Learning methods for machine vibration analysis and health monitoring

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.F. Wakker,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 12 november 2001 om 16:00 uur

door Alexander YPMA

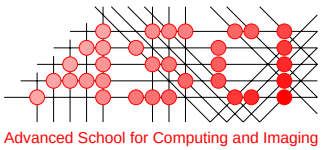
doctorandus in de informatica,
geboren te Leeuwarden.

Dit proefschrift is goedgekeurd door de promotor:
Prof. dr. ir. L. J. van Vliet

Toegevoegd promotor: Dr. ir. R. P. W. Duin

Samenstelling promotiecommissie:

Rector Magnificus, voorzitter
Prof. dr. ir. L. J. van Vliet, Technische Universiteit Delft, promotor
Dr. ir. R. P. W. Duin, Technische Universiteit Delft, toegevoegd promotor
Prof. dr. E. Oja, Helsinki University of Technology
Prof. dr. ir. P. M. J. van den Hof, Technische Universiteit Delft
Prof. ir. K. Smit, Technische Universiteit Delft
Prof. dr. J. N. Kok, Universiteit Leiden
Prof. dr. ir. E. E. E. Frietman, California State University



This work was partly supported by the Dutch Technology Foundation (STW). This work was carried out in the ASCI graduate school, ASCI dissertation series no. 71.

Ypma, Alexander
ISBN: 90-9015310-1

Contents

Summary

**Learning methods for machine vibration analysis and health monitoring -
Alexander Ypma, Delft University of Technology, 2001**

In this thesis we propose a framework for health monitoring with learning methods, that comprises blind demixing, temporal feature extraction, domain approximation, novelty detection and dynamic pattern recognition. We argue that in rotating machine monitoring applications, contributions of several sources are often mixed linearly into an array of acoustic or vibration sensors. Combined with reasonable assumptions on the sources (independence or different spectrum), this allows for blind separation of distorted machine signatures.

We found feature extraction methods based on temporal correlations in a vibration signal suitable for extraction of information about machine health. Several methods for approximation of the normal domain are identified and investigated in three monitoring case studies. Results indicate that proper novelty detection is possible using these methods. Hidden Markov models are then studied for the automatic segmentation into health regimes of time series indicative of a degrading gearbox. We noted several drawbacks of this approach, which calls for novel methods to be devised for this task.

Finally, we demonstrate the feasibility of the learning approach to health monitoring for monitoring of a submersible pump and a gearbox in a pumping station machine, for pipeline leak detection and for detection of eyeblinks and anomalies in EOG and EEG recordings of patients suffering from Tourette's syndrom and Alzheimer's disease, respectively. This has led to the development of a practical monitoring tool based on Self-Organizing Maps, MONISOM, in a collaboration with two industrial companies.

Samenvatting

**Lerende methoden voor machinetrillingsanalyse en conditiebewaking -
Alexander Ypma, TU Delft, 2001**

In dit proefschrift wordt een kader voor conditiebewaking met lerende methoden voorgesteld, dat bestaat uit de onderdelen: blind scheiden van trillingsbronnen, extractie van temporele kenmerken, leren van het domein van een dataset, herkennen van anomalieën en dynamische patroonherkenning. We stellen dat in bewakingstoepassingen met roterende machines vaak een lineair samenstel van bronnen wordt gemeten op een verzameling accelerometers of microfoons. Met redelijke aannames over de bronnen (onafhankelijkheid of het spectraal verschillend zijn) kunnen we dan de wederzijds verstoorde (spectrale) trillingskarakteristieken van machines blind van elkaar scheiden.

Correlatie-gebaseerde kenmerk extractiemethoden blijken geschikt om uit trillingssignalen informatie over de machineconditie te destilleren. Een aantal methoden is aangewezen voor het leren van het domein van een dataset (dat het normaalgedrag van de machine representeert) en onderzocht in drie conditiebewakingsapplicaties. De resultaten suggereren dat een adequate detectie van anomalieën mogelijk is met deze methoden. Daarna worden hidden Markov modellen onderzocht voor het automatisch segmenteren in conditietoestanden van tijdreeksen van een langzaam verslechterende tandwielkast. Experimenten suggereren dat aan deze aanpak nadelen kleven, waardoor een andere benadering noodzakelijk lijkt.

Tenslotte laten we de haalbaarheid zien van de lerende aanpak voor conditiebewaking in het bewaken van een dompelpomp en een tandwielkast in een gemaalpomp, in het detecteren van onderwater gaslekken en in het detecteren van oogbewegingen en anomalieën in het EOG en het EEG van patienten met respectievelijk het syndroom van Gilles de la Tourette en de ziekte van Alzheimer. Dit heeft uiteindelijk geleid tot de ontwikkeling van programmatuur voor praktische conditiebewaking, MONISOM, in samenwerking met twee industriële partners.

List of abbreviations

ANN artificial neural network

AQE averaged quantization error

AR-model autoregressive model

(AS)SOM (adaptive subspace) self-organizing map

BSS blind source separation

cICA contextual ICA

DCA dynamic component analysis

DCT discrete cosine transform

DOA direction of arrival

EEG electro-encephalography

EOG electro-oculography

EPP exploratory projection pursuit

FA factor analysis

FEM finite element model

FFT fast Fourier transform

GOF goodness of fit

HMM hidden Markov model

HOS higher-order statistics

ICA, IFA independent component analysis, independent factor analysis

JADE joint approximate diagonalization of eigenmatrices algorithm for ICA

KL-divergence Kullback-Leibler divergence

KLT Karhunen-Loève transform

k-NN k-nearest neighbour (classifier or noise injection)

LDC linear discriminant classifier

LGM linear Gaussian model

MAP maximum a posteriori

MDL minimum description length

MDOF multiple degrees of freedom

ML maximum likelihood

MSE mean squared error

MST minimal spanning tree

NMC nearest mean classifier

opmode operating mode

PCA principal component analysis

QDC quadratic discriminant analysis classifier

RHS right-hand side

RMS root-mean-square

ROC receiver operator curve

SCORE self-coherence restoral algorithm for ICA

SDOF single degree of freedom

SOBI second-order blind identification algorithm for ICA

SOS second-order statistics

SPM shock pulse method

SRM structural risk minimization

STFT short-term Fourier transform

SV, SVM, SVDD support vector (machine, data description)

TF-analysis, TFD time-frequency analysis, time-frequency distribution

U-matrix unified distance matrix

WTA winner-takes-all

WV-distribution Wigner-Ville distribution

Part I

The learning approach to machine health monitoring

Chapter 1

Machine health monitoring

This thesis is about the use of learning methods for machine vibration analysis and health monitoring. Health monitoring (a.k.a. condition monitoring) is already much practiced in many of today's engine rooms and plants, either by skilled engineers or diagnostic expert systems. However, techniques that rely on automatic pattern recognition have only recently been introduced into this field. Pattern recognition is a research area with a long-standing history, traditionally focused on finding optimal decision functions for static well-sampled classes of data. Besides issues encountered in any pattern recognition problem (feature extraction, small sample sizes, generalization), we face some special issues in health monitoring of rotating equipment. This requires the use of (relatively novel) methods for blind source separation, novelty detection and dynamic pattern recognition. We propose a *learning approach* to machine health monitoring that addresses these issues and investigate the usefulness of our approach in several real-world monitoring applications. First, we illustrate the problems connected to machine health monitoring with an illustrative every-day example.

1.1 Machine health monitoring: an example

Consider a rotating machine that is operating, for example a household mixer or a car engine. These machines produce a kind of noise that seems to be related to their rotating speed, e.g. putting the mixer in a faster mode produces noise at a higher frequency. A car engine is much more complex, since many vibration sources inside the engine contribute to the overall vibration and the engine has a larger and more complex mechanical structure. A car driver gets used to the machine sound during normal operation and may even be able to recognize the car when his or her better half is coming home from a day off. Of course, when cars get older the material wears and faults may develop inside the engine. These (incipient) faults can be recognized by the driver when he suddenly hears a strange noise among the familiar car noises. Initially, this may take the form of occasional clashes or ticks. An incipient fault with a low contribution to the spectrum may be masked by high-energetic frequencies due to other machine components like a waterpump or a properly functioning gearbox. When the fault develops, a clearly distinguishable tone at some unfamiliar frequency can emerge. If the driver knows about car mechanics, he may try to remedy the problem himself. One way

to diagnose the problem is to let the engine run and listen to the vibration inside, e.g. by putting a screw driver at the engine casing in order to track the relevant fault vibration source more closely. After having diagnosed the problem, it may turn out that a minor disturbance was causing the strange noise, e.g. some dust entered the engine or a harmless bolt was loosened a bit. Moreover, wear could have caused the structural characteristics to change over time. In this case, the driver would remove the dust, fasten the bolt, or just conclude that he had to cope with this slightly dissimilar sound until the yearly maintenance would be done at his garage. Noises of the type he had heard in this case are now stored in his "experiential database", making it more easy to spot the origin and its severity next time it appears. However, it can also be the case that there was a small problem with the lubrication and there was a small but growing imbalance present. At this stage, there was no reason for panic but it could lead to a potentially very dangerous situation: decay of bearings or gears (see figure 1.1). The driver should bring his car to the garage to look at the cause of the lubrication problem and e.g. rebalance the shaft. This is an expensive job, but the material damage that would have resulted from a car accident at the highway (let alone the human health damage) would be many times higher. □

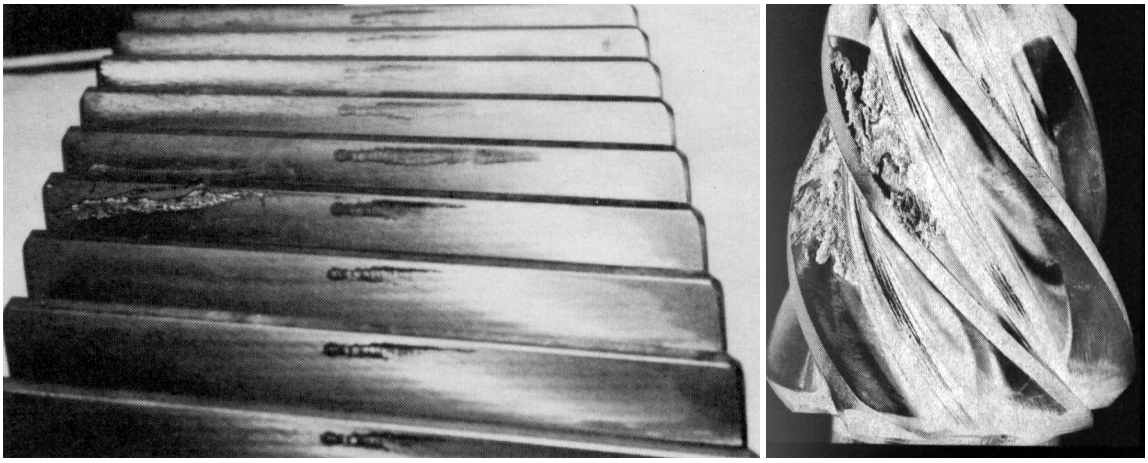


Figure 1.1: Wear and failure in gears, from [Tow91]

The previous example illustrates the purpose of health monitoring in rotating machines: detecting and diagnosing faults in an early stage, which may be feasible since many faults will manifest themselves as pure tones or strange noises in the overall machine vibration. The conventional approach would be to use many heuristics about the structural properties of the machine and look in the vibration spectrum for specific fault-related components with increased amplitude compared to previous measurements. This could be done either manually (the skilled operator) or on the basis of a rule-based system. It becomes immediately clear that in this approach one has to make large fault-databases for each new machine, since every machine vibrates in its own specific manner. Moreover, disturbances from nearby machines or irrelevant vibration sources along with ambiguities in the vibration spectrum may lead to situations that are not in the database, or wrongly assumed to be in the database. A frequency spectrum is often difficult to interpret because of the large amount of overlapping frequency

components, structural resonances, modulating phenomena, noise influences, etc. present in the vibration spectrum.

Research issues

In this thesis, we propose a *learning approach to health monitoring*. In this way, we might be able to tune a health description to the peculiarities of a certain machine. Which are the problems to tackle if one applies learning methods for the problem of machine health monitoring? We address this question from both a machine monitoring *(chapter 1)* and a machine learning *(chapter 2)* point of view. This gives rise to the following research issues:

We need to have a **description of a machine behaving normally** that can be used for early detection of anomalies. This calls for a **proper characterization** of machine health. We identify methods to extract health information from vibration measurements and investigate strengths and weaknesses of these methods as health descriptors *(chapter 3)*.

Having the possibility to **reconstruct a machine signature** from a distorted measurement is very valuable, since increased vibration may be caused by (irrelevant) interfering machine or environmental sources. We take a learning approach for this task as well, by learning the contributions of interfering sources to a multichannel machine measurement blindly. Independent Component Analysis is a well-known method for blind source separation *(chapter 4)*. We investigate which variants of ICA are suitable and to what extent blind source separation can be used in practice *(chapter 5)*.

An important characteristic of the learning approach to health monitoring is the **lack of large sample sizes**; moreover, **fault classes are often largely absent**. This renders traditional pattern recognition techniques less useful and calls for a novelty detection approach. We describe several methods for novelty detection and use them to assess the feasibility of novelty detection for practical health monitoring *(chapter 6)*.

Furthermore, **machine characteristics change as time goes by**: new knowledge about faults and machine wear may be gathered on-the-fly and one may want to monitor the degradation trend. We investigate the use of hidden Markov models to segment time series from a gradually deteriorating gearbox into health regimes *(chapter 7)*.

Finally, we investigate the usefulness of the learning approach in a number of **practical monitoring applications**. Issues like robustness to repeated measurements, fault detection with only one sensor, generalization of the previously mentioned framework to other industrial (gas leak detection; monitoring of large real-world machinery in a pumping station) and medical health monitoring problems are investigated. Finally, a practical tool for *learning health monitoring* based on Self-Organizing Maps, *MONISOM*, is presented *(chapter 8)*.

1.2 Vibroacoustic processes in machines

Nearly every machine will emit a certain amount of noise and vibration: machines are often not perfectly balanced, contact between moving elements in the machine may not be ideal, manufacturing imperfections (like e.g. surface roughness) are often present or it may appear as an inherent consequence of their operation [Epp91]. The *purpose* of a rotating machine is

the transformation of input energy N_{in} into useful new (and different) energy $N_{upgrade}$ (like motion, in a rotating machine) and dissipated energy. In the transformation process, some residual processes (unnecessary motions accompanying the target motion) are also present, like noise, vibration, and acoustical emission [Cem91]. Dissipated energy is subdivided into internally dissipated energy (which is not measurable at the outside of the machine) and externally dissipated energy (like vibration of the machine casing or acoustic emission). The latter quantity can be measured with sensors, and the resulting signals may be analyzed in order to extract a health index. The process of measurement and preprocessing is modelled as a read-out function $\phi(.)$, figure 1.2.

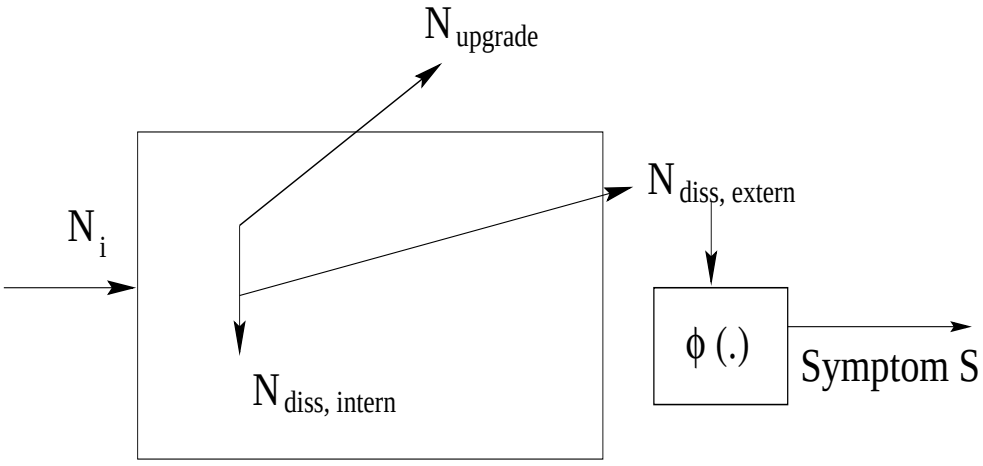


Figure 1.2: Machines are energy transforming systems, from [NC97]

1.2.1 Transmission of vibration sources in machines

Vibration sources in machines are events that generate forces and motion during machine operation. These include [Lyo87]: shaft imbalance, impacts (e.g. due to bearing faults), fluctuating forces during gear mesh (as a result of small variations during machine operation or in local contact stiffness), electromagnetic forces and pressure- and flow related sources. The most common classification of vibration sources in terms of their signal content is a. sinusoidal (e.g. imbalance), b. pulse-like (e.g. impacts due to bearing failure and gear pitting) and c. random (e.g. broadband excitation without temporal shape, as with flow-induced excitation like cavitation).

Transmission of vibration

Vibration in machines takes the form of compressional (longitudinal) and bending waves mainly [Lyo87]. With compressional waves, all frequencies travel at the same speed, whereas with bending waves this is not the case. The latter effect is called *dispersion*; dispersion changes the waveform, since different frequencies travel at different speeds. Moreover, a machine will usually exhibit *reverberation* effects: multiple reflections and propagation along

different paths in the structure. This phenomenon can have the effect that vibration measurements at different positions on the machine are very similar: reverberation in the engine structure can make the vibration almost uniformly spread over the structure. Dispersion and reverberation can severely distort waveforms indicative of faults; design of inverse filters is thereby a highly nontrivial task. The variability in transfer functions that were identified on a class of machines (i.e. a set of different mechanical structures with the same machine blueprint) can be quite large: standard deviations around 8 dB at every frequency were found in [Lyo87]. Hence, identification of inverse filters on one machine will not suffice for other (comparable) machines. In [Lyo87] it is therefore recommended that adaptive systems, which are able to learn the system transfer functions from arrays of accelerometers, are developed.

Coherence loss

The amount of coherence between two vibration measurements at different positions on a machine will vary from completely coherent (if the vibration is dominated by rigid body motion or strong reverberation is present), to completely incoherent (e.g. if one analyzes high-frequency modes with wavelengths that are very small with respect to the intersensor distance). According to [NC97] the transition from complete to no coherence is made in the modes with center frequency f_i that have ratios f_i/f_0 ranging from approximately 0.1 to 10 (see figure 1.3). Here, f_0 denotes the first machine eigenfrequency. The intersensor spacing L is shown between two sensors s_1 and s_2 and the vibration on the casing is due to some force $f(r,t)$ at some time of measurement t applied at some machine position r. The 'regime'

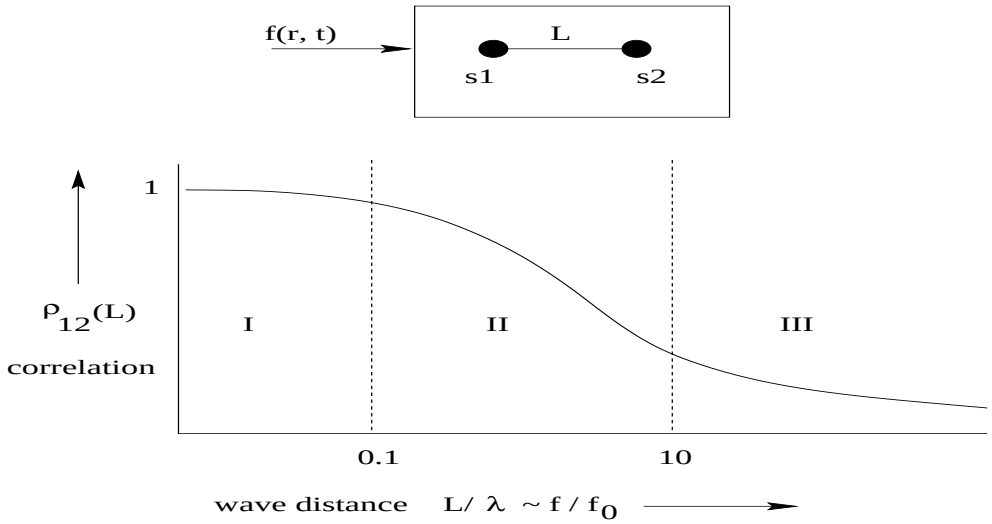


Figure 1.3: Coherence loss in machines, from [NC97]. The coherence of measurements at two different sensors depends on the intersensor distance relative to the analyzed wavelength

labeled with I is the situation where the intersensor spacing L (see figure 1.3, upper part) is approximately 0.1 times the wavelength λ of the frequency under investigation. Here, the prevailing motion type is rigid (or 'whole') body motion, in which case both sensors s_1, s_2 measure virtually the same vibration (coherence near to one). Standing waves are dominant.

For large distances (with respect to the wavelengths under investigation), the coherence drops to zero (see figure 1.3, bottom part). This can be explained by noting that the prevailing wave types are travelling waves, which eventually cancel out each other. Additional forces like impacts due to structural imperfections and also increasing wave distance due to wear development (like cracks in the load path) enhance this effect. From this discussion it is concluded that the intersensor spacing should be chosen approximately as $\alpha\lambda, \quad \alpha = [1,6]$, where the particular choice of α depends on the length and type of the wave under investigation and on the structural properties of the mechanical system. This is the situation where 'spatial diversity and redundancy' exists between a set of vibration sensors on a mechanical system.

Linear model for vibration transmission

In damaged rotating machines, a characteristic signal that is indicative of wear or failure will be repeated during each machine revolution and will be filtered at the machine casing with the local impulse response $h(r,t,\theta)$. The transmission path between a sensor and a vibration source is hence modelled as a linear filter [Cem91, Lyo87]. The modal characteristics of the mechanical structure cause this filtering: reverberation and dispersion of vibrational (bending) waves induce delayed (and scaled) copies of a source and modify the phase of a frequency component, respectively. Moreover, the mode shape (section 3.1) of the structure in relation to the sensor position and measurement direction determines the extent to which a source is transmitted. This determines which resonances will be present and to what extent the effect of an internal source will be present in the response. For the general situation with multiple sensors and multiple vibration sources (both internal to the machine and external interferers), we model the response at position j as

$$x_j(t,\theta) = \sum_{i=1}^{NF} h_{ij}^F(t,\theta) \star F_i(t,\theta) + \sum_{i=1}^{NI} h_{ij}^I(t,\theta) \star I_i(t,\theta) + h_j^S(t,\theta) \star m(t,\theta) + n_j(t,\theta) \quad (1.1)$$

Here, $\star$ denotes the convolution operation. The t and θ parameters denote the two time scales that are relevant in health monitoring: the short timescale t is the scale of seconds and minutes, where the characteristic failure signals and other features for machine health are estimated from a transducer measurement signal and the long timescale θ is the scale where machine health is tracked during its lifetime[1].

Convolutive mixture model

The above expression is close to the expression given in [Cem91]. It is a **convolutive mixture** of NF fault-related vibration sources $F_i(t,\theta)$, vibration from NI interfering machinery components $I_i(t,\theta)$ and filtered modal machine response $m(t,\theta)$ due to all remaining excitation sources arising with normal machine operation and structural imperfections. We introduced a "structural filter" $h_j^S(t,\theta)$ that accounts for the attenuation of a mode due to

[1] On the short timescale, measurement time series are stationary random processes with zero mean and finite variance, on the long timescale, the health time series is a random growth process

sensor measurement direction and position (e.g. putting a sensor on a position of a nodal line). The term $h_j^S(t,\theta) \star m(t,\theta)$ might be rephrased as $m'_j(t,\theta)$, where m'_j now represents the modal response of the structure to the forced excitation due to machine operation, taking into account the mode shapes. By making the attenuation of certain modes explicit, we want to emphasize that the position of a sensor can highly influence the actual measured response. We also included a set of interfering sources into the model. It is not obvious that these interferers (possibly from other machines or remote machine components) appear as an additive contribution. In chapter 5 we investigate the suitability of this model in a setup with two coupled machines. Depending on the purpose of the monitoring setup, a source can be designated as a fault source or an interference source. Finally, the ambient and sensor noise is given by the $n_j(t,\theta)$ term.

1.2.2 Bearing and gearbox vibration

Two central components of rotating machines are bearings and gears. The former components support the rotating parts of the machine, whereas the latter are causing accelerations and decelerations of driving shaft speed. In [Oeh96] vibration analysis of a gearbox with rolling element bearings is discussed. Assuming linear transfer at the casing, the response of a gearbox with a bearing failure to internal excitations (like structural imperfections and circulation of lubrication fluid) and external excitations (like variation in motor behaviour) is modeled as

$$y(t) = y_{mesh}(t) + y_{reb}(t) + y_n(t) \tag{1.2}$$

where y_{mesh} is the component due to gear meshing, y_{reb} is the component due to the rolling element bearing failure. The non-harmonic additive noise part of the vibration signal y_n is due to random excitation. The contribution of bearings and gears to overall machine vibration is discussed below.

Bearing vibration

According to [Epp91] all causes of failures in rolling element bearings (REBs) can be prevented except fatigue. Personal communication with a bearing manufacturer let us believe that the vast majority of bearing failures is due to inappropriate use (e.g. imbalance, improper lubrication) or manufacturing errors, hardly ever to wear. Bearings are critical elements in rotating machines, since they support the rotating structures and much of the energy during operation is transferred at the bearings. Bearing failures may easily lead to other, more expensive damage in a machine. Hence, most machines are periodically serviced, and replacement of the bearings is a routine matter to prevent major damage. However, this (often superfluous) servicing operation may induce faults because of the opening and closing of the machine. If *distributed* defects develop in bearings, sinusoidal vibrations will be produced inside the structure. Distributed defects are faults like misalignment, eccentricity and geometrical imperfections, where the magnitude of the ball-race contact force varies continuously and periodically as the bearing rotates. Examples of geometrical imperfections are:

Table 1.1: Bearing fault frequencies

fault location	fault frequency f
outer race	$N/2\ (1-D_{frac}\cos\phi)\ f_r$
inner race	$N/2\ (1+D_{frac}\cos\phi)\ f_r$
ball	$D_p/D_b\ (1-[D_{frac}\cos\phi]^2)\ f_r$
bearing cage	$1/2\ (1-D_{frac}\cos\phi)\ f_r$

race or element waviness and off-sized rolling elements [Epp91]. Incipient *discrete* defects in bearings can be detected with vibration monitoring: defects in REBs will appear as irregularities or modifications in the rolling surfaces of the bearing, which will lead to (semi-)periodic impacts that will excite the bearing and machine resonances. Deviations from exact periodicity were observed [Epp91] and will lead to continuous phase shifts, i.e. phase randomization. Approximations to the average defect frequencies are shown in table 1.1. In this table, we use the following parameters: f_r = shaft rotation frequency (Hz); D_b = ball diameter (mm); D_p = pitch circle diameter (mm); D_{frac} = fraction of ball to pitch diameter $\frac{D_b}{D_p}$; N = number of balls in the bearing; ϕ = contact angle. The geometry of a bearing is displayed in figure 1.4. In the left subfigure a frontal view is shown. In the side view (right subfigure), several bearing components can be identified: from top to bottom, the outer race, rolling element (ball, in this case), inner race and cage are shown. An important distinction between inner

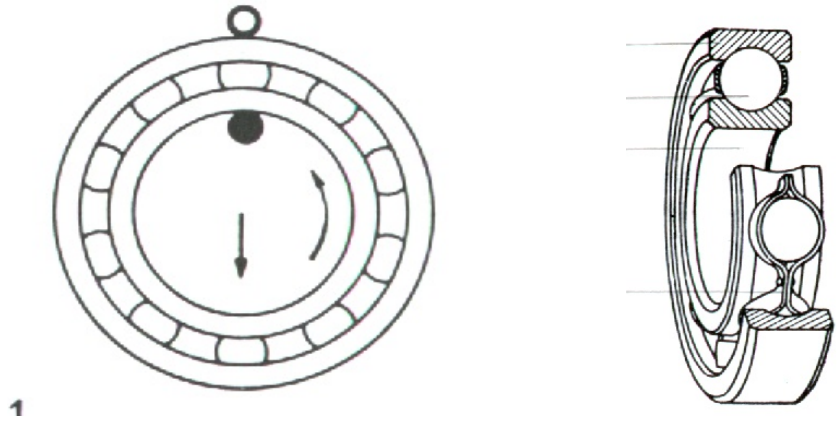


Figure 1.4: Schematic view of a bearing

race- and outer race faults is that significant modulation effects will be present in the former when the rolling element is passing the load zone, whereas these effects will be much smaller in the latter. A detailed analysis of this phenomenon can be found in [Epp91].

The dynamic behaviour of the vibrating machine structure may severely modify the impulsive vibration sources: the vibrations that are measured with accelerometers on the ma-

chine casing are measurements of the *response* of the structure to the underlying excitation *sources*. The transmission characteristic (impulse response between source and transducer) will determine the frequency content of a measured impulse due to a bearing fault. If the (local) damping is high enough, the measured impulses will not overlap in time, i.e. the transients decay before the next transient occurs. Note that each impulse generates broadband energy (since an ideal impulse has a flat spectrum), hence excites the bearing and the surrounding structure to vibrate at their natural frequencies. The observed response is hence a 'train' of transient oscillations, decaying exponentially according to the structural damping, and repeating at a rate that will roughly resemble the frequencies in table 1.1. Modal analysis techniques aim at identification of the transmission characteristic of the structure, by determining the relation between an applied input (excitation, usually a broadband random signal, a frequency sweep, or a single impact signal) and measured output (at the casing), see section 3.1. Transmission effects may also cause additional modulations of bearing impacts [Epp91].

Modelling bearing signals

In case of healthy bearings, the bearing component in the vibration response formula (1.2) will be virtually zero. In case of a failure, the component corresponds to the response of the structure to an amplitude modulated impulse train. The amplitude modulation is present since the forces exerted in the bearing during the rolling process are not isotropic (because of varying loads). Considering a bearing with an outer race that is fixed to the structure, the modulation of the periodic impulse train is modeled as [MS84, Oeh96]

$$y_{reb}(t) = \sin(2\pi f_{bpfi} t)(1 + \beta \sin(2\pi f_r t)) \tag{1.3}$$

where f_{bpfi} is the ball passing frequency at the inner race and f_r is the rotation speed of the shaft at the bearing (i.e. the rotating frequency of the inner race). Moreover, sliding and blocking of the balls and variation in the contact angle causes an additional frequency modulation in the measured signal. It should be noted that a bearing contains many nonrigid structures that introduce nonlinearities in the process. Strong prechargement of the bearing diminishes these nonlinear effects [Oeh96]. Analysis of bearing failures is complex because of the influence of the loading zone. Moreover, the periodicity of the excitation is unknown at the time of analysis, which is the result of not being able to compute the contact angle of the balls in the race beforehand.

Gearbox vibration

During the meshing process, energy is transferred from the driving gear onto the driven gear to establish an acceleration or a deceleration of the driven shaft according to the relation $f_2 = f_1 \cdot \frac{N_1}{N_2}$, where N_1 and N_2 are the number of teeth in the driving respectively the driven gear, and f_1 and f_2 are the respective shaft rotation frequencies in both gears. In this process, two teeth of both gears make contact in several stages. At making contact and taking off two impacts occur, which will always be measured in the vibration response of a gearbox. This

gear-meshing frequency can be computed as $f_{gearmesh} = N \cdot f_r$, where N is the number of teeth in the gear and f_r is the rotating frequency of the shaft connected to the gear. If a fault develops inside the gear (e.g. on the contact surface of a tooth), the generated impacts will be more severe: more energy will be dissipated and radiated in the meshing process. Because of asymmetric loading effects, severe modulations of the gearmesh-frequency (usually with the shaft rotation frequency) will show up. Alternatively, the gearmesh-process may excite the machine structure more severely, possibly leading to stronger modulation of machine resonances with the gearmesh-frequency.

Modelling gear signals

The meshing component in both healthy and faulty gears can be expressed as [BTOR97]

$$y_{mesh}(t) = \sum_{k=1}^{K} A_k(t) \cos(2\pi k f_{mesh} t + \phi_k(t)) \tag{1.4}$$

where

- $f_{mesh} = N_m f_{r,m} = N_{rec} f_{r,rec}$ is meshing frequency, with N_m, N_{rec} number of teeth and $f_{r,m}$, $f_{r,rec}$ rotation speed at both *m*otor (driver) and *rec*eiver (driven) shafts
- $A_k(t) = A_{0,k} + A_{m,k}(t) + A_{rec,k}(t)$ is instantaneous amplitude modulation due to transmitted charge (modulation constant) $A_{0,k}$ and frequencies $f_{r,m}$ and $f_{r,rec}$, respectively
- $\phi_k(t) = \phi_{0,k} + \phi_{m,k}(t) + \phi_{rec,k}(t)$ is instantaneous phase modulation due to transmitted charge (modulation constant) $\phi_{0,k}$ and frequencies $f_{r,m}$ and $f_{r,rec}$, respectively

In *healthy* gears, nonlinearities in the meshing process will cause the presence of higher harmonics of the meshing frequency. Non-constant tooth spacing and nonstationary angular velocity cause phase modulations, whereas amplitude modulations are caused by mechanical load variations due to irregularities on the tooth contact surfaces (e.g. as a result of manufacturing errors) [Oeh96]. Gearbox *faults* can be incorporated into this model as well, since they cause additional phase and amplitude modulation [Oeh96]: damaged teeth lead to quick energetic bursts, which are visible in the spectrum as wideband enhanced modulatory components around (harmonics of) the gearmesh frequency. Spatially distributed faults (like component wear and shaft misalignment) introduce slow nonstationarities, whereas localized faults (cracks) yield abrupt changes. Hence, both normal and wear-out behaviour can be modeled with (1.4), which amounts to learning the signature of the gearbox in admissible conditions; occurence of a crack may then be detected by monitoring the residual of measurements with respect to this model. Difficulties in the interpretation of signals from damaged gearboxes arise because the modulation phenomena are broadband. This may render techniques like envelope detection (section 3.3.2) sometimes less useful [BTOR97, Oeh96].

Table 1.2: Common diagnostical heuristics in rotating equipment. RPM is shorthand for *revolutions per minute*

nature of fault	fault frequency	plane
imbalance	$1 \times$ RPM	radial
misalignment	$1, 2 \times$ RPM, sometimes $3, 4 \times$ RPM	radial, axial
oil-whip, oil-whirl	0.42 to 0.48 $\times$ RPM	mainly radial
mechanical looseness	$2 \times$ RPM and $0.5, 1.5, 2, 5$ etc. $\times$ RPM	radial
bearing faults	bearing fault frequencies, shock pulses between 20 - 60 kHz	radial, axial
gear faults	$f_{mesh} = \#teeth \times f_{gear}$ and harmonics	radial, axial

Other vibration sources

Other typical machine vibrations include shaft misalignment, loose couplings, self-excited bearing vibrations (like oil whirl), cavitation and frictional vibrations. Also, we have to mention the measurement of acoustic emission levels as a useful tool for health monitoring. Acoustic emission is an elastic wave due to a violent release of energy accumulated in the material by the propagating microdamage in the material. However, this measures microstructural fatigue in a structure rather than making use of the rotating nature of a mechanical machine. It is thus very suitable for incipient fault detection with dedicated methods and tools. We will not use acoustic emission as a health indicator in the sequel, since the scale of damage monitoring is too small for generic utilization among a class of machines. In table 1.2 an overview is given of the most commonly used heuristics for fault diagnosis in rotating machinery, based on [FFD96]. More information can be found in [FKFW91]. We can see that different phenomena can be measured in different areas of the vibration spectrum [Cem91]:

very low frequency band (1-100 Hz) vibration of supporting structures (like machine bodies)

low frequency band (10 Hz-1 kHz) vibration of shafts, rotors, slide-bearings

intermediate frequency band (100 Hz-10 kHz) vibration of rolling-element bearings and gears

high frequency band (10 kHz-100 kHz) cavitation (imploding of bubbles, e.g. near the inlet of a pump due to pressure differences)

very high frequency band (10 kHz-100 kHz) acoustic emission in micro-areas

In this thesis, we will restrict ourselves to vibration phenomena in the low and intermediate frequency bands, i.e. vibrational behaviour of typical machine components like bearings,

gearboxes and shafts, that can be measured with accelerometers. This type of behaviour is expected to be common to a broad class of machinery and might enable generalization of knowledge obtained on one machine to other similar machines. Vibration monitoring has been the most popular technique for machine health monitoring, especially for rolling element bearings [Epp91].

1.2.3 Wear and fault development

Several approaches to modelling of machine wear exist, two of which are mentioned here. In the *first* approach, one assumes that fault development shows several stages, and that (severity and timing of) each stage only depends on the previous stage. Hence, the amount of memory in the system is limited to one (previous) health state, like in a first-order Markov process. This can be written down by noting that the probability $p(q(t))$ of being in a certain machine state q at time t only depends on the previous state $q(t-1)$:

$$p(q(t)|q(1,\ldots,t-1)) = p(q(t)|q(t-1)) \tag{1.5}$$

The model can be extended to an nth-order Markov process, where the current state depends on the previous n states. In [LS97] it was noted that modelling a damage function $D(t)$ as a Markov process is a very general, but also quite *challenging* approach since it is very difficult to estimate the conditional probability distribution of the damage increase on the basis of experimental data. In the *second* approach, measurements from a machine are conjectured to give an indirect view of the underlying health state of the system. In [NC97] the symptom-based approach to fault prediction and trending is introduced (cf. figure 1.2). In this approach, an approximate (normalized) **symptom** $\frac{S(\Theta)}{S(0)}$ is introduced (see also figure 1.2) as

$$\phi(\frac{N_{diss,extern}(\Theta)}{N_{diss,extern}(0)}) \sim (\frac{N_{diss,extern}(\Theta)}{N_{diss,extern}(0)})^{-\gamma}$$

The parameters Θ and 0 denote the current position in the machine life cycle and the initial position, respectively (i.e. it may be measured in days or hours). After doing some (reasonable) assumptions on the energy-transforming process, an extracted health symptom S at time t, $S(t)$, is modelled by the **Pareto life curve**:

$$S(\Theta) = S(0)(1 - \frac{\Theta}{\Theta_{down}})^{-\gamma} \tag{1.6}$$

shown in figure 1.5(a) (i.e. left subfigure). This is an analytic approximation to the well-known "bath-tub curve", that is indicative of machine health (figure 1.5(b)). As time proceeds the machine state will change because of wear or replacement of machine components. After the initial stage (where imperfections due to wrong initial conditions, suboptimal settings, etc. will occur), a plateau of relatively constant behaviour will be observed. After this 'smooth

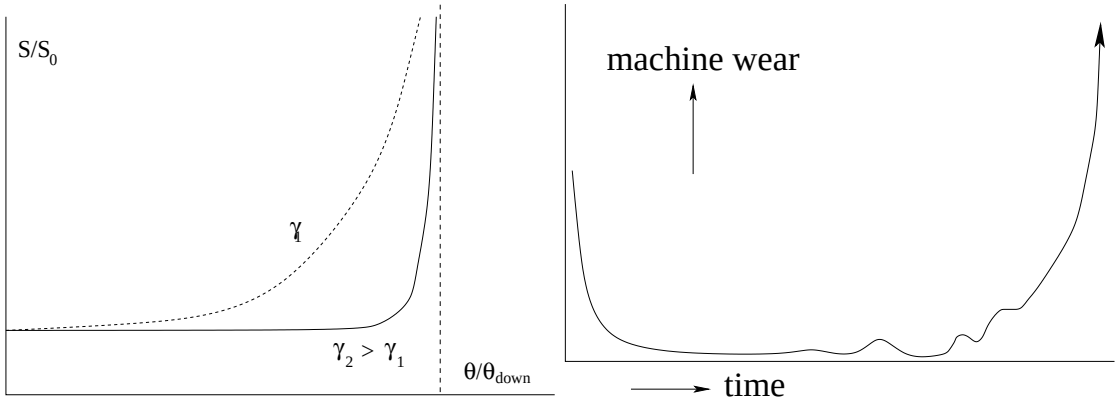


Figure 1.5: Pareto life curve (a) and "bathtub"- life curve (b)

running' period, incipient wear or failures may develop. Sometimes a developing fault may be repaired by the energy-transforming process itself (e.g. smoothing out of the raceways after an initial flaw in a ring of a bearing has developed), which will be called a *self-restoring process*. This is usually only of limited duration, after some time the fault will become more severe, and finally the condition of machine and/ or component can degrade very rapidly. If our machine health index describes the severity of the wear or failure, a relation like drawn in figure 1.5(b) may be observed.

Example 1.1: development of bearing failures

Based on a large number of experimental measurements, Barkov and Barkova [BB96] conclude that the operational life of rolling element bearings can be divided into 4 stages (corroborated by [Mit93, Ber91]). They propose the stages *running-in, trouble-free operation, wear and defect development* and *(rapid) degradation of overall bearing condition*. During the first two stages it is customary to assess the component condition by detection of defects immediately after installation and by monitoring of lubrication and wear levels. In the third stage, wear levels are monitored and remaining lifetime is predicted by dividing the defects into incipient, medium and severe defects. Incipient defects are irreversible changes in condition that do not influence the immediate bearing condition. Medium defects have vibration levels around 10-12 dB above the mean normal vibration levels; severe defects are over 20-25 dB. Measured quantities are vibration levels in the intermediate frequency band (properly enveloped). In the final stage, defect development takes place in three ways: *a.* single severe defects, that show a steady and continuous development of severity; *b.* rapidly appearing and developing faults; the rate of change of a diagnostic parameter associated with the most rapidly evolving fault can be used for estimation of the remaining lifetime; *c.* faults that induce structural changes, accompanied by high levels of shock pulse excitation. □

1.3 Limitations of diagnostic expert systems

Using the fact that a normally vibrating structure has its own "vibration signature" and that faults inside the structure will show up as deviations from this normal pattern and are often visible as distinct frequency components in a vibration spectrum, health monitoring using machine vibration is a feasible approach.

Diagnostical heuristics

The use of the response vibration spectrum for fault detection and diagnosis in rotating machines has a long-practiced history and many heuristics are known. We will mention a few basic indicators of typical rotating machine faults in the sequel [DFM91, Cem91, Mit93, BB96, FFD96, Ber91, Epp91].

Imbalance Consider a machine with an imbalanced shaft, i.e. a shaft whose mass center deviates from its geometric center. This constitutes a periodic excitation at the machine rotating frequency (measured in Hertz or revolutions-per-minute, RPM). An imbalance will usually be visible as running speed component with larger amplitude than in a normal situation (formulated differently: as an increased $1\times$ RPM component).

Low-frequency failures Different machine failures give rise to different types of vibration signals: running speed (RPM) related components of sinusoidal character (such as misalignment, imbalance, mechanical looseness) are usually located in the low-frequency range at multiples of RPM. Usually, low-frequency failure signals are deterministic, and indicate severe problems.

Higher-frequency failures Faults of a repetitive impulsive or random nature (like lubrication problems, gear mesh and bearing failures, pump cavitation) are exhibited in the intermediate and high-frequency range. Nonsynchronous high-frequency components are indicative of incipient failure. Many diagnostic methods rely on being able to identify families of equally spaced components, such as harmonics and sidebands. For this purpose, techniques like cepstrum analysis and envelope detection can be applied [Ang87], see section 3.3.2.

Harmonics As wear progresses, faults become repetitive and eventually related to running speed, e.g. as higher harmonics [Bro84]. Harmonics of machine running speed are present in the spectrum for two reasons: certain phenomena occur several times per revolution, leading to multiples of running speed. Second, the fault signal may not be exactly sinusoidal, because of nonlinear responses in the transmission path [Mit93].

Faults in pumps Within centrifugal machinery (such as pumps) the fluid or gas flows through the rotating member radially, so because of pressure gradients, the vane passing frequency (number of impeller vanes times shaft RPM) and some harmonics (caused by deviations in the pressure profile from a pure sine wave) will also be present in the spectrum. The amplitudes of the vane passing frequencies are related to flow and radial

clearances between impeller and stator. There is also preliminary information that links the vane passing frequency in pumps to *cavitation*, a more or less random phenomenon that is caused by vapour bubbles that are collapsing due to increasing pressure in the impeller.

Limitations of expert systems

Automatic machinery diagnostics is often based on expert systems. Commercial systems of this kind are already widely available, but costs can be prohibitive. This may prevent application of automatic monitoring techniques in practical settings (especially when monitoring small rotating machinery, like a small submersible pump). There are some fundamental limitations of expert systems in this application:

- it is often not feasible to specify all possible fault scenarios beforehand
- explicit heuristics (that form the basis for an expert system) will rely heavily on the operating conditions of the machine
- individual machine vibration behaviour can deviate largely from the expected behaviour
- adaptation-through-time may require a full redesign of the system. Hence, an expert system may be tailored to a specific machine in a specific environment at a specific time (at high cost), but generalizing the knowledge that is present in the system to somewhat different machines or circumstances may be impossible

Due to many problems involved in the interpretation of vibration spectra, a rule-based system is often not capable of an adequate diagnosis, while a human expert with knowledge of the machine is indeed capable of a diagnosis. Many ambiguities are present in the vibration spectrum, since certain components may more or less overlap. e.g.

