.

Most procedures for signal and data analysis assume that sensors are static in terms of their characteristics and they cannot handle the temporal changes caused by drift.

Examples of processes rendering drift are the degradation of sensor surfaces due to ageing or due to exposure of harmful gases. Considering entire sensor systems, drifting may also be due to ageing of amplifier components in auxiliary equipments etc. Moreover, the sensor might be sensitive to changes in environmental parameters such as e. g. air pressure. If such fluctuations are not under control, the unaware user risk to experience drifting signals.

Short-term drift and *memory effects* are phenomena that strictly are not drift

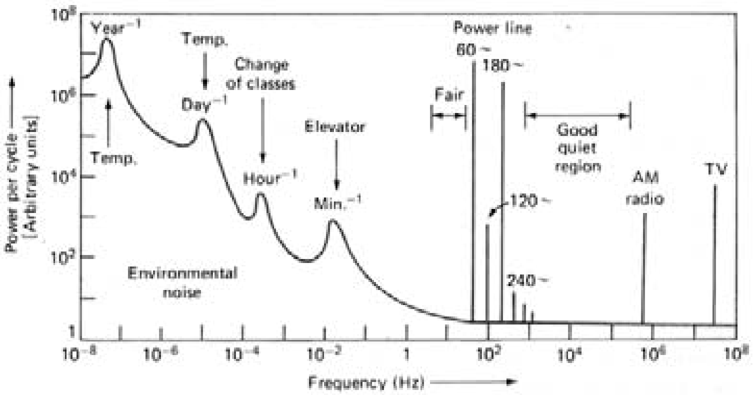


FIGURE 2.3: Flicker noise and some sources of environmental noise in a university laboratory. The spectra was recorded in 1968. Today, the number of sources generating environmental noise is presumably even denser.(Reproduced from T. Choor, J.Chem.Educ.,1968,45,A540)

but add very similar effects to the output. Short-term drift occur in some systems where the instrumentation needs some time to reach its equilibrium state. Until equilibrium is reached, the sensor output is unstable and said to be under influence of short-term drift [3]. Memory effects occur due to species leaving remnants on the sensor, affecting its characteristics. If this occur, the sensor "remember" previous samples meaning that traces of previous measurements can be seen in the signal from fresh measurements. The memory effect may vanish with time and the system will then return to its original state. In some cases, the effect remains for a very long time, or never vanishes, and it becomes impossible to distinguish the memory effects from true drift [3].

In practice, it is rarely necessary to be able to discriminate between drift, short-term-drift, and memory effects. On the other hand, it is many times vital to identify if any of these processes are present and if so use data analysis procedures that are robust to their effects.

2.3.3 Reproducibility

Some sensing principles are such that it becomes troublesome to manufacture sensors without significant sensor-to-sensor variations.

The difficulty to manufacture sensors with reproducible characteristics causes problem when conducting long term experiments, or when striving for commercialization. The reason is that many data analysis procedures establish a model describing the relation between the sensor signals and the measure to acquire. Ideally, a model should be applicable to all sensors of the same kind and hence only need to be established once. However, in cases where sensor-to-sensor differences are significant, it might not be sufficient to apply the same model to all sensor individuals.

Since model building many times is a costly and time-consuming procedure,

Example 2.1: An example of drift

A specific gas mixture has been measured during 60 days with three different sensors. Due to drift, the response is not constant and varies with time. Probably, the same cause of drift affects all three sensors, although the outcome of the effect is different.

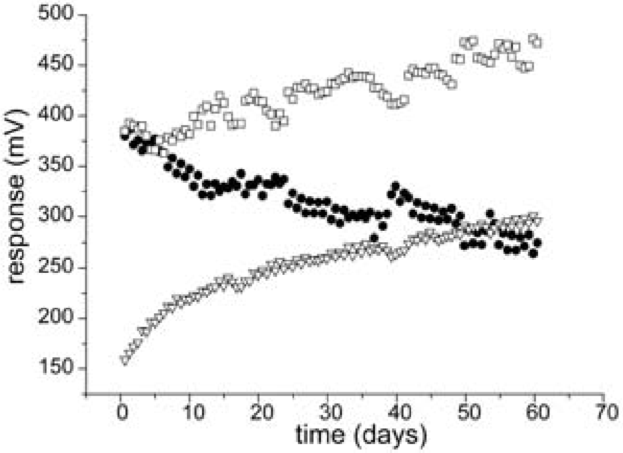


Figure reproduced with permission. T. Artursson et al. , Journal of Chemometrics, 14(2000),pp711-723. Copyright John Wiley & Sons Limited

■

it is rarely an alternative to establish unique models to each sensor individual. Better choices are to mathematically counteract for the sensor differences, or to adapt an already established model to the character of a slightly different sensor. Such techniques will be described later.

2.3.4 Non-linearity

For sake of simplicity, it is often favorable if the sensing mechanism results in a linear relationship between the sensed stimuli and the signal generated by the sensor. Linear sensor responses enable the usage of linear mathematics to analyze data and they are therefore, as compared to non-linear counterparts, not as complex and cumbersome to work with. If it is known that the response is non-linear, data can sometimes be pre-linearized in a pre-processing procedure.

2.4 Exploitable "shortcomings"

The *sensitivity* of a sensor is defined in terms of how much its output changes in response to the state of a specified measurand[1]. If the sensitivity towards *one* particular measurand by far exceeds the sensitivity towards any other, then the current sensor-type is said to be *selective*. Selective sensors are desired in applications aiming at detecting or quantifying single targets. In reality, many sensor types are markedly responsive to several different measurands and therefore considered to be *non-selective*. If a non-selective sensor is used *alone*, uncertainties are introduced by the fact that different combinations of measurand states can generate the very same response. Thereby, it becomes difficult to relate a certain sensor output to the state of a particular measurand.

The trouble experienced with non-selectiveness can be avoided. An obvious approach, although rarely realistic in practice, is to re-design the sensor and thereby reduce the sensitivity towards interfering measurands.

A practical approach to avoid uncertainties from interfering measurands is to make sure they are kept constant throughout all measurement sessions. This approach is sometimes applicable to laboratory setups, but rarely in other situations.

Under certain conditions, non-selectiveness can be counteracted for by assembling several sensors together in a sensor array. The concept of using sensor arrays will soon be outlined in a separate section below.

If a sensor is non-selective, it is many times interesting to learn the character of the non-selectiveness. The least complicated characteristics is when the contributions from each measurand simply add together forming a summary output. *Cross-sensitivity* is a term used to denote when the response depends on an interaction between the contributing measurands. For example, the degree of presence of one measurand could inhibit or amplify the sensitivity towards another. Sensors with excessive cross-sensitivity are in general difficult to handle.

2.4.1 Sensor Arrays

A *sensor array* constitutes a system of locally gathered sensor elements. It could also mean a single sensor with a multidimensional output.

As previously indicated, non-selectiveness can be overcome by assembling arrays of sensors [4]. This can be done whenever the incorporated sensors have different patterns of sensitivity. In a simplified view, the different sensors can be said to measure a sample "from different angles" and the "complete picture" can be put together through joint analysis of the sensor signals. *Pattern Recognition* procedures (PR) are the mathematical tools used for such multidimensional analysis.

Some applications aim at sensing loosely defined parameters such as "air quality". Typically, one sensor alone cannot perceive all aspects of such a complex entity. Fortunately, an elegant benefit of using sensor arrays in conjunction with pattern recognition procedures comes with that also loosely defined parameters can

[1] By the term *measurand* it is meant any physical measure, chemical specimen etc being quantified by the measurement.

be handled. In the same manner as with the counteraction of non-selectiveness, this is possible since the different sensors, *together*, measure a “complete picture” of the environment. The pattern recognizer thereafter extracts pieces of information that are related to the desired parameter.

The two scenarios given above motivate the approach of using various kinds of sensor array assemblies and apply pattern recognition techniques to the generated signals. The remaining part of this chapter will introduce a few sensor technologies and give a couple of examples on sensor arrays that have been used in practice. The thesis will thereafter turn focus and more thoroughly treat mathematical techniques for pattern recognition.

2.5 Chemical Sensors... some examples

A *chemical sensor* is a device that transforms chemical states into readable signals. Large interests are nowadays put on chemical sensors, not least because of increasing demands on environmental monitoring, food quality supervision and safety issues. These and similar applications require small and cost effective devices capable of sensing gases, toxins etc.

Conceptually, chemical sensors are very different from physical sensors, not least because of the range of measurands they cover. Approximately 100 physical properties can be detected using physical sensors, while chemical sensors cover a range of measurands that is several orders of magnitude larger [5]. Among the more widespread and well-known chemical sensors, the pH-electrode and the lambda-sond can be mentioned. The different types of chemical sensors that have been exploited is impressive, and the few sensor types presented below is merely a small selection. A more extensive overview can be found in [5].

2.5.1 Metal Oxide Sensors

Gas sensitive metal oxide sensors MOS are well studied and have been available on the market since 1968 [6]. The basic structure of a MOS sensor consists of a ceramic tube coated with sintered and doped metal-oxide. The gas is sensed by its effect on the electrical resistance of the semiconducting metal-oxide, which is a result of the changes in conductivity caused by reactions with oxygen species on the surface of the metal-oxide particles [7]. Commonly used metal-oxides are SnO_2, TiO_2, ZrO_2, and Ga_2O_3 doped with catalytic metals such as Pd, Pt or Al. The doping enables sensors to get enhanced selectivity toward certain gases.

The SnO_2 based Taguchi-sensor is considered the most important type of MOS sensors with respect to practical applications. A range of different Taguchi-sensors are available on the market, sensitive to measurands like ammonia, alcohols, sulfur compounds, carbon monoxides, methane, hydrogen, CFC etc.

2.5.2 Metal Insulator Semiconductor structures

Field effect sensors are based on metal–insulator–semiconductor MIS structures. The MIS structure can be configured in two ways: as a field effect transistor (MISFET)

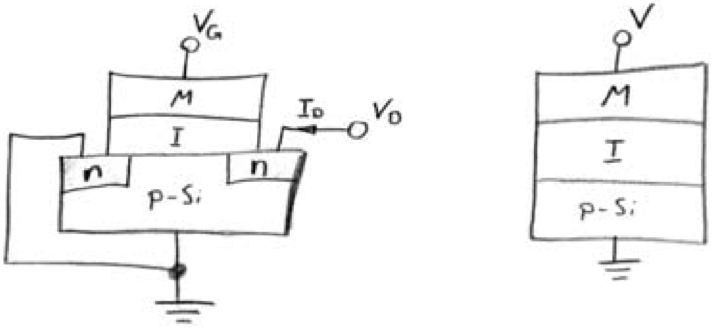


FIGURE 2.4: Schematic illustration of a MIS structure. To the left it is configured as a FET device and to the right as a capacitor.

or as a capacitor (MISCAP) (see FIGURE 2.4). The gas sensing principle is the same for both configurations and relies on a change in the semiconductor's surface potential caused by the sensed gas. In MISFET devices, such change will affect the drain–source current flowing through the semiconductor. For MISCAPs, the capacitance will change as soon as the surface potential is altered.

The metal layer of the MIS structure typically consists of catalysts such as Pt, Pd or Ir, where the particular choice of metal influences the characteristics of the sensor. The physics describing how different gas-metal combinations alter the surface potential, and thereby the response characteristics, will be left out of this thesis. A short example of one such interaction will be given though: In the MISFET configured palladium-gate hydrogen sensor, invented by Lundström et al. [8], hydrogen atoms are generated at the palladium gate due to dehydrogenation of molecules. The hydrogen atoms diffuse through the metal and reach the metal–insulator interface, where they adsorb and generate a dipole-layer. The dipole-layer gives rise to a change in work function between the gate and the semiconductor, causing a change in drain-source current [6, 7].

Field effect sensors have been commercialized [9, 10] and make an active research area. An interesting development is the exploration of alternative semiconductor materials. Wide-bandgap materials like SiC, AlN, GaN, AlGaN and diamond have potential to function in harsh environments [11]. Particularly *Metal Insulator Silcon Carbide Field Effect Transistors* MISiCFET have been utilized and studied at the department of Applied Physics, Linköpings University, Sweden. These devices have proven to function well in harsh environments such as in automobiles and at combustion plants [12].

2.5.3 Scanning Light Pulse Technique

The *Scanning Light Pulse Technique* (SLPT) was introduced in 1983 as a technique for investigating insulator–semiconductor interfaces [13]. In short, a light pulse is used to raise a current due to the formation of electron–hole pairs in the depletion area of the semiconductor. The current depends on the difference in workfunction of the metal and the semiconductor, and also on the applied voltage. It is changes in the workfunction at the metal–insulator interface that is utilized for gas sensing [14]. Note that the current is created locally in the region being illuminated, so the measurement gives a local gas response.

SLPT is a powerful tool since it can be used to scan surfaces and render maps of local gas responses. By scanning a surface with non-uniform properties, an infinite amount of discrete sensor candidates can theoretically be evaluated in a single run. This is exceedingly convenient when testing-out which sensor configuration that yields the best achievable sensitivity, stability, selectivity or reproducibility within a particular application. SLPT can therefore be used as a workbench technique for MIS gas-sensor development.

2.5.4 Electrochemical Sensors

Electrochemistry is concerned with the interplay between electricity and chemistry occurring at an electrode–solution interface [15]. Many sensor technologies for liquid phase applications have been inspired by phenomena observed and described therein. Electrochemical techniques are usually categorized into *potentiometry*, *conductometry* and *voltammetry*. Potentiometry is concerned with the measurement of potential appearing between two electrodes. In conductometry the solutions conductance is measured and traced-back to the movement of charged elements present in the solution. The current arising when a potential is applied between two electrodes is studied in voltammetry. Readers with particular interest in electrochemical techniques are referred to textbooks such as [15, 16].

2.5.5 Sensor Arrays

Electronic Noses

Gardner and Bartlett once defined an electronic nose (e-nose) as [7]:

> "An electronic nose is an instrument which comprises an array of electronic chemical sensors with partial specificity and an appropriate pattern recognition system, capable of recognizing simple or complex odors."

By this definition, the term is restricted to odor recognition only. However, the architecture of the described e-nose has much in common with many other gas sensitive sensor systems and the term has been generally adopted.

One example of an electronic nose is the high temperature electronic nose (HTe-nose) [17, 18, 12] used within the experiments described later in this thesis and in the included papers. In brief, the HTe-nose is developed for harsh environments and consists of three field effect sensors (see Figure 2.5), nine metal oxide sensors, and a lambda sensor.

Many other electronic noses have been described in the literature and some have thereto been commercialized. There is no room to give a fair overview here and the interested reader is recommended to read the thorough overview provided by Pearce et al. [3].

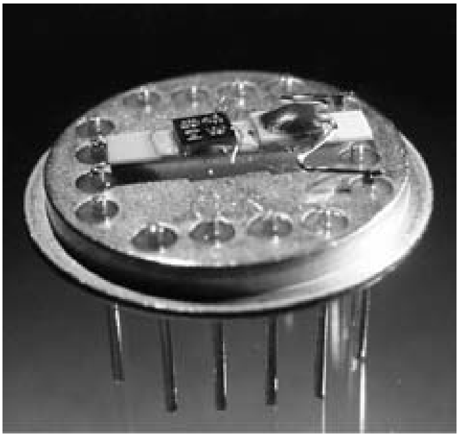


FIGURE 2.5: Three MISiCFET devices integrated on a chip and mounted on a holder together with a heater and a $Pt100$ element. The diameter of the holder is 15mm.

Electronic Tongues

Different electronic tongues (e-tongue) have been described in literature, see [19] for an overview. The e-tongue developed at Linköpings university utilizes an electrochemical technique termed *pulsed voltammetry* [20]. Simplified, the voltammetric e-tongue consists of a set of noble-metal-electrodes onto which pulse-trains of electrical potentials are applied. The applied pulse-train gives rise to a sequence of current pulses, a voltammogram, that can be analyzed. The shape of the voltammogram depends on e. g. the specimen composition, the electrode material, and the applied pulse train.