- a high running speed component usually indicates imbalance, but can also be electrically induced or caused by a fan that is loose on the shaft, or a thermally bowed shaft. Moreover, after machine breakdown for inspection, thermally induced faults will disappear completely
- certain faults may exhibit themselves in different ways in the spectrum. Take for instance rolling element bearing failures, which lead to high nonsynchronous frequencies (i.e. frequencies that are not multiples of running speed). They change with variations in contact angle (caused by changes in load or when sliding occurs) and actual failure frequencies may therefore deviate slightly from calculated values
- high vibration levels may both be caused by forced vibrations due to machine faults and by ordinary excitation of machine resonances, which clarifies the need for understanding the structural behaviour of the machine

- noise may severely complicate detection of early (low-amplitude higher-frequency) failures. Noise enters the scene by means of measurement errors, interference of nearby machines (see chapter 5), flow, cavitation, etc. Direct observation of e.g. bearing defect frequencies at calculated locations in the spectrum may not always be the most suitable approach, since (incipient) spikes are sometimes difficult to observe in the spectrum
- detection of fault-related transients in the time domain may be difficult [WM96]. On rare occasions, apparent nonstationarities are in fact very low-frequency modulates, which further complicates analysis

Techniques that capture knowledge by learning from examples may overcome some of these problems. We give an overview of this class of methods in the next chapter.

1.4 Delft experimental setup

The techniques for health monitoring investigated in this thesis are often applied to one specific machine: a submersible pump that was positioned in the basement of the research institute. Having a machine in a laboratory environment has the advantage of being able to measure what, when, how and how long the experimenter wants. For example, faults could be induced to the machine in a controlled manner. Moreover, the machine was available for modal analysis and experimental modal testing. The submersible pump (figure 1.6) is

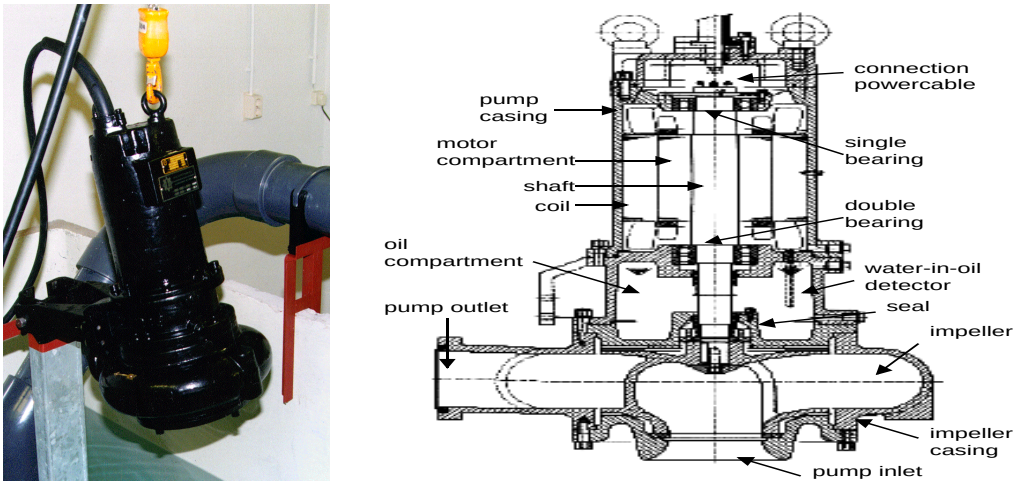


Figure 1.6: One-channel impeller submersible pump (DECR 20-8, Landustrie Sneek, The Netherlands); capacity: 3 KW; rotation speed: 1500 RPM at 50 Hz driving frequency and nominal load; maximum flow: 1.7 M^3/min.

a centrifugal pump and it consists of an impeller with a fixed blade, housed in a suitable shaped casing, the impeller being mounted on a rotating shaft. In order to minimize the influence of the transmission path we chose a machine with only one shaft to which the impeller is mounted. The two ball-bearings in the pump casing keep the shaft in place. The shaft together with these bearings, the seal and the impeller will be responsible for most of

the measured frequency components. Three specific defects can be induced to the machine: loose foundation of the pump, imbalance, and bearing-failures. Cavitation will always be present, and in the actual measurement runs, this phenomenon is being prevented as much as possible. The pump can be made to run in different operating modes: both the running speed and the workload can be varied. Running speed is controlled by a frequency converter, workload can be influenced by opening or closing the membrane (hence determining the amount of water that is pumped around the basin). More (technical) information on the setup can be found in appendix A.

1.5 Other setups

In this thesis we investigate the use of learning methods for practical health monitoring. Besides the previously described test bench with the submersible pump, we address several other monitoring problems as well. We will briefly describe the measurement setup in each of these cases, for easy reference later on in the thesis.

1.5.1 Lemmer experimental setup

Vibration was measured on three identical pump sets in pumping station “Buma” at Lemmer, The Netherlands (figure 1.7). One pump (no. 2) showed severe gear damage (pitting, i.e. surface cracking due to unequal load and wear, an example was given in figure 1.1(a)), whereas the others showed no significant damage (no. 3) and an incipient damage (no . 1), respectively. In the beginning of 1999, the damaged gear in pumpset 2 was replaced by a new faultless gear. All pumps have similar power consumption, age and amount of running hours. The load of the pumps can be influenced by lowering or lifting a sliding door (which determines the amount of water that can be put through). Seven accelerometers were used to measure the vibration near different structural elements of the machine (shaft, gears, bearings). Vibration was measured with 7 uni-directional accelerometers, placed near the driving shaft (in three directions), and upper and lower bearings supporting shafts of both gearboxes (that perform a two-step reduction of the running speed of the driving shaft to the outgoing shaft, to which the impeller is attached). The number of teeth in the first gear wheel was 13. The driving shaft speed was 997 RPM, which results in a gear mesh frequency of $13 * 997/60 = 216$ Hz. The first three channels correspond to three different measurement directions near the incoming axis. Sensors 4 to 7 are mounted on the gearbox casing. Sensors 4 and 5 are mounted near the first set of decelleration gears (4: upper part of the casing; 5: lower part of the casing). Sensors 6 and 7 are mounted near the second set of gears (6: upper part of the casing; 7 lower part of the casing, near the outgoing shaft).

1.5.2 Deteriorating gearbox monitoring

In a laboratory test bench a gearbox was monitored with 8 sensors distributed across the casing. The mounting positions correspond to the bearing positions. During a period of approximately 2 months the machine was monitored continuously; in this period a fault devel-

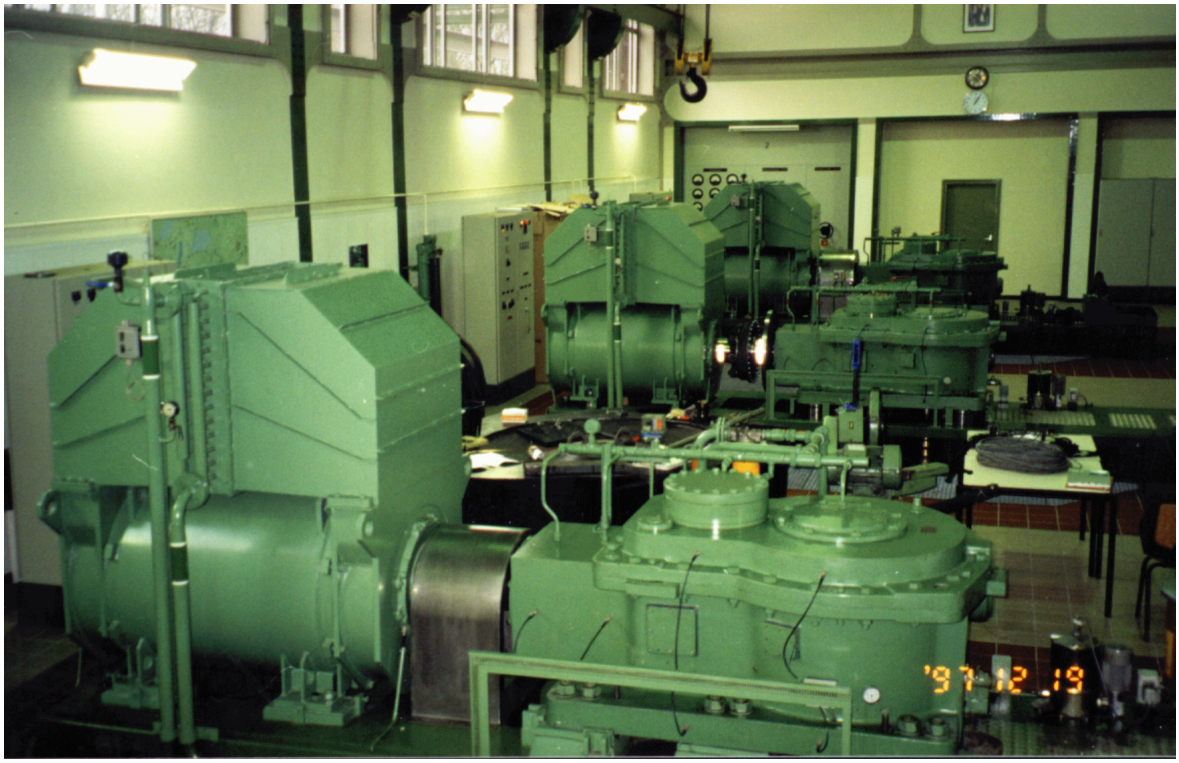


Figure 1.7: Pumping station “Buma” at Lemmer, The Netherlands

oped in one of the gear wheels, which was accompanied by a bearing fault. In the weekends the machine was shut down, but measurement was continued. At the end of the monitoring period the fault progressed quickly; upon component breakdown monitoring was stopped. The measurements on different positions were not synchronized, i.e. exact time of measurement differs between sensors. The number of measurements in the monitoring period also differs per sensor position. Several different quantities were measured (see table 1.3). The minimum frequency in each measured quantity was 0 Hz. The machine running speed f_r was approximately constant at 1000 RPM or 16.7 Hz. The gearmesh frequency in the first set of gears was $20 \times f_r = 333.3$ Hz, the gearmesh frequency in the second set of gears was $40 \times f_r = 666.6$ Hz.

1.5.3 Gas leak detection

Leak activity is measured by hydrophones which are placed in the neighbourhood of an underwater pipeline and which record underwater sounds. When bubbles of escaping gas are detected, an alarm has to be raised, but another demand is that *no false alarms* will be raised. That is, background noise signals are not to be classified as leak sounds. Experimental data was recorded under real conditions at the North Sea near Norway. The data was obtained by simulating a gas leak under water and recording the resulting sounds. First a hydrophone is

Table 1.3: Measured quantities in degrading gearbox setup

quantity	*feature number*	*max. frequency (Hz)*	*resolution (# bins)*
velocity	1	1000	800
acceleration	2	2500	1600
	3	10000	1600
envelope spectrum	4	500	800
	5	2000	1600
	6	8000	1600
acoustic emission	7	500	800
	8	2000	1600
	9	8000	1600

placed at the bottom of the sea, after which a leak is simulated by filling a gas cylinder with pressurized gas. To this gas container a hose is attached and on the other end of the hose a nozzle is placed through which gas can escape. That end of the hose is put underwater at a certain depth and the gas container is opened. First cavitation occurs, but this is disregarded in the following: the measurements made during this phase were not used. After cavitation the gas plume appears and leak signals are recorded. Because the pressure in the gas container decreases, the intensity of the leak signals decreases until only background noise is present. Moreover, before the gas cylinder is opened some background noise is recorded. The complete sequence from recording the background noise to the end of the gas bubble plume when the gas cylinder is exhausted, is called a *blowdown*.

1.5.4 Medical monitoring

Health monitoring is not confined to machines, pipelines or other mechanical structures; a recent workshop, the “1999 IEE Colloquium on Condition monitoring of machinery, external structures and health” was centered around the common issues in seemingly different areas as biomedical and mechanical health monitoring. We think that our framework is suitable for several medical monitoring problems as well and illustrate this in chapter 8 in two particular cases: automatic quantification of Tourette’s syndrom and automatic detection of Alzheimer’s disease.

1.5.5 Acoustic source separation

In an outdoor acoustical measurement setup in a suburb of a Dutch town, noise from a passing car was measured on a logarithmically spaced 5-microphone array [Ypm99b]. Severe disturbances were present like: a rotating machine running stationary around 60 Hz that was positioned at approximately 15 meters from the first sensor in the array at an angle of 45 degrees; an airplane crossing the scene at high altitude; wind bursts from vegetation in the vicinity of sensors 4 and 5 in the array. The purpose of the experiment was to blindly separate the car noise from interferences in the scene.

The setup is drawn schematically in figure 1.8[2].

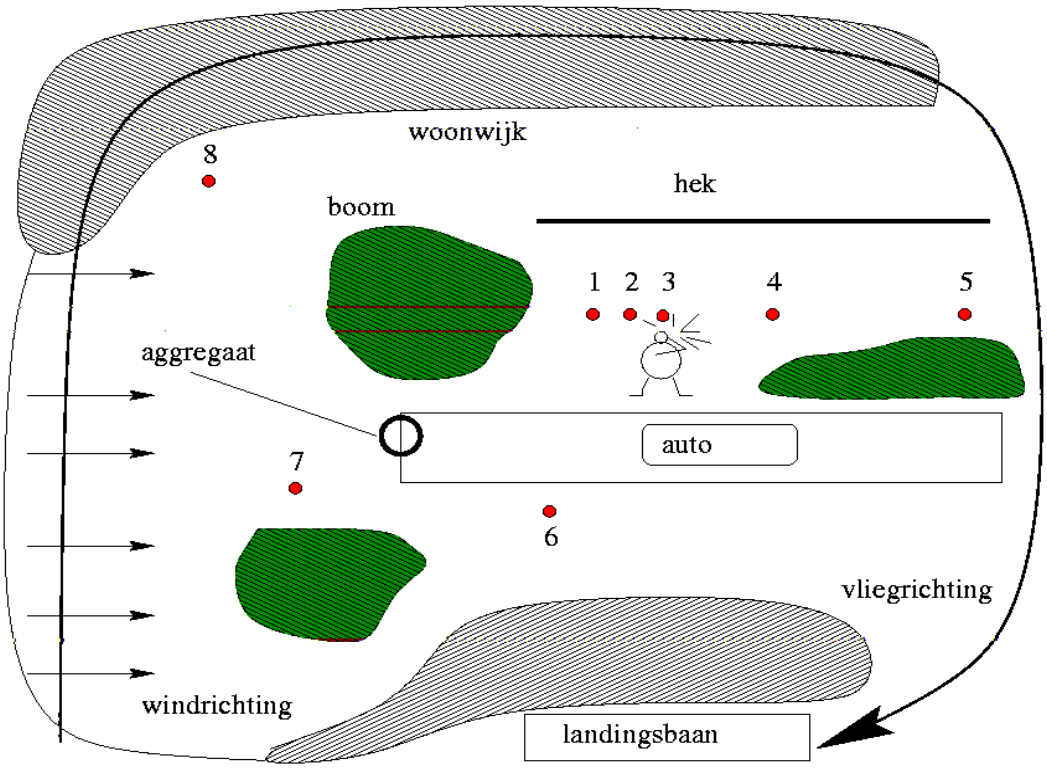


Figure 1.8: Acoustic measurement setup for blind source separation

[2]woonwijk=suburb; boom=tree; hek=fence; aggregaat=rotating machine; richting=direction; baan=track

Chapter 2

Learning from examples: the induction principle

The practical use of methods that learn the structure in machine measurements touches upon the fundamentals of learning theory: how do we know that a learning system has learned the right thing? In other words, how do we know that the system generalizes its learned knowledge succesfully to handle novel (unseen) data? This chapter aims at clarifying the theoretical issues involved in this question. From that, an important application-specific question (addressed in chapters 3, 6 and 8) arises: how to represent the vibration behaviour of a rotating machine, such that this representation can be used for effective health monitoring afterwards?

2.1 The bias-variance dilemma

In a (supervised) learning problem, we face the following dilemma if we try to infer knowledge (general principles) from a set of measured data: we can explain the training data perfectly with a very complex model, but have to pay the price when we encounter new datasets (from the same distribution). First, the finite sample sizes that we encounter in practice inevitably cause 'noise' in the data (structure induced only because of the finite sample size) that will be modeled along with the underlying data structure. Second, measurement noise may distort the samples in the dataset.

The problem described here can be found in the literature under the names *overtraining* (neural networks) and *bias-variance dilemma* (statistics). If we have a learning machine with enough "capacity" to describe all regularities in a finite length dataset used for training the machine (the learning set), the machine will ultimately describe all possible regularities, whether they represent meaningful structure of the underlying data distribution or just coincidental structure due to finite sample sizes or noise. An independent test set that is drawn from the underlying distribution will not possess the coincidental structure and the learning machine will start to perform much worse on the test set (according to some performance criterion, like mean-squared-error (MSE) between a set of desired outputs and the actual predictions of the learning machine).

Bias-variance decomposition

More specifically, the MSE in a learning problem can be decomposed in a *bias* and a *variance* term. Consider the problem of estimating a (single-valued) regression function $f(\mathbf{x};\mathscr{D})$ given a set of examples $\mathscr{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ (the training set). Here, the $\mathbf{x}_i$ are multivariate 'measurements' and the y_i are the corresponding function values. Pairs $(\mathbf{x}_i, y_i)$ are assumed to follow a joint distribution P. If the MSE is chosen as the criterion for determining the estimator's performance, the following decomposition of the error in predicting y from an arbitrary $\mathbf{x}$ can be made [GBD92]:

$$E[(y - f(\mathbf{x};\mathscr{D}))^2|\mathbf{x},\mathscr{D}] = E[(y - E[y|\mathbf{x}])^2|\mathbf{x},\mathscr{D}] + (f(\mathbf{x};\mathscr{D}) - E[y|\mathbf{x}])^2 \tag{2.1}$$

Here, the expectation $E[\cdot]$ is taken over the probability distribution P. The first term on the right-hand-side (RHS) of the above equation does not depend on $\mathscr{D}$ or f (since it is just the variance of y given $\mathbf{x}$). The second term on the RHS is the term that represents the quality of the estimator f. The mean-squared performance $E_{\mathscr{D}}[(f(\mathbf{x};\mathscr{D}) - E[y|\mathbf{x}])^2]$ on the dataset $\mathscr{D}$ can be decomposed as [GBD92]

$$\begin{aligned} E_{\mathscr{D}}[(f(\mathbf{x};\mathscr{D}) - E[y|\mathbf{x}])^2] &= (E_{\mathscr{D}}[f(\mathbf{x};\mathscr{D})] - E[y|\mathbf{x}])^2 + E_{\mathscr{D}}[(f(\mathbf{x};\mathscr{D}) - E_{\mathscr{D}}[f(\mathbf{x};\mathscr{D})])^2] \\ &= \text{bias}^2 + \text{variance} \end{aligned} \tag{2.2}$$

Here, the expectation $E_{\mathscr{D}}[\cdot]$ is taken over all possible datasets $\mathscr{D}$ of fixed sample size N. The first term on the RHS in the above formula is the bias of the estimator. If the estimator f predicts the expected value (over P) of y given $\mathbf{x}$ adequately (on average, over all realizations of $\mathscr{D}$), this term will be low. However, the performance of f may (still) depend heavily on a particular dataset $\mathscr{D}$. This effect is measured by the second term of the RHS, the variance term. It is an indication of the 'sensitivity' of the estimator to a particular realization of the training set. If its performance is highly varying over several training sets, consistent predictions on average (low bias) may still lead to a badly performing estimator (since the estimator is determined on the basis of a single training set)[1].

Remedies to overtraining

Remedies to this problem have been proposed from different viewpoints. One can argue that stopping the learning process before overtraining takes place is a remedy. Bayesian methods incorporate prior knowledge on the (probabilistic nature of the) problem in a sound manner and focus on the estimation of *probabilities* from empirical data. Statistical learning theory uses *structural risk minimization* to enable good generalization with small sample sizes. In classical (parametric) statistical model building approaches and neural approaches this issue is addressed by incorporating *penalties for model complexity* or *regularisation* of the training method. Noise injection during learning (regularization) may prevent the machine from

[1]Note that the above decomposition for regression problems can be generalized to classification by noting that class labels are just indicator functions that are to be learned by the learning machine

learning the coincidental structure in a training set [Bis95]. From an information-theoretic viewpoint the Minimum Description Length (MDL) principle has been formulated, where also an explicit penalty for model complexity is used. These three important approaches to learning-from-examples and their relation will be described in the next sections.

2.2 Three approaches to learning

In the first paragraph of this chapter, we raised a philosophical question that was already known to the ancient Greeks [LV93]: *how to infer a general law or principle from the observation of particular instances?* The process of inferring this law is called **induction**, as opposed to *deduction* in which one derives a thruth or falsehood from a set of axioms and a reasoning system. Deduction can be seen as a special case of induction. It was argued by the philosopher Hume that true induction is impossible, since one can only reach conclusions with known data and methods. One way to remedy this problem is to take a probabilistic reasoning approach (like Bayes' rule, section 2.2.1), but here the learner's initial belief (the prior) is needed, which again should "come from somewhere". The inductive method of Solomonoff is said [LV93] to overcome this problem, since it incorporates three classic ideas that a method for inductive inference should contain:

First principle (*Multiple explanations*, Epicurus): *If more than one theory is consistent with the observations, keep all theories*

Second principle (*Occam's razor*, William of Okham): *Entities should not be multiplied beyond necessity*, often read as [Vap98]: *The simplest explanation is the best*

Third principle (*Bayes' rule*, Thomas Bayes): *Assume we have observed data D. The probability of hypothesis H being true is proportional to the learner's initial belief in H (prior probability) multiplied by the conditional probability of D given H*

In Solomonoff's theory, all hypotheses that are compatible with the data are kept, hypotheses with low Kolmogorov complexity (large compressibility) have high probability and reasoning is done using Bayes' rule. This theory is a 'perfect theory of induction', since it contains all three elements mentioned above. However, it involves the *universal probability*, which cannot be computed. Moreover, it does not address learning with small sample sizes explicitly. Therefore, the inductive principle of *structural risk minimization* was proposed [Vap98].

2.2.1 Bayesian inference

The **first** approach to learning that we will address is Bayesian inference. Consider a dataset $Z = z_1, z_2, \ldots, z_N$ that is obtained by randomly drawing samples from a probability distribution $P(z)$ that depends on the particular choice θ from a set of parameters vectors Θ. We can try to infer the underlying distribution from the observations. The *maximum likelihood* solution to this problem would be to consider the joint density of Z, $P(Z|\theta) = \prod P(z_n|\theta)$ to be the likelihood function $\mathscr{L}(\theta)$ of θ for the observed Z, and find the parameters θ_{ML} that maximize this function

$$\theta_{\mathrm{ML}} = \underset{\theta}{\operatorname{argmax}} P(Z|\theta) \tag{2.3}$$

The Bayesian approach to the same problem would be to *assume a distribution* on the parameters. Initially, a prior distribution of the parameters is fixed and this distribution is then translated into the posterior distribution using the observed data and *Bayes rule*

$$P(\theta|Z) = \frac{P(Z|\theta)P(\theta)}{P(Z)} \tag{2.4}$$

In the Bayesian framework, the probability of an event represents the *degree of belief* that a person has in the occurrence of the event. It turns out that several properties that a degree of belief should possess happen to coincide with the rules of (classical or frequentist) probability, which measures a physical property of the world (e.g. the probability that a dice will show a '6' after it has been thrown) [Hec96].

MAP solution

The parameter vector that maximizes the posterior probability (MAP estimator) is

$$\begin{aligned} \theta_{\mathrm{MAP}} &= \underset{\theta}{\operatorname{argmax}} P(\theta|Z) \\ &= \underset{\theta}{\operatorname{argmax}} \frac{P(Z|\theta)P(\theta)}{P(Z)} \\ &= \underset{\theta}{\operatorname{argmax}} P(Z|\theta)P(\theta) \end{aligned} \tag{2.5}$$

where the last simplification can be made since the term $P(Z)$ does not depend on θ. If all parameter vectors are assumed equally probable (we have a *flat* prior $P(\theta)$) the MAP estimator just maximizes the likelihood function, so the MAP and ML approach coincide. The MAP-solution is in fact the model (parameter vector) that maximizes the posterior probability given the data and a prior over the parameters, and is one of the two solutions that one can get in the Bayesian framework.

Evidence solution

In the *evidence* framework, one obtains the solution by averaging over all possible values of the parameters, weighted by their priors. The desired density function is written as

$$P(z|Z) = \int P(z,\theta|Z)d\theta \tag{2.6}$$

which can be rewritten as

$$P(z|Z) = \int P(z|\theta)P(\theta|Z)d\theta \tag{2.7}$$

We see that the Bayesian evidence approach does not lead to a specific choice for the parameter vector θ, but that the estimated density is found as the result of weighting the chosen density $p(z|\theta)$ with the posterior $p(\theta|Z)$ and averaging over all possible values for the parameter vector. For large sample size the likelihood function becomes more sharply peaked around the true value of θ and the prior becomes 'relatively flat' with respect to the likelihood. Therefore, the Bayesian density estimate will approach the ML estimate for large sample size [Bis95].

2.2.2 Information-theoretic approach

The **second** approach to inductive inference is the algorithmic information-theoretic approach, following the work of Solomonoff, Chaitin, Rissanen and Wallace [LV93]. This approach implements all three inductive principles. It is based on the idea that it is often easier to a encode a hypothesis than devise a prior distribution over the hypothesis space. With each probability distribution a *code* can be identified, i.e. a mapping from the data alphabet A_D to some coding alphabet A_C [2]. Hence, a code is a unique description method of a dataset in A_D. According to the minimum description length (MDL) principle one should select the hypothesis that describes a dataset in such a way that *the length (in bits) of the hypothesis plus the length (in bits) of the data described with help of the hypothesis is minimal*. The expression into bits is not critical: it holds for every (countably infinite) alphabet, not only the binary alphabet. Consequently, the hypothesis that is chosen using the MDL-principle is

$$\theta_{\text{MDL}} = \underset{\theta}{\operatorname{argmin}}\{\mathscr{L}_{C_2}(Z|\theta) + \mathscr{L}_{C_1}(\theta)\} \tag{2.8}$$

Here, $\mathscr{L}$ denotes the codelength (in bits), C_1 is the code to encode the hypothesis and C_2 is the code to encode the data with help of the hypothesis. Hence, with the two-part code of (2.8) one tries to balance the complexity of the hypothesis with the complexity of the error with respect to the data. Note that the hypotheses selected by Bayesian (MAP) learning and MDL-learning coincide if we measure codelength of z as $-\log P(z)$. This code is called the Shannon-Fano code (see section 2.2.4); it has minimum expected codelength for "source words" z from distribution $P(z)$. However, there are many other codes that approximate the shortest expected codelength. Choosing a different code causes MDL and Bayesian inference to differ, as we will illustrate next.

Ideal MDL

In *ideal MDL* one chooses Kolmogorov complexity to measure codelength. This leads to a method that selects a hypothesis such that: the data is 'optimally random' with respect to that hypothesis and the hypothesis is 'optimally random' with respect to the prior over the hypothesis space[3]. This assumes that data can always be described best by assuming a random

[2] An example of a coding alphabet is the binary alphabet $A_C = \{0,1\}$

[3] For a more precise formulation, see [VL99]

residual and a 'maximally ignorant' prior[4]. We can express maximum ignorance by taking as our prior the *universal distribution* $\mathbf{m}(z)$. This distribution multiplicatively dominates every other distribution[5], which means that every hypothesis can be considered random with respect to this prior. It turns out [VL99] that if the *true* prior is computable[6] then using $\mathbf{m}(z)$ is as good as using the true prior. Conditions under which it is allowed to 'plug in' the universal distribution can be found in [VL99]. The universal distribution assigns high probability to simple objects and low probability to complex or random objects. *Simple* and *complex* are phrased in terms of Kolmogorov complexity $K(z)$. The *Kolmogorov complexity* of a string z is the length of the shortest program on a Turing machine that computes the string, i.e. outputs the string on the output tape. This is not a practical measure, since computability of the Kolmogorov complexity would imply the decidability of the Halting problem [LV93]. However, the universal distribution can be related to Kolmogorov complexity as

$$-\log \mathbf{m}(z) = K(z) \pm \mathcal{O}(1) \tag{2.9}$$

The term $\mathcal{O}(1)$ expresses that the equality holds up to a constant. The MDL-principle chooses the hypothesis such that [LV93]

$$\begin{aligned} \theta_{\text{MDL}} &= \operatorname*{argmin}_{\theta} \{-\log \mathbf{m}(Z|\theta) - \log \mathbf{m}(\theta)\} \\ &= \operatorname*{argmin}_{\theta} \{K(Z|\theta) + K(\theta)\} \end{aligned} \tag{2.10}$$

where we ignored the additive constant. This formulation exploits **individual regularities** in data, since we select the hypothesis that allows for the shortest possible description of each particular data item.

2.2.3 Structural risk minimization

The **third** approach to inductive inference is the *structural risk minimization* principle (SRM), formulated by Vapnik [Vap98]. Here, the goal is to minimize the expected risk on **future data** instead of training data, assuming identically and independently distributed (i.i.d.) data and ignoring information about the prior distribution. Instead, a *structure* on the set of learning machines is imposed. The SRM method addresses the problem of *learning with small sample sizes*, where it is often troublesome to get reliable estimates of (posterior) distributions. The philosophy of SRM is **not to solve a more difficult problem than necessary**: if the aim is to perform regression or classification (also referred to as pattern recognition), the use of density estimation (cf. Bayesian methods) as an intermediate step is not necessary. Density estimation is considered a more difficult problem than regression or pattern recognition.

[4] Because of this, ideal MDL has the tendency to underfit data: even with noiseless data from a certain generating model, a *simpler* generating model plus a random residual is assumed

[5] More specifically [VL99]: for each P in the family of enumerable probability distributions EP on the sample space E there is a constant c such that $c\mathbf{m}(z) > P(z)$, for all $z \in E$

[6] The correct term here is *recursive*; it may be interpreted as *computable* in the context of this discussion

VC-dimension and risk minimization

The SRM-method for pattern recognition uses the concept of *VC-dimension* of a set of (indicator) functions, which is the maximum number h in a set Z of l vectors $z_1,\ldots,z_l$ that can be dichotomized in all possible 2^h ways. The VC-dimension of a set of functions $Q(z,\alpha),\alpha\in\Lambda$[7] is a measure for the *capacity* of the set, which *can* be related to the number of parameters (e.g. if the set of functions is linear in their parameters), but *need not* be related. A sample size is considered **small** if the ratio of the number of training patterns l to the VC-dimension of the set of functions induced by the learning machine h is small, e.g. $\frac{l}{h}<20$ [Vap98].

Example 2.1: capacity of functions that are nonlinear in their parameters

Consider a set of indicator functions that is nonlinear in their parameters. For this class of function sets, the number of free parameters is not automatically the factor that determines the capacity of the set. As an example [Vap98], consider a particular set of indicator functions on the interval $[0,2\pi]$,

$$Q(z,\alpha)=\mathscr{I}(\sin\alpha z),\qquad z\in[0,2\pi],\alpha\in(0,\infty) \tag{2.11}$$

where $\mathscr{I}(\cdot)$ denotes the indicator function. For any l and any binary sequence $\delta_1,\ldots,\delta_l$ there can be found l points $Z=z_1,\ldots,z_l$ and a choice for α such that the equation

$$\mathscr{I}(\sin\alpha z_i)=\delta_i,\qquad i=1,\ldots,l \tag{2.12}$$

holds. This equation expresses that the dataset Z can be dichotomized in l ways, regardless the number of samples l. Hence, this set of indicator functions is said to have *infinite* VC-dimension. Similarly, one can construct a neural network such that its VC-dimension is smaller than the number of parameters in the network. A multilayer feedforward neural network (section 2.2.3) is an example of a learning machine that implements a set of functions that is nonlinear in its parameters. This idea also underlies the support vector machine (see example below), where a huge number of parameters is allowed, while the capacity can be kept low. □

A naive approach to learning could be aimed at minimization of the error on the training set (i.e. the *apparent* error). In the SRM context, this quantity is called *empirical risk*. Empirical risk minimization (ERM) bares the danger of overfitting to a particular training set, i.e. adaptation to the 'noise' in the set due to the finite size of the training set. If the training set is large (and if we assume the training set to be representative for the underlying data distribution), this danger will be smaller than if the training set is small (and the 'noise' is large). The latter situation is the small sample size situation, where *sample* now refers to a set of feature vectors. In the SRM-framework, the aim is to minimize the *risk*

$$R(\alpha)=\int L(z,g(z,\alpha))dF(z),\qquad \alpha\varepsilon\Lambda \tag{2.13}$$

[7] This set of functions is induced by the class of learning machines Λ that is chosen

of the set of functions $g(z,\alpha)$ (parameterized by α), taking into account the learning set $z = z_1,\ldots,z_l$. The form of the *loss function* $L(z,g(z,\alpha))$ depends on the particular learning task[8]. The argument z is taken from the subset Z of the vector space $\mathbb{R}^n$, on which a probability distribution $F(z)$ is defined. An instance α^i is the machine that determines the specific function $g(z,\alpha^i)$ in the set of functions $g(z,\alpha)$. One should now look for the instance α^0 such that the expected loss with respect to z, $R(\alpha^0)$, is the minimal risk obtainable when the probability distribution of the data $F(z)$ is unknown (and only a particular sample $z_1,\ldots,z_l$ is given).

Capacity control

It was shown [Vap98] that the risk $R(\alpha)$ can be *bounded* using knowledge about the empirical risk and the capacity of the learning machine only. Bounding means that there is at least $1-\eta$ probability that the particular bound holds for all functions from the set of totally bounded functions $0 \leq Q(z,\alpha) \leq B$ with finite VC-dimension h. The derived bound equals the sum of the empirical risk $R_{emp}(\alpha)$ (i.e. the risk on the training set $z_1,\ldots,z_l$) and a term that depends on both the empirical risk and the 'complexity' of the chosen learning machine:

$$R(\alpha) \leq R_{emp}(\alpha) + \frac{B\varepsilon(l)}{2}\left(1+\sqrt{1+\frac{4R_{emp}(\alpha)}{B\varepsilon(l)}}\right) \tag{2.14}$$

In this formula, the term $\varepsilon(l)$ depends on the parameters h, l and η as

$$\varepsilon(l) = \frac{4}{l}\left\{h\left(\log\frac{2l}{h}+1\right) - \log\frac{\eta}{4}\right\} \tag{2.15}$$

For large sample sizes (i.e. $\frac{l}{h}$ is large), the term $\varepsilon(l)$ is small and the risk in (2.14) will be dominated by the empirical risk. Hence, for large sample sizes it can make sense to minimize the error on the training set. However, with small sample sizes one has to control the capacity of the learning machine in order to achieve good generalization, i.e. make the VC-dimension of the learning machine a controlling variable. Therefore, Vapnik imposes a *structure* $\mathscr{S}$ on the set S of functions $Q(z,\alpha), \alpha \in \Lambda$ such that

$$\mathscr{S}_k = \{Q(z,\alpha) : \alpha \in \Lambda_k\} \tag{2.16}$$

and the subsets of functions S_k are *nested* as $S_1 \subset S_2 \subset \ldots \subset S_n \subset \ldots$. The SRM-method now chooses the element S_k that achieves the smallest bound for the risk (in other words: minimizes the **guaranteed** risk).

Example 2.2: nesting of machines

In an *admissible structure* each subset S_k has a finite VC-dimension h_k. The nesting according to an admissible structure means that for $j \leq k$ the respective VC-dimensions h_j, h_k of the

[8] In regression, the mean-squared-error between estimated and actual function values can be used

subsets S_j, S_k are related as $h_j \leq h_k$. The nesting of machines is depicted graphically [Vap98] in figure 2.1(a)[9]. □

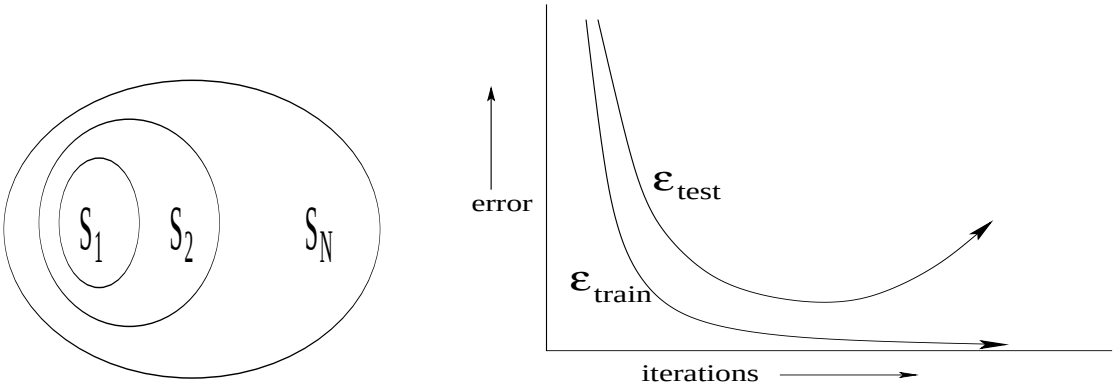


Figure 2.1: Learning theory: nesting of learning machines (a) and bias-variance dilemma (b)

Example 2.3: inductive inference with feedforward neural networks

We include a brief discussion on neural networks in the learning theory context. A feedforward neural network is a network of interconnected neurons, arranged in at least three layers (an input-, hidden- and output layer), see figure 2.2(d). A neuron consists of a summation of the incoming links to a neuron, followed by the application of a nonlinear function (usually a sigmoid activation function), see figure 2.2(b). An adjustable bias is included in each neuron as well. The outputs of neurons in the previous layer are weighted by adjustable weights before their value is used as input to a neuron in the next layer. A feedforward neural network is capable of approximating any function to arbitrary accuracy, provided the number of hidden units is chosen adequately. For example, the network can be taught to learn indicator functions, in which case a classifier is obtained, see figure 2.2(c).

Neural network training

Neural network training is a supervised learning procedure (see figure 2.2(a)) in that the outputs corresponding to a training sample are known and used to adjust the network weights to decrease the error on the output. This is done for multilayer neural networks with the error backpropagation algorithm [Bis95]. The neural network training procedure is a form of empirical risk minimization (ERM) and bares the risk of overtraining. Recall that the performance of a learning machine is determined by its capacity to learn a function (bias) and by the spread in its predictions over several instances of the learning set (the variance). Because of its *universal approximation* property [Cyb89, HSW89], a neural network with zero bias can always be constructed if the architecture is chosen properly.

[9] Note that this structure is comparable to the general-to-specific ordering of hypotheses within the framework of concept-learning [Mit97]

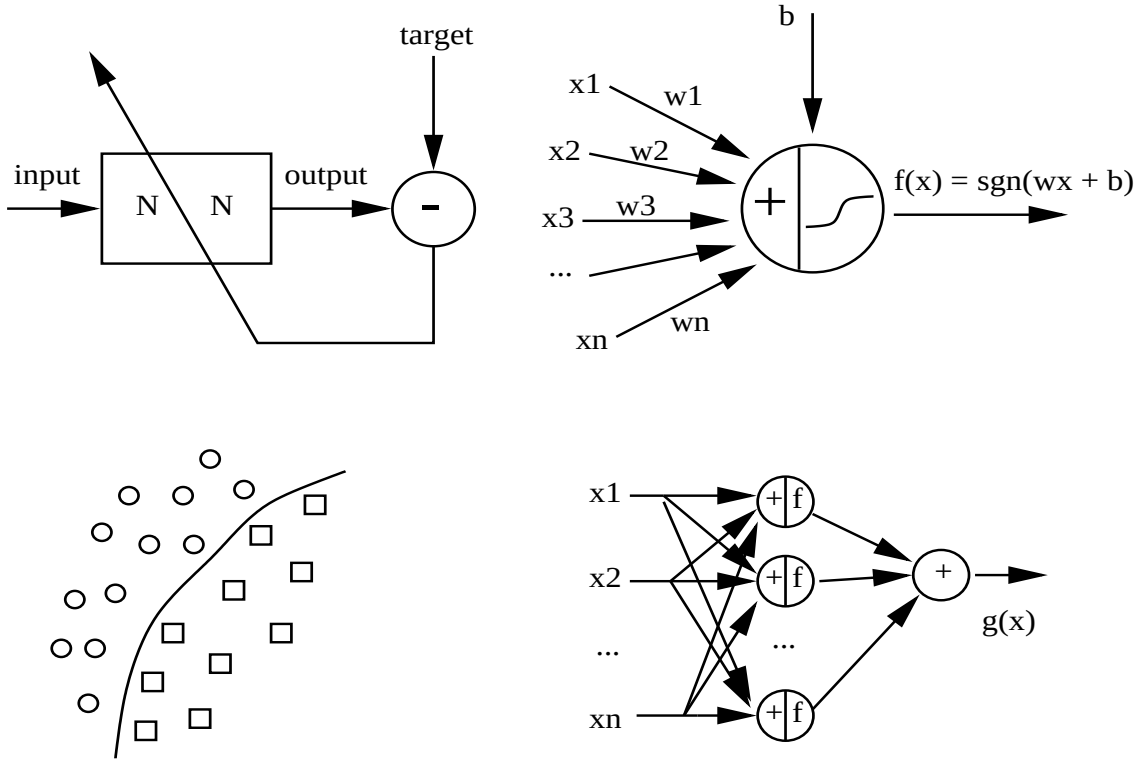


Figure 2.2: Feedforward neural networks: a. (top-left) supervised learning; b. (top-right) artificial neuron; c. (bottom-left) neural classifier; d. (bottom-right) layer of neurons

During the training process, the network transitions from an approximately linear machine (with possibly large bias) to a nonlinear machine (with smaller bias due to the learning procedure, but with possibly larger variance). Hence, the error on a training set ε_{train} and on an independent test set ε_{test} will exhibit the behaviour that is shown in figure 2.1(b). As learning proceeds (increasing number of iterations), the machine becomes more tuned to the training set (which leads to a smaller training error). The test error will also decrease, since tuning to the training set will mean that the regularities in the data (indicative of the regularities in the underlying distribution) are described. When training continues, the coincidental regularities in the particular training set will ultimately be described as well, leading to a machine with smaller bias but larger variance. The penalty for the increased variance will become visible in the increasing test error. □

Example 2.4: support vector classifier

The SRM principle is implemented for classification purposes in the *support vector classifier*. In this method [Vap95], the *optimal separating hyperplane* for two separable classes of labeled data $\{(\mathbf{z}_i, y_i), i = 1, \ldots, l\}$, where $y_i \in \{-1, 1\}$ are the class labels, is given by [Sch97, Vap95]

$$f(\mathbf{z}) = \text{sign}(\sum_{i=1}^{l} \alpha_i y_i (\mathbf{z} \cdot \mathbf{z}_i) + b) \tag{2.17}$$

The coefficients α_i can be computed by a quadratic minimization procedure, and turn out to be nonzero only for the data samples near the classification border (i.e. on the *margin*), the *support vectors*. Using Mercer's theorem, it can be derived [Sch97] that a dot-product in a transformed space (obtained by some nonlinear mapping Φ) can be expressed as the application of a corresponding kernel $k(\cdot)$ in the original space $(\Phi(\mathbf{x}_1) \cdot \Phi(\mathbf{x}_2)) = k(\mathbf{x}_1, \mathbf{x}_2)$. By replacing the dot-product $\mathbf{z} \cdot \mathbf{z}_i$ in (2.17) by the more general similarity measure $k(\mathbf{z}, \mathbf{z}_i)$, discriminants of arbitrary complexity can be obtained. For a separable dataset, the method is illustrated in figure 2.3. The dark objects are the support vectors that span the separating hyperplane. □

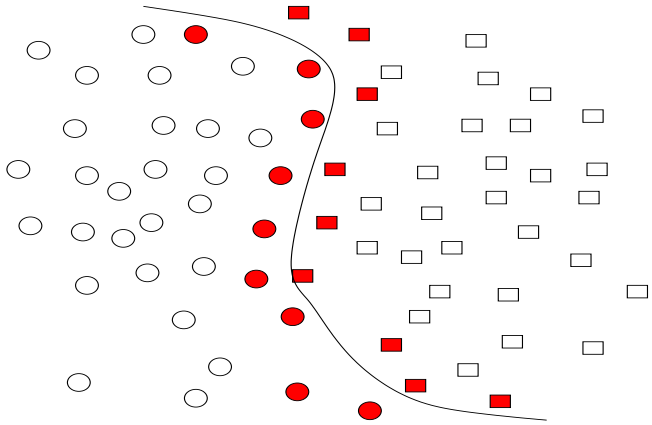


Figure 2.3: Support vector classifier with a separable dataset

2.2.4 Relating the three approaches

Comparing Bayesian learning to MDL

In case one can formulate a (computable) prior for the hypothesis θ, the MDL-approach and the Bayesian MAP-approach coincide. It can be shown that for each distribution there exists a code, the Shannon-Fano code, such that the length of the code for z (in bits) equals $-\log P(z)$ [Grü98]. Intuitively, this means that shorter codes are assigned to more probable outcomes and vice versa. We can now rewrite equation (2.8) as

$$\begin{aligned} \theta_{\text{MDL}} &= \underset{\theta}{\text{argmin}} \{-\log P(Z|\theta) - \log P(\theta)\} \\ &= \underset{\theta}{\text{argmax}}\, P(Z|\theta)P(\theta) \quad = \quad \theta_{\text{MAP}} \end{aligned} \tag{2.18}$$

Choosing the Shannon-Fano code in (2.8) leads to *expected* codelengths that are close to the entropy; the selected hypothesis will reflect the *ensemble* properties of the data. Alternatively,

ideal MDL exploits individual regularities in the data, which may lead to the selection of a different hypothesis than in Bayesian MAP learning.

We note that the Bayesian (evidence) approach to learning implements the first inductive principle, since all hypotheses are retained (and averaged) to estimate the best model from data. Moreover, Bayesian inference implements the third inductive principle if we express Bayes' rule by taking logs (see above). The second inductive principle can be taken into account by formulating a prior that gives simpler hypotheses more probability. Ideal MDL, on the other hand, emphasizes the second and third inductive principles. It also implements the first inductive principle, since all suitable hypotheses for the data are considered (and weighted with their complexity in order to pick the best hypothesis or make the best prediction).

Comparing SRM to Bayesian learning

The Bayesian strategy to statistical inference is to minimize some loss functional by choosing a learning machine that minimizes the expected loss over problems in the problem space. This requires a prior distribution over the set of problems. The SRM-method is an example of the *minimax* loss strategy, where one chooses the learning machine that minimizes the loss for the *worst* problem in the set of problems. For a specific learning problem, the Bayesian strategy hence gives the *best average loss*, whereas the minimax strategy gives the *best guaranteed loss*. Moreover, Bayesian (MAP) inference assumes strong prior information: both qualitatively (in the sense that the set of functions of the learning machine is assumed to 'contain' the set of target functions) and quantitatively (since a prior over hypotheses should be specified). SRM allows for approximation of a set of functions that differs from the set of functions that can be realized by the learning machine, since a structure is imposed on this set of realizable functions.

Comparing SRM to MDL

It can be shown [Vap98] that the (ideal) MDL-principle is a sound inductive principle. For the case of classification (learning a functional relation between samples z_i and corresponding class labels y_i) minimizing the length of the description of this functional relation using some codebook that indexes functionals (in other words: minimizes the compression coefficient), will lead to a minimization of the probability of error. However, the MDL-principle implies a *quantization* of the considered set of functions (that may be continuous in their parameters), since it assumes codebooks with a *finite* number of tables. The chosen quantization affects the compression coefficient, since a coarse quantization allows for smaller codelengths but may hamper a good approximation of set of functions to be learned. In a practical learning problem, one is given a fixed sample (that may be small) and a chosen set of functions. Finding a proper quantization that allows for codebooks with significantly less tables than would be obtained with the 'naive approach' (where no significant compression of each string is obtained) is a highly nontrivial task. In problems where proper codebooks can be formulated, the MDL-principle is a usable (and sound) inductive principle. The SRM

theory does not involve a quantization of the set of functions. Both MDL and SRM impose a structure ('simplicity') on the set of admissible functions and make no (or very weak) prior assumptions. SRM therefore implements the first and second inductive principles.

2.3 Health monitoring with learning methods

In this thesis we look upon machine health monitoring as a learning problem. Several subproblems in the context of the larger problem ("a generic learning method for practical and adequate machine health monitoring") are addressed, and the previously introduced learning concepts enter the scene at several places.

2.3.1 Framework

We study the use of learning methods for automatic processing and interpretation tasks in machine health monitoring. A practical system for on-line machine health monitoring comprises the subtasks shown in figure 2.4. After measuring machine vibration (usually with

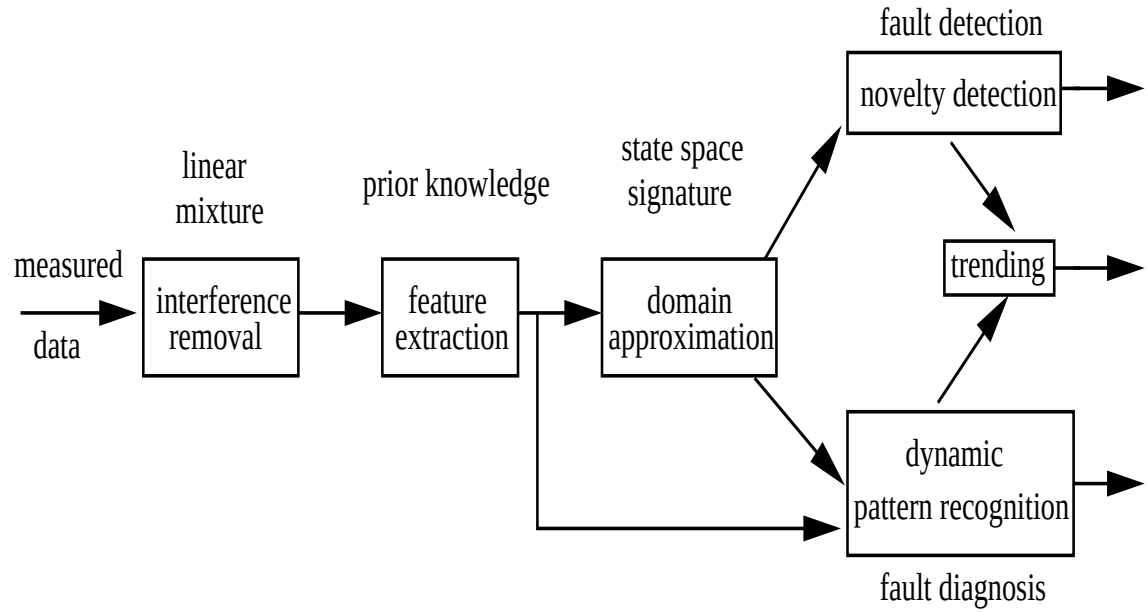


Figure 2.4: Subtasks in machine health monitoring with learning methods

an ensemble of vibration transducers), the instantaneous measurement time series have to be characterized in order to use them for health indication (feature extraction). Prior to this processing of measurement signals in the time domain (temporal processing), we can improve signal-to-noise ratios or focus on fault-related vibration sources by using the spatial diversity and redundancy in a multichannel vibration measurement (spatio-temporal processing). Taking several measurements of a machine that operates normally in varying operating conditions and finding a suitable characterization of the measurements (using monitoring

heuristics and prior knowledge about the machine) leads to a description of the normal behaviour of the machine (state space signature or generalized fingerprint). This description can be used to detect deviating vibration patterns (novelty detection), which may be used as the basis for detection of faults in a machine. Knowledge about the parts of the space occupied by failures can then be used for diagnosis of faults. This knowledge can either be present beforehand, e.g. in the form of heuristics and knowledge from previous case studies, or be acquired during machine operation. Ultimately, one would like to use the transition of the machine to a wear- or 'incipient failure' state as an indicator of the time-to-failure, in other words to analyze the trend in the health index.

2.3.2 Thesis outline

In chapter 3 we study methods to *characterize* the short-time machine *vibration signal*. First, the pump in the test rig is analyzed for its modal properties in order to provide a basic identification of the pump system. Then we discuss correlation-based and parametric methods for signal analysis. The parametric methods in chapter 3 are often based on estimation with maximum likelihood. Particular analysis methods for detecting nonlinearity and nonstationarity are described and illustrated. A method for learning invariants in signals, the ASSOM, is then discussed and applied for machine health monitoring.

In chapters 4 and 5 a multichannel measurement on a coupled machine system is regarded as a mixture of machine sources. We propose the use of *blind source separation* to reconstruct the signatures of individual machines or components. The methods for Independent Component Analysis and blind source separation (chapter 4) can be explained from a maximum likelihood, maximum entropy (another inductive principle that we have not addressed in this chapter) and a Bayesian perspective. In chapter 5 we describe a variant of ICA that is based on the MDL-principle. Next, an algebraic method for blind source separation with bilinear forms is described and investigated. Then, a method for convolutive unmixing of 2-channel mixtures is described. Finally, we investigate to what extent rotating machine sources can be separated blindly from their environment in three real-world case studies. Here we also address the question which mixing model should be assumed when separating acoustical or vibrational machine sources.

In chapter 6 we address methods for description of an unlabeled set of data. The objective is to perform *novelty detection* in situations where the number of data samples is small and is residing in a possibly high-dimensional feature space. Several methods are described and investigated for health monitoring (which features or channels are appropriate for submersible pump monitoring? how to combine information from several sensors?) and gas leak detection.

In chapter 7 we describe two approaches for dealing with the nonstationary nature of a machine dataset. The *dynamics of the system* can be modelled explicitly or merely be tracked. Modelling dynamics can be done with hidden Markov models; we investigate whether this approach is useful to segment time series from a gradually deteriorating gearbox into health regimes. We note that the *Self-Organizing Map* offers the possibilities of tracking, visualization of the (health) state space and novelty detection at the same time.

The feasibility of the learning approach to health monitoring is investigated in chapter 8, where *four real-world monitoring case studies* are presented (pump monitoring, gas leak monitoring, medical monitoring and monitoring of an operational machine in a pumping station). Here we also investigate the robustness of a monitoring system to repeated measurements, i.e. to what extent overtraining to one particular measurement session takes place. Finally, a tool for practical health monitoring called MONISOM is described. The use of MONISOM is illustrated in a case study where a progressing fault in the submersible pump is tracked.

In the concluding chapter 9, we summarize our findings and provide an evaluation of the learning approach to health monitoring.

Part II

Learning methods for machine vibration analysis

Chapter 3

Temporal feature extraction

In this chapter, we describe methods to characterize a mechanical structure (the subtask "feature extraction" in the scheme of figure 2.4). This can be done by modelling the structure itself or by modelling the vibration on the machine casing. In the former approach one addresses the whole system, so an *input-output* (i/o) description is appropriate. In the latter approach, the *response* of the structure to some excitation is measured only. Large deviations in the modal or vibrational description are regarded as indicative of faults (see also chapter 6). From chapter 1 we know that several heuristics exist to diagnose the origin or position of faults based on (deviations in) these descriptions. In effect, our aim is to find a proper description of machine condition that can be used afterwards for fault detection and diagnosis. We introduce the basics of modal modelling and testing [HP86] and will show the use of these techniques in the modal characterization of the test rig of section 1.4. We also introduce the 'classic' ways to characterize the vibration on rotating mechanical structures ("feature extraction") and then proceed to *learning* methods for feature extraction. This includes parameter estimation in models for machine vibration and extraction of invariant temporal features of the signals. Also, we review methods that focus on specific features of machine vibration, like cyclostationarity and transient events.

3.1 Mechanical system identification

Static mechanical systems are usually modelled as a set of coupled idealized springs, masses and dampers that behave in a linear fashion (e.g. viscous damping if damping is proportional to velocity). The relation between response displacement $x(t)$ of a single degree-of-freedom (SDOF) system and an excitation force $f(t)$ at time t is

$$f(t) = m\ddot{x}(t) + c\dot{x}(t) + kx(t) \tag{3.1}$$

where m denotes the mass, k the spring constant and c the damping coefficient in the system. Two parameters are important for describing the behaviour of the system: natural frequency and damping factor. We rewrite equation (3.1) as

$$\ddot{x}(t) + 2\omega_n \xi \dot{x}(t) + \omega_n^2 x(t) = \frac{1}{m} f(t) \tag{3.2}$$

where the *undamped natural frequency* ω_n of the system is

$$\omega_n = \sqrt{\frac{k}{m}} \tag{3.3}$$

and the *damping factor* ξ is

$$\xi = \frac{c}{2\sqrt{km}} \tag{3.4}$$

Moreover, the *damped natural frequency* ω_d of the system is given by [LS97]

$$\omega_d = \omega_n \sqrt{(1-\xi^2)} \tag{3.5}$$

If no excitation is applied (i.e. $f = 0$), the roots of the Laplace transformation of the homogeneous differential equation corresponding to (3.1) are

$$s_{1,2} = -\sigma + j\omega_d \tag{3.6}$$

which corresponds to the time domain solutions $e^{s_{1,2}t}u(t) = e^{-\sigma t}e^{j\omega_d t}u(t)$. Here, $u(t)$ is the step function and the coefficient σ is called the *damping rate* of the system. It determines the speed with which the oscillation of the system decays. For structural dynamics applications only the underdamped case $\xi < 1$ is important (at either critical damping or overdamping, the system experiences no oscillation). If the excitation force is unequal zero, the system will exhibit its *frequency response*. Since this frequency response function (FRF) is a complex quantity, it may be presented in polar coordinates (separate phase and magnitude responses).

Multiple degrees of freedom and mode shapes

More complex mechanical systems can be modelled as a set of coupled spring-mass-damper systems with n degrees of freedom (MDOF systems, systems with multiple inputs and outputs). The SDOF model (3.1) is now extended to its multidimensional equivalent

$$\mathbf{f}(t) = \mathbf{m}\ddot{\mathbf{x}}(t) + \mathbf{c}\dot{\mathbf{x}}(t) + \mathbf{k}\mathbf{x}(t) \tag{3.7}$$

where boldface variables denote n-dimensional vectors. The roots of the free vibration (no excitation) equation again lead to the modal parameters (dampings ξ_i and eigenfrequencies ω_i), and with MDOF systems the *mode shapes* $\phi_i, i = 1,\ldots,n$ have to be taken into account as well. According to [LS97], formula (3.7) can be simplified by writing $\mathbf{x}(t)$ in terms of the eigenvectors $\phi = [\phi_1,\ldots,\phi_n]$ of the matrix $\mathbf{m}^{-1}\mathbf{k}$,

$$\mathbf{x}(t) = \phi\mathbf{z}(t), \qquad \mathbf{m}^{-1}\mathbf{k}\phi = \phi\lambda \tag{3.8}$$

Equation (3.7) can now be rewritten as

$$\gamma^{-1}\phi^T\mathbf{f}(t) = \ddot{\mathbf{z}}(t) + \beta\dot{\mathbf{z}}(t) + \lambda\mathbf{z}(t) \tag{3.9}$$

If β is diagonal, this is a system with uncoupled modes. In this case, we can define the jth modal frequency and damping as $\omega_j^2 = \lambda(j,j), 2\xi_j\omega_j = \beta(j,j)$, and equation (3.9) can be written as

$$\frac{1}{\gamma(j,j)}\sum_{i=1}^{n}\phi(i,j)\mathbf{f}_i(t) = \ddot{\mathbf{z}}_\mathbf{j}(t) + 2\xi_j\omega_j\dot{\mathbf{z}}(t) + \omega_j^2\mathbf{z}_j(t) \tag{3.10}$$

For this situation, the MDOF system reduces to a set of uncoupled SDOF equations scaled by mode shapes ϕ_i. A mode shape is a unique displacement vector that exists for each frequency and damping, and it weights the *local* contribution of all modes to the output at a certain point (depending on the points of excitation and measurement). The mode shape as a whole is a global property of the system, just as the damping- and eigenfrequency-vectors (that are constant throughout the structure); however, each modal coefficient that contributes to the mode shape is estimated from a particular (*local*) measurement point. The mode shape expresses 'to what extent' analytically determined natural frequencies and dampings will be present in the measurements at a certain (machine) position. Experimental modelling of a mechanical structure can be done by analyzing the structure, components/ attachments, its mechanical material properties and designing a *finite element model* of the structure. This gives a prediction of the modal parameters of the structure (eigenfrequencies and mode shapes), which should be validated with measurements on the system.

Example 3.1: troubleshooting

For troubleshooting purposes, a *response model* (set of frequency response measurements acquired in a modal test) is usually adequate. In a modal test, the structure under investigation is excited in order to identify its frequency response. The excitation can be performed with a shaker (broadband random excitation) or with a hammer (burst excitation, an approximation to a delta pulse, which is also broadband). Note that the linear stationary system assumption implies that the response spectrum $Y(\omega)$ is related to the excitation spectrum $X(\omega)$ via the FRF $H(\omega)$ as

$$Y(\omega) = H(\omega)X(\omega) \tag{3.11}$$

The dynamics of the structure dictates the type of excitation. The identification procedure can be executed in two ways: either the response measurement point (usually a displacement or acceleration transducer) is fixed and the excitation position is varied along the structure or the excitation point is fixed and the response is measured along the structure. This means that identifying either a complete row or column in the frequency response matrix $H_{ij}(\omega), i,j = 1,\ldots,n$ (where the ij^{th} entry denotes the response from excitation source i to response point j) suffices for modal identification. Moreover, since the system is assumed linear, the frequency response matrix is symmetric. Deviations (i.e. asymmetry) can be used to detect nonlinearities. From a sufficient set of experimentally determined frequency response functions, the eigenfrequencies, dampings and (unscaled) mode shapes can be estimated. For troubleshooting purposes this usually suffices; for more detailed analyses, the mode shapes should be scaled to extract parameters like modal mass and stiffness. It is noted

in [ML95] that in machine structures the modal overlap is typically small, which in general makes so-called single-mode methods feasible. The parameters are then estimated by looking at the position of the maximum of the mode (the eigenfrequency) and the sharpness of the peak (the damping is related to the width of the peak between the half-power points). □

3.1.1 Experiment: modelling the submersible pump

The modal properties of the submersible pump described in chapter 1 were analyzed experimentally. The design of a finite-element (FEM) model was performed by RND Mechanical Engineering b.v. in Delft [Kei00]. The experimental *validation* of the FEM model was performed with a modal test: the non-operative pump was excited with broadband random noise and the reponse at 3 measurement positions was measured (see figure 3.1(a))[1]. The position

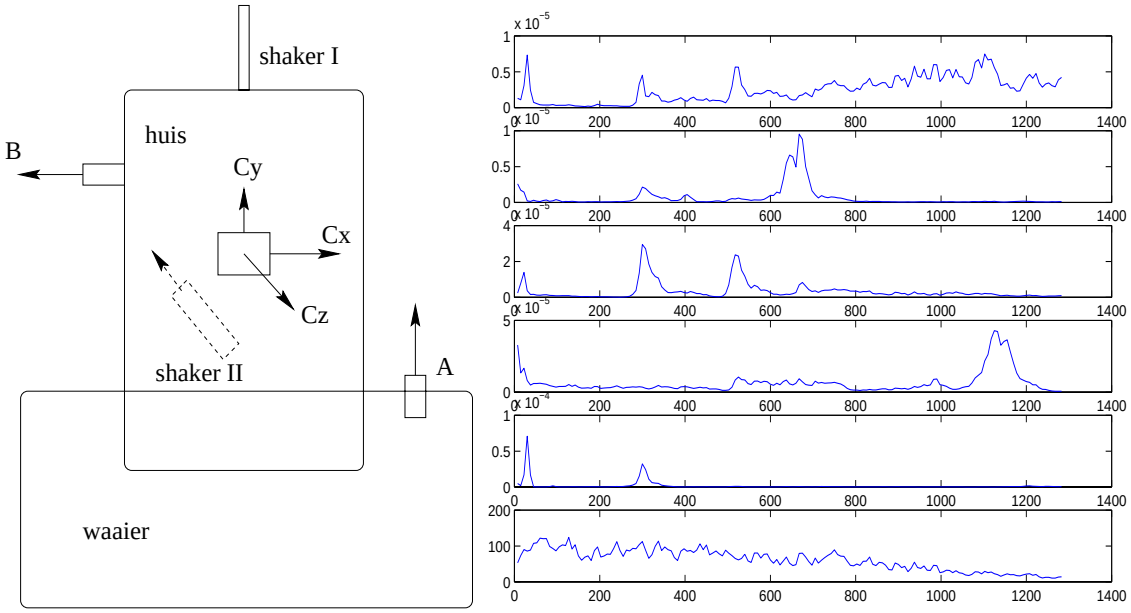


Figure 3.1: Experimental setup with shaker (a) and resulting response spectra (b)

of the shaker was varied. Putting the shaker on the side of the pump (position II in the figure) and setting the cut-off frequency for the excitation signal to 1000 Hz led to the response spectra shown in figure 3.1(b). The first five channels are response spectra (in the order: channel 1 = B, 2 = Cx, 3 = Cy, 4 = Cz, 5 = A), the last spectrum shown (channel 6) is the excitation signal spectrum. Hence, we identified *elements of a row* in the frequency response matrix. The measured eigenfrequencies up to 1200 Hz and the computed frequencies with a FEM model with two different sets of boundary conditions is given in table 3.1.1. *Model 1* is a FEM model that only incorporates constraints with respect to the attachment of the pump; *model 2* also incorporates constraints with respect to the pump outlet. The *measured eigenfrequencies* were determined by visual inspection of the spectra in figure 3.1(b), hence the accuracy is limited to about 20 Hz. We can see that the first eigenfrequency of models

[1] In this figure, 'huis' means pump casing and 'waaier' means vane

and measurement (21 vs. 25 Hz) correspond well. For higher modes there are larger differences, although the mode at 300 Hz is well-represented in both measurements and model. The frequencies are mainly somewhat shifted, which can be caused by a mass distribution in the models that differs from the real distribution. The first modes represent the " whole body motion", whereas material distortions are influencing the higher modes. Computed eigenval-

Table 3.1: Measured and computed eigenfrequencies for submersible pump

measured	model 1	model 2	meas.	mod-1	mod-2	meas.	mod-1	mod-2
25	**22**	**22**		616		**990**	**992**	
		42		627	627		1015	
		98		631				1031
	145		**640**	**635**	**647**		1044	1044
	229	228	**660**		**664**			1068
300	**295**	**305**		684				1072
330	**320**	**335**	**760**	**704**			1078	1076
	362	367	**820**	**809**			1094	
		389		840	838		1133	1133
	447	447			875	**1140**	**1139**	**1137**
530	**548**	**549**		895	895			1156
	552			959			1182	
	597	597		969			1193	1193

ues represent all possible eigenfrequencies that the system might possess. Depending on the measurement position and direction, certain modes will be present in the response spectrum and others will be less present in the spectrum (of even be absent). The results of the FEM models differ at places, but seem to resemble the main resonances in a consistent manner. The bold-faced entries in the table are plotted in figure 3.2. This indicates a fairly random scatter around the straight line with slope 1. Considering the fact that the number of measurement points was only three, along with the fact that there was a fairly large uncertainty in determination of the measured eigenfrequencies, the (locally) linear FEM model seems appropriate for the submersible pump.

3.2 Equispaced frequencies model

For modelling vibration from failures that cause increasing modulatory effects on a carrier frequency, Pajunen et al. [PJKS95] proposed the *equispaced frequencies* model

$$y(n) = \sum_{m=-M}^{M} A_m e^{j(2\pi f_m n + \phi_m)} + \varepsilon(n) \tag{3.12}$$

where $f_m = f_0 + m\Delta f$ are the locations of the fixed distance frequencies around the fundamental frequency f_0, $\varepsilon(n)$ is complex white noise and A_m and ϕ_m are the (constant) amplitude

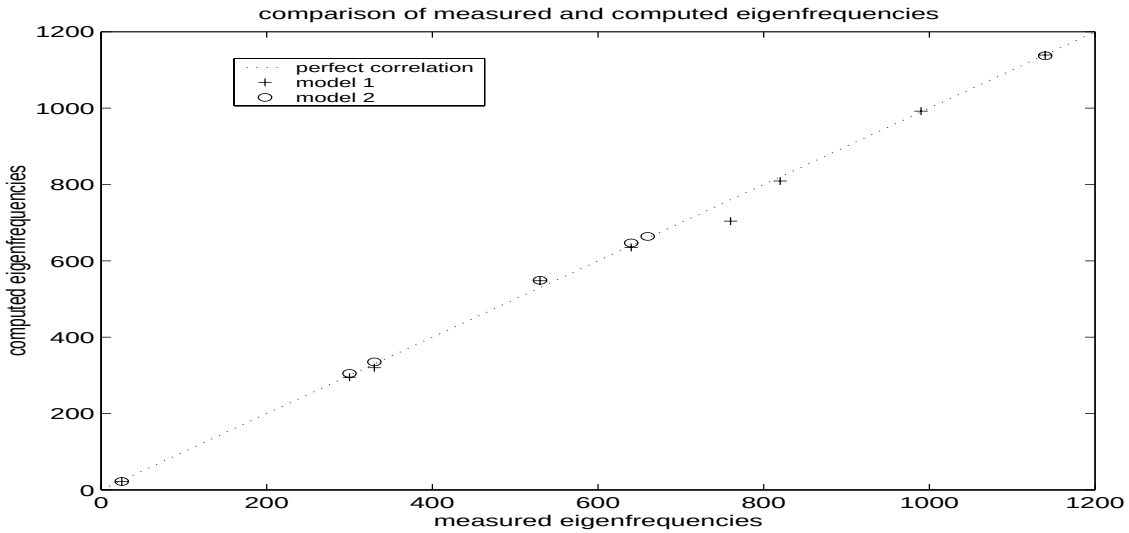


Figure 3.2: Comparison of measured and computed eigenfrequencies of submersible pump

and phase of the *m*th exponential. The time index *n* is taken to be discrete. Real signals can be incorporated by recalling that a real sinusoid at frequency f is the sum of two complex exponentials at frequencies f and $-f$. Hence the sum in (3.12) is summed over frequencies f and $-f$, leading to a cosine term in (3.12) instead of an exponential. Estimation of the parameters in (3.12) assumes knowledge about the fundamental frequency f_0. The amplitude of the rotation frequency component will probably be much larger than the amplitude of the sideband frequencies, so it can be estimated from the power spectrum [PJKS95]. Once the sideband frequencies spacing Δf is known, the amplitudes and phases can be estimated with least-squares methods. The sideband frequencies spacing can be estimated in two different ways, which will be explained in the sequel. With $E[.]$ we denote the expectation operator, whereas $(.)^T$ and $(.)^H$ denote transposition and conjugate transposition, respectively.

3.2.1 MUSIC parameter estimation

Consider the related model for sinusoidal frequencies in additive white noise [PM92]

$$y(n) = x(n) + \varepsilon(n) = \sum_{i=1}^{p} A_i e^{j(2\pi f_i n + \phi_i)} + \varepsilon(n) \tag{3.13}$$

where the noise process has variance σ_ε^2. The *data correlation matrix* R_{yy} is defined as

$$R_{yy} = E[\mathbf{y}\mathbf{y}^H] \tag{3.14}$$

where the expectation is taken over the delay vectors

$$\mathbf{y}_i = [y(i), y(i+1), \ldots, y(i+L-1)]^T \tag{3.15}$$

Define the *signal vectors* $\mathbf{e}_f$ by

$$\mathbf{e}_f = [1, e^{j2\pi f}, e^{j2\pi f 2}, \ldots, e^{j2\pi f(L-1)}]^T \tag{3.16}$$

The *signal correlation matrix* R_{xx} may be written as an expansion on a basis of signal vectors

$$R_{xx} = \sum_{i=1}^{p} P_i \mathbf{e}_{f_i} \mathbf{e}_{f_i}^H \tag{3.17}$$

which stems from the fact that the autocorrelation function of the harmonic series signal $x(n)$ can be expressed in terms of complex exponentials

$$\gamma_{xx}(\tau) = \sum_{i=1}^{p} P_i e^{j2\pi f_i \tau} \tag{3.18}$$

In this formula, $P_i = A_i^2$ is the power of the ith sinusoid. Furthermore, note that

$$R_{yy} = R_{xx} + \sigma_\varepsilon^2 \mathbf{I} \tag{3.19}$$

and that R_{yy} has rank L, while R_{xx} has rank $p < L$.[2] The *signal subspace* is defined as the space that is spanned by the p signal vectors $\mathbf{e}_{f_i}, i = 1, \ldots, p$. This subspace can be estimated by computing the eigenvectors of the signal correlation matrix

$$R_{xx} = \sum_{i=1}^{p} \lambda_i \mathbf{v}_i \mathbf{v}_i^H \tag{3.20}$$

which are an (alternative) orthonormal basis for the signal subspace[3]. From this the expression for the eigenvalue decomposition of the data correlation matrix becomes

$$R_{yy} = \sum_{i=1}^{p} (\lambda_i + \sigma_\varepsilon^2) \mathbf{v}_i \mathbf{v}_i^H + \sum_{i=p+1}^{L} \sigma_\varepsilon^2 \mathbf{v}_i \mathbf{v}_i^H \tag{3.21}$$

The signal subspace is spanned by the p principal eigenvectors, whereas the noise subspace is spanned by the eigenvectors with significantly smaller eigenvalues. In the MUSIC frequency estimation method, one computes spectral estimate $P(f)$ at discrete frequency f, by using the orthogonality of signal and noise eigenvectors:

$$P(f) = \frac{1}{\sum_{i=p+1}^{L} |\mathbf{e}_f^H \mathbf{v}_i|^2} \tag{3.22}$$

This expression tends to infinity when the signal vector at a certain frequency $\mathbf{e}_f$ belongs to the signal subspace. In practice, finite sample sizes lead to an expression that is very large (but finite) at the correct discrete frequencies. Note that expansion of the signal correlation matrix on a signal vector basis (3.17) only holds for (noisy) harmonic signals of the form (3.13). Therefore, MUSIC is only suitable for this type of signals.

[2] We assume that L is chosen sufficiently large, i.e. L cannot be smaller than p

[3] The basis is orthonormal, since the eigenvectors are assumed to be normalized, i.e. $\mathbf{v}_i^H \cdot \mathbf{v}_j = \delta_{ij}$

3.2.2 Maximum likelihood parameter estimation

For short datalengths, maximum likelihood (ML) estimation of the sideband frequencies spacing may be employed, since the maximization can be done in a one-dimensional parameter space (i.e. over Δf) [PJKS95]. By conjecturing Gaussian noise in (3.12), which is denoted for convenience as

$$\mathbf{y} = \mathbf{EA} + \boldsymbol{\varepsilon} \tag{3.23}$$

one can maximize the likelihood of $P(\mathbf{y} - \mathbf{EA})$ being a Gaussian probability density. This method relies on the normal distribution of the measurement errors $\boldsymbol{\varepsilon}$, i.e. $P(\boldsymbol{\varepsilon})$ denotes a Gaussian probability distribution. This leads to a likelihood function

$$L(f) = \mathbf{y}^T \mathbf{E}(\mathbf{E^T E})^{-1} \mathbf{E}^T \mathbf{y} \tag{3.24}$$

where the signal matrix $\mathbf{E}$ depends on the frequencies only.

Remark 3.2: determination of model order

Subspace methods in signal analysis rely on a proper estimate of the signal subspace, which implies that the dimensionality of the data correlation matrix L is chosen sufficiently large. The choice of the signal subspace dimensionality p can be done using well-known model order estimation criteria like Akaike Information Criterion (AIC) and Minimum Description Length (MDL) [LW93]. Explicit expressions for these criteria in a problem similar to sinusoidal frequency estimation are given in [PM92, WK85]. □

3.2.3 Experiment: feasibility of model-based methods

We estimated the sideband frequencies spacing with ML in an artificial signal

$$y(n) = \sum_{m=-5}^{5} A_m e^{j(2\pi f_m n + \phi_m)} + \varepsilon(n) \tag{3.25}$$

with $f_m = f_0 + m \cdot 0.199$ and A_m, ϕ_m and f_0 having some a priori known value. The likelihood as a function of the normalized frequency spacing[4] is plotted in figure 3.3. It is clear from the zoom plot in figure 3.3 that the correct sideband spacing is retrieved, provided one has prior knowledge about the approximate value of the sideband spacing. If prior knowledge is not available, one has to resort to the large-range likelihood plot, from which the correct frequency spacing cannot be estimated easily due to many spurious peaks.

For another signal (15 harmonics, normalized sideband spacing = 0.011), the likelihood plot (figure 3.4) shows a clear drawback of the method: when the number of harmonics is high, not only the target spacing will have high likelihood, but multiples of the sideband spacing as well. This can be understood by noting that a set of three equispaced harmonics can also be interpreted as one pair of harmonics with double frequency spacing. Moreover,

[4] The label on the horizontal axis should read "normalized frequency spacing"

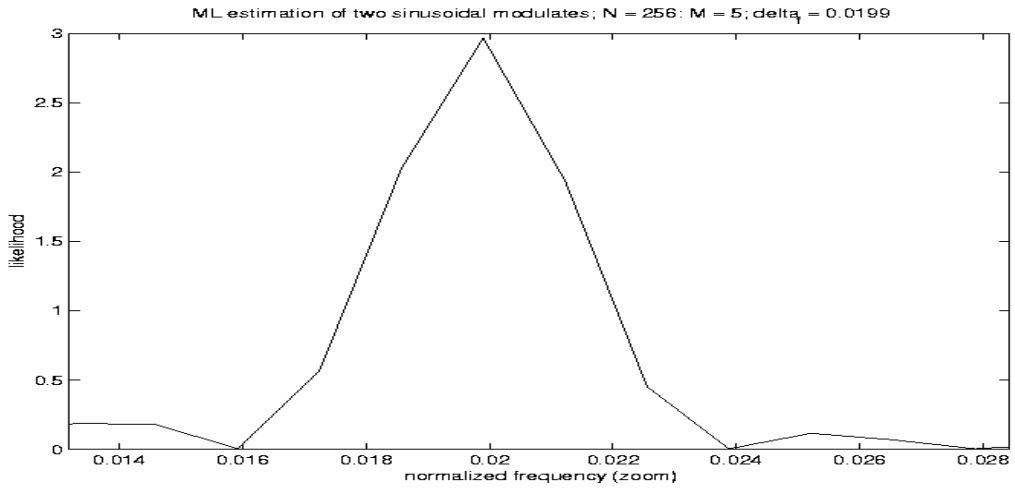


Figure 3.3: ML can be used for modulation frequency estimation if one has prior knowledge about range of the parameter: zoom plot of likelihood around correct frequency spacing

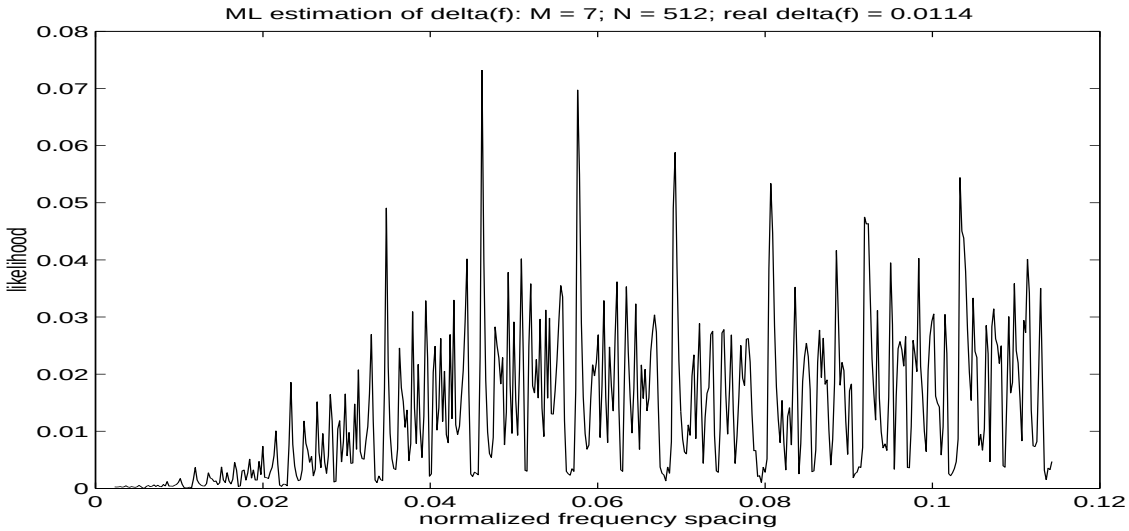


Figure 3.4: ML-estimation of frequency spacing in equispaced frequencies model can lead to peaks at multiples of correct frequency spacing

the likelihood function shows a lot of spurious peaks. Retrieval of the correct sideband spacing might be quite troublesome if one assumes a limited amount of prior knowledge about the modulating frequency.

3.3 Spectrum analysis

Machine vibration has particular characteristics, which can be exploited for diagnostics (chapter 1). Diagnostic information is often obtained by *spectrum analysis* or by using heuristics about features of fault signals.

3.3.1 Heuristic fault indicators

Typical indicators of machine health that have been used extensively by practitioners are the root-mean-square (RMS) value of the power spectrum, and the crest-factor of the time-domain vibration signal. The *RMS* value is just the average amount of energy in the vibration signal (the 'effective amplitude') The *crest-factor* of a vibration signal is defined as $\frac{A_{peak}}{A_{\mathrm{RMS}}}$, i.e. the peak amplitude value divided by the root-mean-square amplitude value[5]. This feature will be sensitive to sudden defect bursts, while the mean (or: RMS) value of the signal has not changed significantly. A technique especially designed for detection of impulsive phenomena is the *shock pulse method* (SPM). Here, one uses the fact that impulsive excitations will excite the mechanical structure with broadband energy. As the fault becomes more pronounced, the amount of energy transmitted to the machine is larger. At some point, there is enough energy to excite the resonance frequencies of the accelerometer that is used for system monitoring. Pronounced increase in the response of the accelerometer is now indicative of impulsive faults, and this can be used for diagnostics. However, the method is prone to false alarms, since many phenomena (not only impulsive faults) may contribute to the excitation of the accelerometer resonances.

3.3.2 Spectrum-based methods

In rotating or reciprocating machines, periodic force variations cause harmonics in the vibration signal. This can be estimated from the signal by spectral estimation. Several heuristics based on Fourier analysis of signals were mentioned in chapter 1.

Spectral estimation and autoregressive modelling

We will not review techniques for spectral estimation here, but refer the reader to [OWY83]. Typical design issues with these methods are the amount of spectral resolution required, available record length and choice of spectral smoothing technique (window choice; amount of overlap between consecutive segments). The obtained spectrum is always an *estimate* of the real spectrum, and spectral leakage (sidelobe generation due to the use of a time-window), aliasing and data lengths determine whether the content of a frequency component is meaningful and the confidence levels associated with a spectral component. We should remember that in the pattern recognition approach we take in this thesis, the amount of energy in a certain frequency bin can become a feature component in a dataset. Spectral leakage can smear the energy over nearby bins, which leads to smearing of activity over several features.

[5] In both cases one often uses the enveloped time signal. Envelope detection will be explained below

If this effect is not accounted for in the pattern recognition procedure (e.g. by assuming correlations between 'nearby features'), enough observations should be available to account for this source of variation between frequency bins.

Parametric spectra have been mentioned as an advantageous alternative to nonparametric (FFT-based) spectral estimation methods. They allow for higher resolution [MM92] and show robustness to the presence of point spectra [LSL93]. In the autoregressive (AR) modelling approach, an autoregressive process of order m is fit to a stationary signal Y_t:

$$Y_t - \alpha_1 Y_{t-1} - \ldots - \alpha_m Y_{t-m} = \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma^2) \tag{3.26}$$

In our pattern recognition approach, each model coefficient will now become a feature component. To avoid the overtraining effect, the order p of the AR-model should not be chosen unnecessarily high. Another remedy to the overfitting problem could be to reduce the dimensionality of the feature vectors obtained with autoregressive modelling, e.g. with Principal Component Analysis (see chapter 4). We remark that AR-model coefficients are insensitive with respect to scaling of a time signal with a constant c:

$$\begin{aligned} c \cdot \varepsilon_t &= c \cdot Y(t) - \alpha_1 \cdot c \cdot Y_{t-1} - \ldots - \alpha_m \cdot c \cdot Y_{t-m} \\ &\Leftrightarrow \\ \varepsilon_t &= Y(t) - \alpha_1 Y_{t-1} - \ldots - \alpha_m Y_{t-m} \end{aligned} \tag{3.27}$$

Cepstrum analysis

Consider the case of one vibration source and one sensor. Ignoring measurement errors, the measured response equals

$$x(t) = h(t) \star s(t) \tag{3.28}$$

with $s(t)$ the source signal and $h(t)$ the local impulse response due to the transmission path. Reconstruction of the source $s(t)$ from the measurement $x(t)$ amounts to a deconvolution. In reciprocating machinery the sources are usually broadband transients, whereas in rotating machinery they are typically narrowband and have durations comparable to the machine rotation period [ML95]. A way to separate the effects of transmission path and source is by using the *cepstrum*. The complex cepstrum of a signal $x(t)$is defined as [ML95]

$$x_{ccep} = \frac{1}{2\pi} \int_{-\infty}^{\infty} \log[X(\omega)] e^{j\omega t} d\omega \tag{3.29}$$

i.e. the inverse Fourier transform of the log spectrum of the signal. In the case that a signal is a filtered version of a source signal (as in equation (3.28)), this gives:

$$\begin{aligned} x_{ccep} &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\log[H(\omega)S(\omega)]e^{j\omega t}d\omega \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\log[H(\omega)]e^{j\omega t}d\omega + \frac{1}{2\pi}\int_{-\infty}^{\infty}\log[S(\omega)]e^{j\omega t}d\omega \\ &= h_{ccep} + s_{ccep} \end{aligned} \tag{3.30}$$

from which it is clear that the convolution in the time domain due to the transmission path appears as an additive effect in the cepstral domain. An application of deconvolution to musical instrument recognition is described in appendix B.

Envelope detection

Machine wear is often accompanied by increased modulations of harmonic components. In gearboxes, shaft misalignment and running speed variations cause modulation of (harmonics of) tooth meshing frequency with running frequency of driving and/ or driven shaft [Bri00]; however, in fault-free gears, these modulation phenomena are small in amplitude and bandwidth compared to faulty gears (section 1.2.2). In [ML96] it was stated that localised gear defects (fractures and incipient cracks) account for 90 % of all gear faults, based on a US Army investigation of 1976. Defects like eccentricity and local tooth faults will be visible in the vibration spectrum as increased modulation sidebands. This can be utilized in specialized signal processing methods. *Envelope detection* prior to Fourier transformation enhances evenly spaced sideband frequencies and harmonics. It consists of a two-stage process: (i) bandpass-filtering the data around the frequencies of interest (e.g. the region containing bearing resonances) and (ii) demodulating the data by applying the *Hilbert-transform*. Consider a real signal $x(t)$ that is obtained from the modulation of signal $s(t)$ with a carrier with frequency ω_c

$$x(t) = \cos(\omega_c t + \phi_c)\cdot s(t) \tag{3.31}$$

The Hilbert transform of the modulated signal $x(t)$ is expressed as

$$\mathscr{H}\{x(t)\} = \tilde{x}(t) = \frac{1}{\pi}\int_{-\infty}^{\infty} x(\tau)(\frac{1}{t-\tau})d\tau \tag{3.32}$$

In the frequency domain, this corresponds to

$$\mathscr{F}\{\tilde{x}(t)\} = X(\omega)\cdot\{-j\,\mathrm{sign}(\omega)\} \tag{3.33}$$

where $\mathscr{F}\{\cdot\}$ denotes the Fourier transform. We see that the Hilbert transform amounts to a 90^o phase shift. The *analytic signal* $x_a(t)$ corrresponding to the real signal $x(t)$ contains the same positive frequencies as the real signal, while the negative frequencies are removed. It is defined as $x_a(t) := x(t) + j\tilde{x}(t)$. For the particular signal $x(t)$ at hand, it holds that

$$\begin{aligned} x_a(t) &= \cos(\omega_c t + \phi_c) \cdot s(t) + j \cdot \sin(\omega_c t + \phi_c) \cdot s(t) \\ &= A(t) e^{j\phi(t)} \end{aligned} \tag{3.34}$$

Now the magnitude $A(t)$ of the analytic signal equals the (modulus of the) amplitude modulating signal, since

$$A(t) = \sqrt{\cos^2(\omega_c t + \phi_c) \cdot x^2(t) + \sin^2(\omega_c t + \phi_c) \cdot x^2(t)} = |x(t)| \tag{3.35}$$

Hilbert-transform demodulation of the ω_c-modulated signal $x(t) = \mathrm{Re}\{A(t)e^{j\phi(t)}\}$ hence comprises the following steps: **1.** bandpass filter around carrier frequency ω_c; **2.** calculate *analytic signal* $x_a(t) = x(t) + j\mathscr{H}\{x(t)\}$; **3.** transform to polar representation $A(t)e^{j\phi(t)}$; **4.** $A(t)$ = AM signal (i.e. signal *envelope*).

3.3.3 Experiment: envelope detection with the submersible pump

Measurements were obtained from the submersible pump of section 1.4 at three machine conditions: normal operation, imbalance and a bearing failure (an outer race flaw of approximately 1.5 mm), while the running speed of the machine was varied in the measurement procedure. Measurements from the triaxial accelerometer near the single bearing in the *axial* direction were selected for the experiments, since spectra from this channel differed most significantly for the three operating modes. The maximum frequency present in the discretized data was 20000 Hz. Using the heuristics from section 1.2.2 we chose the frequency band of interest as 500 - 3800 Hz. We performed a subband filtering using a wavelet transform (section 3.5.1) and retained the level 5 detail coefficients (1250 - 2500 Hz). After envelope detection around this subband, a spectrum containing many evenly spaced sidebands was observed in the measurements from the outer race defect, whereas it was absent in the other two operating modes (figure 3.5). This is clearly an indication that significant modulatory activity is present in vibration from bearings with an outer ring fault; for inner ring faults this effect is expected to be even more pronounced (see chapter 1).