■

3 Multivariate Data Analysis

When several, maybe hundreds, of signals are registered simultaneously it is a delicate task to visualize, explore, and search for results in data. Each signal may potentially be the response from many varying processes and might thereto co-vary with other registered signals. This gives rise to the formation of signal *patterns* and implies that signals must be analyzed jointly and not one-by-one in order to not loose valuable information.

Multivariate data analysis is an important tool to find dependencies between several variables and to learn under which circumstances certain signal patterns are likely to occur. Multivariate data analysis can also be applied to situations when the objective is to relate certain signal patterns to certain properties of the analyzed samples. These, and other similar tasks, are solved by analyzing data solely; equations of physics etc must not be needed and it is hence possible to deal with complex problems where the underlying mechanisms are unknown, see Example 3.1.

Techniques for multivariate data analysis have been developed and applied within numerous scientific areas, where psychology, image processing, bioinformatics, and metrology are some examples. This thesis is related to a fifth example, the utilization of multivariate techniques for sensor applications.

This chapter will define the terms and nomenclatures following through the thesis. Procedures for exploring and reducing datasets will also be presented. Such procedures are often a prerequisite for further processing. General concepts related to learning from data will be given. Methods to make classifications, quantitative assessments, signal separations, and various compensations will then be treated separately in the four following chapters.

Example 3.1: A data analysis problem

A simple data analysis example comes from entry level physics class. A series of known masses (output) are attached to a spring and the spring's elongation (input) is measured for each mass. The experimental data is plotted in a diagram, mass versus elongation, and it becomes apparent that a straight line can be drawn in the diagram that fits nicely to the data. Without knowledge of gravitational laws and Newtonian mechanics a relation has been found that can be used to determine the mass (output) of unknown species by measuring the elongation (input) of that particular spring. A ridiculous example maybe, but the same approach can be applied to far more difficult and multi-variate problems where the property to be estimated depends on many variables simultaneously.

■

3.1 Introduction to terms and nomenclature

Let us consider an assembly of n sensors. Within a narrow (time-)frame k, the sensor responses are recorded and encoded into numerical numbers, whereby an *observation* of the current state of nature is being generated. The numerically encoded observation is stored into a *vector* $\mathbf{x}_k$,

$$\mathbf{x}_k = [x_{k1}, x_{k2}, \ldots, x_{kn}] \tag{3.1}$$

Each *element* x_{ki} of the vector represents the response value, or the signal, from sensor i as observed within the frame k.

A *dataset* is a collection of observations. Assuming that N observations have been made, the observations $\{\mathbf{x}_k\}_{k=1}^{N}$ are compactly collected in a *data matrix*.

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & & \ddots & \vdots \\ x_{N1} & \cdots & \cdots & X_{Nn} \end{bmatrix} \tag{3.2}$$

The literals $\mathbf{x}$, $\mathbf{X}$ are often reserved to denote single observations and datasets of response observations, respectively. In many situations each observation $\mathbf{x}_k$ associates to one or several properties $\mathbf{y}_k$ of the sample under study. Possible sample properties could be quantitative measures, such as a chemical concentration, or qualitative measures, such as a numeric code representing a particular category. The literals $\mathbf{y}$, $\mathbf{Y}$ are often reserved to denote associated sample properties.

Each observation can be thought of as a point in an abstract n-dimensional space, $\mathbb{R}^n$. This space is known as *sensor space*, *response space* or *input space*. Accordingly, the associated properties $\mathbf{y}$ are thought of as points in an m-dimensional *output space*, $\mathbb{R}^m$.

For reasons to become apparent later on, it is sometimes favorable to transform sensor data into another representational form for further processing. Such an operation can be viewed upon as a *mapping* from sensor space $\mathbb{R}^n$ into a new *feature-space* $\mathbb{R}^d$. The term *feature* is generally used when referring to information providing a useful description of the observations.

3.2 Exploratory Analysis

The *exploratory analysis* serves as an initial examination of data and provides aid for settling which data analysis procedures to proceed with.

A typical exploratory analysis consists of two parts: *(i)* plotting data, and *(ii)* calculating summary statistics. By plotting sensor responses, malfunctioning equipment can be detected and information about magnitudes and dynamical ranges can be retrieved. From calculated statistics, by which it is meant estimates of means and covariances etc, it is sometimes possible to identify clusters, to detect obvious *outliers*[1], and to find strong interdependencies between variables.

It is of good practice to not satisfy with plots of *individual* sensor responses, but to proceed and also visualize the complete dataset in a single plot. By doing so the multivariate nature of data can be explored and patterns of joint signal expressions can be found. Naturally, to make such visualization on screen or paper, the multidimensional data must first be given a 2- or 3-dimensional representation, see Example 3.2. Techniques for making low-dimensional representations of data play a crucial role in multivariate data analysis.

3.3 Dimensionality reduction, feature selection and extraction

Dimensionality reduction techniques are helpful for finding a low dimensional representation of multivariate data while retaining as much of the relevant information in the original data as possible. Formally, the concern is to find a mapping from $\mathbb{R}^n$ to $\mathbb{R}^d$

$$G : \mathbb{R}^n \to \mathbb{R}^d, \qquad d < n \tag{3.3}$$

Any procedure for dimensionality reduction must define a criterion $J(G)$ by which it is possible to judge whether a mapping is better than another [21].

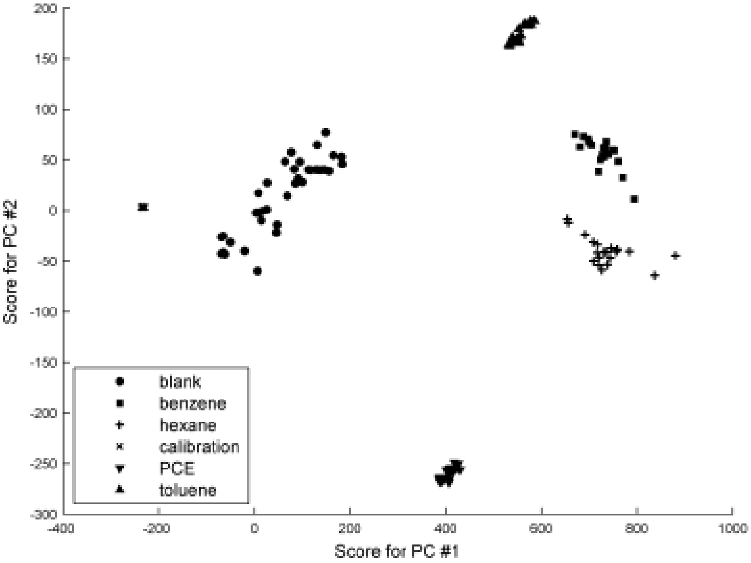
3.3.1 Feature Selection

Given a set of n features, the problem of *feature selection* is to find a subset that contains the $(d < n)$ features that are most suitable for solving the present task. Let $J(\cdot)$ be a criterion assessing the robustness and accuracy of the solution when subset $\mathbf{X}'$ is used, then the most straight forward approach to feature selection would be to first generate all possible subsets and then identify the one

[1] outlier is the term used to describe erroneous observations that strongly deviates from the expected.

Example 3.2: Dimensionality Reduction

An electronic nose (EN3320, Applied Sensors AB), consisting of 23 sensors, were used to measure soil samples contaminated with different toxins (1000ppm). The input space has been reduced into a 2-dimensional representation using a *Principal Component Analysis* algorithm (described later). The reduced 2-dimensional feature space can easily be plotted and it becomes clear that the instrument can be used to discriminate between differently intoxicated samples.



■

rendering the highest value of $J(\mathbf{X}')$. Such *exhaustive search* will find the optimal subset, but the computational burden will be too excessive even for moderately sized datasets. A number of techniques have been described, adding or deleting features sequentially (forward- and backward- selection respectively) avoiding an exhaustive evaluation. Unfortunately, although sequential techniques are computationally efficient and often useful it has been shown that none of them are guaranteed to find the optimal subset [22, 23].

3.3.2 Feature Extraction

Feature extraction methods create a new space of features based on transformations of the original data set. Both linear and non-linear transforms have been reported, although linear projection techniques are more frequently used in practice.

A linear projection technique defines a set of weight vectors spanning a *sub-space* $\mathbb{R}^d$ of the original data space $\mathbb{R}^n$. Geometrically, the weight vectors define the orientation of a d-dimensional hyper-plane inside the original n-dimensional data space. The feature extraction is made by projecting the original data onto the hyper-plane, whereby the image onto the hyper-plane defines the new features. Different sub-space techniques uses different criterions $J(\cdot)$ and thereby yield differently oriented hyper-planes, see FIGURE 3.1 for an illustration.

Principal Component Analysis

The best-known projection technique for feature extraction is *Principal Component Analysis* (PCA) [24, 25, 26, 27]. By analysing the covariance structure of sensor data, PCA determines the d-dimensional sub-space ($d < n$) with *closest fit* to the original data, see FIGURE 3.1. The weight vectors, given the literals $\mathbf{p}_i$, are denoted *loading vectors*. The loading vectors are mutually orthogonal and normed to unit length. By projecting an observation $\mathbf{x}_k$ onto the loading vectors, a d-dimensional *score vector* $\mathbf{t}_k$ is retrieved. The score values defines the extracted features.

$$\mathbf{t}_k = \mathbf{x}_k[\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_d] = \mathbf{x}_k\mathbf{P} \tag{3.4}$$

Geometrically, the score values of an observation are the coordinates of its projected image, within the coordinate system defined by the loading vectors and the hyper-plane they span. The first dimension of this coordinate system is the first *principal component*, and so on. A complete set of data $\mathbf{X}$ can of course also be projected onto the sub-space resulting in a matrix $\mathbf{T}$ of score values,

$$\mathbf{T} = \mathbf{X}\mathbf{P} \tag{3.5}$$

Turning back to what was actually meant by closest fit, PCA minimizes the sum of squared residuals ε_i comparing the original data with the reduced feature set, see FIGURE 3.1 and the expression below

$$\sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{t}_i\mathbf{P}^T\|_2^2 = \sum_{i=1}^{N} \|\mathbf{x}_i - (\mathbf{x}_i\mathbf{P})\mathbf{P}^T\|_2^2 \tag{3.6}$$

The minimization is effectively solved by making an eigenvector decomposition

$$[\lambda, \mathbf{D}] = \mathrm{Eig}(\mathbf{X}^T\mathbf{X}) \tag{3.7}$$

and by identifying the eigenvectors as loading vectors.

Apart from the classic PCA formulation, a range of extensions has been suggested for feature extraction purposes. If data is comprehended from multiple sources, e. g. from several sensor arrays with different modality, Rännar et al. have suggested to deploy *hierarchical PCA* [28] extracting features from each natural subset and pass them to further “top-level” extractions. *Multi-way PCA* [29] is another extension working on multi-mode data [2]. A variety of *adaptive PCA*

[2] generalized matrices with more than two “dimensions”: row,column,...

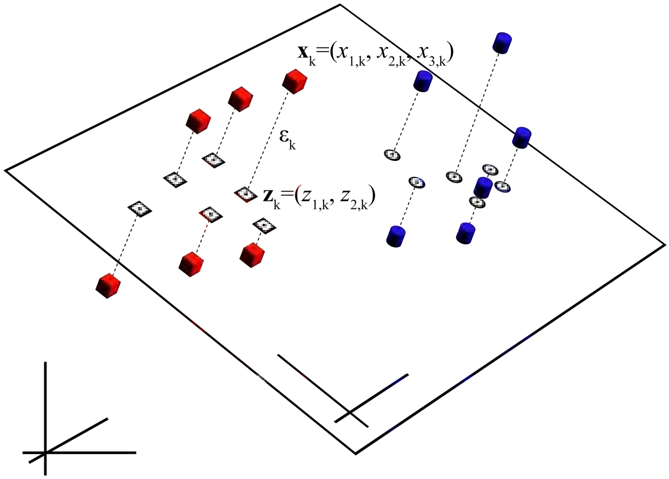

$\mathbf{x}_k=(x_{1,k}, x_{2,k}, x_{3,k})$
ε_k
$\mathbf{z}_k=(z_{1,k}, z_{2,k})$

Figure 3.1: Dimensionality reduction using PCA. The sensor response space is $\mathbb{R}^3$. Two loading vectors have been calculated that define a 2-dimensional plane/sub-space onto which the observations are projected. The sub-space is oriented in such way that the sum of squared residuals, $\sum_{k=1}^{N} \varepsilon_k^2$, is as small as possible.

formulations has also been reported where APEX [30] is one example. Adaptive algorithms make calculations in a recursive fashion requiring only one observation per iteration. This has the effect that arbitrarily large datasets can be analyzed, which is good for on-line purposes, and that the feature extraction continuously adapts to changes in the analyzed data.

Canonical Correlation Analysis

Some applications present samples that are best assessed using a quantitative scale. In those cases, a natural aim is to find features that stand in linear relationship to the desired scale.

Canonical Correlation Analysis (CCA) [31, 24] is a well established linear subspace technique that might be appropriate to use as a feature extractor under the described circumstances. A clear distinction from PCA lies in that CCA requires supervision in finding features to extract from sensor data $\{\mathbf{x}_k\}_{k=1}^{N}$. The supervision is provided in terms of a complementary dataset $\{\mathbf{y}_k\}_{k=1}^{N}$ containing quantitative properties against which each observation should be matched.

CCA provides two subspaces of paired canonical variates,

$$\begin{aligned}\mathbf{U} &= [\mathbf{a}_1,\ \mathbf{a}_2,\ \ldots,\ \mathbf{a}_m]^T\mathbf{X} = \mathbf{A}^T\mathbf{X} \\ \mathbf{V} &= [\mathbf{b}_1,\ \mathbf{b}_2,\ \ldots,\ \mathbf{b}_m]^T\mathbf{Y} = \mathbf{B}^T\mathbf{Y}\end{aligned} \tag{3.8}$$

The technique seeks for vectors $\mathbf{a}_i$ and $\mathbf{b}_i$ that maximizes the correlation

$$\rho_i = \mathrm{corr}(\mathbf{u}_i, \mathbf{v}_i) = \mathrm{corr}(\mathbf{a}_i^T\mathbf{X}, \mathbf{b}_i^T\mathbf{Y}) \tag{3.9}$$

subject to that $\mathbf{u}_i$ and $\mathbf{v}_i$ have unit variance and that the k^{th} solution $(\mathbf{u}_k, \mathbf{v}_k)$ is uncorrelated to all $(k-1)$ previous pairs of canonical variates.

The vectors $\mathbf{a}_i$ and $\mathbf{b}_i$ are found directly from the generalized eigenvector equations,

$$\begin{aligned} \mathbf{S}_{XX}^{-1}\mathbf{S}_{XY}\mathbf{S}_{YY}^{-1}\mathbf{S}_{YX}\mathbf{A} &= \rho^2\mathbf{A} \\ \mathbf{S}_{YY}^{-1}\mathbf{S}_{YX}\mathbf{S}_{XX}^{-1}\mathbf{S}_{XY}\mathbf{B} &= \rho^2\mathbf{B} \end{aligned} \tag{3.10}$$

where $\mathbf{S}_{(\cdot)(\cdot)}$ denote each respective covariance matrix estimate.

Other linear feature extractors

Another common projection technique is *Independent Component Analysis* (ICA) [32, 33, 34]. The method does not rely on any second order statistics (variance–covariance), as PCA does, and is appropriate to use under circumstances where data does not show a structured variance, like in noisy environments etc. ICA is a method strongly related to the problem of source separation and will be described to greater detail in chapter 6.