3.4 Nonlinearity in machines

The transfer of vibration in mechanical structures is usually modelled linearly. The measured signals are often expressed in terms of their spectral (or equivalent: second-order correlation) structure. Uncertainties, model errors and measurement noise are usually modelled as a Gaussian contribution (following the *central limit theorem* and the fact that the sources of vibration and imperfections that are not modelled will be of small magnitude). However, nonlinear effects in the transfer from source to sensor are often present and sometimes lead to very significant model mismatch. As a consequence, it has been stated that the measured signals are more accurately modelled using their higher-order statistics (e.g. with bispectra

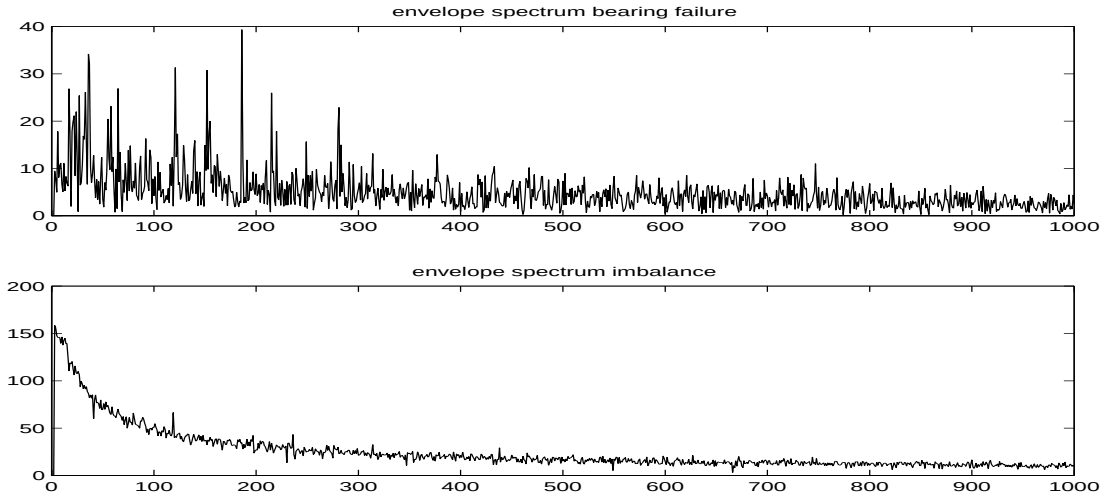


Figure 3.5: Envelope spectra with (upper) and without (lower) impulsive excitations

or trispectra). If the mechanical system under investigation exhibits linear time-invariant behaviour, it will show similar response to different intensities of the excitation. This is in contrast with nonlinear systems, where the response is dependent on excitation force [CPSY99]. Changes appear as frequency shifts or altering mode shapes. It was reported in several places that the higher-order statistics (HOS) of machine signals can be used for diagnostic purposes [MP97, McC98, Gel98]. Extraction of HOS can be done in two ways: *a.* by computing the higher-order moments of the distribution of the signals, and *b.* by computing the higher-order spectra of the signals. We will not describe the second set of methods and refer the reader for details on higher-order spectra to the above literature.

Higher-order moments and phase-coupling

We repeat that the *n*-th order moment $M_n(X)$ of a real stochastic variable X equals the expectation of its *n*-th order power $E[X^n]$. Well-known are the *mean* (1^{st} order moment) and *variance* (2^{nd} order central moment). *Skewness* is the 3^{rd} order central moment of a distribution. It reflects the asymmetry of the distribution, e.g. if there are significantly more large amplitudes in the signal than small amplitudes. *Kurtosis* is the 4^{th} order central moment of a distribution, measuring the 'peakedness' of a distribution. Signals with a Gaussian distribution will have a normalized kurtosis of 0; the normalization consists of a subtraction of 3 from the unnormalized quantity. Signals with heavy-tailed distributions (e.g. impulsive signals) will show larger values.

Nonlinear interaction between harmonic components may cause *phase coupling*. Quadratic phase coupling [SMN98] (coupling at sum and difference frequencies) occurs in a system with a square transfer function (i.e. a nonlinear transfer function). Three harmonics with frequencies ω_k and phases ϕ_k, $k = 1,2,3$ are said to be quadratically frequency coupled if $\omega_3 = \omega_1 + \omega_2$ and quadratically phase coupled if $\phi_3 = \phi_1 + \phi_2$. Often these two types of cou-

plings accompany each other. In machines, nonlinear effects of this type have been reported in [MP97].

3.4.1 Experiment: nonlinear behaviour of the submersible pump?

In this experiment, a shaker was mounted on top of the submersible pump in the test bench. The energy in the shaker-induced excitation signal was set to 0 dB and the response at three positions (5 measurement channels) was measured, figure 3.6. In this figure, the sixth

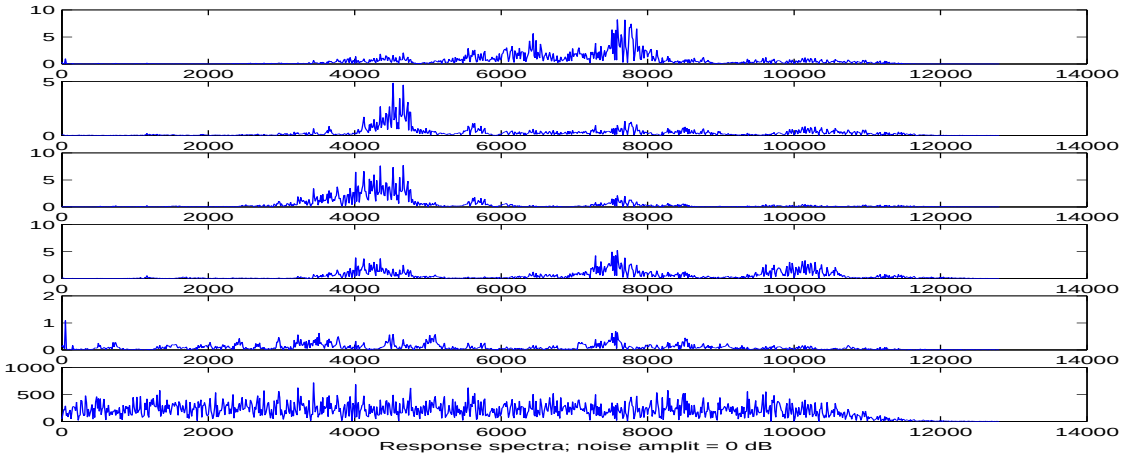


Figure 3.6: Response of submersible pump to excitation with 0 dB power noise. The first five spectra are measurements, the last spectrum is the excitation spectrum

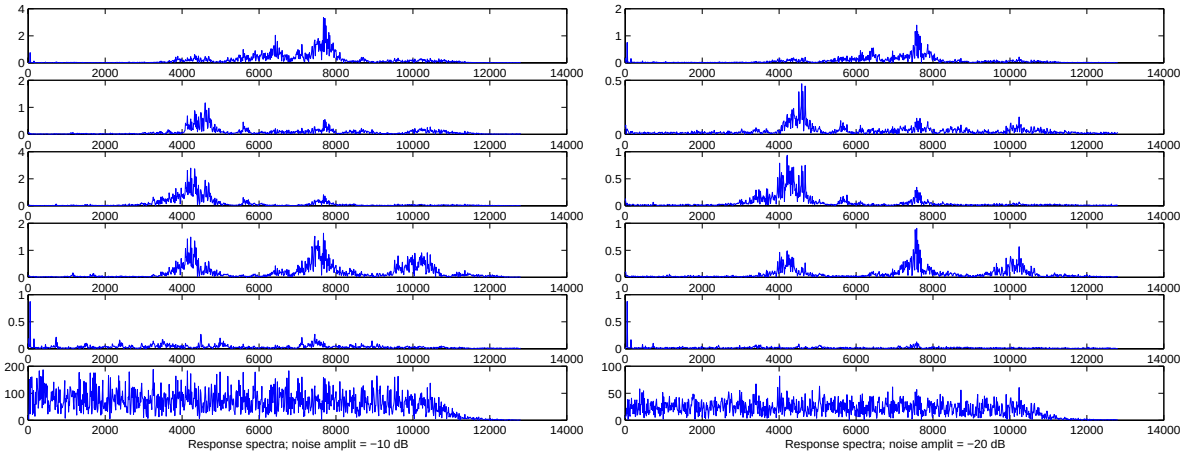


Figure 3.7: Response of submersible pump to excitation with -10 dB (a) and -20 dB (b) power noise. The first 5 spectra are measurements, the last spectrum is excitation

spectrum is the excitation spectrum. Next the energy of the excitation was decreased to -10 dB and -20 dB. The responses (figure 3.7) show no marked differences in modal position upon visual investigation (all modes remain at the same frequencies, with comparable relative

amplitudes), from which we conclude that the system shows reasonably linear behaviour with respect to various excitation intensities. A more thorough test for nonlinear behaviour would be the identification of the complete frequency response matrix (section 3.1) and then detecting possible asymmetries in this matrix, but that test was omitted.

3.5 Analysis of nonstationary machine signals

The stationarity assumption that is widespread in all time series analysis methods may not hold at the within-rotation scale: due to variable loading conditions or faults the signal that is measured during one shaft rotation may be severely nonstationary. From the time-frequency characteristics of such a signal, valuable information about machine or component condition can be extracted.

There have been several claims that incipient and developing faults in machine components can better be characterized in the time-frequency domain [LB97, SWT97, Oeh96]: short-duration transient effects due to intermittent or transient vibration give rise to sudden and brief changes in signal amplitude or phase; this will hardly be visible in the spectrum, because the energy is dispersed in the temporal averaging process. The *spectrogram* (i.e. a time windowed Fourier transform) can be used for time-frequency analysis (*tf-analysis*), but has the disadvantage of either poor frequency resolution or poor temporal resolution. Examples of machine faults that give rise to nonstationary vibratory events are: local gearbox faults (e.g. surface wear spalling, bad tooth contact, cracked or broken tooth [SWT97]), wearing plungers in a cam-operated pump and damaged flutes in a drill [LB97].

A typical feature of rotating (and also reciprocating) machines is the presence of stochastic events that are repeated with a certain frequency. It is known [McC98] that once-per-rotation pulses of noise with random energy (e.g. as the result of a bearing fault) can be modelled as band-limited noise modulated with a periodic rectangular pulse. This is a signal that exhibits *cyclostationarity* (see below). Another example is spalling in gearboxes [CSL00]. Due to load effects and deviation of teeth from their ideal shapes a distinct gear mesh signal will be present in every gearbox. This signal is a function of load, gearmesh frequency and (functions of) angular position of the wheels. The latter parameter (angular position) is a linear function of time if the rotation speed is constant; however, small speed fluctuations will usually be present, which cause phase randomization of the harmonic components of the signal. The resulting vibration signal exhibits (first- and second-order) cyclostationarity.

3.5.1 Time-frequency distributions and wavelets

The *short-time Fourier transform* (STFT) of a signal $s(t)$ is written as

$$STFT_s(\tau,\omega) = \int_{-\infty}^{\infty} s(t)w^*(t-\tau)e^{-j\omega t}dt \tag{3.36}$$

where ω denotes the frequency in a window $w(t)$ around $t = \tau$. The *spectrogram* is the squared magnitude of the short-time Fourier transform of signal $s(t)$:

$$P_{\text{spectrogram,s}}(\tau,\omega) = |\int_{-\infty}^{\infty} s(t)w^*(t-\tau)e^{-j\omega t}dt|^2 \tag{3.37}$$

where $w(t)$ is again a window function. The spectrogram is an example of a *time-frequency distribution* that does not represent a proper density function. It is however possible to find a set of time-frequency distributions that can "satisfy the marginals"; this means that the marginal densities are obtained by summing the joint density of one of the variables *time* or *frequency*. The energy density spectrum can be found as $\int_{-\infty}^{\infty} P(\tau,\omega)d\tau = P(\omega)$ and the instantaneous energy is given by $\int_{-\infty}^{\infty} P(\tau,\omega)d\omega = P(\tau)$. These two expressions constitute the constraints for obtaining proper energy *distributions* over time and frequency. A general class of time-frequency distributions that can satisfy the marginals is the class of Cohen-Posch time-frequency distributions. In the general formulation of Cohen [Coh95] a joint time-frequency distribution (TFD) can be written as

$$P_{\text{tfd,s}}(\tau,\omega) = \frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} s(u+t/2)s^*(u-t/2)\phi(\theta,t)e^{-j\theta(\tau-u)-j\omega t}dudtd\theta \tag{3.38}$$

The choice of the kernel function $\phi(\theta,t)$ determines the properties of the time-frequency distribution. The *Wigner distribution* is obtained if the kernel is chosen equal to identity, $\phi(\theta,t) = 1$, since

$$\begin{aligned}
P_{\text{wv,s}}(\tau,\omega) &= \frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} s(u+t/2)s^*(u-t/2)\cdot 1\cdot e^{-j\theta(\tau-u)-j\omega t}dudtd\theta \\
&\quad \{\ \text{grouping}\ \} \\
&= \frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} s(u+t/2)s^*(u-t/2)e^{-j\omega t}\{\int_{-\infty}^{\infty} e^{j\theta(u-\tau)}d\theta\}dudt \\
&\quad \{\ \int e^{-j\theta\tau}e^{j\theta u}d\theta = \mathscr{F}^{-1}\{\mathscr{F}\{2\pi\cdot\delta(u-\tau)\}\}\ \} \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} s(u+t/2)s^*(u-t/2)e^{-j\omega t}\delta(u-\tau)dudt \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} s(\tau+t/2)s^*(\tau-t/2)e^{-j\omega t}dt
\end{aligned} \tag{3.39}$$

The spectrogram can also be obtained from the general formulation. If a kernel is chosen that is functionally independent of the signal under consideration, the TFD cannot satisfy the marginals for all signals [LB97]. These TFD's are called *bilinear* distributions if the signal is included in the distribution formula (3.38) in a bilinear (i.e. quadratic) manner. In the Cohen-Posch distribution, the kernel is functionally dependent on the signal. Distributions that satisfy the marginals allow the analyst to compute relevant statistics like duration and bandwidth (i.e. standard deviations in time and frequency) from the distributions.

From the previous analysis it can be seen that windowing the Fourier transform yields a constant bandwidth analysis of the input data (i.e. the spectrogram). Determining signal energy at a certain time and frequency is subject to the *uncertainty principle* [Coh95], which

expresses that two interrelated variables cannot be measured simultaneously at arbitrary accuracy. The STFT of equation (3.36) leads to a *time-frequency window*[6] with boundaries $[\mu_t + \tau - \sigma_t, \mu_t + \tau + \sigma_t]$ and $[\mu_f + \omega - \sigma_{\hat{f}}, \mu_{\hat{f}} + \omega + \sigma_{\hat{f}}]$, where μ_t, μ_f and $2\sigma_t, 2\sigma_{\hat{f}}$ are the center and width of the time-frequency window in both domains (see figure 3.8(a)). Note that width $2\sigma_t$ and height $2\sigma_f$ of the windows are constant, independent of the position center of the window, which is undesirable when the objective is to obtain spectral information about limited time intervals in various frequency bands. In order to achieve accurate

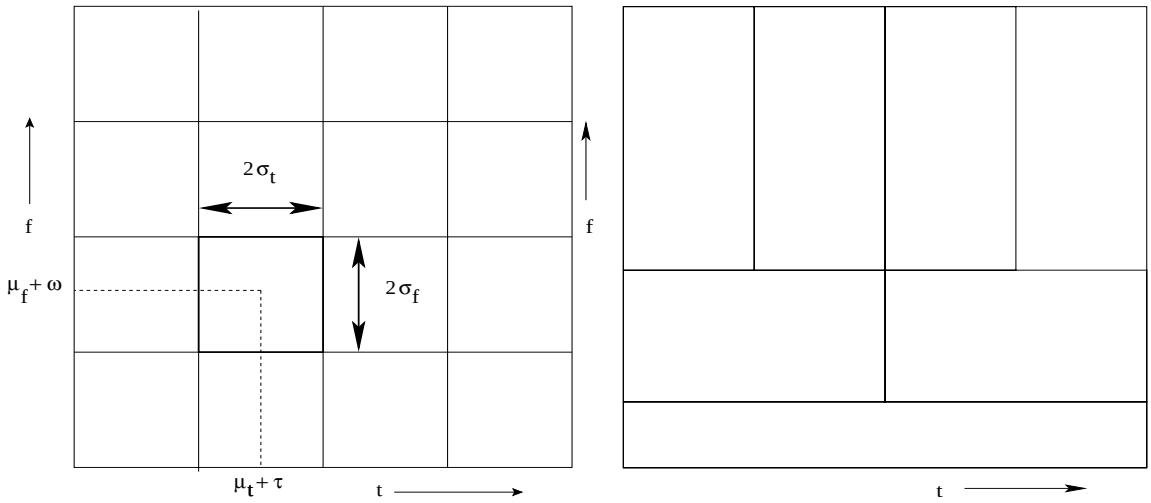


Figure 3.8: Tiling of the tf-plane with the STFT (a) and the wavelet transform (b)

high-frequency information, the time interval should be short, and vice versa. This can be established by introducing the *Continuous Wavelet Transform*

$$W_{s,\psi}(\tau, a) = \int_{-\infty}^{\infty} s(t)\psi_{a,\tau}(t)dt, \qquad a \in \mathbb{R}^+, \tau \in \mathbb{R} \tag{3.40}$$

where $\psi_{a,\tau}(t) = \frac{1}{\sqrt{|a|}}\psi(\frac{t-\tau}{a})$ is a set of functions, called *wavelets*, obtained from a basic wavelet $\psi \in L^2(\mathbb{R})$ by dilation (i.e. scaling) and translation[7]. When ψ and its Fourier transform $\hat{\psi}$ are both window functions, the continuous wavelet transform (3.40) of a signal f gives localized information in $[\tau + a\mu_{\psi,t} - a\sigma_{\psi,t}, \tau + a\mu_{\psi,t} + a\sigma_{\psi,t}]$ and $[\frac{\mu_{\psi,f}}{a} - \frac{1}{a}\sigma_{\psi,f}, \frac{\mu_{\psi,f}}{a} + \frac{1}{a}\sigma_{\psi,f}]$. Now, the width of the frequency window decreases for increasing scale parameter a, and vice versa (see figure 3.8(b)). Since the ratio of center-frequency to bandwidth is constant, the wavelet decomposition gives rise to a constant percentage bandwidth analysis, which is a much practiced technique for machinery vibration analysis [Ran87].

[6] Parts of time and frequency axes where signal s and Fourier transform $\hat{s}$ achieve most significant values

[7] $L^2(\mathbb{R})$ is the space of square integrable functions on the real line, i.e. functions $f(x)$ such that $\int |f(x)|^2 dx < \infty$. The basic wavelet has 'a wave character' since it must satisfy $C_\psi = \int_{-\infty}^{\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty$

3.5.2 Cyclostationarity

A real random signal $x(t)$ exhibits cyclostationarity at order n if its time domain nth order moment is a periodical function of time t [CSL00]. The fundamental frequency α of the periodicity is called the *cyclic frequency* of the signal. The synchronous average of a signal with respect to a period T is

$$E_\alpha[x(t)] = \frac{1}{N}\sum_{k=0}^{N-1} x(t+kT) \tag{3.41}$$

The **first-order** cyclic moment is the discrete Fourier transform of the synchronous average. Stated differently, the mean (the first-order moment) is periodical with T:

$$m(t) := E[x(t)] = m(t+T) \tag{3.42}$$

A signal is **second-order** cyclostationary if its autocorrelation function

$$R_{xx}(t,\tau) = E[x(t+\tau/2)x(t-\tau/2)] \tag{3.43}$$

is periodic in T, i.e.

$$R_{xx}(t,\tau) = R_{xx}(t+T,\tau) \tag{3.44}$$

Note that plain (second-order) stationarity would mean that $R_{xx}(t,\tau) = R_{xx}(\tau)$ is independent of t. The fundamental parameter for second-order cyclostationarity is called the *cyclic autocorrelation function*

$$R^\alpha_{xx,\text{real}}(\tau) = E[x(t+\tau/2)x(t-\tau/2)]e^{-j2\pi\alpha t} \tag{3.45}$$

which can be interpreted as an autocorrelation function with cyclic frequency α. For $\alpha = 0$ this equals the standard (stationary) autocorrelation function. For complex signals, the proper expression for the cyclic autocorrelation function is

$$R^\alpha_{xx,\text{complex}}(\tau) = E[x^*(t+\tau/2)x(t-\tau/2)]e^{-j2\pi\alpha t} \tag{3.46}$$

where * denotes complex conjugation. We see that $R_{xx,\text{real}} = R_{x^*x,\text{complex}}$. A signal with a cyclic autocorrelation function (3.46) or (3.45) is called *spectrally self-coherent at frequency separation* α and *spectrally* conjugate *self-coherent at frequency separation* α, respectively.

Example 3.3: self-coherence and cyclostationarity

Signals that exhibit *spectral (conjugate) self-coherence* are cyclostationary signals [ASG90]. This is illustrated in figure 3.9. An f_0 amplitude-modulated bandlimited analytic signal is correlated with its conjugate that is shifted in frequency over $2 \cdot f_0$. Here, f_0 denotes the carrier frequency of the modulation signal. □

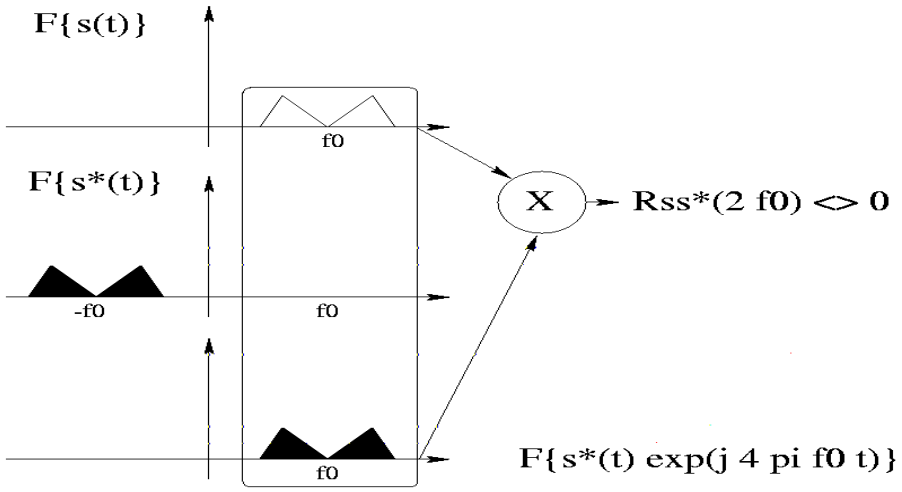


Figure 3.9: Self-coherence in AM-signals, from [ASG90]

3.5.3 Experiment: determination of self-coherence in modulated noise

We generated an analytic AR(1)-signal [BAMCM97], i.e. a signal generated with a first-order autoregressive model to which a complex part was added with the Hilbert transform. The noise model was Gaussian white noise. This signal was modulated with a complex-exponential carrier at (normalized) frequency f_0. As stated above, this signal exhibits cyclostationarity at cyclic frequency $\alpha = 2 \cdot f_0$, since the signal is (conjugate) self-coherent with a $2 \cdot f_0$ frequency-shifted version of its complex conjugate [ASG90]. The self-coherence of a cyclostationary signal (with $f_0 = 0.4$) as a function of the frequency-shift that is applied to the signal's conjugate is shown in figure 3.10(a). Clear peaks in the self-coherence are present at a shift of $2 \cdot f_0$, and deviations from this frequency shift result in pronounced loss of coherence. Note that the periodicity of $2.5 \cdot f_0$ is caused by the sampling process (where the sampling frequency f_s is normalized to 1, which corresponds to $1/0.4 = 2.5 \cdot f_0$).

The process was repeated on a real-AM real AR(1) signal. The (plain) self-coherence as a function of the frequency shift is plotted in figure 3.10(b). In the real case, the spectrum of the sampled real AR(1) source is symmetric around the origin (DC) and around multiples of the sampling frequency. These symmetries lead to the following. Since the spectrum of a sampled cosine (a real exponential) of frequency f_0 consists of symmetric peaks at $-f_0$ and f_0 and symmetric peaks around multiples of of the sampling frequency [OWY83], we see peaks of maximal coherence at multiples of the sampling frequency $k \cdot f_s, k \in \mathbb{Z}$ (i.e. at multiples of $2.5 \cdot f_0$). If the signal is shifted over $2 \cdot f_0$ (the "self-coherence symmetry"), only half of the spectrum of signal and shifted signal match, which explains the coherence peaks with halved magnitude at the values $2 \cdot f_0 + k \cdot f_s$ (since after shifting with $2 \cdot f_0$ an additional shift with multiples of f_s leads to the same coherence magnitude). The same holds for an initial shift with $0.5 \cdot f_0 + k \cdot f_s$, since the initial shift can be generated with $-f_s + 2 \cdot f_0 = 0.5 \cdot f_0$.

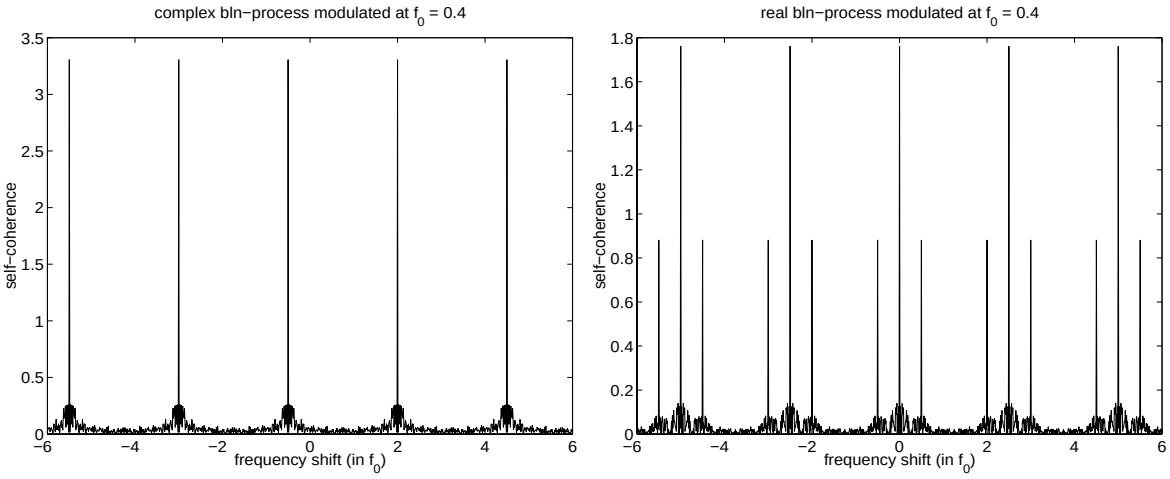


Figure 3.10: Self-coherence as a function of frequency-shift in (a) complex analytic AM-AR(1)-process and (b) real AM-AR(1)-process

3.6 Learning temporal features with the ASSOM

We conclude this chapter by studying a learning method for temporal feature extraction that does not need an explicit choice of feature to be made, the Adaptive Subspace Self-Organizing Map (ASSOM). The method uses correlation as an implicit feature. In the original Self-Organizing Map (SOM) architecture (for a more detailed description see section 6.3.1), an elastic net of locally connected nodes is trained to represent the input data, employing both a vector quantization and dimension reduction, while retaining the topology of the input space as much as possible. When each node consists of a *subspace* (as in the ASSOM), a map detecting invariants of the input data in a self-organized manner emerges [KKL97]. The ASSOM uses the notion that consecutive segments of a signal reside in a common subspace. This implicitly uses the notion of *delay coordinate embedding* of a time series (for an example, see appendix C).

3.6.1 Translation invariance

Consider a sine wave $x(\omega t) = \sin(\omega t)$. This signal can be represented in a delay coordinate embedding, where the one-dimensional signal is transformed into a set of L-dimensional vectors by using delay vectors:

$$\mathbf{x}(\omega t + \phi) = [\sin(\omega t - \tau + \phi), \sin(\omega t - 2\tau + \phi), \ldots, \sin(\omega t - L\tau + \phi)]^T \tag{3.47}$$

We can express a delay vector of the sine wave as

$$\mathbf{x}(\omega t + \phi) = \beta_1(\phi)\mathbf{x}(\omega t + \phi_1) + \beta_2(\phi)\mathbf{x}(\omega t + \phi_2),\ \forall \phi, t \tag{3.48}$$

since it holds that a sinusoid with arbitrary frequency and phase can be generated by taking a linear combination of two fixed-phase sine waves with the same frequency:

$$
\begin{aligned}
\cos(\omega t+\theta) &= \cos\omega t\cos\theta-\sin\omega t\sin\theta, \qquad \forall\theta,t \\
&\Leftrightarrow \{\phi=\theta-\frac{1}{2}\pi\} \\
\sin(\omega t+\phi) &= \cos(\phi+\frac{1}{2}\pi)\cos\omega t-\sin(\phi+\frac{1}{2}\pi)\sin\omega t, \qquad \forall\phi,t \\
&\Leftrightarrow \{\beta_1(\phi)=\cos(\phi+\frac{1}{2}\pi),\beta_2(\phi)=-\sin(\phi+\frac{1}{2}\pi)\} \\
\sin(\omega t+\phi) &= \beta_1(\phi)\cos\omega t+\beta_2(\phi)\sin\omega t, \qquad \forall\phi,t \\
&\Leftrightarrow \{\phi_2=0,\phi_1=\phi_2-\frac{1}{2}\pi\} \\
\sin(\omega t+\phi) &= \beta_1(\phi)\sin(\omega t+\phi_1)+\beta_2(\phi)\sin(\omega t+\phi_2), \qquad \forall\phi,t \qquad (3.49)
\end{aligned}
$$

We see that an arbitrary delay vector from a sinusoid can be represented as a weighted sum of two (90^o) phase shifted delay vectors from the same sinusoid. A delay vector can be considered as one point in an L-dimensional delay space. Taking several delay vectors into account corresponds to generating samples of a *trajectory* of the signal in the delay space. These samples are in a 2-D subspace of the L-D embedding space, since each element of the dataset can be written as a sum of two particular samples (vectors) in the delay space, $\mathbf{x}(\omega t+\phi_1)$ and $\mathbf{x}(\omega t+\phi_2)$. In other words, this basis spans the subspace that is closest (in least-squares sense) to the subspace in which the trajectory in delay space resides (or rather: both subspaces are the same). Note that the frequency of the sine wave is *retained* in the best-matching basis vectors, which can therefore be considered a translation-invariant feature of the signal.

For a general (natural) signal, the above derivations will not hold any more. However, it is known [AH92] that for highly correlated signals or images the Discrete Cosine Transform (DCT) approaches the Karhunen-Loève Transform (KLT $\equiv$ PCA, section D.1). The KLT is a linear transform that takes as basis vectors the eigenvectors of the signal autocorrelation matrix (i.e. it is a signal-adaptive basis). For harmonic signals, we know from section 3.2.1 that the KLT basis spans the same space as the signal vectors (3.16); for general correlated signals, the KLT basis spans the same subspace as a basis of sinusoids as well (the DCT basis). In the experiments reported in [KKL97], Gabor wavelet-like basis vectors emerged when training the ASSOM on either sinusoidal or natural images. The wavelets exhibited a Gaussian modulation, that was *imposed* beforehand by applying a Gaussian window to the data segments before entering the episode. Hence, it appears that the sinusoidal block-transform 'emerging' in ASSOM training can be explained by the fact that the ASSOM tries to find a subspace of the delay space that matches the trajectory of delay vectors in an episode best. This procedure is comparable to performing a PCA/KLT of an episode, especially when the basis vectors are orthogonalized during updating (as practiced in the ASSOM). Moreover, since the ASSOM consists of several connected subspaces, it can be expected that each node will ultimately focus on a subband of the signal spectrum; the topology preservation constraint will dictate an ordered pattern of subband filters on the map, as exhibited in [Koh95].

3.6.2 The ASSOM learning algorithm

A schematic drawing of an ASSOM is made in figure 3.11. Unlike the SOM, adaptation now

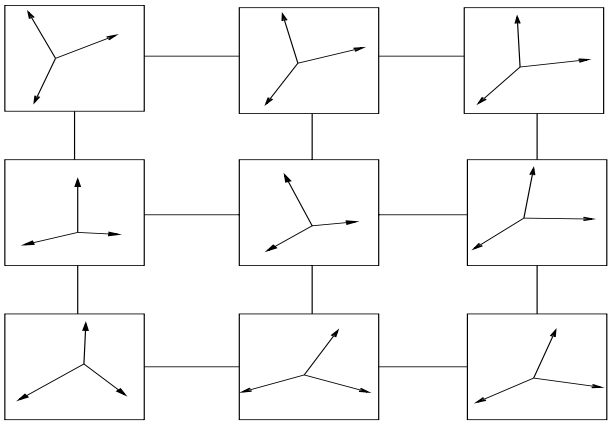


Figure 3.11: The ASSOM consists of topologically ordered subspaces, represented by interconnected sets of basis vectors

takes place with respect to *episodes* of the input data. An episode is a collection of locally shifted finite segments of the input data that contain common (e.g. translation-invariant) features. The winning node can be defined as the node that contains the subspace that has the highest number of matches (segments that are closest to this node) in the episode. Updating is done analogously to the original SOM: the winning node and its neighbourhood are driven towards the input of the ASSOM, in this case by rotating the target subspace towards the episode subspace

$$\mathbf{b}_h^{(i)}(t+1) = \alpha(t)\mathbf{b}_h^{(i)}(t), \quad \forall \mathbf{x}(t_p),\ t_p \in S(t) \tag{3.50}$$

where the update factor $\alpha(t)$ equals

$$\alpha(t) = I + \lambda(t) h_c^{(i)}(t) \frac{\mathbf{x}(t_p)\mathbf{x}(t_p)^T}{\|\hat{\mathbf{x}}^{(i)}(t_p)\|\|\mathbf{x}(t_p)\|} \tag{3.51}$$

In these formulas, we use the following definitions:

$\mathbf{b}_h^{(i)}(t)$ basis vector h of SOM node i at time t

$\mathbf{x}(t_p)$ segment in the episode with time instances $S(t)$, that starts at time t_p

$\|\hat{\mathbf{x}}^{(i)}(t_p)\|$ length of the projection of x onto the subspace of node i

$\lambda, h_c^{(i)}$ (time-dependent) learning rate and neighbourhood function, respectively

The ASSOM approach has the advantage that basis vectors representing features of the input data are formed automatically, i.e. without choosing an a priori basis (like in a wavelet analysis).

3.6.3 Experiment: machine health description using the ASSOM

We trained an ASSOM with 8x1 nodes, each consisting of 5 basisvectors of length 32, for 10000 cycles on episodes of vibration measurements from the submersible pump [YLFD97]. In order to check whether certain parts of the map were being tuned to invariants from a certain class, we calibrated the map by constructing episodes out of the measurements used to train the map and measurements at a different machine running speed (not included in the ASSOM training data). The histograms of the units that produced the best match with the segments in an episode are shown in figure 3.12. Upper left denotes imbalance, upper right normal operation, lower left bearing failure and lower right bearing failure at a different running speed. Similar behaviour was observed when projecting time series from different running speeds (not used in training the ASSOM). A majority voting mechanism can hence be used to calibrate the map, and classification may be performed on the basis of a representative set of episodes from a *raw* collection of measurements.

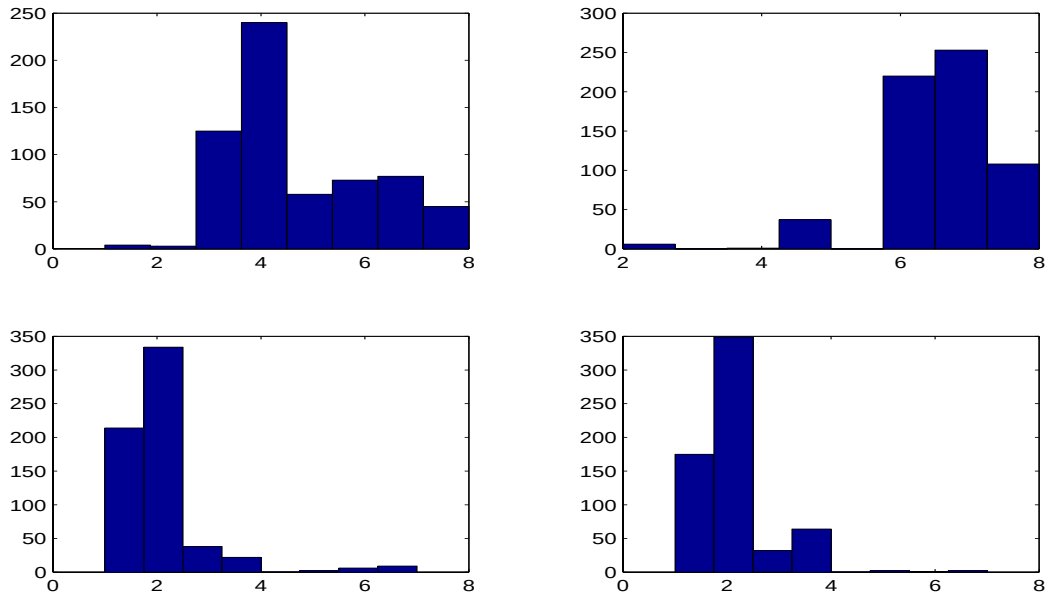


Figure 3.12: Histograms of winning nodes for an episode of measurements from several machine conditions: a. imbalance; b. normal operation; c. bearing failure; d. bearing failure at different running speed, not used in training the ASSOM

3.7 Discussion

In this chapter we studied *methods for description of the machine health condition* on the basis of vibration measurements. The pump in the laboratory test bench was modelled with a (locally) linear *FEM model*, which led to reasonable agreement with measured responses. The response of the system was also considered linear when the pump was excited with noise of different intensities. Several *spectrum-based signal processing methods* were then

reviewed. Using an envelope spectrum, it is possible to distinghuish impulsive patterns in machine vibration from other vibration. This was verified with measurements from the rotating machine in the test bench when an outer race bearing fault and an imbalance were induced to the machine. In this chapter several *parametric models* of machine vibration were presented. For one of the models, the equispaced frequencies model, a drawback of the approach was observed: although it is feasible to search for the correct frequency spacing with maximum likelihood estimation, local maxima in the likelihood appear at multiples of the correct frequency spacing. This might hamper effective diagnostics if prior knowledge about the approximate value of the frequency spacing is lacking.

Inside a rotating machine, *nonlinear and nonstationary* phenomena may occur. Detecting nonlinearity can be done by measuring the response of the system to different excitations (experiment is described above) or using higher-order statistics and spectra of the measurements. A particular form of nonstationarity, cyclostationarity, was studied with a set of artificial signals. It has been reported recently that cyclostationary phenomena can be observed inside bearings and gearboxes. If the proper cyclic frequency is known, diagnostic information may be obtained with this method. Nonstationary phenomena can be detected with time-frequency methods, like wavelets. However, in the submersible pump there was already diagnostic information in the stationary part of the vibration signals: information about machine condition could be obtained from a Self-Organizing Map that learns invariants from raw vibration measurements, the ASSOM. Episodes corresponding to bearing failure measurements at two different operating modes (of which only one was used in map training), showed a similar hit-histogram on the map. This is an indication that the ASSOM may be used as a health indicator by determining a 'health-profile' for projected measurements.

Chapter 4

Independent Component Analysis

When a machine is monitored with several sensors, a number of correlated time series may be obtained. The correlations can be caused by the fact that the same underlying sources are present in all measurements, though in different proportions at each sensor. We propose the use of blind source separation (BSS) to unmix an ensemble of machine measurements. We would like to extract the signature of a machine under investigation, despite the presence of interfering sources. In this chapter we review the concepts behind methods for blind source separation, most notably Independent Component Analysis (ICA). In the next chapter we will apply BSS for the separation of a machine signature from its environment. This corresponds to the subtask "interference removal" in the scheme of figure 2.4. First, we review several concepts from information theory in section 4.1. The basic idea behind ICA is explained in section 4.2. Finally, the similarities and differences between ICA and two related approaches (*linear projection* and *beamforming*) are described in section 4.3.

4.1 Information-theoretic preliminaries

In chapter 2 we introduced information theory as the basis of an inductive principle. Information theory deals with the quantification of the information that is gained when observing realizations of random variables. We give some central definitions, based on [DO96]. Consider a discrete random variable X that has a distribution $p(x)$, where x takes values in the alphabet $\mathscr{X}$.

Definition 4.1.1 *The information $I(x)$ gained after observing an instance x of X is defined as*

$$I(x) = \log \tfrac{1}{p(x)} = -\log p(x)$$

Definition 4.1.2 *The entropy $H(X)$ of X is defined as*

$$H(X) = -\sum_{x \in \mathscr{X}} p(x) \log p(x) = E[\log \tfrac{1}{p(x)}]$$

The entropy is a nonnegative quantity that measures the amount of uncertainty about the outcome of a random variable. Alternatively, it is the average amount of information that is conveyed per message. Zero entropy corresponds to a deterministic process (i.e. a probability distribution that consists of a single value). Maximum entropy is obtained when the outcome of a random variable is maximally uncertain, e.g. with a uniform distribution. Yet another interpretation of entropy is that it represents the minimal expected codelength of a random variable. The optimal Shannon-Fano code approximates the entropy of the random variable to which the code is assigned (cf. chapter 2). Now consider two discrete random variables X,Y that are jointly distributed according to $p(x,y)$ with marginal distributions $p(x)$ and $p(y)$. Here x and y take values in the alphabet $\mathscr{X}$ and $\mathscr{Y}$ respectively.

Definition 4.1.3 *The joint entropy $H(X,Y)$ is defined as*

$$H(X,Y) = -\sum_{x\in\mathscr{X}}\sum_{y\in\mathscr{Y}} p(x,y)\log p(x,y) = E[\log\tfrac{1}{p(x,y)}]$$

Definition 4.1.4 *The conditional entropy $H(Y|X)$ is defined as*

$$H(Y|X) = -\sum_{x\in\mathscr{X}} p(x)H(Y|X=x) = -\sum_{x\in\mathscr{X}}\sum_{y\in\mathscr{Y}} p(x,y)\log p(y|x)$$

Consider again two discrete random variables X and Y. A convenient measure to compare two distributions $p(x)$ and $q(x)$ is the Kullback-Leibler divergence.

Definition 4.1.5 *The Kullback-Leibler divergence $KL(p(x),q(x))$ between two distribution $p(x)$ and $q(x)$ is defined as*

$$KL(p(x),q(x)) = \sum_{x\in\mathscr{X}} p(x)\log\tfrac{p(x)}{q(x)}$$

The Kullback-Leibler divergence is sometimes called relative entropy or cross entropy. We now define the mutual information between two random variables. This concept plays a central role in the theory behind ICA. It expresses the distance of the joint distribution of a set of random variables from a factorized distribution, i.e. it measures the distance from independence of the random variables.

Definition 4.1.6 *The mutual information $I(X;Y)$ is defined as*

$$I(X;Y) = KL(p(x,y),p(x)p(y)) = \sum_{x\in\mathscr{X}}\sum_{y\in\mathscr{Y}} p(x,y)\log\tfrac{p(x,y)}{p(x)p(y)}$$

The relation between entropy and mutual information is given in the following theorem and is further illustrated in example 4.1.

Theorem 4.1.1 *The following equality holds:*

$$\begin{aligned} I(X;Y) &= H(X)-H(X|Y) \\ &= H(Y)-H(Y|X) \\ &= H(X)+H(Y)-H(X,Y) \end{aligned}$$

For continuous random variables, the above definitions can be modified in a straightforward manner by replacing sums with integrals over the support of the random variables and replacing probability distributions with probability density functions.

Example 4.1: mutual information and entropy in a system

Consider a system with input X and output Y, which are both random variables. The relation between entropy H and mutual information I is shown in figure 4.1.

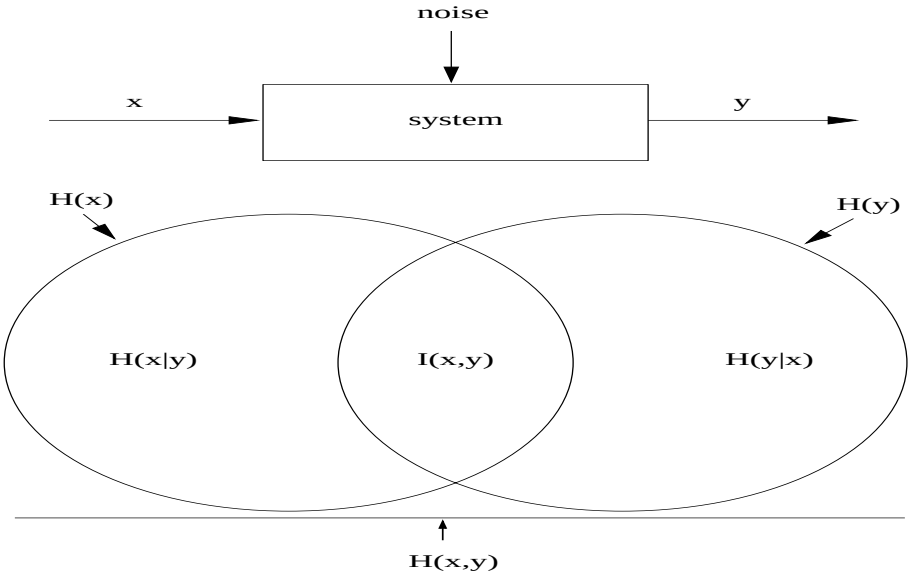


Figure 4.1: Entropy and mutual information, from [Hay94]

The entropy of a random variable X is indicative of the amount of 'surprise' that appears after observing a realisation of X. The 'amount of surprise' about an observation of X given an observation of Y (the conditional entropy of X given Y) is given by

$$H(X|Y) = H(X,Y) - H(Y) \tag{4.1}$$

where $H(X,Y)$ is the joint entropy of X and Y. From the figure is can also be seen that if the intersection of both conditional entropies (i.e. the mutual information) is empty, the joint entropy is maximal. This effect is illustrated on an artifical dataset in figure 4.5. In the current example, the mutual information $I(X;Y)$ between X and Y can be interpreted as the amount of surprise about (the next observation of) X that is resolved by observing Y. This corresponds to the intuitive idea that statistically independent random variables will have zero mutual information. □

4.2 Independent Component Analysis

It is known for more than a decade that a linear mixture of statistically independent sources can be unmixed without identification of the mixing process or detailed knowledge on the sources. The ideas behind those methods will now be described. In the sequel we will use the terms *blind source separation* and *independent component analysis* interchangeably. The first term emphasizes a particular application of the method: separating sources, where information about the sources can be used in the separation. The second term highlights a different goal: decomposition into independent components based on distributional properties, irrespective whether these components represent physical sources or 'interesting multivariate data projections'. At times, both goals (and the employed algorithms) may coincide.

4.2.1 Instantaneous mixing model

Consider the case in which one has access to multiple realizations **x**(t) of a mixture of independent signals **s**(t). Assuming a linear mixing model A and allowing additive Gaussian noise **n**(t) leads to the model

$$\mathbf{x}(t) = A\mathbf{s}(t) + \mathbf{n}(t) \tag{4.2}$$

where boldface variables denote random vectors with zero mean and finite covariance. In a real-world source separation problem, the vectors $\mathbf{x}(t)$ and $\mathbf{s}(t)$ live in $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively, and the number of sources is assumed to be smaller than or equal to the number of measurements ($m \leq n$). The term $\mathbf{n}(t)$ represents white Gaussian noise with covariance matrix λI, where $\lambda = \sigma^2$ is the noise covariance. In blind source separation, the measurements **x**(t) are n-channel time series

$$\begin{aligned} \mathbf{x}(t) &= [x_1(t), x_2(t), \ldots, x_n(t)]^T \\ x_i(t) &= [x_i(1), \ldots, x_i(T)], \qquad\qquad i = 1, \ldots, n \end{aligned} \tag{4.3}$$

which is illustrated in figure 4.2. The same holds for source signals **s**(t) and reconstructions **y**(t), where the dimensionality may differ.

Independence and source separation

The vector **s**(t) is assumed to have statistically independent components. The problem of blind source separation (BSS) is now: given T realizations of **x**(t) (i.e. observations at the time instances $t = 1, \ldots, T$), estimate both mixing matrix A and the m independent components **s**(t). If the noise covariance is known, the independent components can be retrieved, see also section 4.2.3. If this is not the case, estimation errors will be present in the reconstructions, so **s**(t) can only be recovered approximately. The noise term in (4.2) is then discarded, leading to the model

$$\mathbf{x}(t) = A\mathbf{s}(t) \tag{4.4}$$

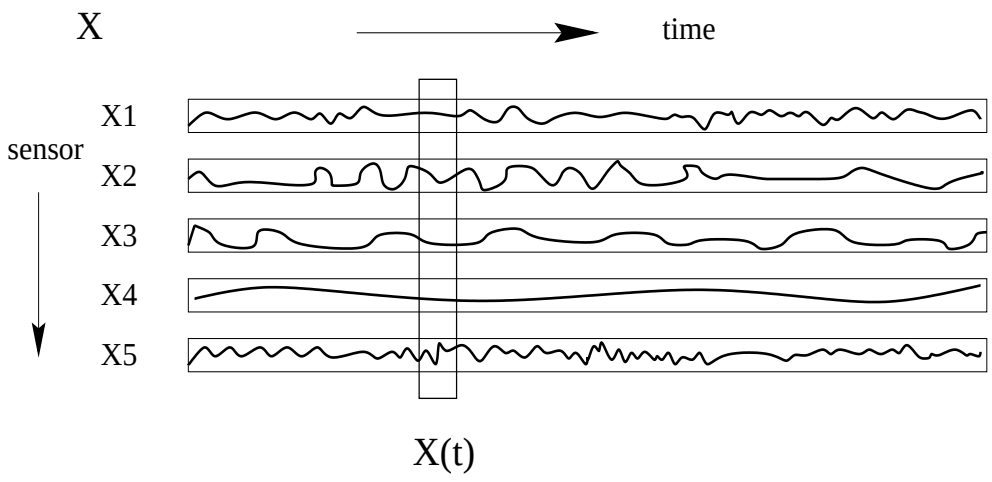


Figure 4.2: An ensemble of measurements and its representation in ICA

Now one tries to recover signals **s**(t) by computing signals **y**(t)

$$\mathbf{y}(t) = W\mathbf{x}(t) \tag{4.5}$$

such that **y** is a random vector whose components maximize a *contrast function*, that is chosen to be maximal when the components of the vector under investigation are statistically independent.

4.2.2 Minimization of mutual information

In the higher-order statistics (HOS) based approach, a criterion function (or *contrast function*) based on HOS is defined, whose optimization leads to the demixing matrix that is necessary for retrieving the independent components. A contrast function sufficient for this purpose is *mutual information*, which measures the KL divergence between the probability density $p(\mathbf{y})$ of a vector **y** and the factorized target distribution $\prod p_i(y_i)$

$$I(\mathbf{y}) = \int p(\mathbf{y}) \log \frac{p(\mathbf{y})}{\prod p_i(y_i)} d\mathbf{y} \tag{4.6}$$

This quantity is zero if the variables y_i are mutually independent and strictly positive otherwise. Hence, minimization of the mutual information between the sources will lead to independent components.

Contrast functions

In the context of ICA (formulated as (4.5)) it can be shown [Paj98b] that for minimizing the mutual information $I(\mathbf{y})$ between the components of the estimated source vector **y**

$$I(\mathbf{y}) = \sum_i H(y_i) - H(\mathbf{y}) = \sum H(y_i) - H(\mathbf{x}) - \log|\det W| \tag{4.7}$$

it is sufficient to minimize

$$J_{\text{MMI}} = \sum_i H(y_i) - \log\ |\det\ W| \tag{4.8}$$

which is a contrast function that only depends on the entropy of the reconstructed sources and the volume of the demixing matrix W. Extending this idea, one can maximize the contrast function

$$\Psi(\mathbf{z}) = -I(\mathbf{z}) \tag{4.9}$$

on whitened data $\mathbf{z}$ (i.e. data with zero first- and unit second order moments, see appendix D) in order to find a vector of independent components

$$\mathbf{y} = Q\mathbf{z} \tag{4.10}$$

and an orthogonal matrix Q. Now the mapping term (that measures the volume of the demixing matrix) can also be discarded, and source separation can be done by optimizing a one-unit contrast for each component. This is the basis for so-called *one-unit learning rules* [GF97]: assuming that one does not need cross statistics between several components in the computation of the cost function, these components can be computed one at a time (a process which is also called *deflation*).

Kurtosis maximization

Since the densities $p(\mathbf{y})$ and $p(\mathbf{z})$ are not known in practice, an approximation to (4.9) is maximized. Using an approximation based on Edgeworth expansions, it has been shown [Com94] that third and higher order moments[1] ($r \geq 3$) of a standardized (i.e. whitened) vector $\mathbf{z}$ are *discriminating* contrasts over the set of random vectors having at most one null moment of order r. This means that including a third or higher order moment in the contrast function $\Psi(\cdot)$ enables retrieval of a matrix A such that the equality $\Psi(p(A\mathbf{s})) = \Psi(p(\mathbf{x}))$ only holds when A is of the form

$$A = \Lambda P \tag{4.11}$$

where $\Lambda = \bar{\Lambda}D$ is the product of an invertible diagonal real positive scaling matrix $\bar{\Lambda}$ and a diagonal matrix D with entries of unit modulus, and P is a permutation matrix. Since the computational cost of computing cumulants of order r increases rapidly with increasing r and the third order cumulant has some awkward properties with certain types of data, one usually chooses the **fourth** order cumulant as the contrast to be maximized. For example, in the case of symmetrically distributed random processes and applications like harmonic retrieval, the third order cumulants are zero. For other processes, third order cumulants are very small, whereas they possess much larger fourth order cumulants [Men91]. The fourth order cumulant often used for this purpose is *kurtosis* [KOW+97]

[1] Actually, the related quantity *cumulants* is used in the method. This term will be used in the sequel

$$cum(y_i^4) = E[y_i^4] - 3E^2[y_i^2] \tag{4.12}$$

In practice one must be aware that short data lengths may give rise to inaccurate estimates of the fourth order cumulant using kurtosis [PTVF92]. Moreover, for prewhitened data one can thus propose the contrast [KOW$^+$97]

$$J(\mathbf{y}) = \sum_{i=1}^{m} E[y_i^4] \tag{4.13}$$

which is minimized for sources with negative kurtosis and maximized for positively kurtotic signals. More elaborate variants of HOS-based contrast functions [HO97, LGS99] can deal with both sub- and supergaussian sources simultaneously.

Remark 4.2: indeterminacies in ICA

Solving the problem (4.10) *blindly*, i.e. without using the parametric structure of the mixing matrix or an i/o identification procedure, implies that we can only recover the original sources up to scaling and permutation. Moreover, with a contrast function that exploits nongaussianity, at most one Gaussian distributed source is allowed (a weighted sum of two Gaussian distributed random variables is again Gaussian distributed!). These properties can be seen heuristically by noting that model (4.4) allows the exchange of a scalar between mixing matrix and sources [BAMCM97]

$$\mathbf{x}(t) = A\mathbf{s}(t) = \sum_{i=1}^{m} \frac{\mathbf{a}_i}{\alpha_i} \alpha_i s_i(t) \tag{4.14}$$

where α_i is an arbitrary complex factor and $\mathbf{a}_i$ denotes the ith column of A. This cannot be resolved without knowledge about source amplitudes or mixing process A, which would make the problem non-blind. Also, a permutation of the sources can be compensated by a corresponding permutation of the columns of A. □

Variants of ICA algorithms

There is a wide variety of methods for ICA. Methods can be distinguished according to the type of information that is used to separate sources (second-order or higher-order statistics) or the minimization procedure that is used (*adaptive approach*: minimization of a contrast function or *algebraic approach*: diagonalization of similarity matrices, see section 5.3.1). For the adaptive algorithms, several methods based on neural networks were proposed [KOW$^+$97, BS95]. An adaptive algorithm that approximates the demixing solution as its fixed-point was proposed in [HO97]. A generalization of the Bell-Sejnowski algorithm was given in [PP96], where (particular) temporal correlations of the sources are used along with higher-order statistics. The combination of higher-order and second-order statistics is also addressed in [ZM98, Hyv00]. A comprehensive overview of ICA methods can be found in [HKO01].

Remark 4.3: performance evaluation

When one is carrying out a controlled experiment, where the mixing matrix A is known, a performance measure developed by [YA97] can be used for tracking the optimization process. The so-called *cross-talking error* is defined as

$$E = \sum_{i=1}^{n}\left(\sum_{j=1}^{n}\frac{|p_{ij}|}{\max_k |p_{ik}|} - 1\right) + \sum_{j=1}^{n}\left(\sum_{i=1}^{n}\frac{|p_{ij}|}{\max_k |p_{kj}|} - 1\right) \tag{4.15}$$

where $P = (p_{ij}) = WA$. This index measures the deviation of WA from a scaled permutation matrix: if W is a proper estimation of the inverse of A (up to scaling and permutation) this index is close to zero; high values mean high cross-talk. The index is not normalized, but increases with the number of mixtures. □

4.2.3 Orthogonal approach to ICA

In the *orthogonal approach* to ICA one uses the fact that, after prewhitening (see appendix D), the ICA-basis is given by a rotation of the (prewhitened) measurements. We mentioned before in section 4.2.1 that in cases with noisy mixtures the sources cannot be recovered exactly. However, if the noise covariance σ^2 is known, we can estimate W_w from noisy data $\mathbf{x}$ since $R_{zz}(0) = R_{xx}(0) - \sigma^2 I$. Imagine we prewhiten the measurements, which are now considered to be complex-valued random variables; real-valued measurements are a special case of this. Prewhitening with matrix W_w gives

$$\mathbf{z}(t) = W_w\mathbf{x}(t) = W_wA\mathbf{s}(t) = Q\mathbf{s}(t) \tag{4.16}$$

Assuming $\mathbf{s}$(t) independent ($R_{ss}(0) = I$) and noting that $R_{xx}(0) = AE[\mathbf{s}(t)\mathbf{s}^*(t)]A^H = AA^H$, we can write $R_{zz}(0)$ as

$$\begin{aligned} E[W_w\mathbf{x}(t)\mathbf{x}(t)^*W_w^H] &= W_wR_{xx}(0)W_w^H \\ &= W_wAA^HW_w^H \\ &= QQ^H = I \end{aligned} \tag{4.17}$$

We see that for the complex case a **unitary** matrix Q remains to be identified after prewhitening. The resulting procedure is depicted graphically in figure 4.3. For this identification, other criteria than instantaneous decorrelation (PCA) are required. In the JADE algorithm [CS93], cancellation of higher-order cross-statistics between independent sources is used as additional criteria for source separation. Alternatively, a technique for identification of the remaining unitary transformation using second-order statistics only was proposed. We will describe this technique briefly in the following.

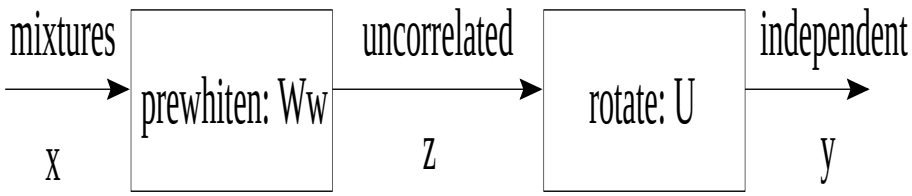


Figure 4.3: The orthogonal approach to ICA

Blind separation with second-order statistics

In the second-order blind identification (SOBI) algorithm by Belouchrani et al. [BAMCM97] the **temporal coherence** of the source signals is exploited. The technique uses the notion of a lagged correlation matrix of **s**

$$R_{ss}(\tau_i) = E[\mathbf{s}(t)\mathbf{s}^*(t-\tau_i)] \tag{4.18}$$

For spatially independent (and similar for spatially uncorrelated) sources with temporal correlations, the whitened measurements **z** have lagged correlation matrices

$$R_{zz}(\tau) = W_w R_{xx}(\tau) W_w^H = Q R_{ss}(\tau) Q^H \tag{4.19}$$

Since sources will exhibit zero cross-correlation at any properly chosen time lag $\tau \neq 0$ (see figure 4.4), we have the property that any whitened time-lagged correlation matrix is diagonalized by Q. This gives additional second-order criteria for separating the sources. Because of finite sample size effects and noise the lagged correlation matrices are never ex-

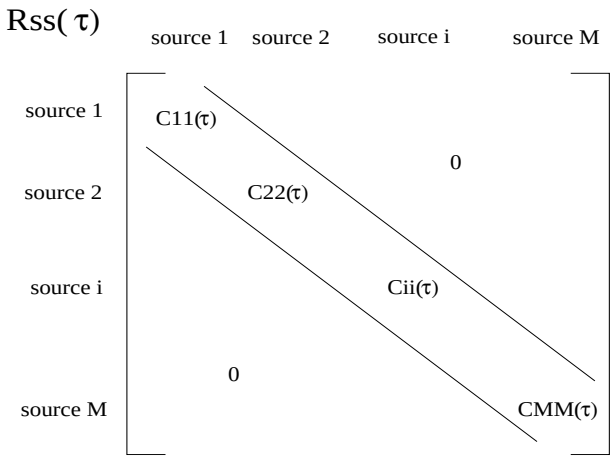


Figure 4.4: Independent sources have diagonal lagged correlation matrices

actly diagonalizable. In the SOBI algorithm, the joint diagonalization procedure by Cardoso [BAMCM97] is used in order to find the unitary matrix Q that (approximately) diagonalizes a set of lagged correlation matrices, corresponding the the a priori chosen set of time lags.

The demixing performance depends on the proper choice of time lags, but it was shown that just taking *many* time lags into account is often sufficient for achieving separation [ZM98]. However, taking noninformative delays into account may lead to numerical problems in the diagonalization procedure, hence decreasing the performance. We will use a variation on this approach in section 5.3.1 to use arbitrary second-order temporal structure for source separation.

Example 4.4: ICA with a skewed uniform distribution

A uniformly distributed 2-D dataset is mixed into a skewed shape, figure 4.5(a). The direction of maximum variance will be approximately aligned with the sum vector of two directions in the data (first: horizontal, second: pointing north-east). The second principal component will be orthogonal to the first basis vector. The projection of the dataset onto the PCA-basis is shown in figure 4.5(b). Prewhitening involves a scaling to unit variance. The projected dataset after rescaling is plotted in 4.5(c). More meaningful would be a projection onto the vectors that are aligned with the skewed distribution. These are the independent components of the data. Projection onto these vectors gives a dataset that is uniformly distributed in the plane, i.e. the independent component basis maximizes the entropy of the projection, figure 4.5(d). □

Optimization in algorithms for ICA and BSS

Yang and Amari [YA97] proposed to use the *natural gradient* in stead of the ordinary gradient in minimization of the cost function, which is a rescaled version of the ordinary Euclidean gradient. The goal is to make the gradient insensitive to the scale on the axes. The authors have shown that the gradient $\frac{\partial J(W)}{\partial W}$ and the weights W each reside in a different metric space (Riemannian and Euclidean, respectively) and that a certain coordinate transformation can be applied to express the gradient in the same space as the weights. In practice this means that it suffices to apply the following coordinate transformation to the previously computed gradient: post-multiplication of the gradient with $W^T W$. The natural gradient does not assume prewhitened data. Maximization by the natural gradient has been shown to improve convergence. Moreover, an on-line approximation of the algorithm has been proposed by [YA97]. The natural gradient is equivalent to the *relative gradient* proposed by Cardoso and Laheld [CL96]; the equivariance property yielded by this method amounts to the fact that the performance of the algorithm is independent from the scaling of the sources. Alternatively, one can exploit the fact that after prewhitening an orthogonal matrix remains to be found. A BSS-algorithm can be implemented as an optimization problem with an orthogonal constraint. Mathematically this is the same as performing an unconstrained optimization over the Stiefel manifold [EAS98]. Making explicit use of this fact by formulating a gradient with respect to this manifold allows for faster convergence in case of the SOBI algorithm [RR00].

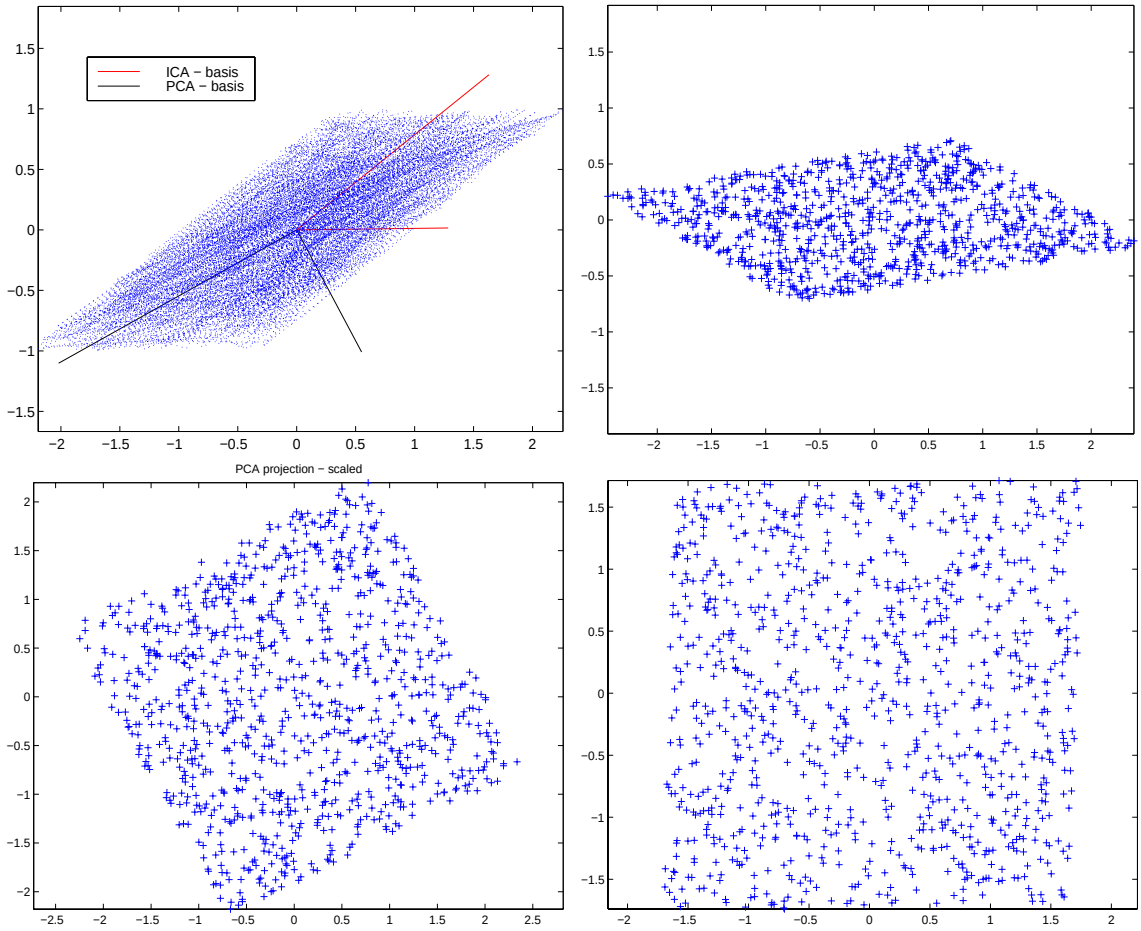


Figure 4.5: Skewed uniform distribution with PCA en ICA basis: (a, top left), PCA projection (b, top right), prewhitened (c, bottom left) and ICA projection (d, bottom right)

4.3 ICA and related methods

We now discuss the relation of ICA to methods in multivariate data analysis and array processing. From this discussion we can conclude that ICA is a technique for multivariate data analysis that can be interpreted as a blind beamforming technique if the multidimensional dataset is a multichannel measurement of mixed source signals. Hence, ICA is a method that can be used for blind source separation.

Projection pursuit The related method of *exploratory projection pursuit* (EPP) aims at finding 'interesting projections' in data by assuming nongaussian distributions of a projection as more interesting than a Gaussian. No noise model is present in this case. We give an overview of EPP and show an application of EPP to multivariate data analysis in appendix D. In ICA, a noise model can be present and also the 'interestingness criterion' may differ. In the HOS-based methods, projections that maximize nongaussianity are used. These methods are very closely related to EPP.

PCA, FA, IFA, cICA Principal component analysis (PCA) and factor analysis (FA) assume Gaussian distributed sources (or variables). With PCA no noise model is present, whereas in factor analysis there is a noise model. The PCA projection method is described in appendix D. In FA the aim is to enforce a diagonal noise covariance matrix, in order to explain as much 'useful information' about the factors (or sources) as possible. The criterion to be optimized may differ, depending on the particular application. In the context of *machine vibration analysis,* PCA and related techniques have been used in attempts to separate vibration sources out of (multichannel) measurements [Lac99]. The main disadvantage of this approach is that PCA tries to find an orthogonal basis that tries to maximize the variance of a (spatial) component. If the underlying vibration sources show marked differences in variance, this may separate the sources [Lac99]. In general, sources may have comparable amplitudes and variances, so that PCA will not decompose satisfactorily into the underlying sources. Independent Factor Analysis (IFA) was proposed by Attias [Att98] as a framework that unifies ICA and FA. It comprises an explicit modelling of the underlying source densities with a mixture of Gaussians (hence can deal with Gaussian and nongaussian sources) and incorporates (possibly anisotropic) additive noise as well. An EM-algorithm is proposed for estimation of demixing coefficients and sources. The earlier proposed contextual-ICA method (cICA) [PP96] uses a mixture of logistic distributions for modelling source densities; it incorporates temporal context by assuming an autoregressive model for the sources and estimates the AR-parameters along with the mixture parameters.

BSS and DCA The class of second-order statistics based separation methods are inbetween the methods described above: they assume Gaussian distributed sources with temporal correlations. As 'interestingness criterion' one can choose projections with minimal cross-similarities between sources (for which an explicit cost function can be formulated as well) or minimum source complexity, see chapter 5. Dynamic component analysis (DCA) is an extension of IFA that allows for convolutive mixing of sources [AS98]. In general, many methods in time- and frequency domain have been devised for BSS in convolutive mixtures, e.g. the Nguyen Thi-Jutten algorithm [NTJ95].

Beamforming The main difference between (informed) beamforming techniques [Boo87, Zat98, vdV98] and methods for *blind* beamforming and signal separation [CS93, Car98, HO97] is the criterion they use for separating a source signal: beamforming uses the directivity of the (main) source and assumes that disturbances make nonzero angles with this source. The direction of arrival (DOA) of an incoming source signal can be computed from an array of sensors. The angular resolution (*AR*) is determined by the frequency of the source signal f and the array dimensions (inter-sensor spacings D)

$$AR = \frac{\lambda}{D} = \frac{c}{D \cdot f} \tag{4.20}$$

where c is the sound velocity (which is approximately 340 m/s in air at 20^oC). In this formula, a planar wavefront is assumed along with zero angle of arrival α of the source (otherwise a factor $\sin\alpha$ enters the formula) and thermal and wind effects in the transmitting medium are being ignored. A narrowband signal $s(t)$ that enters a sensor from the far field will experience

a time delay τ when travelling from one sensor to the next, depending on the angle of arrival α and the interelement spacing (in wavelengths) $\Delta = \frac{D \cdot f}{c}$. If this delay is small compared to the inverse of the bandwidth W of the signal, i.e. $W\tau \ll 1$, the delayed signal can be modeled as a phase shifted version of the 'first' signal, where the phase shift equals

$$\theta = e^{j2\pi\Delta \sin\alpha} \tag{4.21}$$

If we have a sensor array with equispaced sensors at the locations $\Delta_i, i = 1, \ldots, M$ (in wavelengths), the M-dimensional measurement vector $\mathbf{x}(t)$ can be modeled as

$$\mathbf{x}(t) = [e^{j\psi_1}, e^{j\psi_2}, \ldots, e^{j\psi_M}]^T \cdot a(\alpha)s(t) = \mathbf{a}(\alpha)s(t) \tag{4.22}$$

where the *steering vector* $\mathbf{a}(\alpha)$ is expressed in terms of $\psi_i(\alpha)$,

$$\psi_i(\alpha) = 2\pi\Delta_i \sin\alpha \tag{4.23}$$

and the sensor gain pattern $a(\alpha)$. If one considers the case where N independent (narrowband) sources at different locations are impinging on the array simultaneously, a linear mixing matrix between sources and sensors can be postulated:

$$\mathbf{x}(t) = A\mathbf{s}(t), \qquad A = [\mathbf{a}(\alpha_1) \ldots \mathbf{a}(\alpha_N)] \tag{4.24}$$

where the mixing vectors $\mathbf{a}_i$ are attenuated or amplified steering vectors (assuming the multipath propagation model holds). In practice, the narrowbandness assumption has to be verified. A useful heuristic is to check whether the product of source bandwidth W and maximum inter-element delay τ is sufficiently small. The planar wave assumption (i.e. the source is in the far field) may not be valid if the sensor array is close to a source. Moreover, the environment may cause reflections and scattering of the signal into multipath rays. For measurements in open or suburban terrain, the angle and delay spread due to these effects is expected to be small [vdV98]. If one of the above assumptions does not hold, a convolutive mixture model is more appropriate.

In beamforming one tries to find the inverse of the mixing matrix A by using the parametric structure of the matrix imposed by the directivity assumption. We see that for mixing of narrowband acoustical sources, a linear instantaneous mixing model (4.24) holds. However, this is a mixing matrix with *complex* entries. *Blind* beamforming is a restricted version of this approach: one now forgets about the parametric structure of the mixing matrix (e.g. because the directivity assumption cannot be made or the multipath propagation model does not hold) and tries to identify the mixing matrix using only knowledge about the measurements $\mathbf{x}$. Prior knowledge about the sources can be of a parametric nature (e.g. in communication signals), which leads to *deterministic blind beamforming* [vdV98]. If only **statistical** properties of the signals can be used, the methods for *blind source separation* emerge.

Summary The relations between ICA, FA, PCA, EPP, IFA, DCA, beamforming and BSS are summarized in figure 4.6, which is inspired by a figure in [Hyv99].

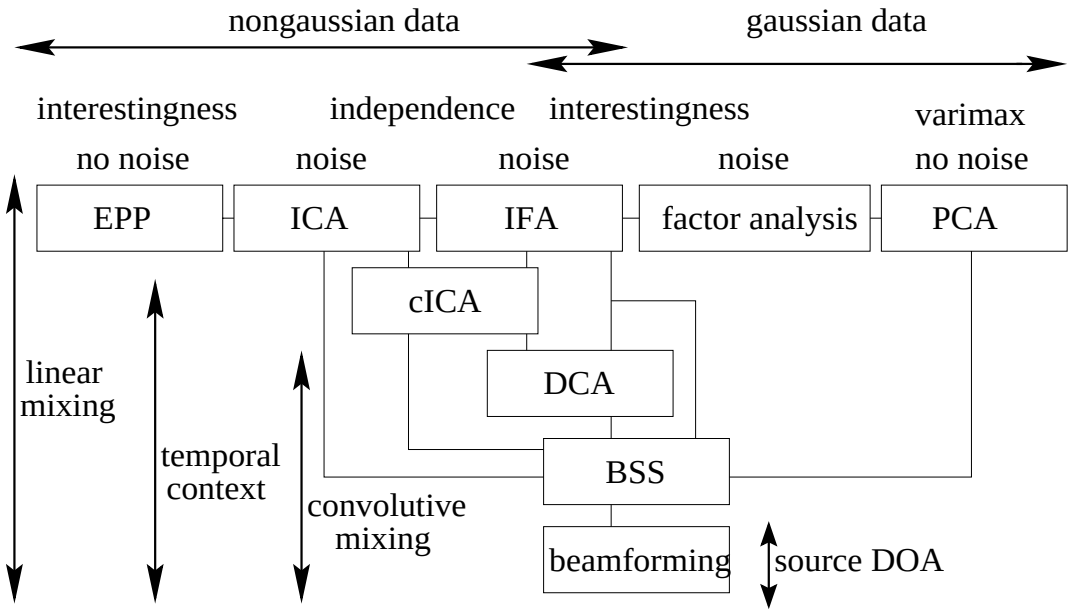


Figure 4.6: ICA, linear projection, BSS and beamforming

Chapter 5

Blind separation of machine signatures

In this chapter, our aim is to separate the vibration of a machine from interfering sources using measurements only. The measurements may be obtained by acoustic or vibration sensors. We will assume certain conditions for the mixing process, but do not identify the mixing parameters explicitly. This general approach is at the expense of stronger assumptions on the sources: that they are statistically independent. We investigate in several case studies which mixing model is appropriate. Moreover, we take into account that the sources to be separated are generated by rotating machines. Hence, the sources will exhibit temporal correlations and may show cyclostationarity or have a particular time-frequency signature (chapter 3).

5.1 Mixing of machine sources

Measurements on machines are often composed of *several underlying sources*. This can be troublesome for fault detection. For example, if multiple machines are coupled (as in ship engine rooms), failure on one machine may not be noticed, since it can be masked by the vibration of interfering machinery, figure 5.1(a). Moreover, measurements from multiple sensors on a machine casing often exhibit *spatial redundancy and diversity*. Forces inside the machine that operate in the radial plane will for example be measured in both radial channels of a triaxial accelerometer. Alternatively, sensors that are distributed on the machine casing will measure a mixture of the underlying vibration components: many fault-related peaks in the spectrum will be visible at several sensors. Moreover, the underlying vibration sources will be measured only as the result of a *filtering operation* from source to sensor and the contribution of each of the sources to a certain sensor will depend on the position of sources and sensors. Sensor placement on suitable positions is a nontrivial task. However, human experts have heuristics on how to do this, for example based on the local rigidity of the structure, direction of exerted forces and the distance to the monitored component. Reconstruction of the underlying sources in a multichannel measurement of rotating machines can overcome the problem of machine interference, whereas simultaneous demixing and deconvolution may allow for the use of time domain (waveform-detection or matching) methods for fault detection. In chapter 1 we noted that the degree of coherence between two vibration measurements at different positions on a machine will vary from completely coherent

to completely incoherent. Inbetween those two extremes the situation arises where 'spatial redundancy and diversity' exists between a set of vibration sensors on a mechanical system; this is the situation we focus on. In subsection 1.2.1 the vibration transmission process in rotating mechanical machines was analyzed. The model for vibration measured on the machine casing is a *convolutive mixture* of machine fault sources, external interferences, modal vibration components and measurement noise. As a first approximation to the convolutive mixture model, an *instantaneous* mixing model can be used. For structures with only narrowband sources (like sinusoids) or mechanical structures without dispersion or reverberation, this model can be a reasonable approximation. Moreover, the instantaneous model is still valid for a convolutive mixture, but only in a certain frequency band, section 5.4. This approach can also be used in an acoustic setting, where the signature (section 2.3) of a rotating machine can also be distorted with environmental interferences. From the previous chapter we know that for acoustic measurements of far-field narrowband sources in non-rural and non-reflective environments, the instantaneous mixing model may be appropriate.

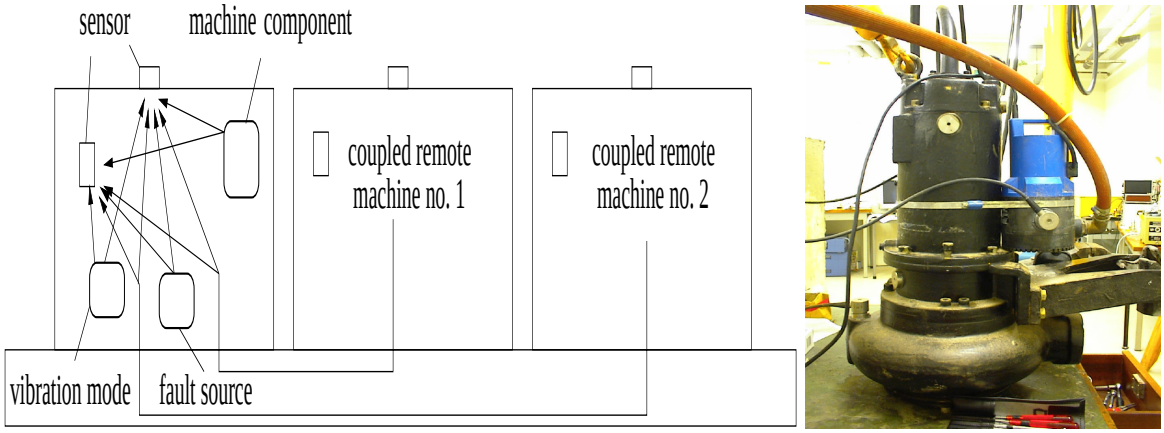


Figure 5.1: a: Contributions to vibration measurements from various machine sources; b: experimental setup with coupled pumps

Previous work and chapter overview

Several authors have addressed the reconstruction of underlying vibration sources from external measurements. In [KDBU94], the number of incoherent sources that contribute to a multichannel machine vibration measurement is determined using the singular value decomposition of the spectral matrix of the measurements. In [DA97] a multichannel measurement is decomposed into broadband sources with the Labrador method. This is a method that rotates the basis obtained with the SVD of spectral matrix according to a number of ad-hoc formulated heuristics. In [KT97], modal and eigenvector beamformers are applied to localize the source of vibration in a vibrating beam. Parallel to our work, Gelle et al. [Gel98, GCD99] have also taken a blind approach to reconstruction of rotating machine sources. Their work builds upon earlier work on blind convolutive demixing of sinusoidal signals using higher-order statistics [SCL97]. The use of second-order methods for source separation is consid-

ered more robust to noise and finite segment lengths compared to higher-order statistics based methods [PS99, SSM+99]. Note that machine vibration of different (nearby) machines will usually be spectrally different, which assures the applicability of SOS based methods in this application.

In this chapter, we present two approaches to second-order blind source separation of instantaneous mixtures. In the *MDL approach* to blind source separation [Paj98b, Paj98a, Paj99, YP98, YP99], a cost function is formulated that is based on second-order *temporal* statistics of the sources. This enables demixing of temporally coherent Gaussian sources. The minimization of mutual information between the reconstructions is exchanged for the minimization of the summed complexity. The *bilinear forms* approach [Les99, YL00, YLD02] unifies several orthogonal second-order source separation algorithms, and allows for incorporation of time-frequency structure or cyclostationary properties of the sources. This is relevant for machine sources, which often exhibit specific second-order structure.

Then we introduce the convolutive source separation problem and describe an algorithm for blind convolutive demixing by Nguyen Thi and Jutten [NTJ95]. The algorithms are investigated on mixtures of simulated sources (which resemble rotating machine signals). Finally, the merits of our blind separation approach for machine monitoring applications are investigated with measurements from three different rotating machine setups. We performed a controlled laboratory experiment with the submersible pump, see figure 5.1(b). We also investigate source separation in two real-life demixing problems, involving both acoustical and vibrational mixing.

5.2 Blind source separation using the MDL-principle

A general assumption about *natural* signals (signals that occur in the world around us) is that they are usually not complex. Remember from chapter 2 that high *complexity* of a signal means: having a description that has approximately the same length as the signal itself. Also note that nearly lossless compression algorithms exist for many natural signals. Since one can look upon a compressed version of a signal as a description of the signal it serves as an upper bound for the algorithmic complexity of that signal. Moreover, in general most (nonnatural) discrete signals or finite alphabet strings are complex indeed. The MDL approach to ICA was introduced in [Paj98a, Paj98b], elaborated on in [Paj99] and employed in a rotating machines context in [YP98, YP99].

5.2.1 MDL-based ICA

Consider a sum x of N 'similar' signals $z_i, i = 1, \ldots, N$. Intuitively, the complexity $K(x)$ of the sum will not be much larger than the complexity $K(z_i)$ of each of the sources separately. For 'dissimilar' signals, the joined complexity $K(x)$ will indeed be much larger then the separate complexities, most probably in the order of $\sum K(z_i)$. This can be used as a basis for signal separation.

Since mixing amounts to a *weighted* combination of the source signals, one should also take the complexity of the mixing mapping (accounting for the weight coefficients) into ac-

count. Remember from chapter 2, that according to the minimum description length principle, the description $\mathscr{L}(D)$ of a dataset D may equal the coding cost of the common production algorithm (or *model*) $\mathscr{H}$ and the residual with respect to this model

$$\mathscr{L}(D) = \mathscr{L}(D|\mathscr{H}) + \mathscr{L}(\mathscr{H}) \tag{5.1}$$

where $\mathscr{L}(.)$ is the codelength (cf. section 2.2.2). This leads to a trade-off between model complexity $\mathscr{L}(\mathscr{H})$ and residual error $\mathscr{L}(D|\mathscr{H})$: the more complex the model, the smaller the error term but the larger the coding cost of the model (and vice versa). The 'mapping term' can be defined as follows. The complexity of a *linear* mixing mapping A (cf. equation (4.4)) can be measured by coding the eigenvectors and eigenvalues λ_i of A. Assuming a fixed quantization and noting that the eigenvectors are of unit norm, this yields a coding length proportional to $\log|\lambda_i|$:

$$c + \sum_i \log|\lambda_i| = c + \log|\det\, A| = c - \log|\det\, W| \tag{5.2}$$

where c is a constant depending on the description accuracy (quantization).

Complexity-based cost function

Separation of linearly mixed sources using (an approximation to) algorithmic complexity is performed by minimization of the MDL-based ICA cost function

$$J_{\text{MDL}} = \sum_i \frac{1}{N} K(y_i) - \log\, |\det\, W| \tag{5.3}$$

It is instructive to see the analogy with the cost function for MMI-based ICA, formula (4.8). In cost function J_{MDL} the Kolmorov complexity $K(\cdot)$ is not computable. For general random variables, entropy $H(\cdot)$ can be used as a measure of complexity. Intuitively, a high entropy indicates high randomness and hence a large description length. This is a different interpretation of the 'signal term' in minimum mutual information based ICA, which is the term $\sum H(y_i)$ in the right-hand-side of equation (4.8). When the sources are random variables *with temporal correlations*, we can exploit the fact that different natural signals reside in different subspaces of the embedding space. This can be quantified by determining the rank of the correlation matrix of each reconstruction and summing the results. If we allow for additive noise in the model, complexity of the reconstructions cannot be measured with $\text{rank}(R_{y_i y_i})$ any more. For this case, the term $\det R_{y_i y_i}$ can be used as measure of complexity of signal $y_i(t)$. This term will be larger if the delay space in which a signal is embedded is more 'filled'.

Example 5.1: complexity in noiseless mixing of sinusoids

The main idea behind MDL-based ICA is that a natural signal $s(t)$ will be concentrated in a subspace of the space that is obtained using a delay coordinate embedding (3.15). Now

consider the case where we linearly mix two sinusoids $s_1(t) = \sin \omega_1 t, s_2 = \sin \omega_2 t$ into two mixtures

$$[x_1(t)\ \ x_2(t)]^T = A \cdot [s_1(t)\ \ s_2(t)]^T \tag{5.4}$$

Each mixed signal $x_i(t), i = 1,2$ has a complexity that is approximately equal to the sum of the complexities of the sources:

$$K(x_1(t)) \approx K(x_2(t)) \approx K(s_1(t)) + K(s_2(t)) \tag{5.5}$$

Hence, the summed complexity of the mixtures would be almost equal to twice the summed complexity of the sources:

$$K(x_1(t)) + K(x_2(t)) \approx 2 \cdot \{K(s_1(t)) + K(s_2(t))\} \tag{5.6}$$

By finding a linear transformation of the mixtures that forces the reconstructions to ly in different subspaces of the signal subspace (i.e. a transformation that minimizes the summed complexity of the reconstructions), we can undo the 'smearing' in the delay space that is due to the mixing operation.

In particular, a sinusoidal signal can be described by two basis vectors (section 3.6.1). i.e. the rank of the signal correlation matrix (3.14) that is obtained with a delay vector of length L will be 2. For the noiseless mixing case at hand we can use rank$(R_{y_i y_i})$ to measure the complexity of the reconstructions. When mixing two sinusoids of different frequency, we will have a summed 'complexity' of the sources of 4 and a summed 'complexity' of the mixtures of 8. It can be seen that finding a linear transformation that minimizes the summed complexity of the reconstructions will lead to demixing. □

5.2.2 Differentiable cost function for MDL-based ICA

The generalized source separation problem using the MDL-principle leads to a global search for the minimimum of complexity, since it is in general impossible to formulate a gradient with respect to the mixing parameters. For the case of (approximately) linear mixing of Gaussian zero-mean random variables with temporal correlations a differentiable cost function has been proposed in [Paj99, YP99]. The complexity of the separated signals $\mathbf{y}_i$ may be measured by coding the correlation matrix $R_i = R_{\mathbf{y}_i \mathbf{y}_i}$ using the delay vectors

$$\mathbf{y}_i(t) = [y_i(t), y_i(t+1), \ldots, y_i(t+L-1)] \tag{5.7}$$

where L is the size of the subspace in which the signal $\mathbf{y}_i$ is embedded (section 3.2.1). Since the entropy of a Gaussian random variable equals [Hay94]

$$\frac{1}{2} \log 2\pi\sigma^2 + \frac{1}{2}$$

and the variances of the principal components may be summed, this leads to a 'signal complexity term'

$$\frac{1}{2}\sum_i \log \sigma_i^2 = \frac{1}{2}\log \prod_i \sigma_i^2 = \frac{1}{2}\log \text{ det } R_i$$

which leads to a cost function that is to be minimized:

$$J_{\text{MDL}} = \frac{1}{L}\sum_i \frac{1}{2}\log \text{ det } R_i - \log|\det W| \tag{5.8}$$

This expression can be differentiated with respect to the rows $\mathbf{w}_i$ of the unmixing matrix. Since R_i only depends on $\mathbf{w}_i$ one can compute $\frac{\partial \log\det R_i}{\partial w_{ik}}$, $1 \leq k \leq m$, where $\mathbf{w}_i = [w_{i1}, \ldots, w_{im}]$. Denoting $F_i' = \frac{\partial F}{\partial w_{ik}}$ it can be derived that

$$(\log \text{ det } R_i)_i' = 2 \text{ tr } \{R_i^{-1} E(\mathbf{y}_i^T \mathbf{y}_i')\} \tag{5.9}$$

where it has been used that $(\det F)' = \det F \times tr[F^{-1}F']$. Moreover, from $\mathbf{y}_i = \mathbf{w}_i X$ it follows

$$\mathbf{y}_i' = [x_k(t), x_k(t+1), \ldots, x_k(t+L)] \tag{5.10}$$

Since the observation matrix X is known, the gradient (5.9) can be computed. There are still a number of practical problems in the minimization of the above cost function. Details on the optimization procedure can be found in appendix E.