Non-linear feature extractors

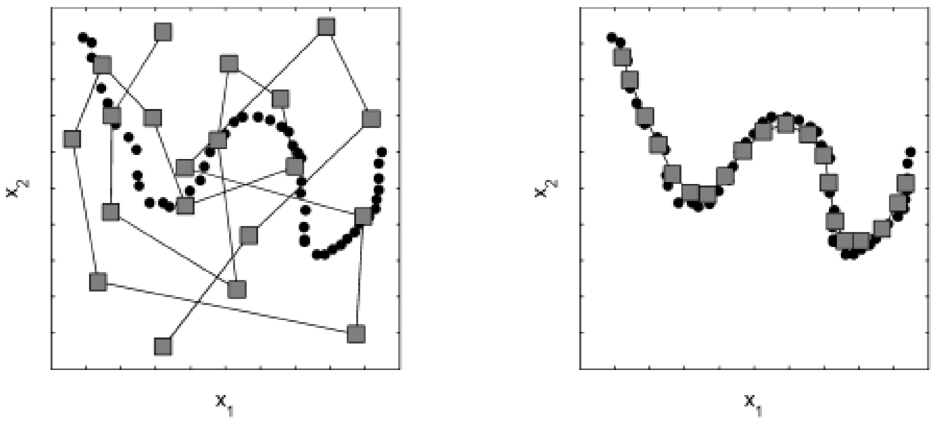
The *Self Organizing Map* (SOM), first described by Kohonen [35], is a non-linear feature extraction technique that might be found conceptually interesting since it is neurobiologically inspired, trying to mimic how the brain maps sensory inputs to different areas in the cerebral cortex [27].

The algorithm is easy to implement, but it has unfortunately been difficult to analyze its general mathematical properties [27]. The SOM can be viewed as a swarm of nodes (points) distributed in the original n-dimensional input space. The nodes are interconnected to their nearest neighbors, typically forming a 1- or 2-dimensional grid, but any general d-dimensional grid is possible. The algorithm iterates as follows: one-by-one, all available observations in $\{\mathbf{x}_k\}_{k=1}^{N}$ are in turn presented to the algorithm and placed in $\mathbb{R}^n$-space. The grid-node being closest to the currently placed observation is designated as "winner" and allows to adapt by moving-up even closer to the observation. Also nodes neighboring the winner, with respect to their position in the grid, are allowed to adapt by moving-up slightly closer. When all observations have been presented sufficiently many times to the algorithm, the adaptations will become smaller and the grid structure settles. Each observation can now be encoded according to its position relative to the position of the nodes in the settled grid, see FIGURE 3.2.

Other non-linear feature extractors include different extensions to PCA, among which *kernel-PCA* [36] stands out due to its computational core is still based on linear algebra.

3.3.3 Notes on selecting between Selection and Extraction

No definite rules exist to decide between feature extraction or selection, but the data analysts must make a wise choice based on experience considering the requirements of the application and the nature of the data. Typically, selection



(a) initial placement of SOM grid

(b) the SOM after adaption

FIGURE 3.2: A 1-dimensional self-organizing map consisting of 20 interconnected nodes (gray boxes) is randomly placed in sensor-space (figure a), where also some measurements can be seen (black dots). After adaption, the map has stabilized capturing the distribution of sensor data (figure b).

techniques lead to future savings in cost and time, since left out features (=sensor signals) do not have to be measured in the future. Since the selected features remain untransformed, selection techniques will also merit by the fact that the reduced feature set retains the original physical interpretation. This might help to understand the process behind the generated patterns. On the other hand, some sensor systems deliver raw-data where the physical interpretability is weak already from the beginning and no significant loss in interpretability is made if the analyst favors to use feature extraction. Features generated by an extractor may also provide better discriminative ability as compared to a subset of selected features.

3.4 Modeling and Learning

Sensors are deployed because we want to use the information they provide to support some kind of *decision*. For instance, it is reasonable to think of an application where action is taken in accordance to the concentration, or the category, of a measured sample. Although the feature extraction techniques just described are helpful tools for visualization purposes and for finding a suitable representation of data, they can generally not be used to foretell e. g. the category of a sample. We will soon look into techniques for providing categorical information (*classification*) and quantitative information (*regression*) out of sensor data, but before doing that a general setting for such procedures will be presented.

3.4.1 Modeling

While modeling, we are concerned with the problem of finding a desired dependence between sensor data, described by the random vector $X \in \mathbb{R}^n$, and a property of interest, described by the random scalar y. Ordinarily, we do not have knowledge of the exact functional relationship between X and y. At this stage, we propose the model

$$y = f(X) + \varepsilon \tag{3.11}$$

where $f(\cdot)$ is a deterministic function and ε is a random expectative error. The error term is introduced to handle the "ignorance" on influential factors such as noise that cannot be accounted for in the model function.

The model function gives an estimate of the actual output

$$\hat{y} = f(X) \tag{3.12}$$

and can take many forms. In some cases, the sensor mechanism is well understood and can be mathematically expressed in terms of physical laws. In other cases, little is known regarding the sensing mechanism. If so, empirical knowledge can be utilized to formulate a purely mathematical model, a procedure known as learning.

3.4.2 Learning

The problem of concern related to *learning* (alternatively *calibration* or *training*) is to, with support from a limited set of observations, chose from a given set of candidate model functions $f(\mathbf{x}, \mathbf{w}), \mathbf{w} \in \mathbb{W}$ the one that best estimates the desired response y. The set of available observations $\{\mathbf{x}_k, y_k\}_{k=1}^{N}$ is hereafter referred to as the set of *training data*. A *loss-function* is defined to measure the degree of miss-fit made by each candidate function and the aim is to find the candidate rendering the lowest overall loss [37]. Different loss functions relate to different pattern recognition procedures. A quadratic loss function on the difference between desired output and model output is normally used while making regressions (see chapter 5)

$$L(y_k, f(\mathbf{x}_k, \mathbf{w})) = (y_k - f(\mathbf{x}_k, \mathbf{w}))^2 \tag{3.13}$$

The pattern recognition techniques that are able to learn from empirical data are often categorized in terms of being parametric vs. non-parametric, supervised vs. unsupervised [3]:

Parametric: Parametric techniques are based on the assumption that the sensor system generates data following a known statistical probability distribution. The aim of an parametric approaches is to estimate the parameters, or statistics, that defines the assumed probability distribution. A majority of the parametric techniques assume that the data follow a Gaussian (normal) distribution.

Non-parametric: Non-parametric techniques do not assume any specific probability distribution of the data and can hence be used in cases that are more general.

Supervised: In supervised learning, a *"teacher"* provide the desired outputs for each observation, $\{y_k, \mathbf{x}_k\}_{k=1}^N$, and the training algorithm seeks the setting that generates the smallest loss, comparing the model's actual output with the desired.

Unsupervised: In unsupervised learning there is no teacher but the algorithm analyses observations only $\{\mathbf{x}_k\}_{k=1}^N$. The aim is to find a setting that is optimal according to criterions implicitly or explicitly defined within the algorithm.

3.4.3 Generalization

The *training error* is the magnitude, or the frequency, of the errors made by a model during the training session. The *generalization error* is the magnitude, or the frequency, of the errors made by the same model function when it is applied onto observations it has not seen before. The practical usefulness of a model is essentially determined by its ability to yield a low generalization error.

The ability to *generalize* is foremost influenced by three interdependent factors: the complexity of the problem *(i)*, the complexity of the model architecture *(ii)*, and the number of representative observations available for training *(iii)*.

The balance between the complexity of the problem and the architecture of the model should be given careful consideration. By architecture is meant the structure of the predefined functions $f(\cdot, \mathbf{w}),\ \mathbf{w} \in \mathbb{W}$ that the learning algorithm chooses from during the learning phase. For relatively easy problems, it might be sufficient to rely on linear models and to choose from the set of *linear functions* defined by

$$f(\mathbf{x}, \mathbf{w}) = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n = \mathbf{x}\mathbf{w} \tag{3.14}$$

Linear models are easy to handle, both computationally and analytically, but are sometimes not capable of capturing the structure of the studied problem. *Non-linear functions*, like the ones used for constructing single-layered neural- networks (see page38)

$$f(\mathbf{x}, \mathbf{w}) = \sum_{j=1}^{M} (1 + \exp(a\mathbf{x}\mathbf{w}_j))^{-1} \tag{3.15}$$

are better suited for describing complex non-linear relations, but are also more demanding to handle.

Is it then a good idea to strive for the most complex model that can be handled? Unfortunately, it is not so! Complex models tend to require a larger number of observations to be trained properly and are more prone to *overfitting*. An overfitted model adapts too hard to the particular set of observations used for the learning task. The model thus sacrifices proper approximation of the general behavior in pursuit of making the smallest possible error on the unique "prints" contained in trainingdata due to random processes such as noise. Consequently, the training error of an overfitted model is typically very low, but the generalization error is higher than necessary. The goal is to match the complexity of the model with the complexity of the problem, making the model capable to describe the

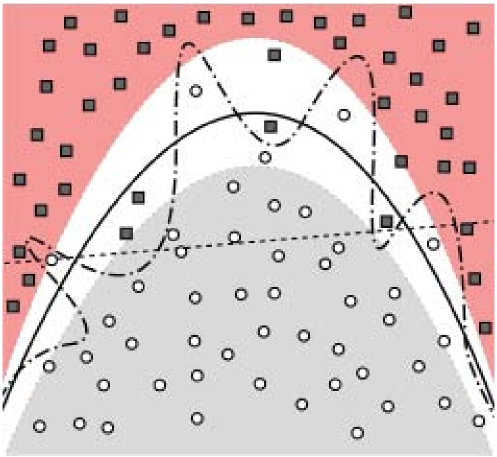


FIGURE 3.3: An illustration on the impact of model complexity. There are two populations, circles and squares, which normally fall within two separate regions. Due to noise and other unpredictable processes, observations occasionally fall outside these regions. To draw a line is too simplistic and cannot separate the populations (- - -). An overly complex curve is capable of separating all observations contained in the set of training data, even those that are un-normal. For other sets of data, containing similar but not identical observations, the overly complex curve might fit poorly and might even separate normal observation in a false manner (–·–). A sufficiently complex model is balanced to the complexity of the problem and capable of separating the normal processes, but unable to separate un-normal observations. Thereby, the balanced model has a higher generalization ability and yields a good overall performance on future unseen observations(—).

general functional relationship, but incapable of learning the unique "prints" of the particular set of data, see FIGURE 3.3.

3.4.4 Validation

The process in which it is estimated if a model generalizes well and has a complexity balanced to the problem at hand is known as *validation*. Validation is typically performed using trainingdata to establish models with increasing degrees of complexity. The average error yielded by each of the models on a set of validation data, the *validation error* is calculated and put in a graph. At a certain point in the graph, when the complexity of the model starts to out-balance the complexity of the problem, the error starts to increase again after an initial phase of decrease, see FIGURE 3.4. Good practice is to settle on the model complexity generating the lowest average error of validation.

Producing sets for training and validation

The set of observations used to validate a model should ideally come from relevant sessions carried out 'in-field'. A model that passes a proper validation made

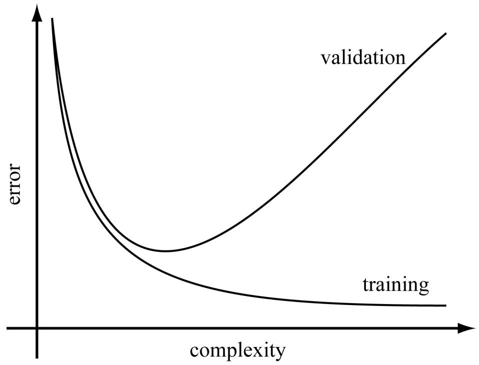


FIGURE 3.4: An illustration on the influence model complexity typically has on the average training- and validation error respectively.

with such data has a good chance to be robust to both expected and unexpected conditions of the application and stands a good chance to be useful in practice.

In reality, a single set of data is many times all that is available and it must hence be used both for training and validation. Several approaches have been suggested for partitioning a pool of samples into representative subsets such that all partitions captures the functional relationship of the problem but has independently attached contributions from errors and disturbances. *Random sampling*, in which observations are randomly split into sub-sets, is a popular technique because of its simplicity and because the statistical distribution of the subsets follows the statistical distribution of the entire set [38]. The retrieval of a large pool of samples suitable for random sampling might still be a too costly and time-consuming procedure. To partition an already small data set into even smaller subsets of training- and validation-data degrades the performance and the reliability of the training procedure even further and is therefore not recommendable. To make the best out of such a situation it is then better to use refined validation protocols based on re-sampling techniques. *Bootstrap* [39] and *cross validation* (CV) [40] are techniques commonly used to deal with the outlined problem.

■

4 Classification

Classification is to sort observations into labeled classes. The aim is to find a *classifier* implementing a *decision rule* that can be used to assign class labels to unknown observations. As before, let $\mathbf{x}_k$ represent an n-dimensional observation of an sample belonging to a particular, but unknown, *class* or *category*. Let c_i represent the *label* of that class. The classifiers output, $\hat{y}_k$, is a discrete valued variable providing a guess on the correct labeling. There are q different classes represented in the *class library*, so the guess can take on any of q possible values. Formally, the task of a classifier can now be described as finding a mapping,

$$\begin{aligned} &\hat{y} = f(\mathbf{x}) \\ &f : \mathbb{R}^n \rightarrow \mathcal{C}, \qquad \mathcal{C} = \{c_1, c_2, \ldots, c_q\} \end{aligned} \tag{4.1}$$

A common representation of $f(\cdot)$ is to define a set of discriminant functions

$$g_i(\mathbf{x}), \quad i = 1, \ldots, q \tag{4.2}$$

and define $f(\cdot)$ as [26]

$$f(\mathbf{x}) = c_i \quad \text{if} \quad g_i(\mathbf{x}) > g_j(\mathbf{x}), \quad \text{for all } i \neq j \tag{4.3}$$

See Figure 4.1 for an illustration.

The chance of succeeding with a classification task depends on many factors, among which the variability between observations within classes compared to the variability between classes is one example. Altogether, the information contained in the observations $\{\mathbf{x}_k\}_{k=1}^{N}$ is rarely sufficient to make a foolproof mapping and there will always be a risk of making miss-classifications. Therefore, probability measures are often integrated into the design of classifiers and the objective is to find the classifier yielding the lowest probability of making miss-classification.

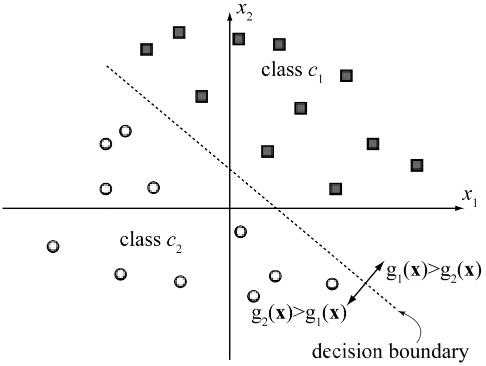

x_2
x_1
class c_1
class c_2
$g_1(\mathbf{x})>g_2(\mathbf{x})$
$g_2(\mathbf{x})>g_1(\mathbf{x})$
decision boundary

FIGURE 4.1: Classification using discriminants

4.1 Bayesian decision theory

Bayesian decision theory is a fundamental statistical approach to the problem of classification. The theory is extensive and contains important strategies for making classifications. A brief overview will be given here.

The main idea is to assign an observation $\mathbf{x}_k$ to the class being most likely,

$$f(\mathbf{x}) = c_i \qquad \text{if} \qquad p(c_i|\mathbf{x}) > p(c_j|\mathbf{x}), \qquad \text{for all } i \neq j \tag{4.4}$$

and a crucial part of Bayesian classification is to determine the the *a posteriori* probability[1] $p(c_i|\mathbf{x})$, i. e. the probability of the state of nature being c_i given the knowledge provided by the observation $\mathbf{x}$. The a posteriori is sometimes hard to learn directly, but a reformulation can be made using *Bayes formula*,

$$p(c_j|\mathbf{x}) = \frac{p(\mathbf{x}|c_j)p(c_j)}{\sum_{i=1}^{q} p(\mathbf{x}|c_i)p(c_i)} = \frac{p(\mathbf{x}|c_j)p(c_j)}{p(\mathbf{x})} \tag{4.5}$$

in which the *a priori*[2] probability $p(c_k)$, the *unconditional* probability $p(\mathbf{x})$, and *class conditional* probability $p(\mathbf{x}|c_k)$ are used instead.