Remark 5.2: pros and cons of MDL-based ICA

Since departure from Gaussianity is never used as a criterion, multiple Gaussian sources are allowed. Moreover, note that there is no assumption on the linearity of the (mixing) mapping in this approach, nor that the components will be statistically independent. This may allow for dealing with (slightly) nonlinear mixing and correlated sources. However, we note that for a larger number of sources, minimization of the MDL-ICA cost function still yields unsatisfactory results (see next section). □

5.2.3 Experiment: separation of harmonic series using MDL

We compare the MDL-based source separation algorithm to several convential (HOS-based) ICA algorithms for separating artificially mixed harmonic series.

Data description

The data consists of an ensemble of source signals, artificially mixed using a mixing matrix A with entries drawn randomly from the uniform distribution on the interval $[-1,1]$, i.e. $\mathscr{U}[-1,1]$. The individual source signals are harmonic series with additive noise $\varepsilon(n) \sim \mathscr{N}(\cdot)$, generated according to equation (3.12). The harmonic series depend on the amplitudes A_m, phases ϕ_m, center frequency f and sideband frequency spacing Δf. All parameters were drawn randomly, according to the following distributions:

$$A_m \sim \mathscr{U}[0.5,1] \qquad \phi_m \sim \mathscr{N}[-\pi,\pi] \qquad f \sim \mathscr{U}[0.02,0.35] \qquad \Delta f \sim \mathscr{U}[0.01,0.02]$$

Note that we used normalized frequencies and sideband frequency spacings. Since the position and width of each harmonic series varies with each new source signal realization, there is no way to distinguish more or less complicated mixtures beforehand. We consider a mixture as *more complicated* if several source signal have harmonic series at similar frequency bands. Moreover, the larger the number of sources in the mixture, the larger the probability that the mixture will be complicated. The number of sinusoids M in a harmonic series is a parameter to be specified. This choice defines the size of the signal subspace corresponding to the individual source signals. In the experiments with artificial data we used a signal length $N = 1000$ for each source signal.

Choice of delay vector length L

The learning behaviour of the algorithm depends on the cost function that is being minimized and the manner in which the gradient descent search is performed. We used MDL-based ICA with deflation[1] and the $\log J$ cost function from equation (E.5). The learning rate was taken to be high and exponentially decaying, whereas no momentum was used. In figure 5.2 the results for harmonic series with 3, 7 and 21 (spectral) components are plotted for the case of 2 mixtures (of 2 sources). The size of L was varied and the mean cross-talking error along with its variance (upper and lower bars) was computed over 5 repetitions of the algorithm on the same data. For two mixtures, separation is fairly accurate for small number of sinusoids (cross-talking error smaller than one). Remember that the cross-talking error is not a normalized error, so the performance figures are only indicative of the separation quality. We can however use the measure to compare the results of different separation experiments for different values of L, since worse separation leads to higher cross-talking errors (for the same set of sources). Furthermore, note that the upper and lower bars indicate the *variance* over 5 repetitions, which explains the value smaller than zero in figure 5.2(a). We usually noted a large variation in demixing results over several repeated demixing trials with the same mixing matrix. In this experiment the mixing matrix was also randomized in each repetition, leading to extra variation. This explains the large variability in each set of 5 repetitions. The choice of L, however, does not seem to influence the separation results significantly. For three mixtures, however, performance degrades severely. Again, the particular choice of delay vector length does not influence the result significantly. For higher number of mixtures (5, 8, 10; not shown), separation performance is on average even worse. We conclude that the choice of L does not influence the separation performance of MDL-based ICA significantly.

Comparison of algorithms

We generated an ensemble of nonoverlapping harmonic series with 5 (spectral) components. First of all, we investigated 1-D projection pursuit (EPP) for separating the mixtures (figure 5.3(b)) of independent sources shown in figure 5.3(a). The prewhitened mixtures are shown in figure 5.3(c). The PP separated sources are shown in figure 5.4(a). We deflated two projection pursuit directions, which resemble two underlying components of the mixtures.

[1] Iteratively determining one basis vector after another, see appendix D

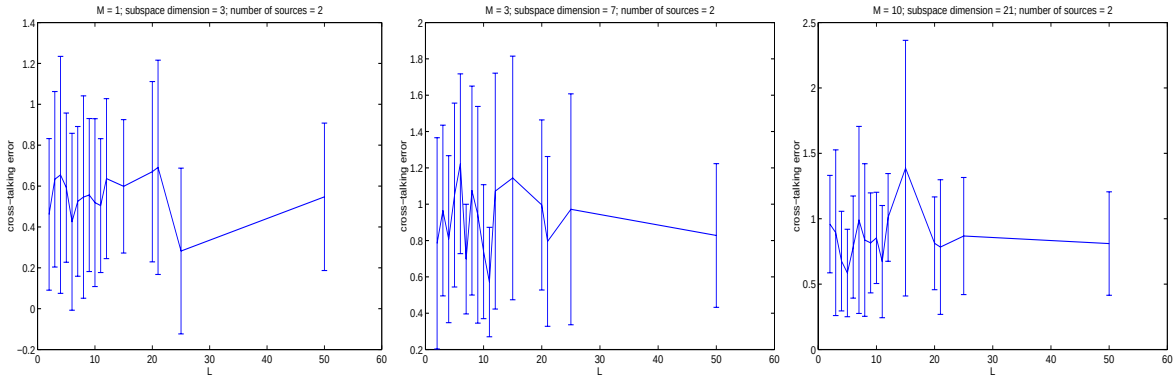


Figure 5.2: Influence of choice of L on separation performance: 2 sources, (a) target dimension = 3, (b) target dimension = 7 and (a) target dimension = 21

Moreover, removing only first- and second-order correlations from the mixtures already separates the sources quite reasonable. There is still some mixing present between the second and third component with respect to the first and second (in terms of frequency bands) harmonic series. Then we applied the fixed-point algorithm (also called *fast-ICA*) by Hyvärinen [HO97], to the mixtures. Symmetric fast-ICA with sigmoid nonlinearity and no dimension

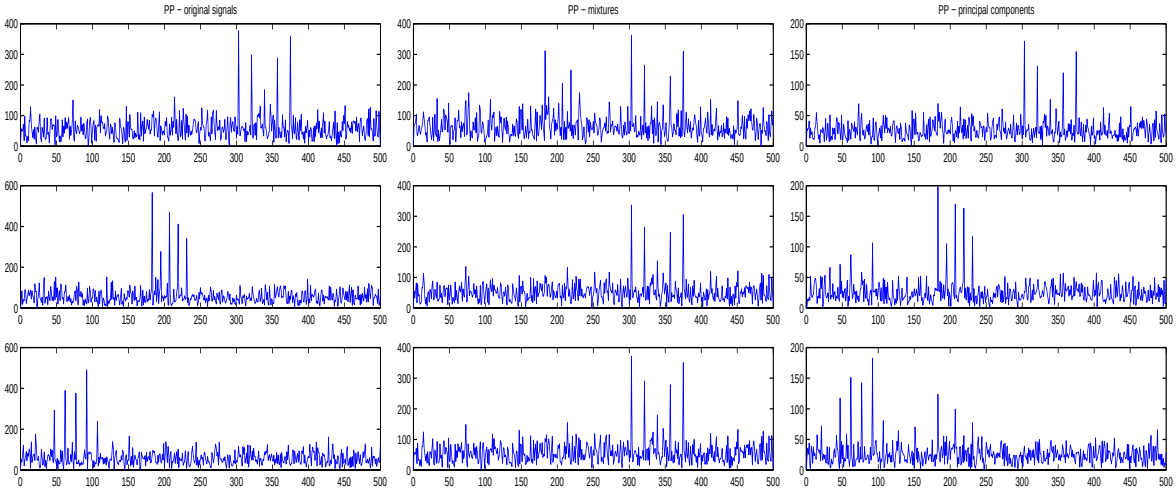


Figure 5.3: Separation of harmonic series: original sources (a), mixtures (b) and PCA (c)

reduction gives in general bad results: out of ten repetitions, approximately 5 runs do not converge in 30 steps, approximately 2 converge to the wrong solution (i.e. a cross-talking error in the order of 5.0) and three make a good separation (cross-talking error $\ll 1$). This may indicate that the requirements for fast-ICA are not fulfilled with this type of signals. Inspection of the kurtosis of the sources reveals values of -0.13 , -0.24 , -0.01 respectively. Too small differences in nongaussianity may cause problems for HOS-based separation. Using fast-ICA with the deflation approach, however, yields convergence to a fairly reasonable solution in every case, figure 5.4(b). Apparently, restricting the search for a new component

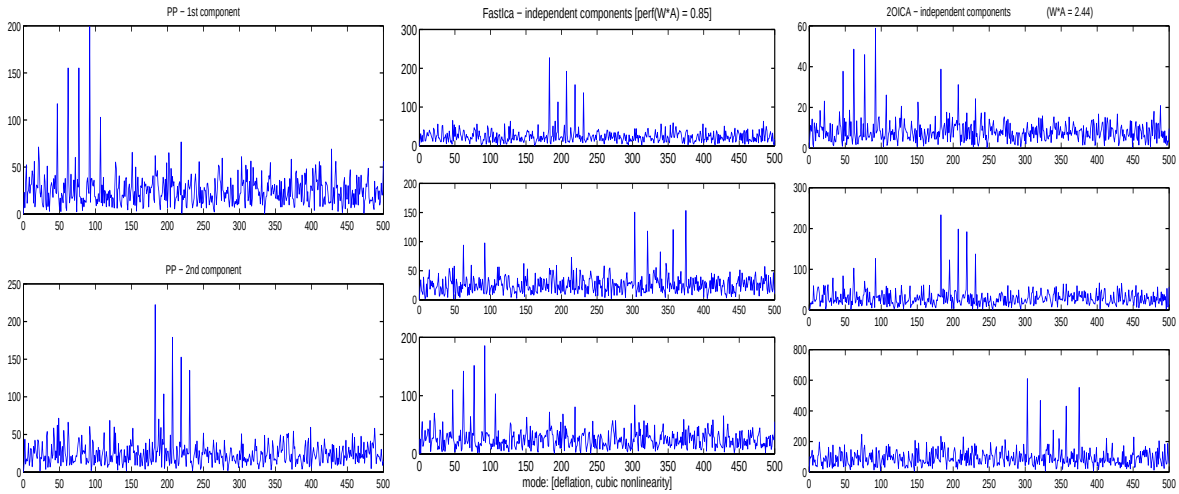


Figure 5.4: Separation of harmonic series: EPP (a), fast-ICA (b) and MDL-ICA (c)

in the space orthogonal to the subspace of previously found components is to be preferred to a repeated search for all components, and orthogonalizing the basis afterwards. Finally, MDL-based ICA (with deflation) results in a slightly worse but still reasonable separation, figure 5.4(c). However, the unmixing procedure is much faster than projection pursuit unmixing, whereas it has the advantage over fixed-point ICA that there is no assumption with respect to the nongaussianity of the sources.

Overlapping harmonic series

Then we generated an ensemble of harmonic series with 5 (spectral) components where two harmonic series were overlapping. In this case, neither projection pursuit nor fixed-point ICA succeeded in finding an adequate solution, figures 5.5(a) and 5.5(b), whereas the MDL algorithm still performs adequately, figure 5.5(c). We used MDL-based ICA with a constant initial learning rate (0.9), with 25 iterations per deflated component and no momentum. The MDL-based ICA method outperforms the HOS-based methods on these mixtures. We stress that the previous results are representative of the typical behaviour of the algorithms.

5.3 Blind source separation with bilinear forms

The bilinear forms framework is an extension of existing second-order approaches to BSS. It was first published in [Les99] and subsequently employed in a rotating machine context in [YL00] and [YLD02]. We present the main idea and then move to the choices of particular bilinear forms. In the framework, an arbitrary set of second-order signal similarities may be used for source separation.

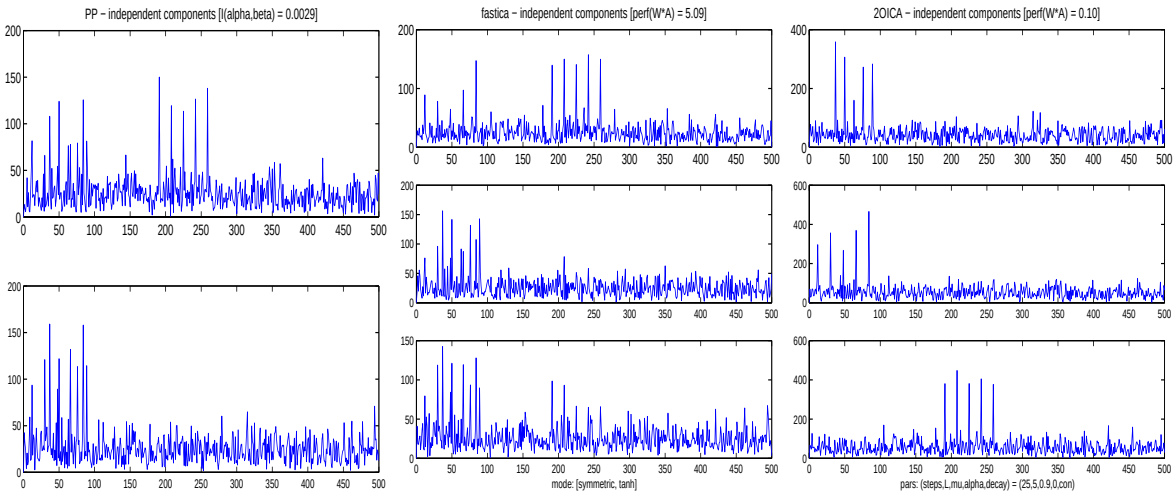


Figure 5.5: Separation of overlapping harmonic series: EPP (a), fast-ICA (b), MDL-ICA (c)

5.3.1 Bilinear forms and source separation

A bilinear form g is a function $g(\cdot,\cdot)$ of two signals that expresses the degree of similarity between the signals in terms of a generalized cross-correlation. Let U and V be vector subspaces over $\mathbb{C}$.

Definition 5.3.1 *A function $g(u,v) : U \times V \rightarrow \mathbb{C}$ is called a bilinear form if it satisfies the following properties:*

1) $g(u_1+u_2,v) = g(u_1,v)+g(u_2,v)$ *for all* $u_1,u_2 \in U, v \in V$.

2) $g(\alpha u,v) = \alpha g(u,v)$ *for all* $u \in U, v \in V$ *and* $\alpha \in \mathbb{C}$.

3) $g(u,v_1+v_2) = g(u,v_1)+g(u,v_2)$ *for all* $u \in U, v_1,v_2 \in V$.

4) $g(u,\alpha v) = \alpha^* g(u,v)$ *for all* $u \in U, v \in V$ *and* $\alpha \in \mathbb{C}$.

An example is the inner product, denoted by $g_0(x_1,x_2)$

$$g_0(x_1,x_2) = \lim_{N\rightarrow\infty} \frac{1}{N} \sum_{t=0}^{N} x_1(t)^* x_2(t) \tag{5.11}$$

where now the * operator denotes complex conjugation. We will describe other bilinear forms in the sequel. In order to be suitable for source separation, a bilinear form g must satisfy

$$g(y_k,y_l) = c_k \delta_{kl}, \qquad 1 \leq k,l \leq m \tag{5.12}$$

where the $y_i, i = 1,\ldots,m$ are components of the estimated source vector $\mathbf{y}$. We have now dropped the time index t. Moreover, by δ_{kl} we denote the Kronecker delta and c_k is nonzero

for all k. Both sources and mixtures are now considered to be complex-valued random variables. When applied to a multidimensional vector $\mathbf{y}$, a bilinear form gives rise to a "similarity matrix"

$$g(\mathbf{y},\mathbf{y}) = R^g_{yy} = \begin{bmatrix} g(y_1,y_1) & \cdots & g(y_1,y_m) \\ \vdots & & \vdots \\ g(y_m,y_1) & \cdots & g(y_m,y_m) \end{bmatrix} \tag{5.13}$$

that is diagonal if the reconstructions y_i are proper estimates of the sources. Bilinearity implies that

$$g(\sum \alpha_k x_k, \sum \beta_l x_l) = \alpha^H R^g_{xx} \beta \tag{5.14}$$

for a linear mixture $\mathbf{x}$ with components $x_i, i = 1,\ldots,n$. In this formula, R^g_{xx} is defined analogous to equation (5.13), and parameter vectors α and β are written as $\alpha = [\alpha_1,\ldots,\alpha_n]^T$ and $\beta = [\beta_1,\ldots,\beta_n]^T$.

Bilinear forms can be used as criteria for source separation. A proper demixing matrix W should separate the sources, which results in diagonal similarity matrices for all chosen bilinear forms. It is known that in the noiseless case, prewhitening of the data leaves a unitary transformation Q to be estimated in order to retrieve the demixing matrix W [BAMCM97]. We need at least one more bilinear form (other than the inner product) to identify this unitary transformation. The separation algorithm now consists of two steps:

prewhitening Let $V_s = \text{span}\{s_1,\ldots,s_m\}$. Find a basis $z_1,\ldots,z_m$ of V_s such that $g_0(z_k,z_l) = \delta_{kl}$. The $z_1,\ldots,z_m$ components are called prewhitened measurements, since no cross-correlation exists beween these components

estimate Q The remaining unitary transformation Q is found, such that
$\mathrm{Q}\,[z_1,\ldots,z_m]^T = [s_1,\ldots,s_m]^T$

The first step can be performed using principal component analysis. The second step is implemented through the use of joint diagonalization. To solve for the unitary factor we compute $R^{g_l}_{zz}$ for $l = 1,\ldots,M$, and search for a unitary matrix Q such that $QR^{g_l}_{zz}Q^H$ are simultaneously (approximately) diagonal. This can either be implemented using the JADE algorithm [CS93] or the joint Schur decomposition [vdVP96]. In JADE, a set of consecutive Jacobi rotations is performed in order to diagonalize the set of matrices $R^{g_l}_{zz}$ approximately; because of finite sample size effects and noise these similarity matrices are never exactly diagonalizable. After the JADE operation, each of the matrices individually need not be diagonalized exactly, but the criterion function

$$\mathscr{C}(\mathscr{R},Q) = \sum_{l=1}^{M} \text{off}(QR^{g_l}_{zz}Q^H) \tag{5.15}$$

is minimized. Here, the function off$(\cdot)$ of a matrix is the sum of the squared nondiagonal elements, and $\mathscr{R}$ is the set of matrices to be (approximately) diagonalized. The consequence of prewhitening is that infinite weight is put to the data correlation matrix, which may lead to estimation errors in cases with small sample sizes or noise. Recently, an attempt has been made to approximately diagonalize a set of matrices with an arbitrary (not necessarily

unitary) matrix Q [Yer00]. We will not consider this approach, although it would fit well into our framework.

5.3.2 Examples of bilinear forms

The previous formulation allows the incorporation of several existing algorithms for source separation, depending on the choice of bilinear form.

Temporal correlation The SOBI algorithm [BAMCM97] is obtained by choosing

$$g_i(x_1,x_2) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=0}^{N} x_1(t)^* x_2(t-\tau_i) \tag{5.16}$$

Here, inclusion of covariance matrices at different lags can be interpreted as implicitly filtering the mixtures with different filters, hence providing multiple conditions for identifying the unitary factor Q [ZNCM00, MS94b].

Subband filtering If one has prior knowledge about the spectral bands in which the sources are different, linear filters h_1, h_2 can be designed such that the corresponding bilinear form

$$g_{h_1,h_2}(x,y) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=0}^{N} \{ \sum_{k=0}^{N} x(k)^* h_1(t-k)^* \cdot \sum_{k=0}^{N} y(k) h_2(t-k) \} \tag{5.17}$$

is separating the sources. These filters can be *matched filters* that capture templates supposedly present in the sources [KOBF98]. Only sources that exhibit this template will be separated in this approach. Alternatively, the filters may be tailored to the mixture spectra in such a way that (hidden) cross-correlations (i.e. before filtering) between mixtures are maximal. If the sources can be assumed spectrally different in each frequency band, this will separate the sources. However, spectral modeling with autoregressive models can lead to unstable filters [KO99]. A solution based on FIR filters was proposed in [KO00]. Another variation of this idea can be found in [KO98], where the filters are chosen as a fixed orthogonal wavelet filter bank. The level of the wavelet decomposition contains the amount of 'zoom' into the low-frequency region, and determines the potential gain in focussing on particular sets of frequencies. In this approach the amount of adaptivity to source characteristics is limited.

Cyclostationarity In the phase-SCORE algorithm [ASG90] the cyclic cross spectral matrix $R^{\alpha}_{xx}(\tau)$ is defined by $R^{\alpha}_{xx}(\tau) = \sum_{t=0}^{N} x(t)^* x(t-\tau) e^{-j2\pi\alpha t}$. Then the generalized eigenvalue problem

$$\lambda R_{xx} w = R^{\alpha}_{xx}(\tau) w \tag{5.18}$$

is solved. The eigenvectors corresponding to the larger generalized eigenvalues are used as beamforming (demixing) weight for the various signals. The solution to the generalized eigenvalue problem arising with the phase-SCORE algorithm [ASG90] is equivalent to joint

diagonalization of the bilinear forms

$$g_{(\alpha,\tau)}(x_1,x_2) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=0}^{N} x_1(t)^* x_2(t-\tau) e^{-j2\pi\alpha t} \tag{5.19}$$

This allows for an extension of phase-SCORE: in the case of multiple features (α_i, τ_i), one only has to jointly diagonalize the corresponding bilinear forms [Les99].

Time-frequency distributions Belouchrani's time-frequency algorithm [BA97] is obtained by choosing

$$g_{t,f}(x_1,x_2) = \sum_{m,l\in Z} \varphi(m,l) x_1(t+m-l)^* x_2(t+m+l) e^{-4\pi j f l} \tag{5.20}$$

using time-frequency atoms (t, f) and smoothing kernel $\phi(m,l)$. In later experiments in section 5.5 we use a Wigner-Ville distribution as the bilinear form, corresponding to a smoothing kernel identically 1. When using only the significant time-frequency atoms in the separation, we expect cross-terms in the time-frequency representation to be sufficiently suppressed. The

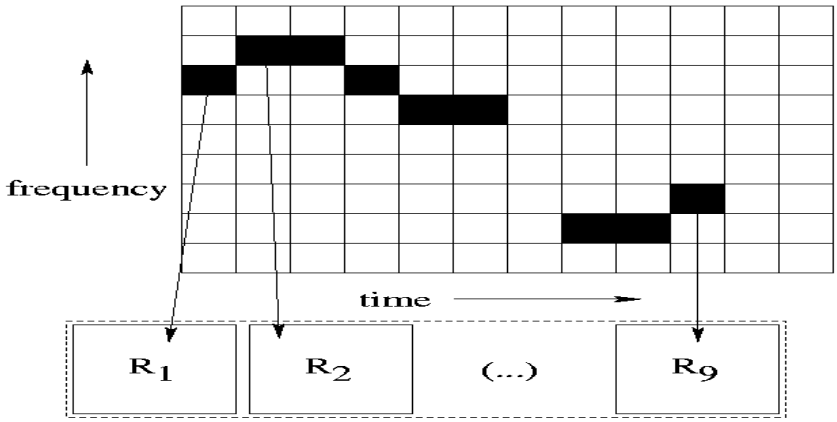


Figure 5.6: Time-frequency source separation: choice of time-frequency atoms determines which similarity matrices are being diagonalized

choice of included time-frequency atoms (see figure 5.6) determines the number of matrices to diagonalize. If this number becomes too large or the matrices are largely uninformative, this may lead to an intractable or suboptimal diagonalization procedure. A solution may be to use a subset of these matrices, e.g. the matrices that represent the time-frequency atoms with the highest energy. The proper choice of spectral and temporal resolution proved to be critical for successful separation of an ensemble of linearly mixed artificial signals with different time-frequency content (see the following example).

Example 5.3: time-frequency source separation

The algorithm is demonstrated in figure 5.7 on a set of synthetic signals with distinct time-frequency signatures. To highlight the time-frequency behaviour of these synthetic signals,

consider the time-frequency plot of two reconstructed sources (figure 5.8). They show respectively a linear chirp (linear increasing frequency, the 'cross'-like figure is due to the symmetric way the spectrum is displayed) and a signal exhibiting a Doppler-shift. We will see later that in acoustic monitoring Doppler shifts may be present. □

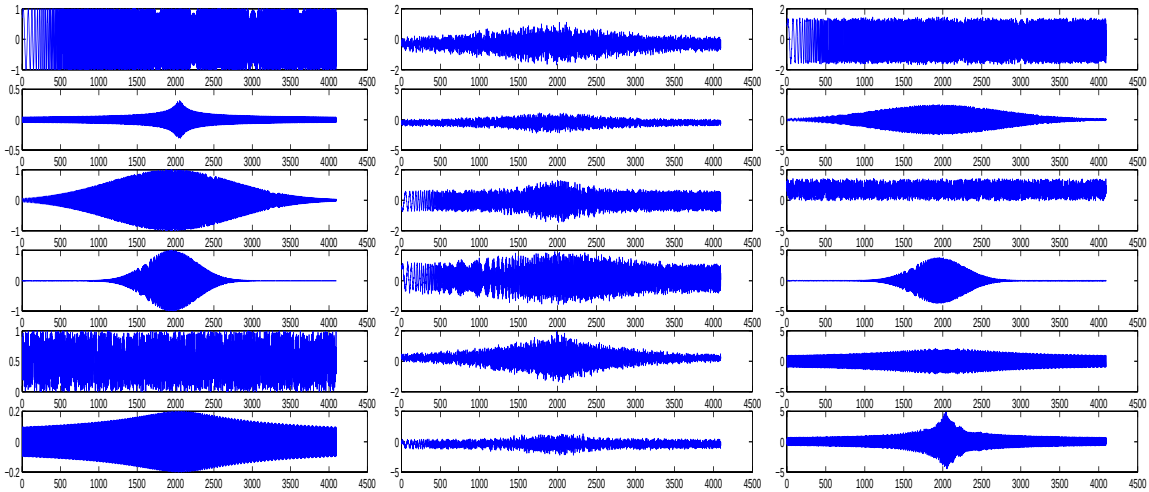


Figure 5.7: Sources (a), mixtures (b) and reconstructions (c) with tf- characteristics

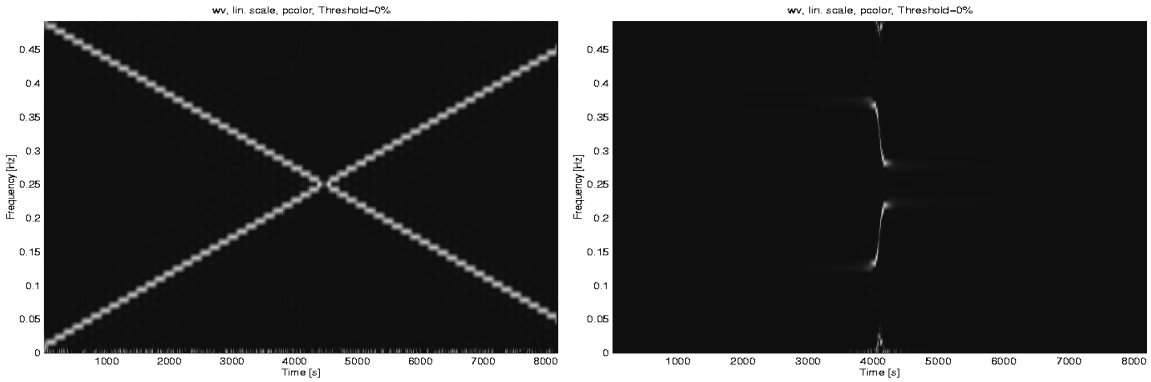


Figure 5.8: Time-frequency behaviour of two reconstructed sources

5.3.3 Experiment: combining multiple separation criteria

In the bilinear forms approach one may include multiple second-order criteria simultaneously for source separation. We investigate the merits of this approach with simulated directional narrowband sources that exhibit both temporal coherence and cyclostationarity. Moreover, a heuristic extension can be made to include higher-order and second-order statistics simultaneously in the separation procedure, analogous to [MPZ99]. We investigate the suitability of this approach again with signals that resemble a setup where a rotating machine is monitored acoustically.

Temporal coherence and cyclostationarity

We mixed two cyclostationary sources analogous to experiment 3.5.3, i.e. two complex modulated analytic AR(1)-sources with random initial phase. These sources were mixed in order to simulate impinging on an array with 5 sensors with directions-of-arrival of -12 and 13 degrees. The length of the signals was 512 samples. Apart from direction-of-arrival, the source signals only differ with respect to their modulation frequency $f_1 = f_0 + \Delta f$. We chose f_0 to be equal to $0.3 \cdot f_s$, where the sampling frequency f_s was defined to be 1. The maximal frequency shift Δf was $0.1 \cdot f_s$. We compare the results obtained with SOBI, SCORE and a combination of both methods, if we vary the frequency shift Δf. We repeat 250 runs of the source separation algorithm (SOBI, SCORE or their combination), and the median *signal-to-inference-plus-noise ratio* (SINR, in dB), averaged for both sources, is computed. We chose the set of delays as $\{1,\ldots,25\}$, i.e. in both SOBI and SCORE we used 25 similarity matrices (and 50 matrices in the combination algorithm). In figure 5.9a the result is plotted for a source SNR of -5 dB. It can be seen that for small spectral difference only SCORE

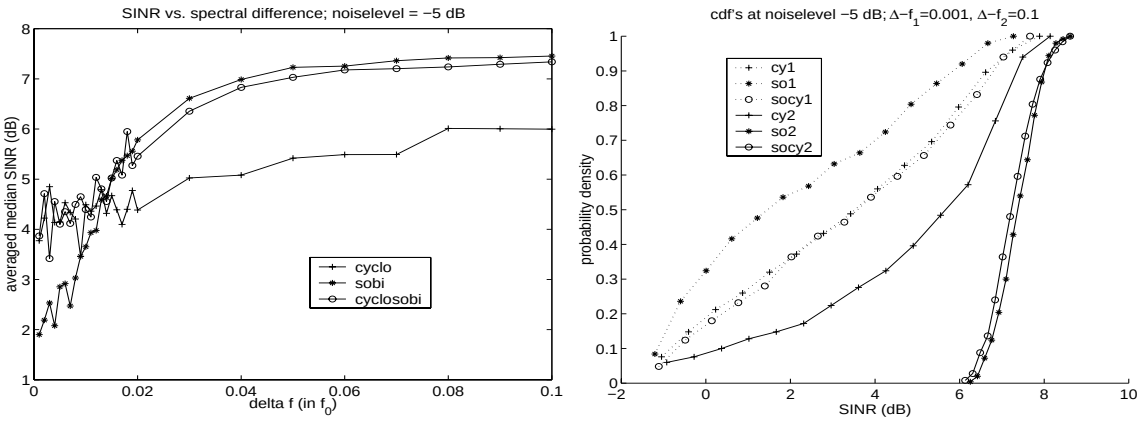


Figure 5.9: a: Average median SNR vs. Δf for SOBI, SCORE (*cyclo*) and their combination, SNR = -5 dB, delays = $\{1,\ldots,25\}$; b: empirical cumulative distribution functions for small (dotted, denoted by *cy1*, etc.) and large (solid, *cy2*, etc.) frequency shift

(denoted as *cyclo* in the figure) will be able to do the separation. For large spectral difference ($\Delta f > 0.01$) there is sufficient information in the lagged covariance matrices, which results in a better performance of SOBI. The combination algorithm (denoted as *cyclosobi*) follows the better performance of both methods: for small spectral shift it behaves like SCORE, whereas for larger shifts it approaches SOBI. We noticed a large variation in 250 repetitions of an experiment. In order to quantify this, we determined the cumulative density of the averaged SINR for both sources over 250 repetitions. In figure 5.9b the empirical cumulative distribution for small and large spectral shift ($\Delta f = (0.001, 0.1)$) are shown for SOBI, SCORE and their combination separately. For the small shift the combination behaves like SCORE, whereas for the large shift it behaves like SOBI. In both cases it follows the maximum performance. From several experiments we noticed that the benefit of combining SCORE with SOBI depends on a proper choice of time lags. This can be seen in figures 5.10 to 5.12, where

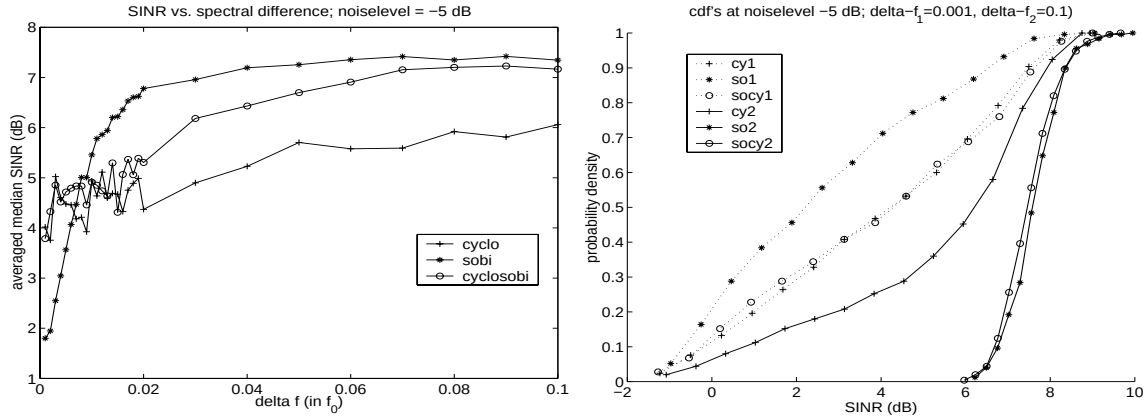


Figure 5.10: a: Average median SNR vs. Δf for SOBI, SCORE (*cyclo*) and their combination, SNR = -5 dB, delays = $\{1,\ldots,5\}$; b: empirical cumulative distribution functions for small (dotted, denoted by *cy1*, etc.) and large (solid, *cy2*, etc.) frequency shift

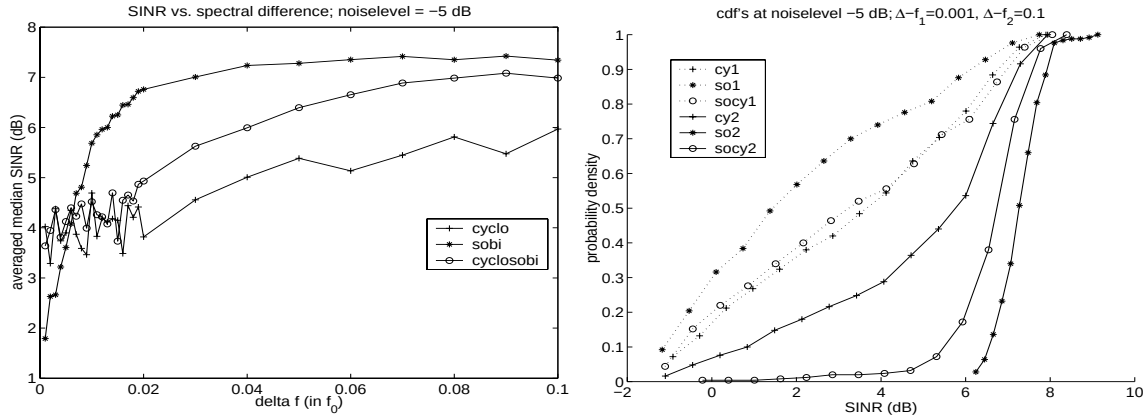


Figure 5.11: a: Average median SNR vs. Δf for SOBI, SCORE (*cyclo*) and their combination, SNR = -5 dB, delays = $\{2\}$; b: empirical cumulative distribution functions for small (dotted, denoted by *cy1*, etc.) and large (solid, *cy2*, etc.) frequency shift

the previous experiment was repeated for delay sets $\tau = \{1,\ldots,5\},\{2\}$ and $\{25\}$. The cumulative distribution was determined for the averaged results for both sources, except for the $\{1,\ldots,5\}$ case (where only the result for the first source was used). However, reconstruction results for both sources were comparable in that experiment. In [ZNCM00] it was shown that for SOBI the proper choice for the delay set τ can be related to the subbands where the sources are spectrally differing, which may be used in the future as a criterion for choosing the set of time lags. We postulate that with suitable choices for the set of time delays, combination of SCORE and SOBI for separation of sources that exhibit both types of structure allows for approximation of the individually best performances in cases where the dominant signal structure (cyclostationarity or temporal coherence) is not known a priori.

In the previous experiments, the two sources were registered on 5 sensors. Now we study

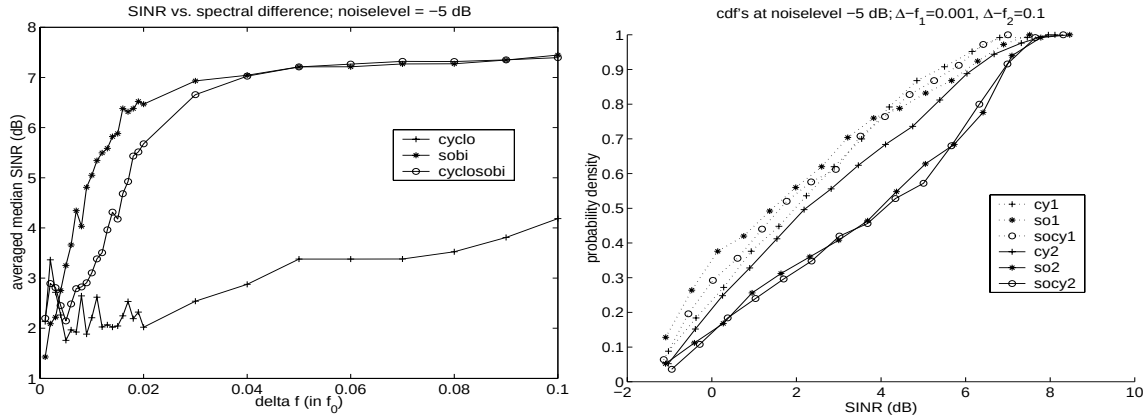


Figure 5.12: a: Average median SNR vs. Δf for SOBI, SCORE (*cyclo*) and their combination, SNR = -5 dB, delays = $\{25\}$; b: empirical cumulative distribution functions for small (dotted, denoted by *cy1*, etc.) and large (solid, *cy2*, etc.) frequency shift

the influence of the number of sensors on the combination performance. The results for 3 sensors and two different SNRs (-5 and 5 dB) are shown in figure 5.13. In all experiments, the set of delays was chosen as $\{1,\ldots,5\}$. Only the empirical distribution functions are shown.

In none of the cases performance improvement is possible with combining. Moreover,

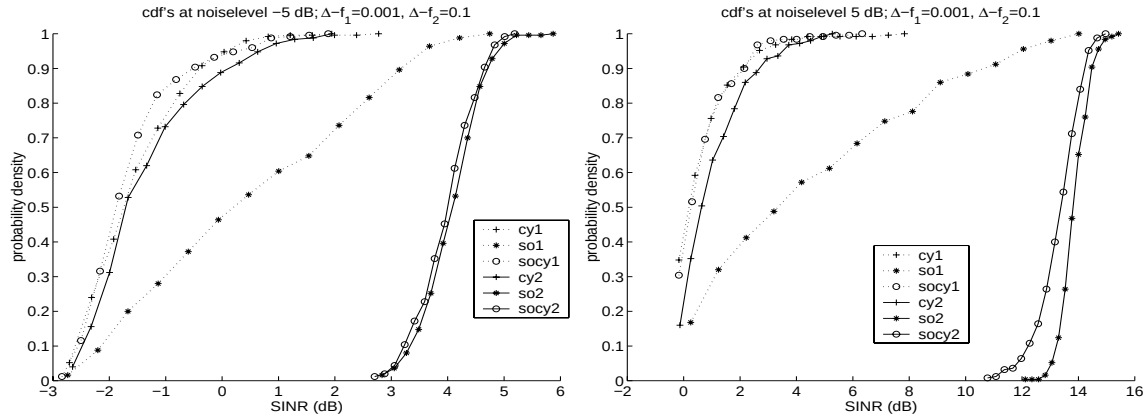


Figure 5.13: Empirical cumulative distribution functions of averaged SNR for small (dotted, denoted by *cy1*, etc.) and large (solid, denoted by *cy2*, etc.) frequency shift. The number of sensors is 3, delays = $\{1,\ldots,5\}$. a: SNR = -5 dB; b: SNR = 5 dB

SOBI outperforms SCORE even in the range of very small spectral shifts. The results for 10 sensors are shown in figure 5.14. In these experiments, the set of delays was now chosen as $\{2\}$ and $\{1,\ldots,5\}$ and the SNR was -5 dB. Again we observe that combination leads to approximation of the individually optimal curves, even for the delay choice that in earlier experiments proved to be very suboptimal ($\tau = 2$). Results for high SNR (20 dB, not shown) indicate that for bad delay choices the combination algorithm is far worse than SOBI for large frequency shifts, while it approximates the SCORE performance for small shifts. For

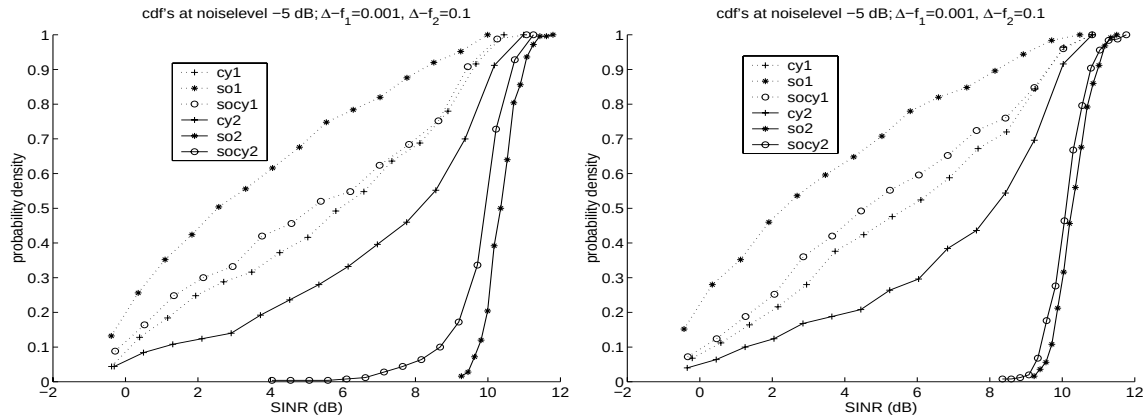


Figure 5.14: Empirical cumulative distribution functions of averaged SNR for small (dotted, denoted by *cy1*, etc.) and large (solid, denoted by *cy2*, etc.) frequency shift. The number of sensors is 10, SNR = 5 dB; a: delays = $\{2\}$; b: delays = $\{1,\ldots,5\}$

good choices, the combination algorithm approaches the best individual performance more closely.