It is implicitly understood that the reason for categorizing samples is to use the gained information to make decisions about what actions to take. To take an action can be associated with a certain *cost*. Taking a wrong action is costly and taking the right action is not. With this in mind, a loss function $\lambda(\alpha_i|c_j)$ is introduced describing the cost associated with taking action α_i if the state of nature is c_j. For a given observation, it is then possible to estimate the *conditional risk* of taking action α_i,

$$R(\alpha_i|\mathbf{x}) = \sum_{j=1}^{q} \lambda(\alpha_i|c_j)p(c_j|\mathbf{x}) \tag{4.6}$$

[1] 'a posteriori' denotes knowledge once the outcome of the observation is taken into account
[2] 'a priori' means knowledge present before a particular observation is made

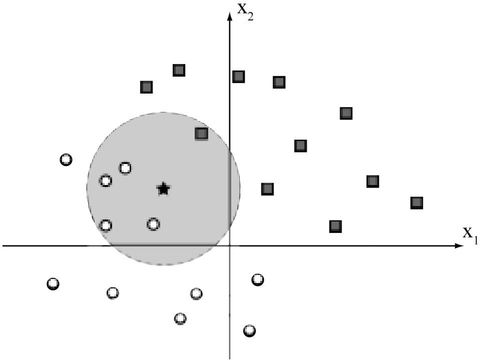


FIGURE 4.2: An observation with unknown class-membership (★) is being classified with the k-NN algorithm ($k = 5$). It is decided that the sample must belong to the ○-class.

and from there also define a decision rule suggesting taking the action minimizing the overall risk.

$$\text{take action } \alpha_i \quad \text{if} \quad R(\alpha_i|\mathbf{x}) < R(\alpha_j|\mathbf{x}) \quad \text{for all } i \neq j \tag{4.7}$$

The minimum overall risk is called the *Bayes risk* and is the best performance that can be achieved [26].

Many classification algorithms make use of Bayesian decision theory trying to estimate class-conditional densities from data. Traditionally, the densities are estimated either by assuming special parameterized functional forms, such as assuming multivariate normality, or by using non-parametric approaches. In between these two traditional approaches come mixture models (see e. g. [41]), in which very general functional forms with an adaptive number of parameters are used.

4.2 k-Nearest Neighbors

k-Nearest Neighbors (k-NN) is a simple non-parametric technique that has found wide use in practice [41, 42]. The classification of a sample, here encoded as $\mathbf{x}_i$, is made by placing the center of a hyper-sphere in the position of $\mathbf{x}_i$ and expand the sphere until k of the nearest training samples are contained within. The unknown sample can thereafter be classified according to a majority vote of the class-memberships of the k nearest neighbors, see FIGURE 4.2.

The algorithm might appear "rough" but can in fact be seen as a non-parametric probability density estimator being plugged-in into a Bayes classifier. Suppose the training data contain N_k samples in class c_k and N samples in total. Let V denote the volume of the sphere encompassing the k nearest training samples. Out of these k nearest neighbors k_k of them belongs to class c_k. The class conditional density, the unconditional density, and the priors can now be approximated as (see

e. g. [43] for details)

$$p(\mathbf{x}|c_k) = \frac{k_k}{N_k V} \tag{4.8}$$

$$p(\mathbf{x}) = \frac{k}{NV} \tag{4.9}$$

$$p(c_k) = \frac{N_k}{N} \tag{4.10}$$

All of the above put together in Bayes formula (eq. 4.5) gives

$$P(c_k|\mathbf{x}) = \frac{k_k}{k} \tag{4.11}$$

and it is readily seen that we should assign sample $\mathbf{x}_i$ to the class for which the ratio k_k/k is the largest.

Sophisticated variants of the k-NN strategy can be found in literature, among which many try to alleviate for the disadvantage that the native algorithm requires that all training samples are stored in memory.

4.3 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA), also known as Fishers Discriminant Function, originally evolved as a geometric approach. Loaning the original contributor's own words [44]:

> "When two or more populations have been measured in several characters, $x_1, \dots, x_n$, special interest attaches to certain linear functions of the measurements by which the populations are best discriminated."

In *binary classification*, where observations are classified to either of two possible classes $\{c_1, c_2\}$, LDA seeks a projection $\mathbf{w}$ in which the projected means of the classes are maximally separated. To account for variability in data, separation is not measured in the standard Euclidian norm but with the *Mahalanobis distance*. The Mahalanobis distance differs from the Euclidean distance in that it takes into account the correlations of the data and is invariant to the scale of the measurements [45]. The objective boils down to finding the projection $\mathbf{w}$ maximizing the ratio

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathbf{T}}\mathbf{S}_B\mathbf{w}}{\mathbf{w}^{\mathbf{T}}\mathbf{S}_W\mathbf{w}} \tag{4.12}$$

where $\mathbf{S}_W$ and $\mathbf{S}_B$ are the estimated between- and within- class covariance matrices [24]

$$\mathbf{S}_B = (\bar{\mathbf{x}}_2 - \bar{\mathbf{x}}_1)^T(\bar{\mathbf{x}}_2 - \bar{\mathbf{x}}_1) \tag{4.13}$$

$$\mathbf{S}_W = \sum_{j=1}^{2}\sum_{i\in c_j}(\mathbf{x}_i - \bar{\mathbf{x}}_j)^T(\mathbf{x}_i - \bar{\mathbf{x}}_j) \tag{4.14}$$

and $\bar{\mathbf{x}}_j$ is the mean of all observations belonging to c_j.

The solution for $\max J(\mathbf{w})$ can be solved as a generalized eigenvalue problem [26]. The problem can be generalized and used also in scenarios where multiple classes are considered [43, 26, 24].

Strictly, LDA is not a classifier but a supervised feature extraction technique. Assuming samples are to be classified to any of several populations having Gaussian distributions and equal covariance, then LDA can however be turned into a classifier based on the principles of Bayes. The decision rule can be set to [24].

$$f(\mathbf{x}) = c_i \quad \text{if} \quad \mathbf{w}^{\mathbf{T}}(\mathbf{x} - \bar{\mathbf{x}}_i)^2 < \mathbf{w}^{\mathbf{T}}(\mathbf{x} - \bar{\mathbf{x}}_j)^2, \quad \text{for all } i \neq j \tag{4.15}$$

Looking at LDA as a pure feature extractor, and comparing it with PCA, then it can be said that PCA extracts the most representative features while LDA extract the ones most discriminatory. LDA has a tendency to over-fit in small-sample-size problems while PCA is more robust to such tendencies [46].

4.4 Support Vector Machines

Support Vector Machines (SVM) are a family of learning methods invented during the 90^{ties} [47]. In its basic setting, SVM considers the problem of binary classification between two separable classes. It is possible to extend this setting and handle non-separable classes, multi-class problems, and the problem of regression, see e. g. [37, 27, 26].

Fundamentally, the SVM maps input data vectors $\mathbf{x}_k$ into a high dimensional feature space $\mathbb{Z}^d$ through some non-linear mapping

$$\mathbf{z}_k = \varphi(\mathbf{x}_k) \tag{4.16}$$

In feature space, a *linear* decision boundary is constructed in form of a hyper-plane.

$$\mathbf{w}_0^T \mathbf{z} + b_0 = 0 \tag{4.17}$$

Once the boundary is established, observations with an unknown class-membership can be classified in accordance to which side of the boundary their respective feature vectors $\mathbf{z}_k$.

The decision boundary is established from training data by the principle of maximum margin separation. The *margin of separation* can be thought of as the margin being created when two parallel hyper-planes are inserted in between the classes and then being pushed-up against each respective class, see FIGURE 4.3. The decision boundary is, per definition, the hyper-plane falling precisely in the middle of the margin.

Intuitively, good class separation and low generalization error is achieved when the margin can be made large. From principles in statistical learning theory, it shows that the generalization error of any classifier is bounded from above by a measure known as the *guaranteed risk*. The guaranteed risk depends on two terms; *(i)* the training error-rate, and *(ii)* a term that depends on the number of

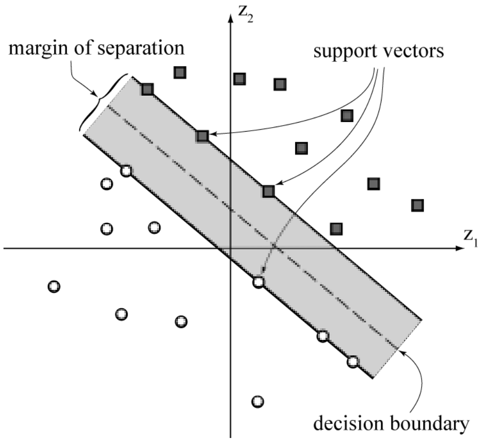


FIGURE 4.3: Support vector classification. If a margin is inserted between observations from two different populations and the margin is expanded as much as possible, then the observations falling on the edge of the margin are the support vectors. The mid-line of the margin can be used as a decision boundary.

observations available for training and the capacity[3] of the classifier. While many methods only aim at minimizing the first term of training error-rate, SVM puts zero value on this term and minimizes the second term. As a consequence, an SVM provides good generalization properties although everything has been learnt from observations and nothing has been assumed about probability distributions etc. This property is unique to SVM [27].

Considering the practical aspects of SVMs, to find the widest margin of separation is a *quadratic programming problem* [48] solved efficiently with numerical methods. The problem at hand is to find the optimal solution to

$$\max_{\arg\alpha} Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j c_i c_j \mathbf{z}_i \mathbf{z}_j^T \tag{4.18}$$

subject to

$$\alpha_i \geqslant 0, \ \forall i \quad \text{and} \quad \sum_{i=1}^{N} \alpha_i c_i = 0$$

where c_k is a class label taking values $\{+1, -1\}$ depending on whether $\mathbf{z}_k$ belongs to one of the classes or the other. Once the optimal solution α^o is found, the decision boundary can be set to

$$\mathbf{w}_0 = \sum_{i=1}^{N} \alpha_i^o c_i \mathbf{z}_i \tag{4.19}$$

[3] more precicely on the classifiers VC-dimension, see e. g. [27]

TABLE 4.1: Some commonly used kernel functions satisfying the necessary requirements indicated in text.

Name	Kernel Function, $K(\mathbf{x}_i, \mathbf{x}_j)$=
polynomial	$(\gamma \mathbf{x}_i \mathbf{x}_j^T + 1)^d$
radial basis	$e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$
gaussian radial basis	$e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}}$
sigmoid	$\tanh(\kappa \mathbf{x}_i \mathbf{x}_j^T + c)$

Details on the writings above and on how to come to their solutions can be found elsewhere, e. g. in [27, 37, 48]. It turns out, although not seen from this description, that for the optimal solution the α^o-values take non-zero values only for those feature vectors $\mathbf{z}^{(\mathbf{s})}$ falling exactly on the edge of the margin. The optimal decision boundary is hence upheld by a only a subset of the vectors contained in the training data. These vectors are called *support vectors*, see FIGURE 4.3. Note that the support vectors are those observations in trainingdata that are closest to the competing class and as such they are the ones hardest to classify. With this in mind, it is natural to see why SVM:s pay prime focus on these while setting the decision boundary.

On behalf of the non-linear mapping (eq. 4.16), the linear decision boundary constructed in feature space is non-linear in input space, and can potentially separate very complex scatter formations. Normally, problems risk to be introduced when making non-linear mappings to high dimensional feature spaces. It has been observed, though, that for constructing the linear decision boundary in feature space *explicit* mapping can be *avoided* by the introduction of the *kernel trick*. The kernel trick, first introduced by Aizerman et al. [49], transforms any linear algorithm that solely depends on dot products between vectors into a non-linear counterpart. This is seen from Mercer's theorem, see e. g. [27, 37], stating that any continuous, symmetric, positive semi-definite kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ represents the dot product of a possibly high-dimensional and non-linear function $\varphi(\mathbf{x})$,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j) \tag{4.20}$$

As an example, the dot products between feature vectors in eq. 4.18 becomes

$$\mathbf{z}_i^T \mathbf{z}_j = \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + 1)^d \tag{4.21}$$

where the polynomial kernel, see TABLE 4.1, was used at the far right hand. Note that the kernel functions themselves do not define the non-linear mapping from input- to feature- space, but are merely explicit writings of dot-products of the implicitly defined mapping $\varphi(\mathbf{x}_i)$. It can be noted that the non-linear functions generating a Gaussian radial basis kernel are infinitely-dimensional [27].

In summary, SVMs are supervised learning methods that map data into high dimensional feature spaces. In feature space, a linear decision boundary is constructed, rendering a non-linear counterpart in original input space. The boundary is entirely defined by support vectors, a subset of the vectors contained in training data. The support vector machine can provide good generalization properties although no problem-domain knowledge is given and everything must be learned from training examples only.

■

5 Regression

The term *regression* describe techniques used for the modeling of one or several *continuous* response variables[1] as a function of one or several continuous observations[2]. Regression techniques are often applied to map complex sensor responses to some quantitative property of the analyzed sample, like its concentration for an example.

Following previous terminology, the task given is to find a model $f(\mathbf{x}, \mathbf{w})$ that can map observations $\mathbf{x}_k$ to a corresponding quantity y_k making the smallest possible error. A quadratic loss function

$$L(y, f(X, \mathbf{w})) = (y - f(X, \mathbf{w}))^2 \tag{5.1}$$

on the difference between desired output and model output is normally used as error measure. The objective of regression becomes to find the parameter vector $\mathbf{w}$ yielding the minimum sum of error over the range of observations $\{\mathbf{x}_k, y_k\}_{k=1}^N$ available for training

$$\min_{\arg \mathbf{w}} \sum_{k=1}^{N} (y_k - f(\mathbf{x}_k, \mathbf{w}))^2 \tag{5.2}$$

This is the basis for all *least squares*-procedures.

[1] also known as dependent variable

[2] also known as independent- or explanatory variables

5.1 Linear Regression Techniques

A classic regression model is introduced by assuming a linear dependence between the observation $\mathbf{x}_k$ and the target property y_k

$$f(\mathbf{x}, \mathbf{w}) = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n = \mathbf{wx} \tag{5.3}$$

whereby the least squares approach becomes to minimize

$$\min_{\arg \mathbf{w}} \sum_{k=1}^{N} (y_k - \mathbf{wx}_k)^2 \tag{5.4}$$

The minimum solution can be found using the normal equations[3]. In matrix notation the solution reads

$$\mathbf{w} = \frac{\mathbf{X}^T \mathbf{y}}{\mathbf{X}^T \mathbf{X}} \tag{5.5}$$

If needed, it is straightforward to simultaneously handle several independent y-variables, whereby $\mathbf{y}$ and $\mathbf{w}$ above becomes $\mathbf{Y}$ and $\mathbf{W}$ instead. This is *Multiple Linear Regression* (MLR).

A shortcoming with this rather straightforward approach emerges when the variables within the observations $\{\mathbf{x}_k\}_{k=1}^{N}$ are mutually correlated and not independent. Then, the matrix inverse $(\mathbf{X}^T\mathbf{X})^{-1}$ becomes ill-conditioned or might not even exist. The problem can be circumvented by deriving suitable features from input data and use the features as input to the regression model.

5.1.1 Principal Component Regression

The technique *Principal Component Regression* (PCR) avoids ill-conditioned matrix inversions by applying a PCA on $\mathbf{X}$-data. Due to the properties of PCA, the principal components give a good representation of the original data and are moreover de-correlated. The procedure is to replace the original $\mathbf{X}$-data with the PCA score values $\mathbf{T}$ and apply an MLR on these instead. Since the score vectors are orthogonal, the matrix inverse $(\mathbf{T}^T\mathbf{T})^{-1}$ is well conditioned.