Higher-order and second-order statistics

We generated a set of sinusoidal sources that are impinging on a sensor array from different directions. The (direction dependent) attenuation factors, sensor spacings and direction-of-arrival were set in conjunction with the measurement setup in section 5.5.1. We chose 5 sources to be present simultaneously, where each source was harmonic and had a different frequency (64, 500, 100, 200, 150 Hz, respectively). Their direction-of-arrival were (86, 90, -45, 0, and -25 degrees, i.e. the first two sources virtually in end-fire position, the fourth emitting broadside) and they were attenuated with factors (0.1, 0.9, 0.7, 0.4, 0.3). We took only real signals and mixing matrices into account. White Gaussian noise was added to each sensor, where the magnitude of the noise determined the resulting signal-to-noise ratio. With each replication of an experiment, a dataset was generated with new noise components. A single measurement consisted of 10000 time samples. We studied the influence of additive noise on the separation performance of several algorithms for linear instantaneous ICA. The performance of an algorithm was measured by the cross-talking error[2] in the product of demixing and mixing matrices. We investigated the source separation algorithms JADE, SOBI and a combination of these algorithms. Each separation experiment was repeated 10 times and the average, worst and best results in a run are plotted for each algorithm in figures 5.15 to 5.17. On the horizontal axis the SNR is plotted, on the vertical axis the values for the the reconstruction quality, now expressed in terms of the cross-talking error of equation (4.15). The combination algorithm seems to enable a somewhat robuster separation algorithm than the SOBI algorithm, see figures 5.15(b,c), where covariance matrices at delays

[2]A drawback of the cross-talking error is that is is a nonnormalized quantity. However, it will be zero for perfect separation and it increases with the number of unmixed sources in the ensemble

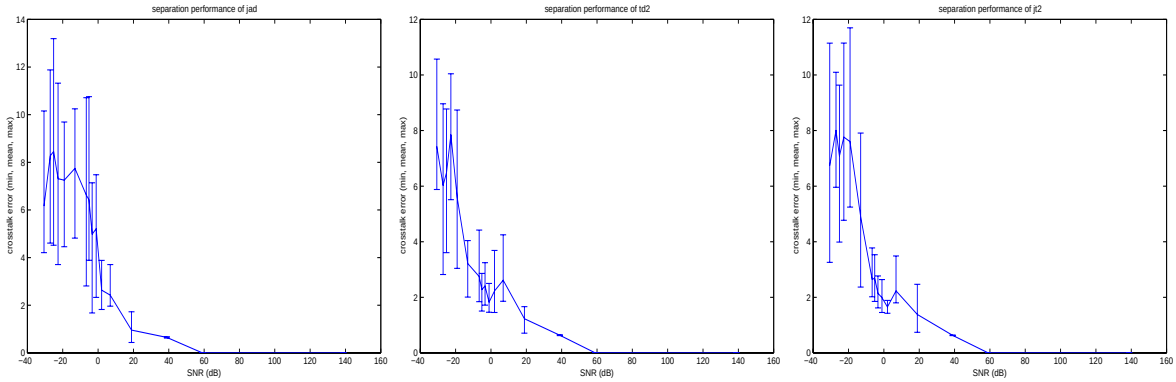


Figure 5.15: Performance on directional sources: JADE (a), SOBI (b), combination (c)

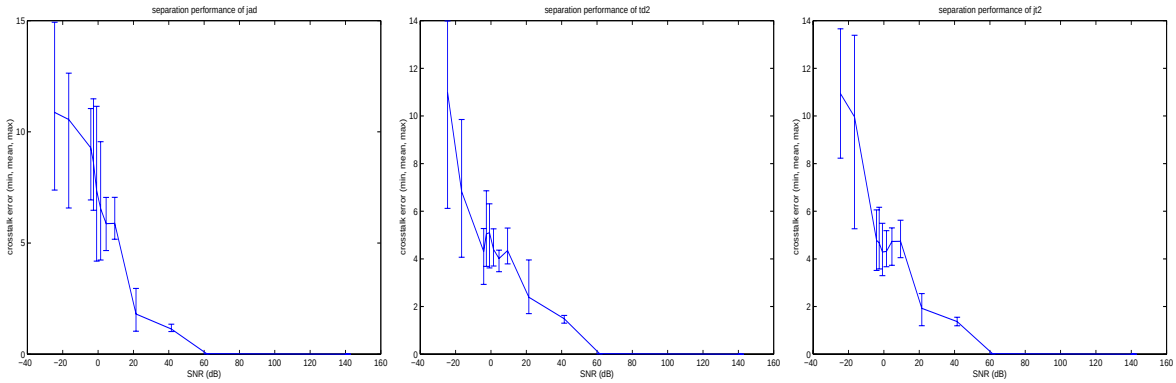


Figure 5.16: Performance on equal angles sources: JADE (a), SOBI (b), combination (c)

$\{1, 6, 11, \ldots, 46\}$ were taken into account. The JADE algorithm produced comparable results. For sources coming from only two different angles (sources 1 and 2 from 90 degrees, sources 3 to 5 from -45 degrees) and with identical attenuation factor for sources 1, 3, 4 and 5, JADE performs much worse, whereas SOBI and the combination algorithm show comparable results (figure 5.16). For sources with equal frequency content (every source contains five

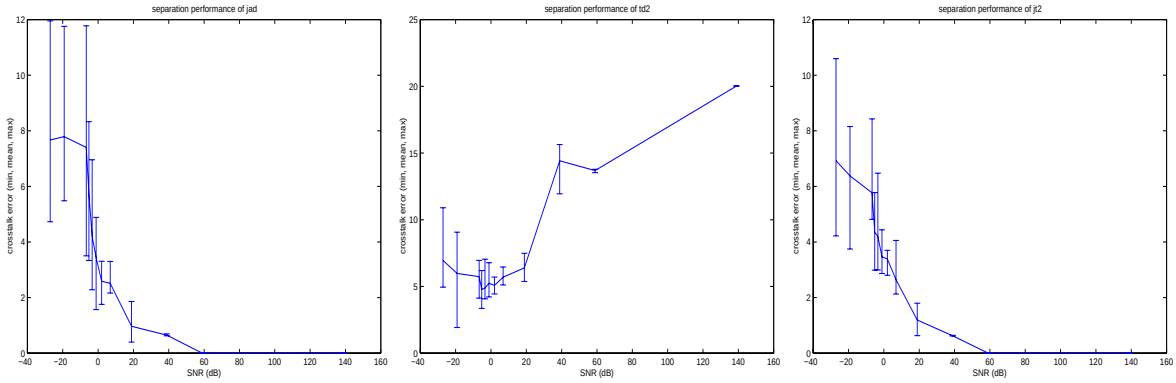


Figure 5.17: Performance on equal spectrum sources: JADE (a), SOBI (b), combination (c)

sinusoidal "subsignals", but the order of the five segments and the attenuation factors are different for all 5 sources), the combination algorithm is still capable of separating the sources, whereas SOBI fails (figure 5.17). Hence, if one does not know a priori if higher-order or second-order structure is a discriminating feature for linearly mixed acoustical narrowband sources, a combination algorithm that takes both types of information into account may be a good choice.

5.4 Separation of convolutive mixtures

We have seen in section 1.2.1 that sources in vibrating structures will both be filtered and mixed. The *convolutive mixture model* is then appropriate:

$$x_i(t) = \sum_{j=1}^{n} \{ \sum_{\tau=0}^{N} \mathscr{A}_{ji}(\tau) \cdot s_j(t-\tau) \}, \qquad i = 1, \ldots, m \tag{5.21}$$

where $\mathscr{A}$ is now a matrix of FIR-filters $\mathscr{A}_{ji}$ and the sources and mixtures are again considered real-valued random variables (see figure 5.18). An additive noise term can easily be included in the model. The aim is to find the matrix of FIR-filters that jointly demixes the measurements (to undo the spatial mixing operating) and equalizes the sources (to undo the temporal filtering of the transmitting medium).

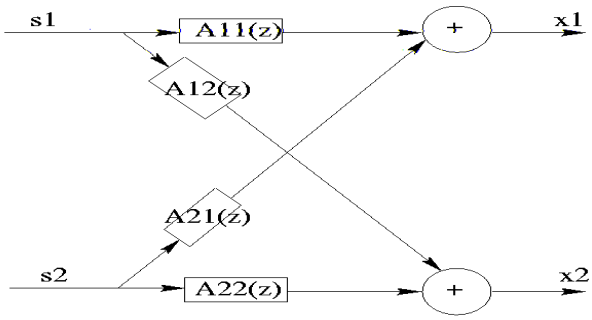


Figure 5.18: Convolutive mixtures in a 2×2 (2 sources, 2 sensors) scenario

This problem can be solved in both the time domain [PS99, MIZ99, YW96, NTJ95, GCD99] and the frequency domain [FSL98, FS99, AG99, Sma98, MP99]. Drawbacks of the (entropy-maximization) *temporal approach* are that [Sma98] the temporal whitening operation (equalization) often contributes more to the entropy increase than the unmixing (it is much easier to find temporally whitening filters than spatially whitening filters, while the filters should perform both operations at once!) and that noncausal filters may be obtained. This calls for constraints on the optimization procedure or the use of (less stable) IIR filters, which leads to complex optimization procedures. The *frequency domain* methods often use the fact that a convolutive mixture corresponds to an instantaneous mixture in each (sufficiently narrow) frequency band [FS99, MP99]. One decreases the $W\tau$-product (see section 4.3) by decreasing the effective bandwidth W to make the mixture approximately instantaneous in each band.

However, the permutation indeterminacy now occurs at each frequency band, so one has to reorder the reconstructions per frequency band before an inverse Fourier-transformation is done, e.g. by using assumptions on the smoothness of the spectral envelope [MIZ99].

Algorithm by Nguyen Thi and Jutten

An example of the time-domain approach to convolutive demixing is the algorithm by Nguyen Thi and Jutten [NTJ95]. In this algorithm, a convolutive mixture of two sources s_1, s_2 measured with two sensors x_1, x_2 (also referred to as a 2×2 convolutive mixture) is modelled in the Z-domain as

$$\begin{cases} x_1(z) &= \mathscr{A}_{11}(z) \cdot s_1(z) + \mathscr{A}_{21}(z) \cdot s_2(z) \\ x_2(z) &= \mathscr{A}_{12}(z) \cdot s_1(z) + \mathscr{A}_{22}(z) \cdot s_2(z) \end{cases}$$

where the filters $\mathscr{A}_{ji}(z)$ are linear and causal FIR filters of order M

$$\mathscr{A}_{ji}(z) = \sum_{k=0}^{M-1} a_{ji}(k) \cdot z^{-k} \tag{5.22}$$

The model is simplified by assuming that each sensor is close to one source, which leads to filters $\mathscr{A}_{11}(z) = \mathscr{A}_{22}(z) = 1$. The aim is now to estimate the inverse filter matrix

$$\mathscr{B}(z) = \begin{bmatrix} 1 & \mathscr{B}_{21}(z) \\ \mathscr{B}_{12}(z) & 1 \end{bmatrix} \tag{5.23}$$

such that

$$\mathscr{B}(z)\mathscr{A}(z) = \begin{bmatrix} \mathscr{H}_{11}(z) & 0 \\ 0 & \mathscr{H}_{22}(z) \end{bmatrix} \tag{5.24}$$

Note that the solution can be determined blindly only up to a permutation and a filtering, i.e. a matrix $\mathscr{B}(z)$ that results in a product matrix $\mathscr{B}(z)\mathscr{A}(z)$ with zeros and FIR filters $\mathscr{H}_{ii}$ interchanged with respect to the above matrix also separates the sources, while any remaining FIR filter $\mathscr{H}_{ii}$ suffices. The inverse matrix to be identified consists of FIR filters only and the inverse 'close sensor' filters are constrained to unity. The algorithm performs an adaptive cancellation of higher-order cross-statistics, e.g. by finding the zeros of $E[\hat{s}_i^2(t)\hat{s}_j(t-k)]$ with the update rule

$$c_{ij}(t+1,k) = c_{ij}(t,k) + \mu \hat{s}_i^3(t)\hat{s}_j(t-k) \tag{5.25}$$

where $c_{ij}(t,k)$ is the estimated demixing filter coefficient at tap k and time t (using a recursive demixing architecture [NTJ95]) and μ is the (positive) step size. It was shown in [CSL96] that convolutively mixed harmonic signals may be separated on the basis of their higher-order statistics.

5.5 Experiments: separation of rotating machine noise

In this section, we investigate the feasibility of blind source separation for rotating machine monitoring applications. Three different experiments are performed: a controlled experiment with the submersible pump and experiments with two real-world machine setups (all described in chapter 1). Both vibrational and acoustical demixing is encountered in these case studies.

5.5.1 Separation of acoustical sources

The measurement setup for this experiment has been described shortly in section 1.5.5. The measurements were lowpass-filtered, so the sources remaining in the measurements were wind-, car- and machine noise. The inter-element spacing was such that beamforming techniques could not be used for source separation (severe spatial aliasing was present). Moreover, the fact that vegetation and car noise were impinging on the sensor with similar directions-of-arrival was expected to prohibit the use of beamforming for source separation. In the measurement setup, no significant reflections from the environment or from the ground were expected. A calculation of the maximum $W\tau$ product for the sources and the inter-sensor delays, assuming a maximum frequency of interest of 75 Hz and a maximum direction-of-arrival of 45 degrees revealed that the instantaneous mixing assumption might hold approximately for the first 4 sensors in the array ($W\tau \approx 0.7$). A first attempt at source separation using delay-and-sum beamforming (we time-aligned the signals with respect to the most dominant source using cross-correlation between the sensors) corroborated our expectation that beamforming is less suitable for this setup. In the spectra corresponding to the aligned measurements, mixing can still be observed, figure 5.19(a). We attempted further separation by using blind instantaneous source separation with bilinear forms. The

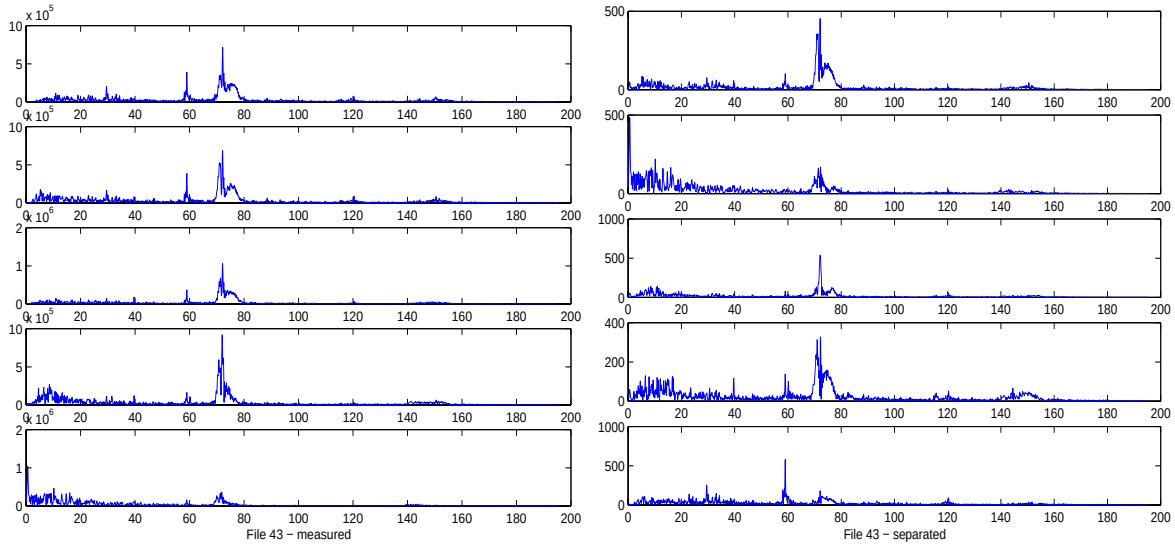


Figure 5.19: a: Spectra of outdoor acoustical measurements; b: separated components

time-frequency plot of a typical measurement is shown in figure 5.20(a). The rotating machine (stationary component around 60 Hz) was not visible in any time-frequency plot of the measurements. After prewhitening we obtained spectra that contained several components where machine and car were severely mixed. Since one of the sources is moving (the car), we can take the nonstationarity of the signal due to this source into account. We applied the time-frequency algorithm of equation (5.20) with kernel $\phi(m,l) = 1$ (so we used a Wigner distribution) to the measurements, using the 20 matrices that represent the most energetic tf-atoms. The signals were downsampled to plm. 200 Hz and the time-frequency atoms had a width of 50 time samples and approximately 1.5 Hz. This resulted in the spectra of figure 5.19(b). The rotating machine is separated into the fifth component: harmonic peaks at 30 and 60 Hz are clearly discernible. Its time-frequency signature is plotted in figure 5.20(b).

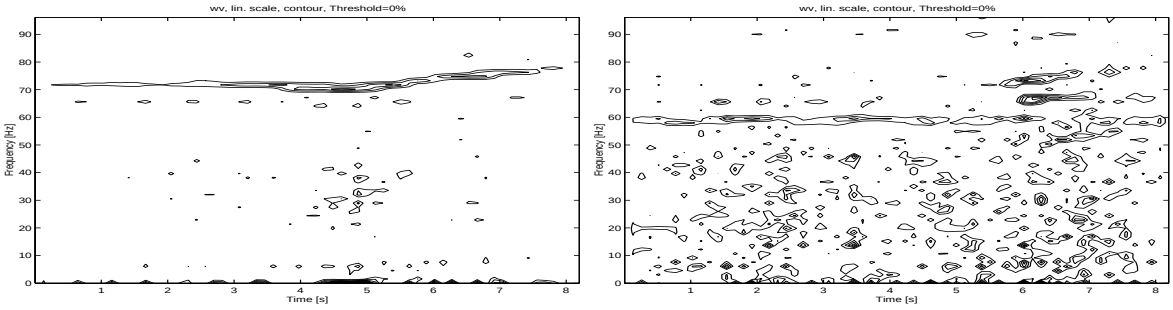


Figure 5.20: Time-frequency plot of (a) a typical measurement, and (b) a component consisting of rotating machine plus wind

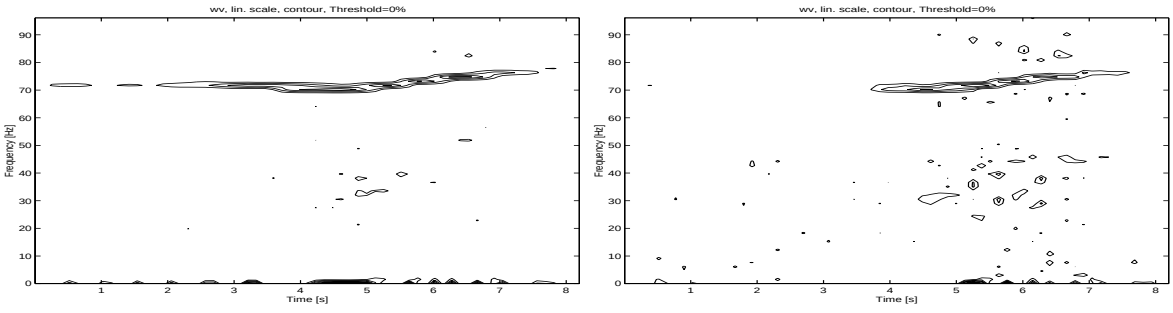


Figure 5.21: Time-frequency plot of 2 independent components in acoustic measurements. a: car signature; b: spurious incomplete car signature

The second component is predominantly a wind component and the first component contains mainly the car signature, see figures 5.19(b) and 5.21(a). The component due to sensor 5 seems to be ignored in the separation procedure: comparison of the tf-plots corresponding to component 5 in figure 5.19(a) and component 2 in figure 5.19(b) revealed nearly identical tf-signatures. This can be explained by the fact that sources registered on this sensor cannot be considered as phase-shifted versions of registrations at other sensors. However, care

should be taken that the size of the signal subspace is estimated correctly. In our experiment we assumed 5 spatially coherent sources, where there are only two present (along with the coloured noise due to wind and vegetation). The time-frequency signature corresponding to component 4 is shown in figure 5.21(b). Since we forced a solution into 5 components with distinct time-frequency behaviour (according to the most energetic time-frequency descriptors), different temporal parts of the car signature are repeated in spuriously reconstructed components 3 and 4. The important point to note is, however, that the rotating machine component (the main spatially coherent interference source) could be largely separated from the car noise. We remark that similar decompositions could be obtained by different choices for temporal bandwidth (10 and 500 time samples, respectively) and number of included tf-atoms (100) in the time-frequency algorithm. Also, application of conventional separation algorithms like SOBI and JADE led to comparable results.

5.5.2 Demixing of two connected pumps

In a laboratory test bench, a small water pump was attached to a larger submersible pump, see figure 5.1(b). The small pump runs at 100 Hz. The running speed of the larger pump is 1500 RPM at a nominal driving frequency (of the frequency converter driving the machine) of 50 Hz. Due to slip and load differing from the nominal load, this results in an effective fundamental frequency around 28 Hz at a driving frequency of 60 Hz. In the spectrum of the submersible pump, harmonics of this fundamental are clearly visible. In figure 5.22(a) the uppermost spectrum corresponds to a measurement on the small pump, where the small pump is the only operating machine. The other spectra are measurements at two different positions on the large pump, where only the large pump was running. Here, channel 1 denotes measurements from the sensor on the small pump, channels 2 - 4 represent orthogonal measurement directions from a triaxial sensor that is mounted on the middle of the large pump and channel 5 is from a sensor at the bottom (the vane) of the large pump. When both

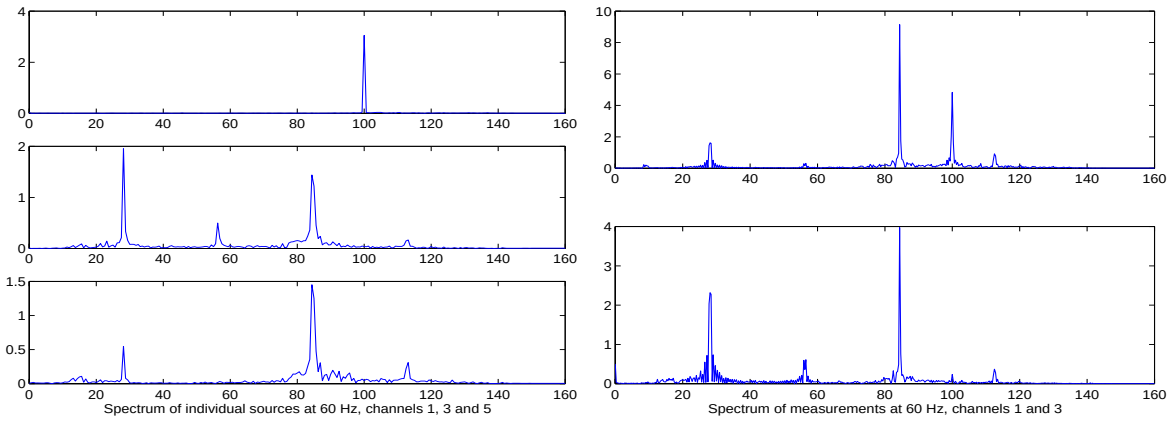


Figure 5.22: a: Measurements at driving frequency 60 Hz while one pump is running, on small pump (top) and two positions (middle: channel 3; bottom: channel 5) on large pump; b: measurements while both pumps are running simultaneously, on small pump (top) and on large pump (bottom, channel 3)

pumps are running simultaneously, the measured spectra on the small machine and on the position corresponding to channel 3 on the large machine are shown in figure 5.22(b). Since there are only two sources present, we applied the Nguyen Thi-Jutten algorithm to pairs of mixed sources. Each time, the mixture signal from the small pump was paired with a mixture signal from the large pump. The demixing result using filter order 14 and channels 2 and 3 is shown in figure 5.23. It can be observed that the harmonic series due to the large pump is

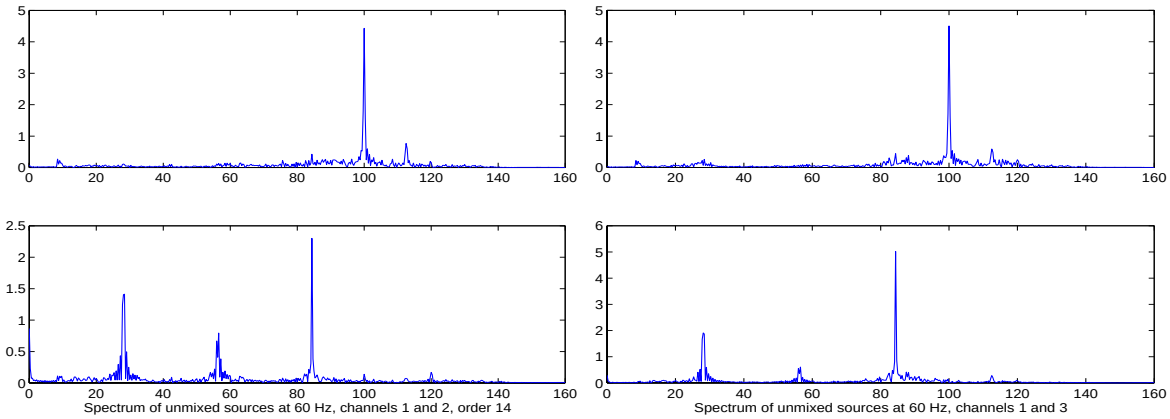


Figure 5.23: Reconstruction of small pump (top figures) and machine signature (bottom) on two different positions (a. channel 2; b. channel 3) on large pump

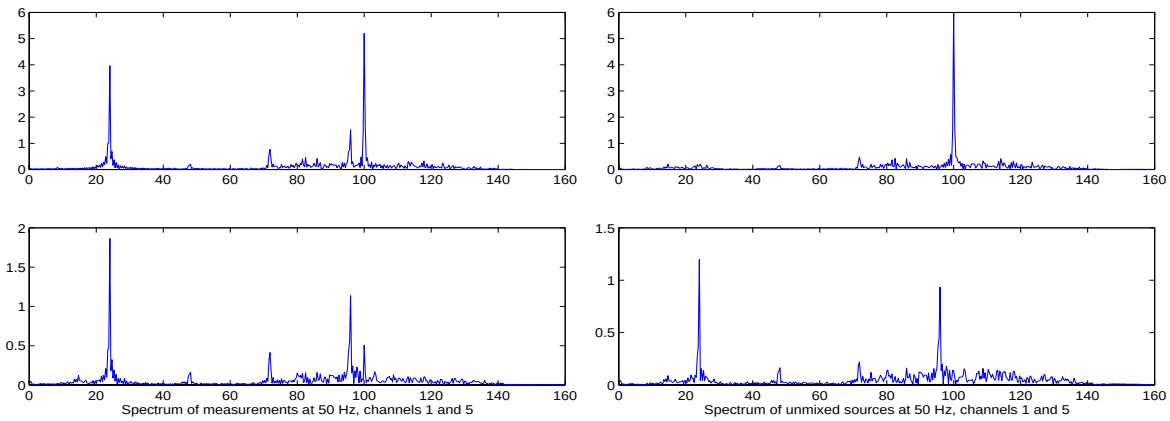


Figure 5.24: a. Measurements at driving frequency 50 Hz while both machines are running simultaneously, on small pump (top) and on large pump (bottom, channel 5); b: reconstruction of small pump (top) and machine signature (bottom, channel 5)

separated from the small pump contribution. We repeated the experiments on measurements with the large pump running at 23 Hz (driving speed 50 Hz). The measured spectra when both machines are running are shown in figure 5.24(a). Note that 1st, 3rd and 4th harmonics of running speed are clearly discernible at both machines. The separated signatures using a filter order of 14 are shown in figure 5.24(b). These decompositions are representative of most decompositions we obtained using different sensor combinations. Sometimes there re-

mains a small amount of cross-talk in the reconstruction obtained with channel 5, which is possibly due to the fact that channel 5 is more remote to the oilpump than channels 2 and 3. In general, it appears that the main benefit is in demixing the big pump signature from the smaller pump (gains up to 2 dB can be obtained); demixing of the (usually small) contribution of the small pump to the big pump can lead to a somewhat distorted big pump signature, which may offset the demixing gain (in terms of SINR).

5.5.3 Vibrational artifact removal

Vibration was measured on a pumpset in a pumping station, in which a gearbox fault (severe piting in a gearwheel) was present, section 8.4.1 Mounted onto the gearbox casing was a small oil pump. We measured the vibration on gearbox and oilpump casing, when *only the oilpump* was running. In figure 5.25(a) the corresponding vibration spectra are shown. Channel 3 (top subplot) is measured on the gearbox, channel 7 (bottom subplot) is measured on the oilpump. Measurement channels 1 and 3 correspond to a sensor position near the driving shaft of the gearbox, channels 4 and 5 correspond to positions at the top and bottom of the gearbox casing. The oilpump predominantly emits vibration at multiples of 25 Hz.

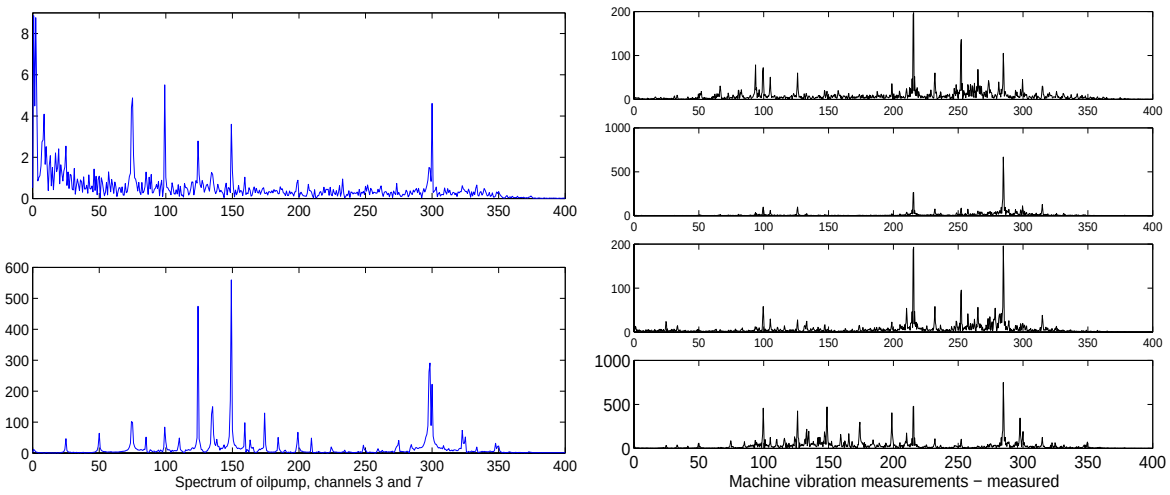


Figure 5.25: a: Vibration spectrum due to oilpump, measured at gearbox (top, channel 3) and oilpump (bottom, channel 7); b: vibration with both machines running simultaneously, measured at gearbox (top three plots, channels 1, 3, and 5) and oilpump (bottom, channel 7)

The oilpump contribution is observable on the large machine casing as well, though small in amplitude. If both machines are running simultaneously, we measure the spectra of figure 5.25(b) on several positions at the large machine (top three plots) and at the oilpump (bottom plot).

Instantaneous separation

The signals were downsampled to approximately 400 Hz and an inspection of the eigenvalues of the data covariance matrix revealed that the signal subspace contained 2 to 3 components,

see figure 5.26(a). After performing time-frequency ICA on the multichannel measurement (spectral resolution in algorithm was 1024 bins and 50 most energetic atoms were used), the decomposition shown in figure 5.26(b) was obtained. The first component consists mainly of the gearmesh related component at 285 Hz, whereas the second component contains predominantly structural vibration of the oil pump. The third component contains mainly the fundamental gearmesh frequency (216 Hz), which was high due to the gearbox fault. This component was measured on the oil pump as well, see figure 5.25(b), 4th channel, but suppressed in the 'cleansed' oil pump measurement, see figure 5.26(b), 2nd channel. Prewhitening is clearly insufficient for source separation, see figure 5.26(a), since the peaks at 216, 285 and around 100 Hz are present in all three components. We noticed that proper choice of frequency band and time shift was critical for the performance of the algorithm: bad choices did not allow for significant improvement over PCA. In this decomposition, the relevant gearmesh-related frequencies are removed from the oilpump signature. However, both frequencies due to gearmesh are distributed over two different components, which may indicate that significant time delays are present between different registrations of the gearmesh components on the sensor array. This suggests that for the mechanical structure at hand the mixing process should be modelled as convolutive.

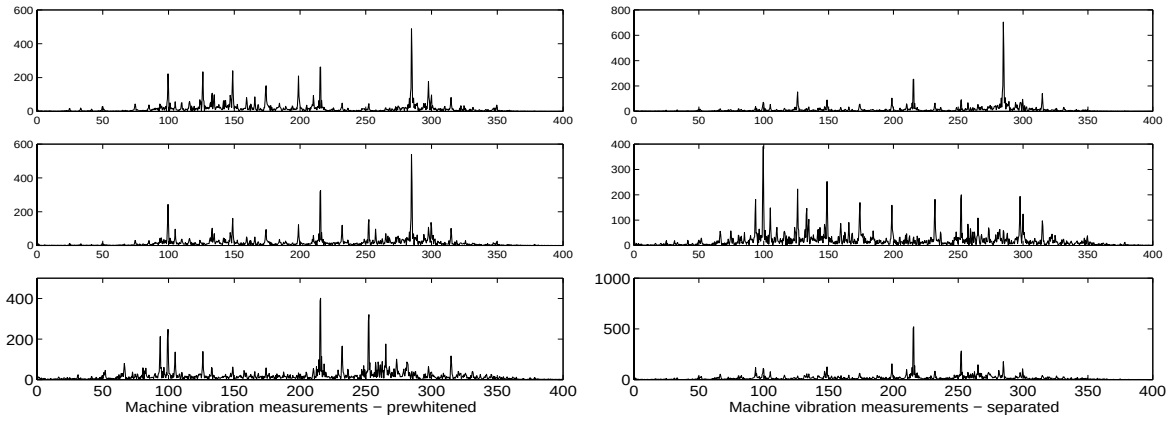


Figure 5.26: a: Spectra of prewhitened vibration measurements; b: separated components using instantaneous tf-separation algorithm

Convolutive separation

We applied the Nguyen Thi-Jutten algorithm to pairs of mixtures, analogous to the laboratory experiment with the two connected pumps. In figure 5.27(a) the mutual cross-talk on both machines is again visible from the spectra measured at the same positions as in figure 5.25, where both machines were running simultaneously. A convolutive demixing with filter order 22 yielded the reconstructions shown in figure 5.27(b). The oilpump contribution to the large machine is suppressed slightly; however, the removal of the large peaks at 216 and 285 Hz (related to gearmesh) from the oilpump signature is much more significant. Note that the small peak around 210 Hz in the oilpump reconstruction should not be confused with the

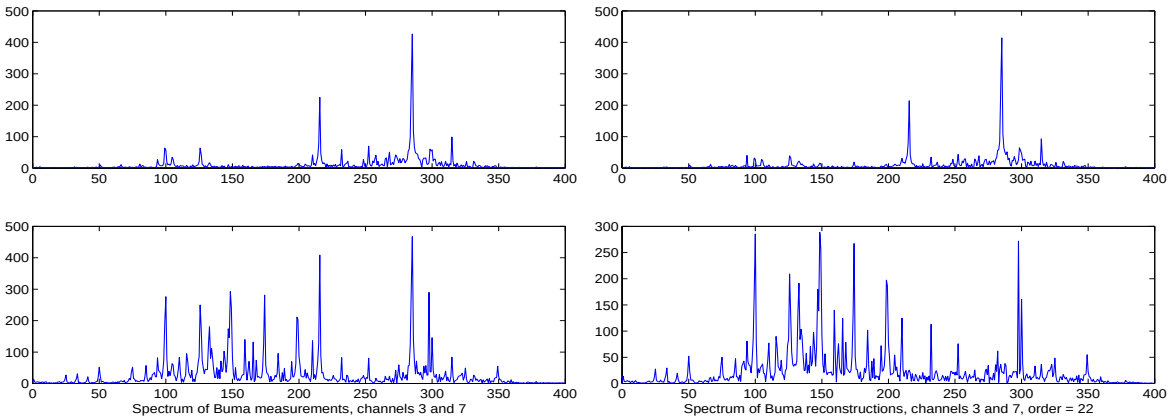


Figure 5.27: a: Measurements while both machines are running simultaneously, on gearbox casing (top) and oilpump (bottom); b: reconstructed gearbox signature (top) and oilpump signature (bottom)

gearmesh component at 216 Hz. The demixing results depend on a proper choice of the filter order. Demixing with a filter order of 45 gives comparable results. Demixing with much smaller or much larger filter order yields unsatisfactory or unstable results for the 2×2 convolutive demixing case, which gives an indication of the time constants in the mixing process of the system. The (spurious) separation of the 216 and the 285 Hz components in the 2×3 (2 sources, three sensors) instantaneous demixing case of the previous section indicates that a proper estimation of the number of underlying sources and use of the proper mixing model is important for blind separation of machine signatures.

5.6 Discussion

When monitoring rotating machines, it is valuable to determine a machine signature that is free from interfering noise due to neighbouring machines or due to the environment. This will allow a focus on the relevant spectral signature without spectral masking effects caused by interferences (that may cause false alarms). We use the spatial diversity and redundancy in a multichannel measurement of machine noise to separate the machine signature blindly from disturbing noises in the scene. Based on an *analysis of the mixing process* that takes place in mechanical structures and in sound propagation, we indicate that both instantaneous and convolutive mixtures may be obtained in machine monitoring.

When separating sources based on the *MDL-principle*, elements of SOS-based and HOS-based approaches are combined. The structure of the cost function is very similar to HOS-based approaches, where the *summed source entropies* term is now replaced with a *summed source complexities* term. Assuming Gaussian random variables with temporal coherence, the dimensionality of the subspace in which each reconstructed source signal resides is used to measure complexity. A differentiable cost function can be derived using this criterion [Paj99, YP98]. With MDL-ICA we were able to separate mixtures of overlapping harmonic series, where conventional HOS-based algorithms were not adequate. Proper choice of the

time lag L (which determines the dimensionality of the time delay embedding of each signal) was not important to separate mixtures of harmonic series. However, for a larger number of sources the MDL-based ICA algorithm suffers from convergence problems.

We showed that using the *bilinear forms framework* it is possible to include several types of temporal signal structure simultaneously for separation of instantaneously mixed sources. This can be beneficial when the sources exhibit several types of structure and it is not clear beforehand under which conditions each criterion can be used for separation. Specifically, it allows the use of particular rotating machine related signal structure like cyclostationarity and time-frequency structure. For simulated directional sources, we showed that combination of different signal characteristics may give rise to a more robust separation procedure. As a heuristic extension, HOS may be included simultaneously with a bilinear form to enhance the robustness of the separation algorithm, e.g. when sources have similar spectra but impinge from different directions.

In three real-world case studies *we showed experimentally that the relevant machine component can be separated* out of a multichannel acoustical or vibration measurement. In the first case, a (stationary) rotating machine source could be separated from the contribution due to a (nonstationary) car using the time-frequency algorithm, which can be seen as an instance of the bilinear forms framework. In the second case (involving convolutive mixtures), demixing results improve significantly when a convolutive demixing algorithm is employed. Both observations are in accordance with our prior analysis of the mixing processes in rotating machine applications. We conclude that BSS is a feasible approach for blind separation of distorted rotating machine sources.

Part III

Learning methods for health monitoring

Chapter 6

Methods for novelty detection

6.1 Introduction

A central task to be performed in a method for automatic machine health monitoring is the detection of a fault or anomaly. Typically, only measurements at a machine's normal operating conditions are available. In rare cases there exist histories of fault development in a machine or measurements of a complete set of possible anomalies. The number of measurements on one particular machine can be large, but the *total number of machines* under investigation is usually very limited. Characterizing the health state of a machine may require high-dimensional feature vectors. Hence, in designing a method for detection of faults in machines one will often encounter a small-sample size problem [TNTC99]. When building a machine-specific system, the number of 'effective samples' is usually much smaller than the total number of feature vectors extracted from a measurement signal. Measurements that are sustained for several minutes can be segmented in smaller frames, since the rotating machine vibration signal is usually stationary over several machine revolutions. The set of feature vectors resulting from this procedure is indicative of the health state under a particular operating mode and is expected to cluster around a prototype feature vector. The position of the clusters is determined by the health state and the operating mode; since normal machine behaviour can be fairly dependent on the operating mode, all modes that may occur in practice should ideally be included in the measurements. This will often lead to a (relatively) small set of prototype vectors in a high-dimensional feature space.

In this chapter we investigate methods for the detection of samples that do not resemble the set of learning samples from the normal domain. In this approach, no measurements from faulty situations are needed. This parallels the approach taken at Los Alamos National Laboratory (USA) to damage identification in structural and mechanical systems. It was noted [FD99] that the statistical pattern recognition approach to fault detection is more apt than the classical model-based approach. Often it is difficult to learn an explicit model of the structure under investigation; however, damage will significantly alter stiffness, mass and energy dissipation properties of the system, which will be reflected in measurements of the dynamic response of the system. Moreover, the proper learning paradigm was reported to be *unsupervised learning*, since data from damaged structures are often not available. At

LANL, damage identification in bridges is undertaken, and inducing damage to a bridge in a controlled manner is rarely possible. For the problem at hand, it is difficult to acquire a representative learning set that completely describes the normal behaviour of a machine or structure. It is costly to measure the system response at all operating modes of a machine; a priori knowledge about the subset of modes to measure is often not available. As stated before, effective sample sizes may be low. Therefore, our approach will be to study methods that describe the *domain* of the learning set, rather than use density estimation methods. After a description of the normal machine behaviour, anomalies are expected to show up as significant deviations from this description. Hence, we address the subtasks "domain approximation" and "novelty detection" in the scheme of figure 2.4.

We start with a short review of the model-based approach to fault detection and describe a neural network structure that has been proposed in the literature for this purpose, the wavelet network. Based on initial experiments, we observe two drawbacks of the wavelet network for fault detection. Then we proceed to the description of three methods for novelty detection that are suitable for description of datasets with small sample sizes. First, we describe the use of Self-Organizing Maps for novelty detection. Second, we introduce the $k-centers$ method, which approximates the domain of a dataset by extracting a subset of the samples in the learning set. Third, we briefly describe a method for data description using the support vector paradigm [Tax01]. Then we study the feasibility of novelty detection in a rotating machine application, where we consider the choice of features and sensors for machine health monitoring. Finally, we use Self-Organizing Maps for leak detection in pipelines using hydrophone measurement signals.

6.2 Model-based novelty detection

The common approach to model-based fault detection and isolation (FDI) is to model the input-output relationship of the system under investigation in a certain health state. If only measurements from the system in normal conditions are available, this leads to a fault detection method that is called *residual generation*: changing system behaviour due to wear or faults will lead to large errors between predicted outputs and real outputs, i.e. the previously learned system transfer function becomes obsolete [PLTU99, AMLL98, Sta00, MH99, NC97].

6.2.1 System modelling with wavelet networks

Neural networks have been proposed for modelling of dynamical systems, e.g. [PLTU99]. Usually, the dynamics of the system is represented in a delay vector (section 3.2.1), and the system is assumed to produce outputs

$$Y_t = [y_t, \ldots, y_{t-T}] \tag{6.1}$$

as a function of previous inputs U_t

$$U_{t-1} = [u_{t-1}, \ldots, u_{t-1-T}] \tag{6.2}$$

and previous outputs, according to a nonlinear system transfer function f,

$$y_t = f(Y_{t-1}, U_{t-1}) \tag{6.3}$$

The nonlinear function f is then approximated with a neural network, which is a feasible approach because of the *universal approximation* property of a feedforward neural network (see chapter 2). Since the dynamics of real-world systems often spans several scales, the *wavelet neural network* has been proposed for system modelling.

Function approximation with wavelets

Learning a function with neural networks involves the positioning and scaling of transfer functions in the network along with weighting of the inputs to the network. For system modelling, the inputs and outputs to the network are samples of the dynamic function to be learned, mentioned in equation (6.3). A wavelet network is a multilayer network having wavelet basis functions in the hidden layer.

The computation obtained with a wavelet network is similar to the reconstruction of a function from a finite set of wavelet coefficients. The wavelet transformation was introduced in section 3.5.1. A wavelet transformed signal $s(t)$ can be recovered from its wavelet components $\psi_{a,\tau}(t)$ by the inversion formula [SN96]

$$s(t) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W_{s,\psi}(\tau, a) \psi_{a,\tau}(t) \frac{da d\tau}{a^2} \tag{6.4}$$

We can look upon the signal $s(t)$ as a *function* of t that is to be approximated. In the context of dynamic system modelling, we aim at reconstruction of a system transfer function f and we will use this terminology in the sequel. In formula (6.4) the integrals run over infinity. In a practical function approximation task, however, f has to be recovered from a finite number of wavelet components. In other words, f has to be reconstructed from *samples* of the continuous wavelet transform. This means that in equation (6.4) only a discrete number of wavelets can be used for reconstruction. When these wavelets constitute an *orthonormal* basis of $L^2(\mathbb{R})$, perfect reconstruction is still possible. When a discrete set of wavelets $\psi_{j,k}^{b_0} = 2^{-j/2}\psi(2^{-j}x - kb_0)$ forms a *frame*, i.e.

$$A\|f\|_2^2 \leq \sum_{j,k \in Z} |< f, \psi_{j,k}^{b_0} >|^2 \leq B\|f\|_2^2 \tag{6.5}$$

for some constants $A, B \in \mathbb{R}_+$, and $b_0 \in \mathbb{R}_+$, only *approximate* reconstruction can be done. The approximation to f is then given by

$$f_{app} = \frac{2}{A+B} \sum_{j,k \in Z} < f, \psi_{j,k}^{b_0} > \psi_{j,k}^{b_0} \tag{6.6}$$

where $< \cdot, \cdot >$ denotes the inner product in $L^2(\mathbb{R})$. The approximation becomes better when B/A is closer to one (i.e. the frame becomes more tight).

The wavelet theory can be extended to functions $f : \mathbb{R}^n \mapsto \mathbb{R}$, for which a wavelet $\psi : \mathbb{R}^n \mapsto \mathbb{R}$ must be found such that the family of n-dimensional wavelets [ZB92]

$$\Phi = \{|\sqrt{D_k}|\psi[D_k(\mathbf{x} - \mathbf{t}_k)]\} \tag{6.7}$$

forms a frame. Here, $\mathbf{t}_k \in \mathbb{R}^n, D_k = \text{diag}[\mathbf{d}_k], \mathbf{d}_k \in \mathbb{R}^n_+$ and $k \in \mathbb{Z}$. Moreover, $|.|$ denotes the determinant of a matrix and boldface symbols indicate n-dimensional variables. Then the approximation

$$g(\mathbf{x}) \approx \sum_{i=1}^{N} w_i \psi[D_i(\mathbf{x} - \mathbf{t_i})] + \bar{g} \tag{6.8}$$

can be used to reconstruct f. It can be shown [ZB92] that the frame property is satisfied when we set ψ to the function $\prod x_i e^{-\frac{1}{2}x_i^2}$, the direct product of one-dimensional Gaussian derivatives. Note that for the multi-dimensional case the matrix D_k plays the role of the scaling parameter a in the scalar case: it consists of entries $d_{ij} = \frac{1}{a_{ij}},\ i = 1,\ldots n,\ j = 1\ldots,N$, where N is the number of wavelets used for the approximation.