5.1.2 Partial Least Squares... or Projection to Latent Structures

Partial Least Squares (PLS), originally developed by H. Wold during the 70$^{\text{ies}}$, is a method to solve linear least squares problems. The method has been widely used within chemometrics and lately also within the chemical sensor community [50]. Originally, the method was described as a heuristic method but has thereafter been given a statistical analysis [51]. Since introduction, PLS has evolved and nowadays constitutes a toolbox of related methods (see e. g. [52]).

In similarity to PCR, PLS avoids collinear X-data through a subspace projection. The assumption using PLS is that the system or process under study is influenced by just a few underlying *hidden* variables. These variables are known

[3] derivation of the normal equations should be found in any textbook on linear algebra

as *latent variables*(LV's) or *latent structures*. Hopefully, the LV's are concealed within the space $X \in \mathbb{R}^n$ spanned by the sensor responses and the objective is find a smaller subspace spanning just the LV's. Practically, the objective is to find a subspace of X that has high covariance with Y.

Assuming there are c different LV's to extract, then the linear combination that potentially defines their subspace is written as

$$\mathbf{t} = \mathbf{W}\mathbf{x} \quad (\mathbf{T} = \mathbf{W}\mathbf{X}) \tag{5.6}$$

where $\mathbf{W}^{c \times n}$ is a matrix of *weights*. The $\mathbf{t}^{n \times 1}$ vectors are known as X*-scores* and are estimates of the LV's, or their rotation. The X-scores are, when multiplied by the *loadings* $\mathbf{p}^{n \times 1}$ a good summary of the original X-space

$$\mathbf{X} = \mathbf{P}\mathbf{T} + \mathbf{E} \tag{5.7}$$

where $\mathbf{E}$ is a small residual. The X-scores are also a good predictors of Y-space

$$\mathbf{Y} = \mathbf{C}\mathbf{T} + \mathbf{F} \tag{5.8}$$

where $\mathbf{F}$ is a small residual and $\mathbf{C}^{m \times c}$ is a weight matrix. In similarity to eq. 5.7, the weights together with a set of related Y*-scores* $\mathbf{u}$ can be used to give a good summary of Y-space

$$\mathbf{Y} = \mathbf{C}\mathbf{U} + \mathbf{G} \tag{5.9}$$

where $\mathbf{G}$ is a small residual.

A combination of eq. 5.6 and eq. 5.8 finally gives

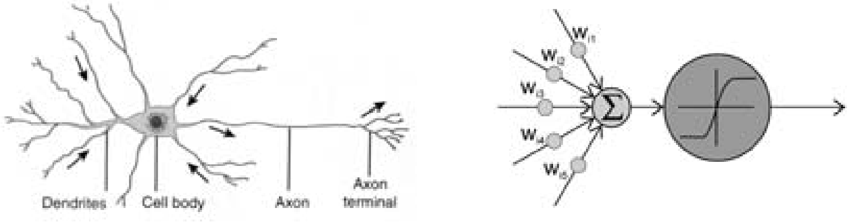
$$\mathbf{Y} = \mathbf{C}\mathbf{T} + \mathbf{F} = \mathbf{C}\mathbf{W}\mathbf{X} + \mathbf{F} = \{\mathbf{B} = \mathbf{C}\mathbf{W}\} = \mathbf{B}\mathbf{X} + \mathbf{F} \tag{5.10}$$

To run the necessary calculations, the NIPALS algorithm (see ALGORITHM 1) is mostly used. The algorithm (see e. g. [53, 54]) is an iterative procedure in which each step aim at satisfying the maximization

$$\max[\text{cov}(\mathbf{t}, \mathbf{u})]^2 \tag{5.11}$$

ALGORITHM 1: The NIPLS algorithm

```
Input: $\mathbf{X}$ and $\mathbf{Y}$
Output: $\mathbf{W}$, $\mathbf{P}$, $\mathbf{C}$ and $\mathbf{U}$
while a validation indicate that more components can be extracted do
    Get a starting vector $\mathbf{u}_0$, usually one of the columns in $\mathbf{Y}$;
    repeat until convergence
        $\mathbf{w}_i = \frac{\mathbf{X}^T\mathbf{u}_i}{\mathbf{u}_i^T\mathbf{u}_i}$;
        norm $\mathbf{w}_i$ to $||\mathbf{w}_i|| = 1$ (optional);
        Calculate $\mathbf{X}$-scores: $\mathbf{t}_i = \mathbf{X}\mathbf{w}_i$;
        Calculate $\mathbf{Y}$-weights: $\mathbf{c}_i = \frac{\mathbf{Y}^T\mathbf{t}_i}{\mathbf{t}_i^T\mathbf{t}_i}$;
        Update $\mathbf{Y}$-scores: $\mathbf{u}_i = \frac{\mathbf{Y}\mathbf{c}_i}{\mathbf{c}_i^T\mathbf{c}_i}$;
        Calculate relative change in $\mathbf{t}$: $\Delta\mathbf{t} = \frac{||\mathbf{t}_{(i-1)} - \mathbf{t}_i||}{\mathbf{t}_i||}$;
    until $\Delta\mathbf{t} < \varepsilon \quad , \varepsilon \approx 10^{-6}$;
    Deflate the present components from $\mathbf{X}$ and $\mathbf{Y}$:
    $\mathbf{p} = \frac{\mathbf{X}^T\mathbf{t}}{\mathbf{t}^T\mathbf{t}}$;
    $\mathbf{X} = \mathbf{X} - \mathbf{t}\mathbf{p}^T$;
    $\mathbf{Y} = \mathbf{Y} - \mathbf{t}\mathbf{c}^T$;
end
```



(a) Nervous neuron: Stimuli from other neurons are, via the dendrites, propagated to the cell nucleus. The nucleus responds to the pattern of signals and transmits, through the axon, an electrical impulse to the interconnecting neurons

(b) Artificial neuron: The inputs signals are multiplied by individual weights and thereafter summed together. The weighted sum passes through a function, mostly non-linear, and the output of the neuron is thereby calculated. The neuron adapts by updating the weight coefficients.

Figure 5.1: An illustration on the conceptual similarity between nervous and artificial neurons.

5.2 Artificial Neural Networks

Linear models are not capable to describe strong non-linear input-to-output relations accurately. If such a situation is at hand, non-linear modeling techniques are suitable to use. *Artificial Neural Network* (ANN) is the collective name of a family of biologically inspired techniques capable of handling non-linear behaviors. The ideas behind ANN's will be reviewed below, primarily since they represent an important category of non-linear modeling techniques, but partly due to their inspiring character. Other alternatives for non-linear modeling include support vector regression [37] and various counterparts of classic linear techniques, such as e. g. non-linear PLS.

It is easy get to inspired by human brain's capability of processing information. Physicists tell us that the basic building block of nervous tissue is the neuron cell, illustrated in Figure 5.1(a). Signals are sensed by the dendrons of a neuron leading them via electrical impulses to the cell nucleus. The nucleus responds to the pattern of signals by transmitting an electrical impulse through the axon. The axon terminal is interconnected to other neurons, through their dendrites. In this way, massive neural networks are grown in the brain. The average human brain (1350 g) contains about 85 billion neurons [55]. An organism learns by strengthening and weakening interneuron connections (dendrites and axons), causing suppressions of certain patterns of nerve impulses and an amplification of others [27, 56].

Inspired by the structure of nervous brain-matter, scientists have been tickled to mimic brain functionality by the implementation of artificial neural networks. A descriptive description on neural networks has been made by Aleksander and Morton [27]

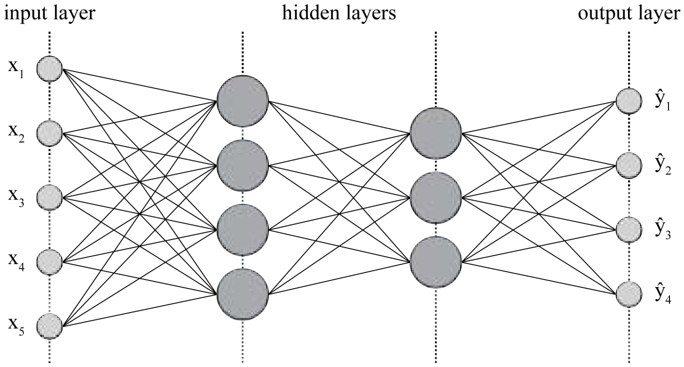


Figure 5.2: Artificial Neural Network; A layer of input nodes is cross-connected to a single or a series of hidden layers. The hidden layers consist of artificial neurons having the outputs of the previous layer as their inputs. Their own outputs are feed forward to the proceeding layer. The last layer consists of output nodes.

> “A (artificial) neural network is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experimental knowledge and making it available for use. It resembles the brain in two respects. (*i*) Knowledge is acquired by the network from its environment through a learning process.(*ii*) Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge”

In practice, the *nodes* are the basic building blocks of an ANN and represent the digitalized versions of the biological neurons, see Figure 5.1(b). The inputs of the nodes are multiplied by individual weights and summed together. The summary signal is passed through a function, mostly non-linear, whereby the output of the node is calculated. The nodes are structured into interconnected layers forming small networks, see Figure 5.2. Viewing the network as an entity, input vectors $\mathbf{x}_k$ can be fed into, and propagated through, the network to produce an output vector $\hat{\mathbf{y}}_k$. By adjusting the weights $\mathbf{w}$ of the neurons (which corresponds to strengthening and weakening the biological interneuron connections), the network can learn to produce the desired output for a given input.

The learning can be formulated as an optimization problem with the goal of finding the parameters $\mathbf{w}$ that minimize the difference between the network's actual output and the desired. Basic numerical optimization techniques, such as the gradient decent approach, can be used in the search for an optimum, although slightly more advanced procedures like the Levenberg-Marquardt method are commonly used in practice [27, 48]. A difficulty with using ANN is that the function to optimize easily gets complicated which results in a high risk of getting stuck in one of many local optima and thereby miss the global minimum. If configured correctly, an ANN network has the ability to make highly complex and non-linear input to output mappings. The networks can adapt themselves to the given data and to changes in the data structure [27, 23].

The criticism toward neural networks point to that relatively large data sets are mostly required training the network correctly [57]. There is risk in adapting the network to hard to data (overfitting), which results in poor generalization properties. It is also meant that an ANN cannot be interpreted in depth [57] and that the input–output mappings are of a “black-box” nature. However, most of the well known neural network configurations are implicitly equivalent or similar to classical statistical approaches [23] and it has been said that

> “...neural networks are statistics for amateurs...”

A good overview of artificial neural networks and statistical inference has been written by Howard Hua Yang et al. [58]

■

6 Source Separation

Source separation problems are those in which several signals have been mixed together and the objective is to find means to sort out the original source signals from the mixed signals.

Blind source separation is to conduct source separation without information (or with very little information) about the source signals or the mixing process. Blind source separation is thus a more difficult than regular source separation, but also more interesting due to its applicability to many practical problems.

6.1 Introduction

A classical illustration to the blind source separation problem is the "*cocktail party problem*". In the cocktail party problem, a number of people are talking simultaneously in a room (like at a cocktail party), and one is trying to follow one of the discussions. The human brain can handle this sort of source separation problem elegantly, yet it turns out to be a difficult problem to solve technically by means of data analysis and signal processing.

A technical setting for the problem will now be given. Consider a situation in which a number of source signals are transmitted by some sources. Denote the source signals $s_i(t)$. Assume there are a number of sensors or receivers which each registers a signal, here by denoted $x_j(t)$. There is a mixing process causing crossover effects from transmission to reception, so that each received sensor signal is a mixture of source signals. In the cocktail party problem, each speech represents a source signal. The sensor signals could be represented by a number of microphones that have been placed out in the room. Let us say there are three sources and three sensors, whereby the scene mathematically can be expressed as

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) + a_{13}s_3(t) \\ x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t) + a_{23}s_3(t) \\ x_3(t) &= a_{31}s_1(t) + a_{32}s_2(t) + a_{33}s_3(t) \end{aligned} \tag{6.1}$$

In the model above, it has been assumed that the mixing process is linear and allows to be represented by linear weights a_{ij} determined from the different positions of the microphones relative to the positions of the guests.

The questioned considered next is how to estimate the unknown source signals $s_i(t)$ from the observed sensor signals $x_j(t)$.

6.2 Blind Source Separation

The signal model above (eq. 6.1) can easily be extended to include more sensors and sources. In matrix notation the signal model becomes

$$X(t) = \mathbf{A}S(t) \tag{6.2}$$

where X is an n-dimensional vector representing the sensor signals, S is an m-dimensional vector representing the blind source signals, and $\mathbf{A}^{n\times m}$ is a *mixing matrix* representing the linear mixing process.

If the mixing matrix $\mathbf{A}$ is known, it is a quick operation to find its inverse[1] and extract the mixed sources from the sensor signals.

$$S(t) = \mathbf{A}^{-1}X(t) \tag{6.3}$$

On the other hand, the problem is considerably more difficult to solve if $\mathbf{A}$ is unknown. The problem of *blind source separation* (BSS) is to analyze a set of observable sensor signals and find an *unmixing matrix* $\mathbf{W}_{m\times n}$ estimating the inverse $\mathbf{A}^{-1}$. Once found, the blind sources could be estimated by making the calculations

$$\hat{S}(t) = \mathbf{W}X(t) \tag{6.4}$$

Since nothing is known about the sources, or the mixing process, some assumptions must be made to aid in the search for the unmixing matrix $\mathbf{W}$. A common approach is to assume the source signals have a definable regularity or are non-redundant in some meaning. For example, the signals may be assumed mutually statistically independent or decorrelated. Blind source separation thus separates a set of sensor signals into a set of other signals, such as e. g. the redundancy between the new signals is minimized.

6.3 Independent Component Analysis

Independent Component Analysis (ICA) is a statistical technique for revealing hidden latent variables in measurements and signals [33]. It is sometimes seen as

[1] Assumed that the mixing matrix $\mathbf{A}$ is invertible

an extension to PCA, able to extract features PCA, per definition, cannot find. The technique will be briefly presented here due to its close association with blind source separation problems.

ICA is defined as the problem to find a linear transformation, given by $\mathbf{W}$, such that the signals $z_i(t)$ below are as statistically independent as possible.

$$\begin{aligned} z_1(t) &= w_{11}x_1(t) + w_{12}x_2(t) + w_{13}x_3(t) \\ z_2(t) &= w_{21}x_1(t) + w_{22}x_2(t) + w_{23}x_3(t) \\ z_3(t) &= w_{31}x_1(t) + w_{32}x_2(t) + w_{33}x_3(t) \end{aligned} \tag{6.5}$$

It can be shown that the problem is well defined and can theoretically be solved if the hidden sources $s_i(t)$ are *non-Gaussian* [33]. If this assumption holds, the signals $z_i(t)$ can be made statistically independent and equal the hidden source signals $s_i(t)$.

6.3.1 Estimating statistical independence

An important part of ICA is to find an appropriate method to estimate statistical independence. Statistically independent random variables are uncorrelated, but uncorrelated variables are not necessarily independent. For that reason, methods finding un-correlated linear combinations, like PCA, are not sufficiently effective. A range of different estimation methods have been suggested in the literature. A very brief introduction to some of these will be presented below.

Measure of non-Gaussianity

The central limit theorem, a fundamental result in probability theory, tells that the distribution of sums of random variables tends toward a Gaussian distribution. A sum of two independent variables is thus more Gaussian than the two terms separately. To split up a mixture of sources, one strategy is to find the matrix $\mathbf{W}$ that maximizes the non-Gaussianity of the z_i variables. The *kurtosis* is a term from statistics and measures the “peakedness” of a probability distribution. It is defined as

$$\mathrm{kurt}(z) = E\{z^4\} - 3(E\{z^2\})^2 \tag{6.6}$$

The kurtosis is zero for for Gaussian variables and non-Gaussianity is typically measured by the absolute value of the kurtosis [33].

Another usable measure of non-Gaussianity is the *entropy*. Entropy is an information-theoretic concept (see e. g. [27, 26]) and can be interpreted as the degree of information that is provided by making an observation of a random variable. The more “random”, or unstructured, the variable is, the higher entropy. A Gaussian variable has the largest entropy among all random variables and entropy can hence be used to estimate non-Gaussianity.

Minimization of mutual information

Mutual Information (MI) is yet another concept from information theory (see e. g. [27, 26]). Basically, MI expresses the reduction of uncertainty of one variable, z_1, due to the knowledge of the other, z_2. In other words, it is a measure on the amount of information that is shared between two of more variables. Statistically independent variables share no information and there is consequently no mutual information between them. From this result the strategy follows to find the linear combination (eq. 6.5) minimizing the mutual information between the z_i signals.

Calculating the estimates

A practically important aspect of ICA is to find means to compute the needed calculations numerically. The estimates of statistical independence are non-linear functions and cannot be expressed using linear algebra. To find the maximum independence between variables hence risk to become a cumbersome task. Numerical optimization algorithms are therefore an integral part of ICA. The basic approach is to resort to some classical approach like gradient decent [48], but methods have been described that are particularly tailored to fit the structure of ICA problems. One such method is the FastICA algorithm [59, 60].

6.4 An alternative solution

ICA uses statistical dependence to find linear combinations of the observed signals that minimize the redundancy between the set of extracted signals. Alternative approaches exist, where one is to use autocovariance [59].

Sensor signals are mostly time signals with a smooth time structure. A simple measure on time structure is the linear autocovariance. For a time signal $x(t)$, the autocovariance is defined as

$$c_\tau^x = \mathrm{cov}(x(t), x(t-\tau)) \tag{6.7}$$

where τ is a time-lag constant. Considering multidimensional signals $\mathbf{x}(t)$, the time-lagged covariance matrix is defined as

$$\mathbf{C}_\tau^{\mathbf{x}} = E\{\mathbf{x}(t), \mathbf{x}(t-\tau)\} \tag{6.8}$$

Note that the autocovariance of a signal $x_i(t)$ can be found as the i^{th} diagonal element of the time lagged covariance matrix.

To find a linear combination maximizing time structure of the resulting signals, while simultaneously minimizing the time structure between signals, is equivalent to making all off-diagonal elements of the time-lagged covariance matrix zero, and the diagonal elements as large as possible. This problem is perfectly solved with canonical correlation analysis (see section 3.3.2, page 20) if the set of time-lagged data $\mathbf{X}(t-\tau)$ is used as $\mathbf{Y}$-data, i. e.

$$\mathrm{CCA}(\mathbf{X}(t),\ \mathbf{X}(t-\tau)) \quad \rightarrow \quad \mathbf{U}(t) = \mathbf{A}^T\mathbf{X}(t) \tag{6.9}$$

where $\mathbf{U}(t)$ serve as estimates of the source signals and $\mathbf{A}$ is the corresponding demixing matrix (compare to eq. 3.8). It is possible to extend the approach by e. g. considering multiple time-lags $\{\tau_1, \tau_2, \ldots\}$ simultaneously. This will not be done here.

A few notes are in place commenting on the difference between the CCA approach and ICA. ICA analyses random variables and does not consider the structure between consecutive samples. ICA is a general and powerful method, but if a time structure exists, ICA will not exploit it as the CCA-approach does. Further, ICA cannot separate Gaussian signals, so in cases where the signals are Gaussian but correlated over time, the autocovariance approach is an important alternative. ■

7 Drift Counteraction and Calibration Transfer

The modeling techniques presented in this thesis adopt the principle that by learning from a set of observations, it is possible to generalize and apply the gained knowledge to other, previously unseen, observations. The fundamental assumption that needs to be made while using such a principle is that the characteristics of the sensors remains stable throughout learning and throughout utilization.

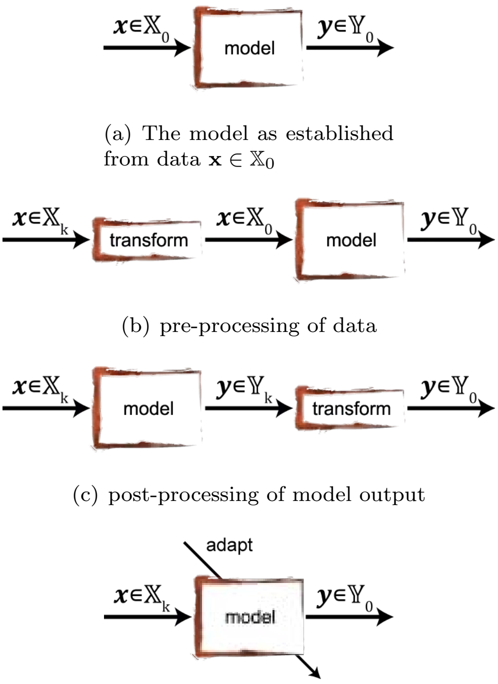
As discussed earlier, the characteristics of a sensor may suffer from changes induced by drift (section 2.3.2) or reproducibility issues (section 2.3.3). If so, the critically important assumption on stability might become violated which may result in loss of performance and in an increased risk of making misinterpretations of data.

This chapter introduces techniques counteracting for effects caused by drift and irreproducibility. Two terms will be introduced; *drift counteraction* and *calibration transfer*. In loose terms, drift counteraction can be seen as counteraction of gradual *temporal* changes. Calibration transfer can be seen as the counteraction of *spatially* induced changes, e. g. between different sensor systems. In practice, the terms and thereto-related techniques are overlapping.

7.1 Overview

To organize and overview different strategies for counteraction, this thesis identifies three different 'dimensions' along which different approaches can be sorted. The dimensions are: *effort (i)*, *where (ii)* and *when (iii)*.

The *efforts* needed to counteract for unwanted changes is of practical importance. Counteraction procedures which autonomically identify changes and



(a) The model as established from data $\mathbf{x} \in \mathbb{X}_0$

(b) pre-processing of data

(c) post-processing of model output

(d) model adaptation

Figure 7.1: The different approaches for drift counteraction and calibration transfer.

take appropriate action constitute the ultimate scenario. In worst case, the complete pattern recognition system must regularly be re-modeled by hand and from scratch. Many procedures need to invoke *recalibration measurements* in order to identify malicious changes. These measurements are often made on predefined and well-known *reference samples*. Unfortunately, reference samples tend to be logistically difficult to handle and from a practical viewpoint, it is more convenient if well known but otherwise arbitrary samples can be used instead. In either case, the goal is to run as few and simple recalibration measurements as possible and to re-use the information contained within the original model as much as possible.

A second distinction between counteraction methods relates to *where* the counteraction is applied, see Figure 8.2. Some methods apply a *pre*-processing that counteracts for deficiencies in sensor data before it is analyzed by the pattern recognizer. Here, sensor signals can be corrected as a joint group (multivariate approach) or one by one (univariate approach). Another approach is to let the pattern recognizer process the deficient sensor data and then *post*-process the outcome. A third alternative is to counteract *in* the process by properly adjusting the main model.

The third dimension of categorization involves *when* it is determined on how to counteract. Some methods try to find features unaffected by drift, whereby only an *initial* assessment is made. Adaptive methods for drift counteraction *continu-*

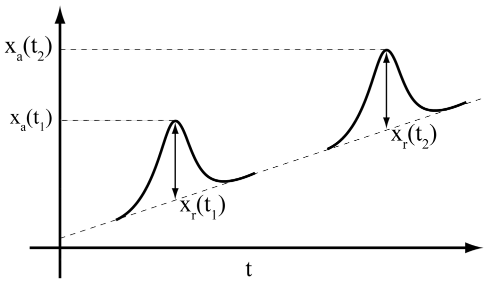


FIGURE 7.2: In systems where a *baseline* is available, the response relative to the (drifting) baseline might be a parameter that is insensitive to drift.

ously update the counteraction model. *Time-discrete* updates of the counteraction model made at certain intervals are the strategy used by calibration transfer algorithms and by many drift counteraction procedures.

7.2 Counteraction Procedures... some examples

The numbers of techniques for drift counteraction and calibration transfer are rare compared to the wealth of techniques described to solve the classical problems of e. g. classification and regression. Within the field of sensor signal processing, the methods described below are frequently being refereed to.

7.2.1 Drift-free parameters

One approach to drift counteraction is to identify features within the data that are insensitive to drift. In some regards, the *Component Correction* algorithm, described later, is such a method. Another example is to utilize the baseline of a signal, if such is available. If the base-line response is also affected by drift, the sample response relative to the base-line response might be a parameter free from drift, see FIGURE 7.2. In effect, this latter example makes a variant of the *additive drift correction.*

7.2.2 Additive drift correction

An easy-to-use univariate recalibration method, running on reference samples, has been described by Fryder et al. [61]. The method, known as *additive drift compensation*, assumes that drift makes an *additive* component $d(t)$ that is independent from the response component $s(y)$ of the sensor, i. e.

$$x(y,t) = s(y) + d(t) \tag{7.1}$$

A *master* reference is constructed at time t_0 by recording the reading x_r^0 while measuring on the reference sample y_r,

$$x_r^0 = x(y_r, t_0) = s(y_r) + d(t_0) \tag{7.2}$$

Similar reference readings are then made at time t_k and the difference is calculated

$$\Delta d_k = x_r^k - x_r^0 = d(t_k) - d(t_0) \tag{7.3}$$

Within a period shortly after time t_k it can be assumed that the approximation $d(t) \approx d(t_k)$ is just. The drift counteracted sensor signal $x(y,t)'$ can then be calculated as

$$x(y,t)' = x(y,t) - \Delta d = s(y) + d(t) - d(t_k) + d(t_0) \approx s(y,t) + d(t_0) \tag{7.4}$$

The calculation above transforms the sensor signal back to its state at t_0, making it possible to apply data acquired approximately at $t = t_k$ to a model established from data acquired at t_0.

7.2.3 Multiplicative drift correction

Multiplicative Drift Correction [62, 63] assumes *multiplicative* drift, i. e. where drift affects not the bias but the sensitivity of the sensor

$$x(y,t) = s(y)d(t) \tag{7.5}$$

In a fashion similar to additive drift correction, drift quotients are calculated comparing the magnitude of the master and the k^{th} reference response

$$q_k = \frac{x_r^0}{x_r^k} \tag{7.6}$$

Corrections are then made according to

$$x(y,t)' = g_k x(y,t) = \frac{x_r^0}{x_r^k} s(y)d(t) \approx s(y)d(t_0) \tag{7.7}$$

Applicable to both additive- and multiplicative drift correction, it may be preferable to gather a time-series of $\{\Delta_k, \Delta_{k+1}, \ldots\}$ or $\{q_k, q_{k+1}, \ldots\}$ values. By fitting a curve to the time-series, such a procedure makes it possible to use interpolation to calculate correction factors to use in between adjacent reference measurements.

7.2.4 Component Correction

Component Correction (CC) [64] is a multivariate technique taking all sensors into account simultaneously (An example on the use of component correction can be seen in FIGURE 7.3). Two different versions of the algorithm has been suggested, one based on PCA and the other based on PLS. Here the PCA version will be described.

The CC-algorithm runs in two phases. The first phase uses PCA to analyze reference samples that have been measured over time and thereby are affected by drift. Except from noise, drift is the only source of variation within the set of references and the first loading vector $\mathbf{p}_r$ describes its main direction. In the

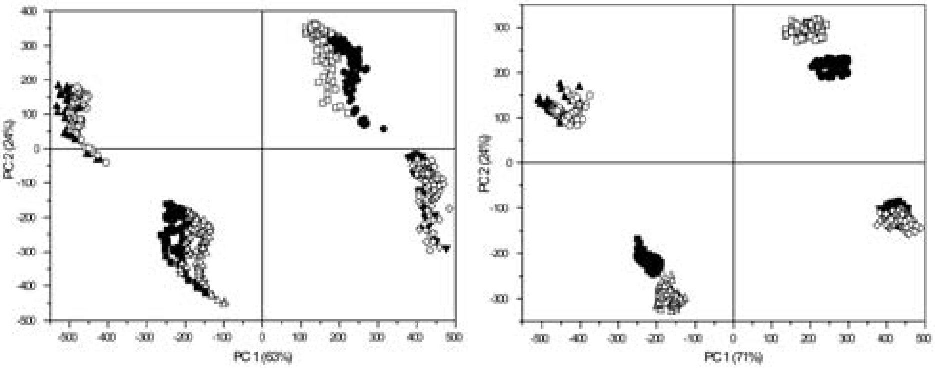


Figure 7.3: Component Correction on different gas mixtures as measured by a drifting sensor system. The top figure shows a PCA plot prior to CC. The "tails" are caused by drift and it can be seen in the bottom figure that these are efficiently removed by CC. (T. Artursson et al. , Journal of Chemometrics, 14(2000),pp711-723. Copyright John Wiley & Sons Limited. Reproduced with permission.)

second phase, data is corrected by removing all variations along the drift direction. Algebraically, this is achieved by projecting data on the drift direction, rendering the score vector

$$\mathbf{t} = \mathbf{X}\mathbf{p}_r \tag{7.8}$$

Data is then corrected through subtraction of the bilinear expression $\mathbf{t}\mathbf{p}_r^T$, whereby the final expression yields

$$\mathbf{X}' = \mathbf{X} - \mathbf{t}\mathbf{p}_r^T \tag{7.9}$$

Note that the CC-algorithm is a reference sample method due to its fist phase. It also runs according to the principle of identifying and exploiting drift free features. Component Correction assumes that all samples, including the references, drift in the same direction. It is also assumed that no "important" information is found along the drift direction, because if so this information will be removed also. Non-linear drift can be counteracted to some extent by extracting and removing more directions. Purely non-linear techniques might be used when severe non-linear drift is experienced. See for instance [65] where ANN:s are used as a direct method for calibration via references.

7.2.5 Self adapting models

Data analysis procedures have been described that are able to self-adapt and thereby automatically counteract for drift. Such models require minimum of interaction and are beneficial to use on-line and in mass-market applications. A downside with self-adaptation is the risk that the model adapts to, and hence suppresses, the useful information. This might occur e. g. in cases where both the samples and the sensor system change systematically over time.

Self organizing systems based on SOM (section 3.3.2) have been described and used for adaptive drift rejection purposes [66, 67]. The basic idea is to place a SOM-net in response space and assign the different net-nodes to the different classes of the samples. While in use it is then possible to slightly adapt the network to follow the changing sensor responses and the distribution of incoming samples. The different nodes retain their assigned labels, although they move in space, and new samples can be classified according to which nodes that are the closest neighbor. Further development on this theme has been made in which multiple SOM:s were used to track one class each [68].

■

8 Summary of Work

This chapter will try to picture the environment from where this thesis evolved. The projects from which data has been captured will be introduced and a background will be given to the the work eventually leading to the papers included in this thesis.

8.1 The research environment

The work preceding this thesis has mostly been conducted at S-SENCE (see below) and the Department of Applied Physics, Linköpings University. The department has tradition in gas sensing and thereto dedicated sensors, such as e. g. the MISFET. During the nineties, there was a growing interest in chemical sensor arrays, which resulted in the development of an electronic nose and later the electronic tongue. During this era, in 1995 to be precise, The Swedish Sensor Centre (S-SENCE) was launched (it then settled in 2005). S-SENCE was a center of excellence dedicated to bio- and chemical sensing. The motive for starting the center was to create an environment in which academic research could fertilize with commercial interest to bring forward an interesting development of bio- and chemical- sensing applications. The members of the center were the Governmental Agency for Innovation Systems, Linköping University (on behalf of the department of Applied Physics), and a selection of companies finding advanced sensor systems interesting and useful. S-SENCE was sub-divided into three different sensing areas: *gas sensing*, *liquid phase sensing*, and *bio sensing*. A fourth group, the *data evaluation* group, supported the other groups in data evaluations and signal processing issues. The respondent of this thesis was part of the data evaluation group.

During his time at S-SENCE, many valuable insights were given to the respondent through cooperation with the industry. Aside from the work at academia, the respondent was also able to take a short leave and work for a company commercializing gas sensor systems. Under this period, practical work related to the academic research was on hold. Nevertheless, many valuable experiences were made and gave, in retrospect, the respondent an emphasized view on practical difficulties related to the current field of research.

8.2 Paper I

Facilities like schools, small industries, offices, and households are quite commonly equipped with moderately sized boilers for heat production. The combustion in such boilers is often inefficient, producing flues with high levels of carbon monoxides, hydrocarbons (HC) and produces ashes containing unburned charcoal. Efficient control of the combustion would gain the environment and render a better fuel economy, but would on the other hand require proper on-line monitoring of the boiler. Legalizations and popular opinion have forced through combustion control systems at larger plants, but the technology is not transferable to small-scaled boilers. The hope is that small and cost effective sensors, in near future, can be used to control systems for small- and medium-scaled boilers.

From the viewpoint of Applied Physics, the suggested application offers an opportunity to contribute with sensor devices operating in harsh environments and under elevated temperatures. The main focus has been to develop and evaluate field effect devices based on silicon carbide. Compared to silicon, silicon carbide is more chemically inert and has a wider band-gap, which in theory makes it more appropriate considering the given conditions [11]. Much work has been done trying to evaluate and refine the ideas [17, 18]. Important work has also been done for the automotive industries, a closely related field of application. The aim here is to develop alternatives to lambda-sonds and sensors for car exhaust monitoring.

Prologue

Prior to the respondents engagement for research, the department had initiated collaborations with the Swedish power- and heat producer 'Vattenfall' and experiments had been run on a 100 MW boiler located in Nyköping, Sweden. All details regarding the experimental setup and the achieved results are reported in [17]. The boiler was used for generating power and heat to local households and industries. The experimental setting was based on a high temperature electronic nose (HTe-nose) specially developed for harsh environments. The HTe-nose consisted of three MISiCFET sensors, nine metal-oxide sensors and a linear lambda sensor [17].

The outcome of the experiment was promising. By using PCA it was possible to identify clusters in sensor data that could be related to different operating modes of the boiler. An attempt was made to model for CO concentration in the flues under normal operating conditions by using the PLS algorithm. The constructed model performed well with training data, but less satisfactorily with data measured

14 days and 39 days after establishment of the model. It was suggested that the observed model degradation was due to drift of the chemical sensor array.

The respondent engaged into the project and wanted to contribute to it by making further improvements on the data analysis procedures. As drift had been pointed out as an issue to be dealt with, attempts were now made to identify and counteract for its effects. The experimental data that had been recorded did not, due to practical limitations, include reference samples made on controlled samples. By that, there were no natural means to identify and track effects possibly caused by drift. At this stage, the taken approach was to replace the missing reference samples with *pseudo-references* to be found within the available data.

To identify a limited sub-set of samples that could play the role as pseudo-references, the following strategy was implemented. In parallel to each sensor array measurement, the experimental setup included measurements made with accurate laboratory equipment to track CO_2, O_2, NOx and HC. The instrumental measurements were analyzed with PCA. Within a 2-dimensional PCA-plot, a small and dense region of samples with even temporal distribution could be identified. Assuming that the instruments gave reliable drift stable results, samples falling within the defined region was selected to represent the pseudo references.

Sensor array measurements made on the identified pseudo references were now studied. Unfortunately, a PCA analysis of the sensor readings did result in any distinguishable patterns or trends that could be related to systematic changes such as drift. A component correction [64] procedure was also applied, using the pseudo-references, but no improvement was achieved. By now it was concluded that drift was either not present or not detectable with the used methods. The work was important, providing insight to the application, but did not contain enough essence to bear a publication on its own. In retrospect, the pseudo reference method may present little novelty, but should be well worth to practice in applications requiring drift compensation.

Work directly related to the paper

A deficiency with the experiments run in Nyköping was that they were run with the boiler in operational production, which resulted in a dataset with little variation. It was believed that a wider coverage of operating conditions would make a better dataset for modeling. The department was now able to initiate a collaboration with the commercial research institute 'Termiska Processer'(TPS), Studsvik. TPS could offer a 200kW boiler being used for experimental purposes only. This opened up for experiments with vivid alteration of settings, enabling the combustion process to run under a wide span of conditions. An experimental setup similar to the one used in Nyköping was used and experiments were run for 4 days, resulting in data with a relatively wide and evenly distributed span of sensor responses. As before, the main objective of the project was to find means to measure the composition of flue gases. This time, the prioritized targets were CO, HC and O_2.

The respondent wanted to engage into the project and make a contribution devoted to data analysis issues. The following reasoning defined the outline of the data-analysis related sub-project: Flue gas is a composition of e. g. CO, HC and

O_2. Each sensor in the sensor array is non-selective (see section 2.4) and respond to a spectra of specimens present in the flue. Since each sensor is unique in its response characteristics, each signal represents a unique mixture of the contributions given by the specimens in the flue gas. The objective of the data analysis must be to find means to demix the mixed signals as to get the contribution from each specimen.

Blind Source Separation techniques (see chapter 6) suit very nicely into the outlined objective and the foremost choice was to evaluate the experimental data using an ICA algorithm (see section 6.3). Unfortunately, the result of ICA was poor, possibly due to strong Gaussian processes within the analyzed signals. Another BSS approach, based on the principle of using second order statistics and auto-correlation (see section 6.4), was therefore used instead. The latter approach was able to predict the target gases quite well. The work, included as Paper I in this thesis, was reported in a scientific journal.

Epilogue

During the progress of the work presented above, the respondent got interested in how to proceed with the control and monitoring of boiler combustion processes. Techniques on *statistical process monitoring* and *statistical process control* were studied. Such techniques are often based on PCA, PLS etc and have been applied into various process industries. The respondent made some trials on data already collected. Tests were also made to split-up time-continuous data, like online boiler measurements, into different time-scales and then process each scale separately. The separation was made using wavelet techniques. Unfortunately, the explorations did never pay-off and, at the time, no threads could be identified that were promising enough to support a detailed study. In retrospect, this is certainly an interesting field in which more can be done. One of the difficulties moving into this field is to get hold on resources that enable a supply of relevant data to study.

8.3 Paper II

Due to experiences gained through collaborations with Vattenfall and TPS, it became interesting to better understand the response characteristics of the MISiC-FET devices being used. A study was initiated and experiments were run in a controlled laboratory environment. The purpose of the study was to examine in which way a set of MISiCFET devices responded to different mixtures of HC, CO and O_2. The respondent did not take part in the experimental work, but became engaged in the exploration of the retained datasets.

Work directly related to the paper

It was soon discovered that the tested sensors were cross sensitive (see section 2.4) to CO and O_2. Different models were constructed aiming at describing the cross-sensitive response patterns mathematically. At that time, the respondent had recently been on a leave, working for a commercial company developing gas sensor systems. The leave gave insights into many commercialization aspects of the sensor development and the respondent decided to make attempts to utilize the mathematical models to tackle the problem of calibration transfer (see chapter 7). From the derived models, it was possible to parameterize sensor- to-sensor differences. An idea evolved to use these parameters as 'data-sheets' describing the characteristics of each sensor. Potentially, the data-sheets were easy to distribute and yet contained information making it possible to compare the characteristics of different sensors. A calibration transfer procedure was suggested that emphasized ease of handling and applicability to large scaled sensor production. The work resulted in a manuscript submitted to a scientific journal. The manuscript is enclosed in this thesis as paper II.

Comments on the work

In retrospect, the respondent recognizes that the suggested approach is merely a first suggestion to the quest of finding an acceptable strategy for large scale calibration transfer. To find a simple but powerful method is a very difficult task and, to the respondents awareness, no similar work has been published directly trying to tackle these kind of problems. Therefore, the suggested approach makes a justifiable contribution to the field, brings up the encountered problems to the surface, and invites to further development.

8.4 Paper III

As indicated in chapter 2, the Scanning Light Pulse Technique is a technique providing means to evaluate different sensor configurations conveniently, without having to physically produce test sensors. Scientists at the department of Applied Physics were, and still are, active within the field of SLPT research and development. During informal discussions, the respondent and the SLPT scientists saw an opportunity to make a joint effort in combining the SLPT technique with multivariate data analysis procedures. Ideas were refined and it was settled that an interesting approach would be to support SLPT with techniques for variable selection. The strongpoint of such a combined approach would be that potential sensor candidates could be screened and a sensor array could be optimally configured for a predetermined application, without having to physically produce, assemble, and evaluate sensors for test purposes. An experimental plan was set up and measurements were made.

Work directly related to the paper

In SLPT, a surface is prepared with non-uniform properties. Under exposure of predetermined test gases, the surface is scanned in $x \times y$ discrete positions rendering a set of $x \times y$ variables. Due to the geometric relationship between the scanned positions, and due to the smoothness of the non-uniform surface, each variable is closely related to its neighbors. The first intention of the respondent was to construct an algorithm that accounted for the geometric structure and selected interesting *regions* rather than interesting *positions* (=variables). The procedure was to place Gaussian-shaped kernels on the response images. Each kernel represented the region it covered and its use was to give a local weighted summary response. The number of kernels, their position, their shape, and their orientation was to be optimized in such a way that the joint response from all kernels together carried as much information on the test gases as possible. The search for optimum was made with *genetic algorithms*, a special regimen of optimization procedures. As it turned out, the kernel based optimization problem did not solve easily. For each run, the optimization procedure found a new local minima and the respondent was unable to get consistent results.

To resort from the discovered difficulties, the kernel approach was abandoned and a more standard variable selection method using forward selection was implemented. The alternative approach was successful and resulted in a manuscript submitted to a scientific journal. The manuscript is included in this thesis as paper III.

8.5 Paper IV

As one of the last engagements as a Ph.D-student, the respondent was given the chance to run a project at the Swedish Defence Research Agency. The outcome of that work resulted in a manuscript included in this thesis as paper IV.

To assemble sensors into distributed networks is a topic gaining interest. Security related applications such as perimeter- and area surveillance are often mentioned. Other applications such as e. g. pollution- and municipal water distribution monitoring have been suggested.

The Swedish Defence Research Agency takes part in the development of distributed sensor systems. In 2003, a field campaign was made in which a network of acoustic and seismic sensors were distributed in a small geographic area, covering a road segment. Different types of vehicles were then let to drive along the road segment while recordings were made. The main objective of the campaign was to illustrate and study problems related to area surveillance. Interesting sub-tasks included to study and evaluate techniques able to track and identify vehicles.

Work directly related to the paper

The respondent was given access to the described field campaign data. At start, no particular aim was set and time was spent on basic exploration. Previously, successful tracking of passing vehicles had been made using the received acoustic signals. Successful classification of passing vehicles had also been demonstrated, by using the received spectral signatures as input to an support vector machine classifier. Classification on vehicles in an environment where multiple vehicles passes simultaneously had not been studied before, though, and the respondent started to study the problem. Techniques for blind source separation were studied and judged against each other. After initial testing, it was decided that a combination of spatial beam-forming and an ICA based blind source separation of frequency domain signals was the appropriate method. A thorough evaluation of the method showed promising results and indicated its practical usefulness. Details on the study can be found in the manuscript enclosed as Paper IV in this thesis.

8.6 Additional work and results

During the course of the work leading to the presented papers, many additional ideas have been tried. Some of these have been abandoned and others have been put aside as interesting seeds to let grow at other occasions. A pair of these ideas will be presented here as additional work and results.

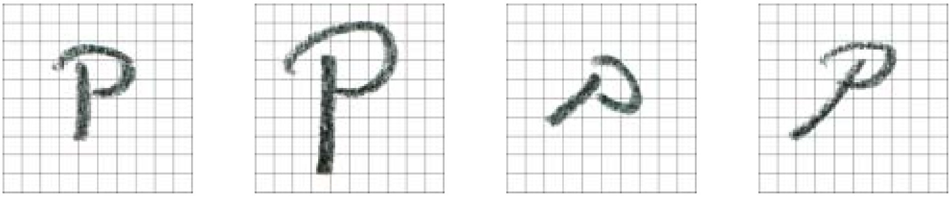
8.6.1 Analysis of impedance spectroscopy data

Electrochemical impedance spectroscopy is a powerful tool for evaluating chemical and physical processes in solutions and in solids. The respondent has participated in a project in which the purpose was to use the technique to measure soot and diesel contamination in engine oils. The final result, in which an ordinary PLS algorithm was used to predict the degree of contamination from impedance spectra, is reported in [69].

To evaluate the impedance spectra, attempts were first made to parameterize the data instead of using the raw measurements as basis for further data analysis. Impedance is an electromagnetical property and each electrochemical impedance spectra should, to some extent, be comparable to the spectra generated by an equivalent electrical circuit of resistors, coils and capacitors. The respondent and collaborators studied the possibility to develop a method to find the equivalent circuit that best matched an impedance spectroscopy measurement and then use the equivalent circuit to parameterize the data. If successful, such parametrization would increase the physical interpretability of the analyzed sample, would possibly increase measurement robustness, and would compress the dataset. A similar approach, using a fixed model of two exponential decay functions, has been described and applied to compress electronic tongue data [70]. To not constrict to a fixed model, system identification theory and filter theory were studied and various optimization procedures were evaluated. Unfortunately, no approach could be found that gave better interpretability or performance, as compared to a simple PLS model on raw data. The equivalent circuitry idea was abandoned. Additionally, an approach using Orthogonal-PLS [71] was tested and showed to perform well. However, the main focus of the project was never meant to be onto advanced signal processing and it was decided to use the more well known PLS algorithm as basis when publishing the study.

8.6.2 Tangent Distance

Tangent distance classification, introduced by Simard et al. , is a methodology originating from the application of handwritten character recognition [72]. The tangent distance implements the idea to make methods invariant to definable



(a) normal (b) scaled (c) rotated (d) skewed

FIGURE 8.1: In handwritten character recognition, a classifier should be able to see through certain characterizable alterations and properly compare characters being written with a different size, rotation, skewness etc to the "normal" character contained within the class library.

changes of the analyzed observations $\mathbf{x}_k$. In character recognition, the concept is used to make classifiers insensitive to alterations in e. g. the scale, rotation or skewness of written letters, see FIGURE 8.1. This can be achieved by implementing a similarity measure being invariant to a set of predefined transformations. Consider $d_t(\mathbf{x}, \mathbf{y})$ as the similarity measure between the vectors $\mathbf{x}, \mathbf{y}$ and define e. g. a scaling transform as

$$L_\alpha : \mathbf{x} \mapsto \mathbf{x} + \alpha\mathbf{x} \tag{8.1}$$

With this setting, the similarity measure must not change upon changes in α, that is

$$\frac{\partial}{\partial\alpha} d_t(\mathbf{x} + \alpha\mathbf{x}, \mathbf{y}) = 0 \tag{8.2}$$

The described ideas were adopted by the respondent and fitted into a PCA-algorithm specially suited for a voltametric electronic tongue (section 2.5.5). Such a device consists of a number of electrodes upon which voltage pulse-forms are applied and currents are registered and used as measurements. The current amplification of each electrode can sometimes be considered as a parameter that changes over time and the goal was to find a PCA decomposition that was invariant to these changes.

Considering an electronic tongue with four electrodes, variations in electrode amplification can be modeled by the transform

$$L_\alpha : \mathbf{x} = [\mathbf{x}_a \ \mathbf{x}_b \ \mathbf{x}_c \ \mathbf{x}_d] \mapsto [\mathbf{x}_a \ \mathbf{x}_b \ \mathbf{x}_c \ \mathbf{x}_d] + [\alpha_a\mathbf{x}_a \ \alpha_b\mathbf{x}_b \ \alpha_c\mathbf{x}_c \ \alpha_d\mathbf{x}_d] \tag{8.3}$$

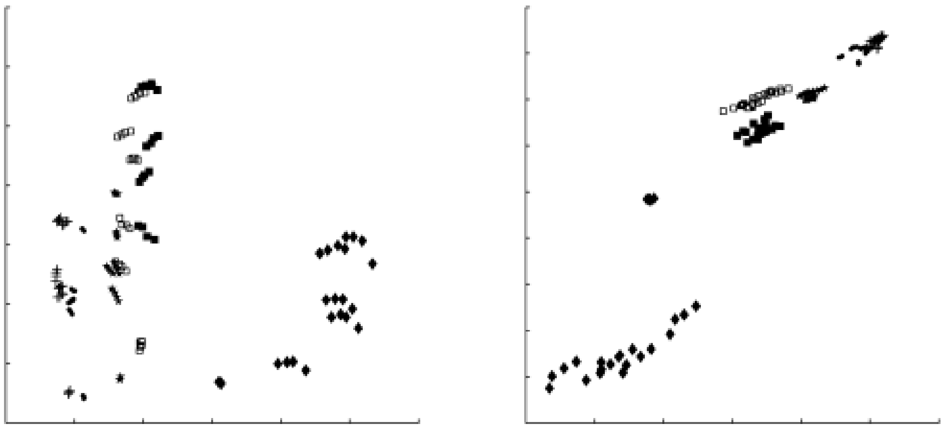
Inserting the transform into the score vector decomposition $\mathbf{t} = \mathbf{xP}$ gives

$$\mathbf{t} = \mathbf{xP} \mapsto \mathbf{t} = [\mathbf{x}_a \ \mathbf{x}_b \ \mathbf{x}_c \ \mathbf{x}_d]\mathbf{P} + [\alpha_a\mathbf{x}_a \ \alpha_b\mathbf{x}_b \ \alpha_c\mathbf{x}_c \ \alpha_d\mathbf{x}_d]\mathbf{P} \tag{8.4}$$

To have score vectors that are invariant to changes in amplification it requires that

$$\frac{\partial\mathbf{t}}{\partial\alpha_i} = 0 \quad \text{for } i = \{a, b, c, d\} \tag{8.5}$$

The adapted PCA algorithm tries to find the subspace (through the loading vectors in $\mathbf{P}$) minimizing the sum of squared residuals (see eq. 3.6, page 19), constrained to that the derivatives of eq. 8.5 are as small as possible.



(a) ordinary PCA

(b) modified PCA

FIGURE 8.2: Fruit juice measurements made with an electronic tongue. The figures shows a comparison of ordinary PCA and a PCA approach modified to stand invariant to electrode changes giving rise to variations in the degree of current amplification. The figure indicate that the modified approach yield less scattered scores and that the four sub-sets of data that are included are more tightly gatehered (most easily seen for the ♦-samples)

The approach was tested on measurements made on various fruit juices and preliminary result indicated that suggested adaptation could give slight improvements, see FIGURE 8.2.

8.7 Final comments on the conducted work

This final chapter of the thesis has presented the background to the four main contributions of paper I–IV and thereto also summarized the work behind some additional results. It is now time to conclude by providing a general perspective on the graduate studies leading to the thesis.

Initially, during the first time as a graduate student, the respondent felt that every discovered article and technique was ingenious and great ideas evolved on how to apply the described algorithms to his current projects. It was soon discovered, however, that many obstacles were in between the respondents vision and reality. Most of the times, the obstacles were related to practical difficulties and due to the fact that data was taken from live applications.

In live applications, it is difficult to gain control over environmental factors. There are restrictions in testing various operating conditions, and there are limitations in how to make measurements. The captured data is noisy and contains errors. The sensors sometimes run under harsh conditions and occasionally break down during operation. It must be presumed that drift is present within the data. Poor precision of the reference instrumentation and time-lags between sensor data and reference data are factors that must be considered. In summary, much time must be spent on organizing data, to learn its characteristics, and to counteract for defects and malfunctioning equipment. Even if counteractions are made, there is a high probability that data will be colored by deficiencies in the application. Is it then worth to work with data captured under live conditions? It depends! Small ideal datasets, perhaps artificially generated, are good for learning properties of new algorithms etc. When it comes to test whether chemical sensors are useable in an application, tests on real data probably give results that are more reliable. Personally, the respondent has found it more interesting to work on real data, due to its closer relation to useful and foreseeable applications.

As time passed, and as a consequence of experiencing the difficulties described above, the respondent started to grew an interest in how to handle practical limitations. It was realized that learning the mathematics of a method is one thing, and learning to understand under which circumstances it can be used and how to take proper preparatory actions is another. In later work, the reader can perhaps discern a better awareness of the respondent to issues related to putting data analysis procedures into practice.

In conclusion, hope is that the work leading to this thesis contributes with useful suggestions on how to apply data analysis procedures to applications (paper I), how to make the integration of chemical sensors into applications efficient (paper II and paper III), and how to improve the performance of an algorithm by first counteracting for artifacts raised by the application (paper IV).

References

[1] D.A. Skoog and J.J. Leary. *Principles of instrumental analysis.* Saunders Collage Publishing, 4 edition, 1992.

[2] M. Holmberg and T. Artursson. *Handbook of machine olfaction*, chapter Drift compensation, standards, and calibration methods, pages 325–346. Wiley-VCH, 2003.

[3] T.C. Pearce, J.W. Gardner, S.S. Schiffman, and H.T. Nagle. *Handbook of machine olfaction.* Wiley-VCH, 2003.

[4] A. Hierlemann, M. Schweizer-Berberich, U. Weimar, G. Kraus, A. Pfau, and W. Göpel. *Sensors Update*, volume 2, chapter Pattern recognition and multicomponent analysis, pages 121–176. VCH Publishers INC., 1996.

[5] H. Nanto and R. Stetter. *Handbook of machine olfaction: electronic nose technology*, chapter Introduction to chemosensors, pages 79–104. Wiley–VCH, 2003.

[6] W. Göpel, J. Hesse, and J. Zemel. Sensors - a comprehensive survey, 1991.

[7] J.W. Gardner and P.N. Bartlett. *Electronic noses: principles and applications.* Oxford University Press, first edition, 1999.

[8] I. Lundström. Hydrogen sensitive MOS-structures, part I: principles and applications. *Sensors and Actuators*, 1:403–426, 1981.

[9] AppliedSensor GmbH, Germany. www.appliedsensor.com.

[10] Sensistor Technologies AB, Sweden. www.sensistor.com.

[11] A. Lloyd-Spetz, S. Nakagomi, and S. Savage. *Advances in silicon carbide: processing and applications*, chapter High-temperature SiC-FET chemical gas sensors. Artech House INC., 2004.

[12] M. Andersson, H. Wingbrandt, H. Petersson, L. Unéus, Henrik Svenningstorp, M. Löfdahl, M. Holmberg, and A. Lloyd-Spetz. *Encyclopedia of Sensors.* American Scientific Publishers, 2006.

[13] O. Engström and A. Carlsson. Scanned light pulse technique for the investigation of insulator-semiconductor interfaces. *Journal of Applied Physics*, 54:5245–5251, 1983.

[14] I. Lundström, U. Frykman, E. Hedborg, A. Spetz, H. Sundgren, S. Welin, and F. Winquist. Artificial 'olfactory' images from chemical sensor using a light pulse teqhnique. *Nature*, 352:47–50, 1991.

[15] J. Wang. *Analytical electrochemistry.* Wiley-VCH, 1994.

[16] C.H. Hamann, A. Hamnet, and W. Vielstich. *Electrochemistry.* Wiley-VCH Verlag GmbH, 1998.

[17] L. Unéus, T. Artursson, M. Mattson, P. Ljung, R. Wigren, P. Mårtensson, M. Holmberg, I. Lundström, and A. Lloyd-Spetz. Evaluation of on-line flue gas measurements by MISiCFET and metal-oxide sensors in boilers. *IEEE Sensors Journal*, 5(1):75–81, 2005.

[18] M. Andersson, P. Ljung, M. Mattson, M. Löfdahl, and A. Lloyd-Spetz. Investigations on the possibilities of a MISiCFET sensor system for OBD and combustion control utilizing different catalytic gate materials. *Topics in Catalysis*, 30–31:365–368, 2004.

[19] F. Winquist, C. Krantz-Rülcker, and I. Lundström. *Electronic tongues and combinations of artificial senses*, volume 11(1), pages 279–306. VCH Publishers INC., 2002.

[20] F. Winquist, P. Wide, and I. Lundström. An electronic tongue based on voltammetry. *Analytica Chimica Acta*, 357:21–31, 1997.

[21] M. Pardo and G. Sberveglieri. Learning from data: a tutorial with emphasis on modern pattern recognition methods. *IEEE Sensors Jorunal*, 2(3):203–217, 2002.

[22] T.M. Cover. The best two independent measurements are not the two best. *IEEE Transactions on Systems, Man, and Cybernetics*, 4:116–117, 1974.

[23] A.K. Jain, R.P. Duin, and M. Jianchang. Statistical pattern recognition : a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, 2000.

[24] R.A. Johnson and D.W. Wichern. *Applied multivariate statistical analysis*. Prentice-Hall, 5 edition, 2002.

[25] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2:37–52, 1987.

[26] R.O. Duda, P.E. Hart, and D.G. Strok. *Pattern classification*. John Wiley & Sons Inc., 2nd edition, 2001.

[27] S. Haykin. *Neural networks: a comprehensive foundation*. Prentice-Hall Inc., 2nd edition, 1999.

[28] S. Rännar, J.F. MacGregor, and S. Wold. Adaptive batch monitoring using hierarchical PCA. *Chemometrics and Intelligent Laboratory Systems*, 41:73–81, 1998.

[29] S. Wold, P. Geladi, K. Esbensen, and J. Ohman. Multi-way principal components and pls-analysis. *Journal of Chemometrics*, 1:41–56, 1987.

[30] S. Kung, K. Diamantaras, and J. Taur. Adaptive principal component extraction (APEX) and applications. *IEEE Transactions on Signal Processing*, 42(5):1202–1217, 1994.

[31] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

[32] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13:411–430, 2000.

[33] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley & Sons, 2001.

[34] P. Comon. Independent component analysis, a new concept? *Signal Processing*, 36:287–314, 1994.

[35] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[36] B. Schölkopf, A. Smola, and K.R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.

[37] V.N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag Inc, 2^{nd} edition, 1999.

[38] R.K.H. Galvão, M.C.U. Araujo, G.E. José, M.J.C. Pontes, E.C. Silva, and T.C.B. Saldanha. A method for calibration and validation subset partioning. *Talanta*, 67:736–740, 2005.

[39] R. Wehrens, H. Putter, and L.M. Buydens. The bootstrap: a tutorial. *Chemometrics and Intelligent Laboratory Systems*, 54:35–52, 2000.

[40] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):111–147, 1974.

[41] S. Brahim-Belhouari and A. Bermak. Gas identification using density models. *Pattern Recognition Letters*, 26:699–706, 2005.

[42] M. Kuske, R. Rubio, A.C. Romain, J. Nicolas, and S. Marco. Fuzzy k-nn applied to moulds detection. *Sensors and Actuators B*, 106:52–60, 2005.

[43] C.M. Bishop. *Neural networks for pattern recognition*. Oxford University Press Inc., 1 edition, 1995.

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

[45] R.D. Maesschalck, D. Jouan-Rimbaud, and D. Massart. Tutorial:the mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, 50:1–18, 2000.

[46] M. Wang, A. Perera, and R. Gutierrez-Osuna. Principal discriminants analysis for small-sample-size problems: applications to chemical sensing. *Proceedings of IEEE Sensors*, 2:591–594, 24-27 October 2004.

[47] B.E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Computational Learing Theory*, pages 144–152, 1992.

[48] J. Nocedal and S.J. Wright. *Numerical optimization*. Springer-Verlag, 1999.

[49] M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.

[50] P. Geladi and B.R. Kowalski. Partial least squares regression: a tutorial. *Analytica Chimica Acta*, 185:1–17, 1986.

[51] I.E. Frank and J.H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993.

[52] S. Wold, J. Trygg, A. Berglund, and H. Antti. Some recent developments in pls modelling. *Chemometrics and Intelligent Laboratory Systems*, 58:131–150, 2001.

[53] A. Höskuldsson. PLS regression methods. *Journal of Chemometrics*, 2:211–228, 1988.

[54] S. Wold, M. Sjöström, and L. Eriksson. Pls-regression: a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 58:109–130, 2001.

[55] R.W. Williams and K. Herrup. The control of neouron number. *The Annual review of Neuroscience*, 11:423–453, 1988.

[56] G.J. Tortora and S.R. Grabowski. *Principles of anatomy and physiology*. John Wiley and Sons, Inc., New York, USA, 2000.

[57] C. Di Natale, F. Davide, and A. D'Amico. Pattern recognition in gas sensing: well-stated teqhniques and advances. *Sensors and Actuators B*, 23:111–118, 1995.

[58] H.H. Yang, N. Murata, and S.I. Amari. Statistical inference: learning in artificial neural networks. *Trends in Cognitive Sciences*, 2(1):4–10, 1998.

[59] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, 1999.

[60] E. Bingham and A. Hyvärinen. A fast fixed-point algorithm for independent component analysis of complex valued signals. *International Journal of Neural Systems*, 1:1–8, 2000.

[61] M. Fryder, M. Holmberg, F. Winquist, and I. Lundström. A calibration technique for an electronic nose. In *Eurosensors IX*, 1995.

[62] J.-E. Haugen, O. Tomic, and K. Kvaal. A calibration method for handling the temporal drift of solid state gas-sensors. *Analytica Chimica Acta*, 407:23–39, 2000.

[63] O. Tomic, T. Eklöv, K. Kvaal, and J.E. Haugen. Recalibration of a gas-sensor array system related to sensor replacement. *Analytica Chimica Acta*, 512:199–206, 2004.

[64] T. Artursson, T. Eklöv, I. Lundström, P. Mårtensson, M. Sjöström, and M. Holmberg. Drift correction for gas sensors using multivariate methods. *Journal of Chemometrics*, 14:711–723, 2000.

[65] R. Goodacre and D.B. Kell. Correction of mass spectral drift using artificial neural networks. *Analytical Chemistry*, 68:271–280, 1996.

[66] F. Davide, C. Di Natale, and A. D'Amico. Self-organizing multisensor system for odour classification: internal categorization, adaption and drift rejection. *Sensors and Actuators B*, 18-19:244–258, 1994.

[67] C. Di Natale, F. Davide, and A. D'Amico. A self-organizing system for pattern calsssification: time varying statistics and sensor drift effects. *Sensors and Actuators B*, 26-27:237–241, 1995.

[68] M. Zuppa, C. Distante, P. Siciliano, and K. Persaud. Drift counteraction with multiple self-organmising maps for electronic noses. *Sensors and Actuators B*, 98:305–317, 2004.

[69] C. Ulrich, H. Petersson, H. Sundgren, F. Björefors, and C. Krantz-Rülcker. Simultaneous estimation of soot and diesel contamination in engine oil using electrochemical impedance spectroscopy. *Sensors and Actuators B*, 127:613–618, 2007.

[70] T. Artursson, P. Spångeus, and M. Holmberg. Variable reduction on electronic tongue data. *Analytica Chimica Acta*, 452:255–264, 2002.

[71] J. Trygg and S. Wold. Orthogonal projections to latent structures (O-PLS). *Journal of Chemometrics and Intelligent Laboratory Systems*, 16:119–128, 2002.

[72] P.Y. Simard, Y.A. Le Cun, J.S. Denker, and B. Victorri. Transformation invariance in pattern recognition: tangent distance and tangent propagation. *International Journal of Imaging Systems and Technology*, 11(3):181–197, 2001.

Index