Wavelet networks

Reconstruction formula (6.8) can be emulated with a multilayered Perceptron (see figure 6.1) consisting of one hidden layer of "wavelons", i.e. units i with activation $D_i(\mathbf{x} - \mathbf{t_i})$ and transfer function ψ, and one output with inner product activation and linear transfer function. A bias $\bar{g}$ is included to deal with functions with nonzero mean.

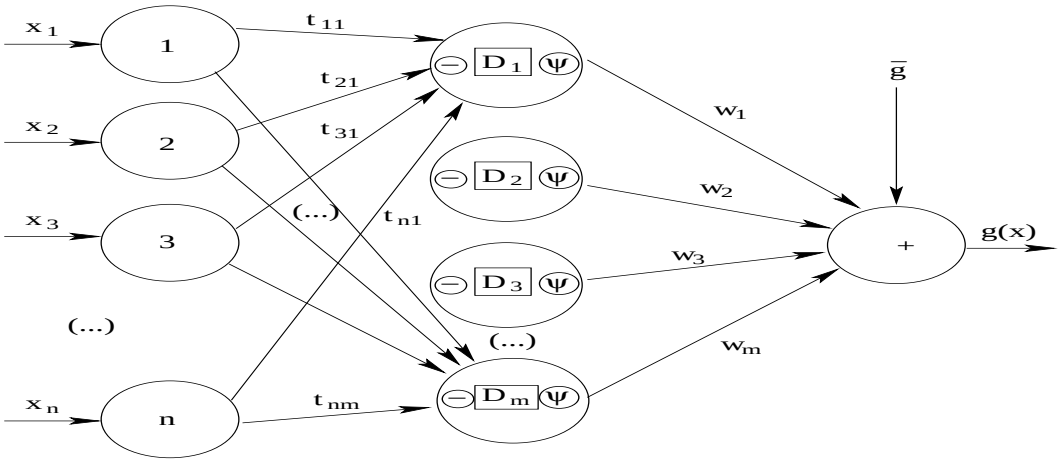


Figure 6.1: Wavelet network architecture

Comparable to radial-basis function networks (where weights and kernel positions and spreads can be adapted with a supervised learning procedure [Hay94]) the free parameters θ of the wavelet network, i.e. output bias, weights, and wavelet positions and supports, can be tuned to the desired function using a least-mean-squares algorithm. No error backpropagation is necessary, since the solution is linear in the basis function expansion. The gradients with

respect to all parameters in the network can be computed directly from the output error. For parameter updating, the following (gradient) method can be used. Consider a set of learn samples $\{\mathbf{x}_k, y_k = f(\mathbf{x}_k) + v_k\}$, where $\{v_k\}$ is observation noise. Define by $g_\theta(\mathbf{x})$ the output of the network parameterized by θ. Then the parameters have to be selected such that the mean-squared error $C(\theta) = \sum(g_\theta(\mathbf{x}_k) - y_k)^2$ between network predictions and targets over a batch of training samples is minimized. This can be done iteratively using gradient descent, i.e. computing the gradient $\nabla\theta$ of the function to be minimized with respect to all the parameters of the model, and updating by [Hay94]

$$\theta(n+1) = \theta(n) - \gamma\nabla\theta \tag{6.9}$$

where γ is the learning rate. When updating is done after presentation of each single training sample $(\mathbf{x}_k, y_k)$, a *stochastic* gradient descent procedure is obtained, which can be helpful to avoid local minima in the error surface [Hay94].

Example 6.1: function approximation with wavelet networks

A one-dimensional test function with a pronounced discontinuity [ZB92] can be approximated with a wavelet network, see figure 6.2 and [YD97b]. Training the network for several epochs results in gradually better approximation performance. The number of epochs is increased when going from figure 6.2(a) to 6.2(b).

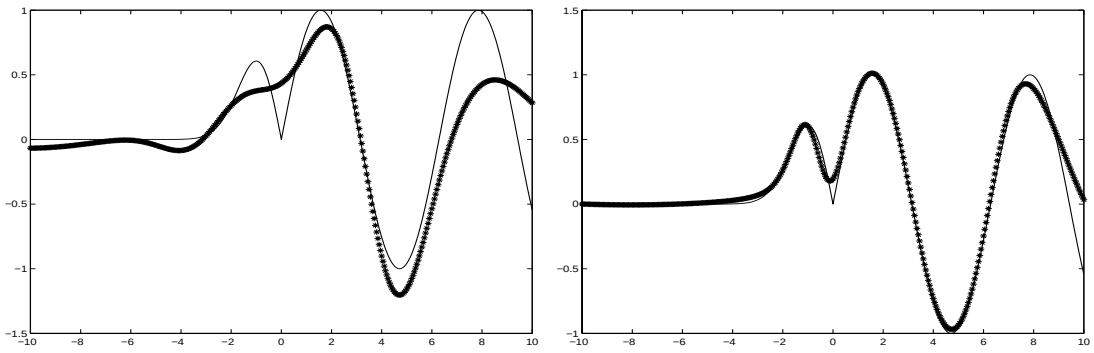


Figure 6.2: Approximation of 1-D test signal with wavelet network

However, in experiments with a two-dimensional curved surface, approximation proved to be much harder. We think that this can be ascribed to the increased dimensionality and the (heuristic) initialization procedure that we used in this experiment (analogous to [ZB92]). In [DJB96] it was noted that training a wavelet network may be notoriously slow, since the approximation class is nonlinear in the adjustable parameters and the gradient descent optimization criteria are highly nonconvex functions that exhibit many local minima. Moreover, it is known [Bis95] that constructive learning methods suffer from the "curse of dimensionality": increasing the input dimension would require a vast increase in the number of training patterns to cover the input space. The rate of convergence decreases with higher dimensional inputs [JHB+95]. In later work by Zhang [Zha93], a different initialization procedure (backward elimination selection of candidate wavelets) was proposed as a good alternative to the

heuristic initialization procedure employed in [ZB92]. From the literature on this subject and our experiences with function learning, a proper initialization is expected to be highly important for adequate network training. □

6.3 Feature-based novelty detection

In general there are no measurements of system *inputs* available, only system *responses* are measured. When these measurements are used for fault detection, a representation with high-dimensional feature vectors may be necessary. We mentioned in the previous section and in chapter 2 that many learning methods suffer from a curse-of-dimensionality. In the sequel we describe three methods for *domain approximation* in datasets with small sample sizes.

6.3.1 Self-Organizing Maps

The Self-Organizing Map (SOM) was proposed by Kohonen in the early Eighties in an attempt to mimic the topographic maps that are present in the human cortex. In the brain, certain sensory inputs (auditory, motor, somatosensory, etc.) are mapped onto certain (corresponding) brain areas in an orderly manner [Hay94]. The SOM can be considered an elastic net, that unfolds itself in the input space according to the distribution of the learning samples. It is *sensitive* to the input distribution (more samples in a sub-region will attract more nodes), but does not perform a *density estimation*. We say that the SOM performs a *domain approximation*, where the result is a lower-dimensional and quantised representation of the input data. Because of the fixed interconnections of the nodes (and the SOM learning method), the SOM also preserves the topology of the learning set to a certain extent. More specifically, nodes that are close on the map will represent learning samples that are also close in the input space.

The SOM algorithm

In SOM training, the map node m_c that is closest to an input sample x is determined (*winning node*): $c = \text{argmin}_i \|x - m_i\|$, and this node along with its neighbouring map nodes are updated towards this sample (see figure 6.3(a)):

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \tag{6.10}$$

where $h_{ci}(t)$ is the neighbourhood kernel and $t = 0, 1, 2, \ldots$ denotes discrete time. An example of the neighbourhood kernel is the rectangular kernel,

$$h_{ci}(t) = \begin{cases} \alpha(t), & d_{\text{map}}(i,c) <= N_r \\ 0, & d_{\text{map}}(i,c) > N_r \end{cases} \tag{6.11}$$

The interpretation of this kernel is that neighbouring neurons (i.e. having map distance $d_{\text{map}}(i,c)$ to winning neuron c up to N_r) are updated with a learning rate of $\alpha(t)$; neurons outside this neighbourhood set are not updated (see figure 6.3(a)). The map distance expresses

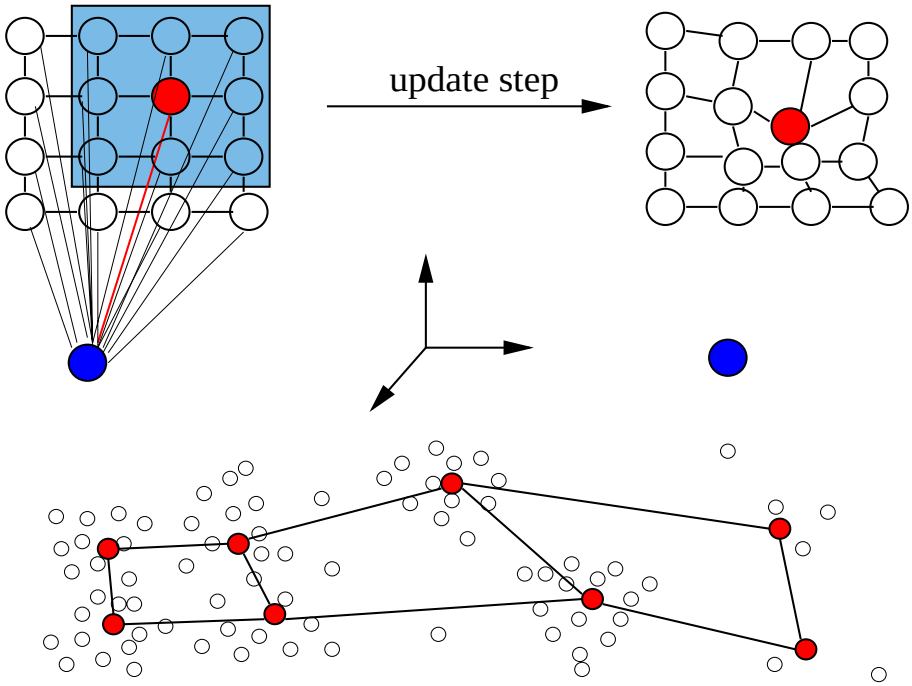


Figure 6.3: Self-Organizing Map training. The map unit that matches an input sample best is pulled towards the sample, along with the units in its neighbourhood set (top figure, a). After training, the SOM approximates the domain of the dataset (bottom figure, b)

the closeness of two nodes on the map grid. A common choice for the *neighbourhood set* is the set of nodes that are connected to the winning node via maximally N_r links. A different example of a kernel function is the (smoother) Gaussian kernel, which shows a more gradual decrease of the learning rate applied to neighbouring neurons.

The SOM training procedure is split into two phases [Koh95]. In the *unfolding phase*, the neighbourhood and learning rate are chosen large for the first, say, 300 cycles. In the next 2500 cycles, the learning rate is still large, but the neighbourhood decreases (e.g. linearly) to one. In the *fine-tuning phase*, the neighbourhood is fixed at a small value, and the learning rate decreases (e.g. linearly) to a small value. This phase may take some 10000 to 40000 cycles, depending on map size and problem complexity. After proper unfolding and finetuning, a Self-Organizing Map tends to approximate the input domain (figure 6.3(b)). It has been observed [Koh95] that random initialization of the network weights in conjunction with the two-stage learning procedure just mentioned usually leads to proper unfolding and data quantization. However, when the SOM is initialized with samples from the learning set or when the SOM grid is initially positioned along the principal components (directions) in the learning set, the unfolding stage is less important for successful SOM training.

Self-Organizing Maps for novelty detection

The Self-Organizing Map involves both a vector quantisation and a nonlinear projection of the input space to some (usually lower-dimensional) output space. Note that the effective dimensionality of the representation is equal to the dimensionality of the SOM. In many cases, a two-dimensional SOM is used for reasons of easy visualization, etc. However, this may give rise to mapping errors in the case of datasets with larger intrinsic dimensionality than the SOM dimensionality. For determining the accuracy of the SOM, often only the first task (vector quantization) is taken into account. The goodness of a map is then given by the *average quantisation error* (AQE, or mean-squared error MSE) between samples in a dataset and their best-matching units.

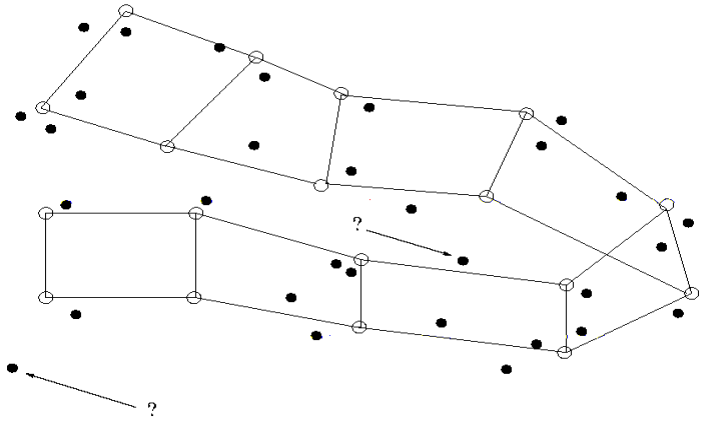


Figure 6.4: Novelty detection with Self-Organizing Maps

Since the SOM will approximate the domain of the dataset by minimizing the average quantization error between data samples and prototypes (under the topology preservation constraint) the distance between a dataset and a map may be used as a criterion for novelty detection [YD97a, MM98]. Data samples that resemble samples in the training set (the set on which the SOM is trained) will have a small distance to the map, whereas deviant samples will be more remote. This is illustrated in the figure 6.4. The compatibility between the unknown samples (denoted with the $'?'$ label in the figure) and the map can be judged by comparing the Euclidean distance of the sample to the averaged quantization error of a test set (not used in training the map) with respect to the SOM. Additionally, the topology of the SOM may be used in this task by taking the *goodness-of-fit* (GOF, described in appendix F) as distance measure: a dataset is now considered as *similar* to another dataset, if the SOM that represents the trainset captures both domain and topology of the test set adequately. We illustrate the goodness-of-fit measure in the following example.

Example 6.2: topology preservation

We generated a dataset according to a two-dimensional parabolically shaped distribution. A 1-dimensional SOM of 100 units was trained for 10000 cycles and the final neighbourhood

(in the convergence phase of the training procedure) was varied from 1.0 to 5.0 and 10.0 (figure 6.5). Increasing the final neighbourhood in adaptation of a SOM amounts to increasing its

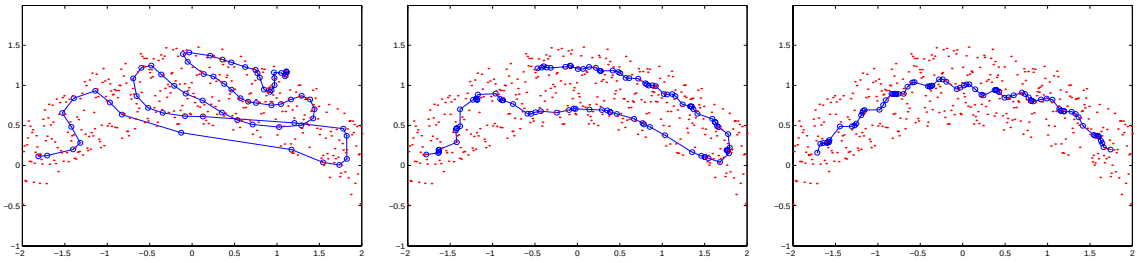


Figure 6.5: Map goodness in 1-D SOM with varying stiffness: triples (final neighb. width, AQE, GOF) have values a. (1, 0.02, 0.673); b. (5, 0.032, 0.164); c. (10, 0.062, 0.133)

stiffness, since its becomes less sensitive to local characteristics. A broader neighbourhood kernel causes less local distances to contribute to the error function. Intuitively, the SOM in figure 6.5(c) gives the best representation of the input space: it does not fold itself unnecessarily into the input space, whereas the general form of the distribution is still being tracked. The SOM in figure 6.5(a), however, is much less stiff and exhibits a kind of "overfitting": every local characteristic is being tracked, enabling excessive tuning to local disturbances and folding. Indeed, it can be observed that going from less to more stiffness causes a slight increase in quantization error, from 0.020 in the least stiff case, to 0.032 in the intermediate case and 0.062 in the most stiff case (which is still reasonably low), while the goodness (F.1) of the map with respect to the input distribution, denoting the regularity of the map, decreases with stiffness from 0.673, to 0.164 and finally 0.133. □

We continue with a different method for domain approximation, the $k-centers$ algorithm [YD98], which is inspired by the minimax-approach to learning (chapter 2) and by the relational approach to pattern recognition [Dui98].

6.3.2 Domain approximation with the k-centers algorithm

We start by noting that distances within a dataset can be looked upon as *features* of the dataset. More specifically, an $m \times m$ distance matrix $D = d(\mathbf{x}_i, \mathbf{x}_j), i, j = 1, \ldots, m$ corresponding to a dataset of size m can be treated as a set of m samples in an m-dimensional space [Dui98]. This may allow for pattern recognition in situations where it is difficult to define features for the objects to be classified, but distances (or similarities) between objects can be derived intuitively. This approach suffers from the problem that we end up in a situation where we have as many samples as we have features, where the expected generalization error shows a peak [Dui98]. As a remedy, both the number of features or the number of samples can be reduced. When the rows in the distance matrix are considered as samples, and the columns as features, reducing the number of samples (data editing) results in a matrix with for each remaining sample (the *representation objects*) the distances to each of the original objects (e.g. see table 6.1). The rows in the edited distance matrix can be interpreted as a

Table 6.1: Edited distance matrix: dataset $\mathbf{x}_i$ with four samples is represented by three entries in the distance matrix. This results in three samples in a four-dimensional space. Note that no features are used to represent a data object

	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$
$\mathbf{x}_1$	0	1	5	8
$\mathbf{x}_2$	1	0	2	5
$\mathbf{x}_3$	5	2	0	1

subset of data samples that captures information about the distances in the dataset. This can be used for approximation of the domain of a dataset.

Domain approximation: *k-centers* **algorithm**

Consider a dataset $X = \{\mathbf{x}_i\}, i = 1,\ldots,m$. For *domain approximation*, every object in the representation set $J = \{\mathbf{y}_j \in X\}, j = 1,\ldots,k \leq m$ is now given a *receptive field* in $\mathbb{R}^m$ of radius r. A sample $\mathbf{x}_i$ in X is *assigned* to the receptive field R_p of a representation object $\mathbf{y}_p$ when $p = \text{argmin}_j \; d(\mathbf{x}_i, \mathbf{y}_j)$ and $d(\mathbf{x}_i, \mathbf{y}_j) \leq r$. Here, we take Euclidean distance for the distance measure $d(\mathbf{x}, \mathbf{y})$. The set of k representation objects J is found such that the corresponding ra-

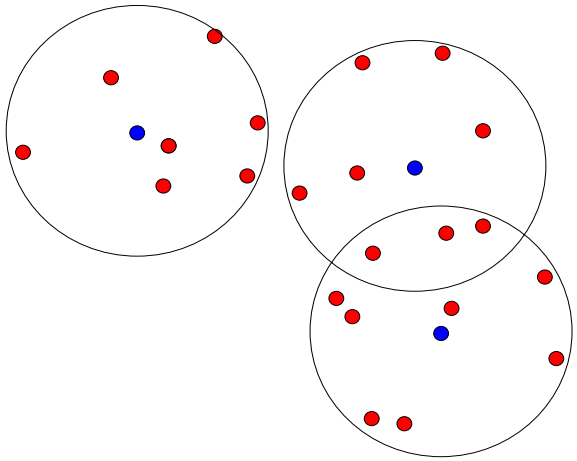


Figure 6.6: Domain approximation with *k-centers* algorithm; each sample lies in the receptive field of at least one representation object and is *assigned* to the closest center

dius $r(J)$ is minimized, while all original objects are assigned to some representation object's receptive field (figure 6.6, dark objects in sphere center),

$$J = \underset{S_k}{\text{argmin}}\, r(S_k) \tag{6.12}$$

where the cardinality $|S_k|$ equals k. Furthermore, S_k is a subset of X of length k (i.e. $S_k \in 2^X$)

and the corresponding receptive field radius $r(S_k)$ is given by

$$r(S_k) = \max_i d(\mathbf{x}_i, \mathbf{y}_j), \qquad \mathbf{x}_i \in R_j, \mathbf{y}_j \in S_k \tag{6.13}$$

We use a variant of the *k-means clustering* algorithm (see [Hay94]) that is adapted for the domain approximation problem (referred to as the *k-centers* algorithm) to select the subset of data samples that minimizes expression (6.12). During the algorithm, each representation object represents the center of the samples in its receptive field. If a better center can be found within its receptive field (i.e. such that the radius can be decreased), swap the former and the latter objects, until the subset can not be improved any more. Ultimately, the best subset over several trials is retained. For choosing the initial subset, one can repeat different trials with different random choices and keep the best subset over several repetitions. The effect of different random initializations is studied later in this section. Note that in this method we assume that there are no serious outliers present in the data. Moreover, proper choice of k is important for the quality of the description.

Determining the number of centers k

There is a trade-off between complexity of the solution (representation set size) and generalization capability (figure 6.8). Estimation of an appropriate representation set size can be aided by the data *clustering structure*. Alternatively, a linear search over the number of centers can be done. In the *first approach* we use a *successive approximation* scheme: the representation set size is increased from 1 to the number of samples, and the optimal subset at size k is used as initial subset for size $k+1$. The sample most remote to its receptive field center is initially taken as the additional representation object. As convergence criterion we use the relative improvement in radius when increasing the representation set from size k to $k+1$. We investigate this approach in experiment 6.3.3. A *second approach* to estimating k is suitable in cases with separable clusters. We use a *minimal spanning tree* (MST) [CLR90] for describing the interpoint distances in the dataset. An MST can be interpreted as a hierarchical clustering method [BD89]: if we apply it to the interpoint distances in a dataset, the distances at the *medial axis* of the tree (the longest path along the tree, i.e. the path with highest cost) will represent jumps from one cluster to another cluster. Counting the number of significant peaks on the tree axis can be used to get a prior estimate of the appropriate number of centers (see the example below). As a *third* approach, we mention the possibility to determine the number of centers using cross-validation. This requires an *a priori* specification of a fraction $0 \leq \alpha \leq 1$ of the 'normal set' that is to be accepted. We split the normal set into a 'normal training set' and a 'normal test set'. The training set is described with k centers and the error (percentage of a dataset that is not accepted by the description) on the test set is determined. This procedure is repeated for several splits, leading to an average test error for a certain k. Repeating the procedure for several k will then lead to an optimal k that corresponds to the a priori defined acception rate. The final description can be made using the samples that were never rejected in the test-set evaluation. This may also alleviate the problem that the presence of outliers in the normal set can seriously degrade the quality of the description.

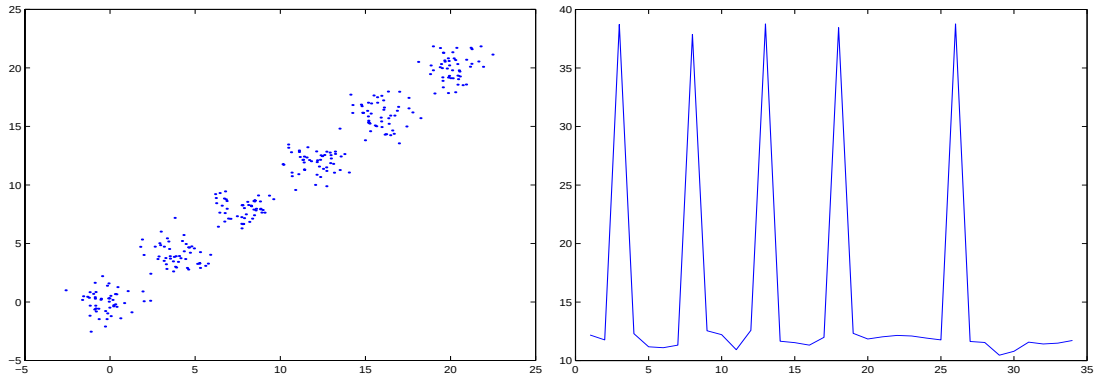


Figure 6.7: Six Gaussian clusters (a); distances on medial axis of the corresponding MST (b)

Example 6.3: description of Gaussian clusters using an MST

We generated six Gaussian clusters of two-dimensional data, where the mean of the Gaussians was markedly different for each cluster, figure 6.7(a). A minimal spanning tree was fitted to the dataset, by using the distances from a sample to all other samples as the weights on the fully connected graph (which then represents the dataset). The resulting MST was analyzed by extracting the medial axis of the MST. We expect that the inter-cluster distances are included on the medial axis. From figure 6.7(b) it is clear that the inter-cluster distances correspond to the peaks, so that 5+1 = 6 clusters can be inferred for this dataset. □

6.3.3 Experiment: evaluation of k-centers algorithm

Determining the number of centers

The *k-centers* method was applied to a set of measurements from the submersible pump test rig of chapter 1. In the dataset three classes were present (normal behaviour, imbalance and bearing failure). The dataset was obtained by representing segments of machine vibration as a 256-bins power spectrum, which was normalized with respect to mean and variance. Some 40 % of the variance in the dataset is retained in the first two principal components. The effect of domain approximation of this dataset with a varying number of centers is shown in figure 6.8. We compared the successive approximation algorithm to the *k-centers* algorithm with random initialization (1 and 5 trials), where we varied the representation set size. We monitored the value of the final radius in the description as a function of the number of spheres. In figure 6.9(a) the successive approximation algorithm (the bottom solid line in the graph) can be seen to yield smaller final radius (vertical axis), while the radius is monotonically decreasing with representation set size (horizontal axis). The radius obtained with 1 trial random initialization fluctuates randomly around the 5 trial random initialization results. It is clear that the stabilizing effect of more trials is at the expense of increased computing time, whereas successive approximation needs only one run of the algorithm for each representation set size. Moreover, using more spheres leads to significant improvement with successive approximation, whereas the random initialization variants reach a plateau. This

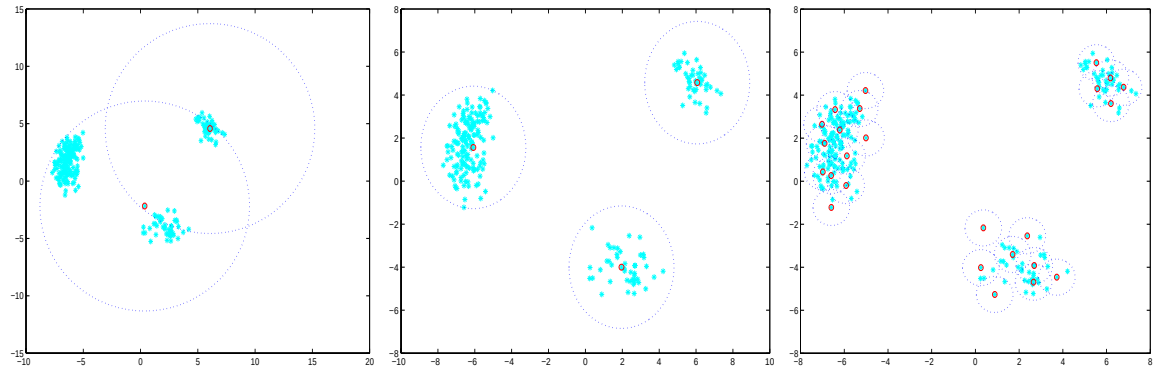


Figure 6.8: Trade-off in k-centers domain approximation: large tolerance (a), intermediate tolerance and complexity (b) and large complexity (c)

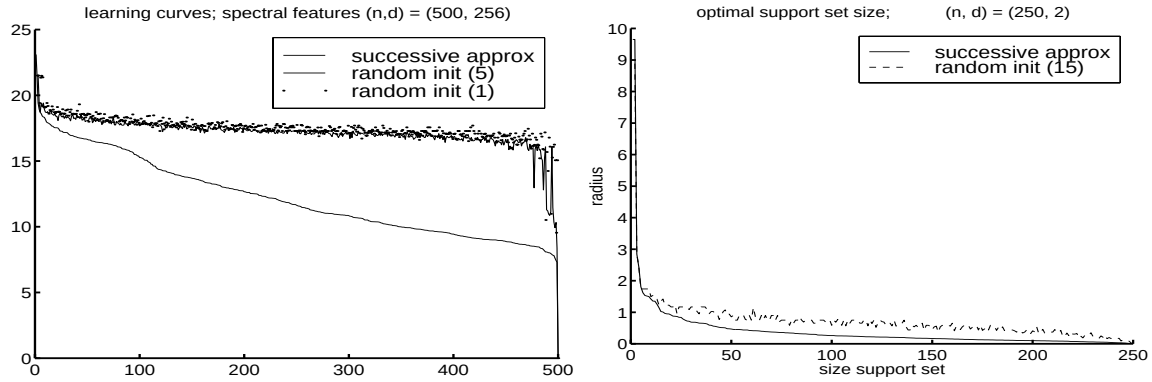


Figure 6.9: Successive approximation vs. random initialization: a. using a 256-D dataset; b. using a 2-D dataset

indicates that there is a relatively homogeneous distribution of distances in high-dimensional spaces, since the sphere radius can be constantly decreased. With increasing representation set size, the chance that a random guess for the initial subset is adequate decreases, hence the radius will be less improved using random initialization. When using data consisting of the first two principal components of the previous dataset, the successive approximation algorithm proved again superior, figure 6.9(b), but the difference is much smaller. Due to the three clusters in the data, combined with low dimensionality of the space, a sudden decrease in the radius occurs with three spheres (to a very small value). Moreover, the clusters are already represented adequately by three spheres, resulting in only marginal improvement in radius using more spheres.

Novelty detection with k-centers algorithm

We analyzed whether k-centers domain approximation can be used for the rejection of gradually more dissimilar data. New data samples were generated around existing data samples with *k*-nearest neighbour data enrichment [Sku01, SYD00], see also figure 6.10(a). The off-

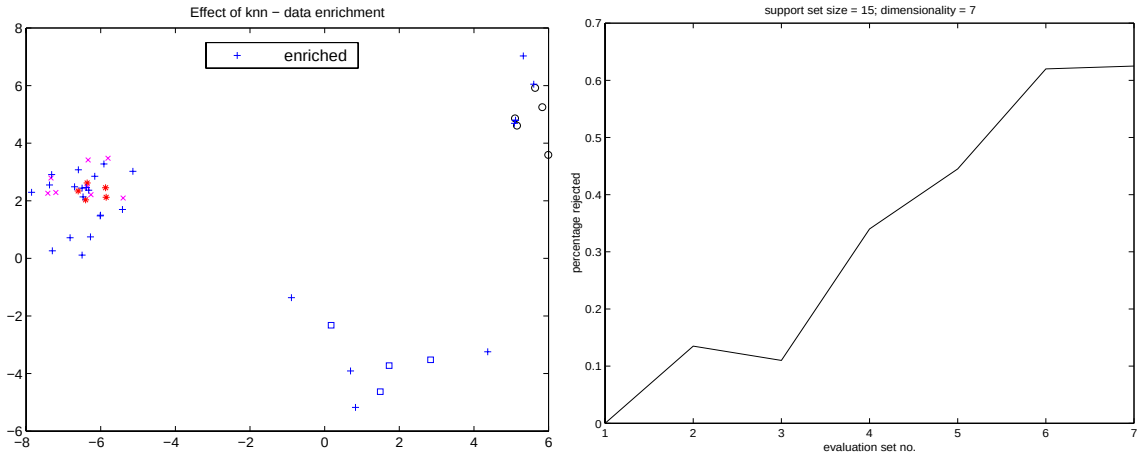


Figure 6.10: a. Data enrichment using k-nearest neighbours; '+' signs denote new samples generated around existing samples ('x', '*', etc. symbols); b. rejection of (enriched) novelty data; on the horizontal axis the id. of the dataset is plotted: 1 = train, 2 = test, 3 - 7 = evaluation data

set of a new sample from its corresponding original sample is Gaussian distributed with zero mean and standard deviation s times the mean signed difference between the point under consideration and its nearest neighbours in the original set. For s running from 0 (original set) to 5, different validation sets were contructed. To track generalization, an independent test set of the same origin as the original set was used. Note that the use of multidimensional spheres can give rise to excessive tolerance in singular directions when the data lies in some (possibly nonlinear) subspace, hence we used the principal components of the spectral features dataset in this experiment. It was observed that increasing feature size and representation set size decreased generalization drastically (i.e. test samples and validation data for moderate values of s were frequently rejected). Results for reasonable values for dimensionality (7) and number of spheres (15) are shown in figure 6.10. The first set is the training set, the second set the test set, and sets 3 to 7 are validation sets with $s = 1, \ldots, 5$. Since the domain is fitted as tight as possible, a test set already shows some rejections. Then the rejection rate increases with the amount of novelty, up to a point where always a certain fraction of the new samples lies somewhere on the original domain (the distribution of the offset has zero mean).

Another method for description of the domain of a dataset using a subset of the data was proposed in [TD99], the Support Vector Data Description (SVDD) method. Here, the domain of the dataset is captured with a single sphere with minimum volume. Analogous to the *support vector method* by Vapnik [Vap95], one can extend this idea to describe an arbitrarily shaped region in the original feature space.

6.3.4 Support vector data description

Assume a dataset contains N data objects, $\{x_i, i = 1, .., N\}$, and a sphere that includes the data is represented by a center a and a radius R. Tax [Tax01] minimizes an error function that represents the *volume* of the sphere. The constraint that objects should reside *within* the

sphere is imposed by applying Lagrange multipliers:

$$L(R,a,a_i) = R^2 - \sum_i a_i\{R^2 - (x_i^2 - 2ax_i + a^2)\} \tag{6.14}$$

with Lagrange multipliers $a_i \geq 0$. This function has to be minimized with respect to R and a and maximized with respect to a_i. Setting the partial derivatives of L to R and a to zero, gives:

$$\begin{aligned} \sum_i a_i &= 1 \\ a &= \frac{\sum_i a_i x_i}{\sum_i a_i} = \sum_i a_i x_i \end{aligned} \tag{6.15}$$

The center of the sphere a is expressed in terms of a linear combination of data samples x_i. Resubstituting these values in the Lagrangian results in a function that should be maximized with respect to a_i:

$$L = \sum_i a_i (x_i \cdot x_i) - \sum_{i,j} a_i a_j (x_i \cdot x_j) \tag{6.16}$$

with $a_i \geq 0$, $\sum_i a_i = 1$. In practice this means that a large fraction of the a_i become zero. The vectors that have values of $a_i > 0$ are the support vectors, i.e. the vectors that span the description. A sample z is *accepted* (i.e. considered as a part of the domain of the dataset) when

$$(z-a)(z-a)^T = (z - \sum_i a_i x_i)(z - \sum_i a_i x_i) \leq R^2 \tag{6.17}$$

Description with kernels

In general this does not give a very tight description. Analogous to the support vector method, the inner product $(x_i \cdot x_j)$ in equation (6.16) can be replaced by a kernel function $K(x_i, x_j)$. For example, when the inner products are replaced by Gaussian kernels one uses the substitution

$$(x_i \cdot x_j) \rightarrow K(x_i, x_j) = \exp(-||x_i - x_j||^2 / s^2) \tag{6.18}$$

Equation (6.16) now changes into:

$$L = 1 - \sum_i {a_i}^2 - \sum_{i \neq j} a_i a_j K(x_i, x_j) \tag{6.19}$$

and the formula to check if a new object z is accepted, equation (6.17), becomes:

$$1 - 2\sum_i a_i K(z, x_i) + \sum_{i,j} a_i a_j K(x_i, x_j) \leq R^2 \tag{6.20}$$

Controlling the error rate

The Gaussian kernel contains one extra free parameter: the width parameter s in the kernel (equation (6.18)). As shown in [TD99] this parameter can be set by fixing the maximal allowed rejection rate of the target set (i.e. the error on the target set) a priori. Analogously to the support vector classifier method, this error can be estimated by the number of support vectors:

$$E[P(\text{error})] = \frac{\#SV}{N} \tag{6.21}$$

where $\#SV$ is the number of support vectors. This number of support vectors can be controlled by the width parameter s and therefore the error on the target set can be put at a prespecified value. Note that no prior restrictions on the error on the outlier class can be made. One only assumes that a good representation of the target class is available; the outlier class is then considered to be "everything else".

6.4 Experiments: detection of machine faults and gas leaks

6.4.1 Choosing features and channels for gearbox monitoring

The purpose of the following experiment was to assess the appropriateness of several feature extraction methods for description of machine health. This was assessed by determining the amount of overlap that exists between normal and abnormal data patterns, while approximating the domain of the normal patterns only [TYD99a]. The normal behaviour of a healthy machine (machine no. 3) was described using the SVDD method and several features were used to represent the machine state. Then, measurements from damaged machine no. 2 were transformed into feature vectors and tested for being on the normal machine domain. The machines under investigation were two different pumps in a pumping station (section 1.5.1), one with a defect gearbox and one with a healthy gearbox.

Rationale

Three different feature sets were constructed by joining measurements from different sensors into a set:

configuration C1 one radial channel near the place of heavy pitting,

configuration C2 two radial channels near both heavy and moderate pitting along with an (imbalance sensitive) axial channel, and

configuration C3 inclusion of all channels except for the sensor near the outgoing shaft (which might be too sensitive to non-fault related vibration)

By putting the measurements of different sensors into one dataset, the dataset increases in size, but information on the exact position of an individual measurement is lost. As reference datasets, we constructed (first) a high-resolution logarithmic power spectrum estimation (512

bins), normalized with respect to mean and standard deviation and (second) its linear projection using Principal Component Analysis on a 10-dimensional subspace. Three channels were included, which leads to a set that is roughly comparable to the second configuration C2 previously described. In all datasets we included measurements at various machine loads, e.g. samples corresponding to measurements from the healthy machine operating at maximum load were added to samples corresponding to less heavy loads to form the total normal (target) dataset. The same holds for data from the worn machine.

We compared several methods for feature extraction from vibration data, by comparing the amount of overlap that the corresponding datasets for several feature choices exhibited. We used a feature dimensionality of 64 and compared the features: power spectrum, so-called *classical* features (kurtosis, crest factor and RMS-value of spectrum), autoregressive modelling and MUSIC spectrum estimation (section 3.2.1). To compare the different feature sets, the SVDD is applied to all target datasets. Because also test objects from the outlier class are available (i.e. the fault class defined by the pump exhibiting pitting, see section 1.5.1), the rejection performance on the outlier set can also be measured. In all experiments we have used the SVDD with a Gaussian kernel. For each of the feature sets we have optimized the width parameter s in the SVDD such that $1\%, 5\%, 10\%, 25\%$ and 50% of the target objects will be rejected, so for each dataset and each target error another width parameter s is obtained. For each feature set this gives a acceptance-rejection curve (or ROC-curve) for the target and the outlier class.

Novelty detection results

We first consider sensor combination C3, which contains all sensor measurements. In this case we do not use prior knowledge about where the sensors are placed and which sensor might contain most useful information. In figure 6.11(a) the ROC-characteristic is shown for

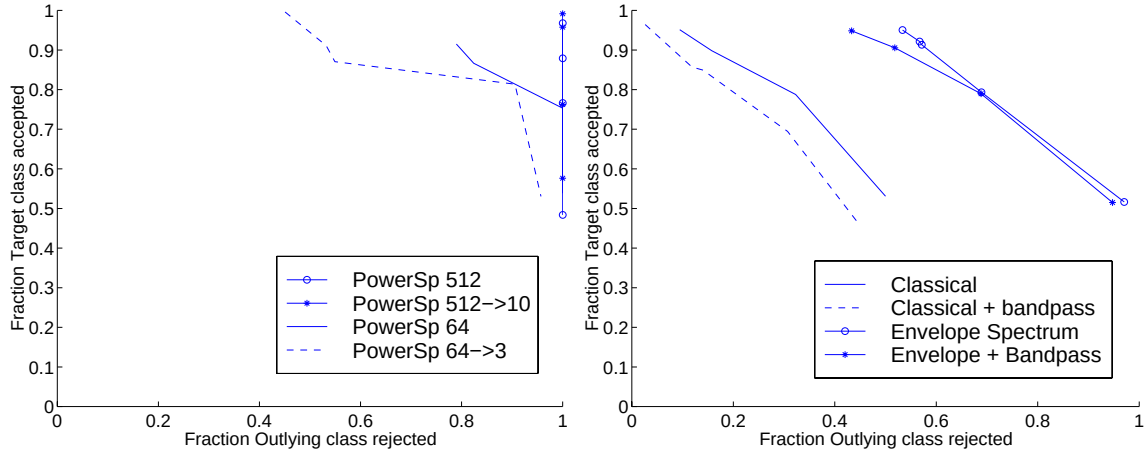


Figure 6.11: Acceptance/rejection performance of SVDD: a. using 512-bin and 64-bin power spectrum features on sensor combination C3; b. using classical features and the envelope spectrum on sensor combination C3

the power spectrum dataset. If we look at the results for the power spectrum with 512 bins we see that for all target acception levels we can always reject 100% of the outlier class. This is the ideal behavior we are looking for in a data description method and it shows that in principle the target class can be distinguished from the outlier class very well. Reducing this 512 bin spectrum to just 10 features by applying a Principal Component Analysis (PCA) and retaining the ten directions with largest variation, we see that we can still perfectly reject the outlier class. Using a less well sampled power spectrum of just 64 bins results in a decrease of performance. Only when 50% of the target class is rejected, more than 95% of the outlier class is rejected. Finally, when using just the three largest principal components, SVDD performance worsens almost everywhere.

In figure 6.11(b) the envelope spectrum feature set is compared with the classical features. Both the envelope spectrum and the bandpass filtered envelope spectrum features outperform the description based on classical features. The differences between the bandpass filtered and the original envelope spectrum features are small. Looking at the results of the classical method and the classical method using bandpass filtering, we see that the target class and the outlier class overlap significantly. When we try to accept 95% of the target class only 10% or less is rejected by the SVDD. Also considerable overlap between the target and the outlier class is present when envelope spectra are used. When 5-10% of the target class is rejected, still about 50% of the outlier class is accepted.

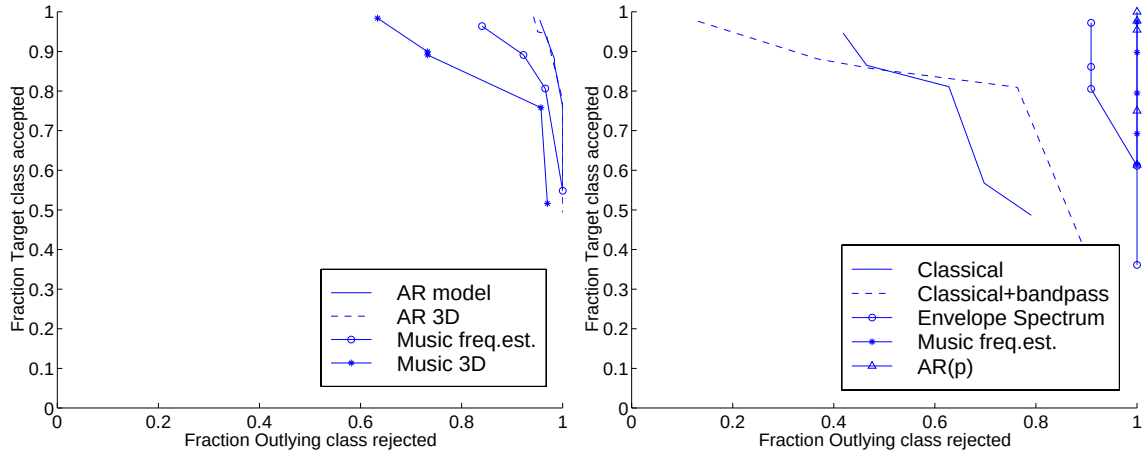


Figure 6.12: Acceptance/rejection performance of SVDD: a. using AR-model and MUSIC features on sensor combination C3; b. using various features for sensor combination C1

Finally, in figure 6.12(a) the results on the AR-model feature set and the MUSIC feature set are shown. The MUSIC estimator performs better than the already shown classical features, the envelope spectra with and without bandpass filtering. Taking the 3-D PCA severely degrades the performance, especially for smaller target rejection rates. The AR model outperforms all other methods, except for the 512 bin power spectrum, cf. figure 6.11(a). Even taking the 3-D PCA does not deteriorate the performance. Only for very small rejection rates of the target class, we see some patterns from the outlier class being accepted. From these

experiments we conclude that measurements from healthy and worn machines 3 and 2, respectively, can be separated very well using a proper domain approximation method (SVDD) and a proper feature extraction method (AR or power spectrum modelling). However, this can not be interpreted as a detection of gearbox wear, since the separability of the classes may also be explained by the different vibration characteristics of each machine (regardless the condition). Significant inter-machine variability in vibration behaviour is expected to be present in this setup. Repetition of the above experiments with data from machine no. 1 (that exhibited intermediate gearbox wear) revealed that patterns from this machine could also be separated perfectly from the other two machines.

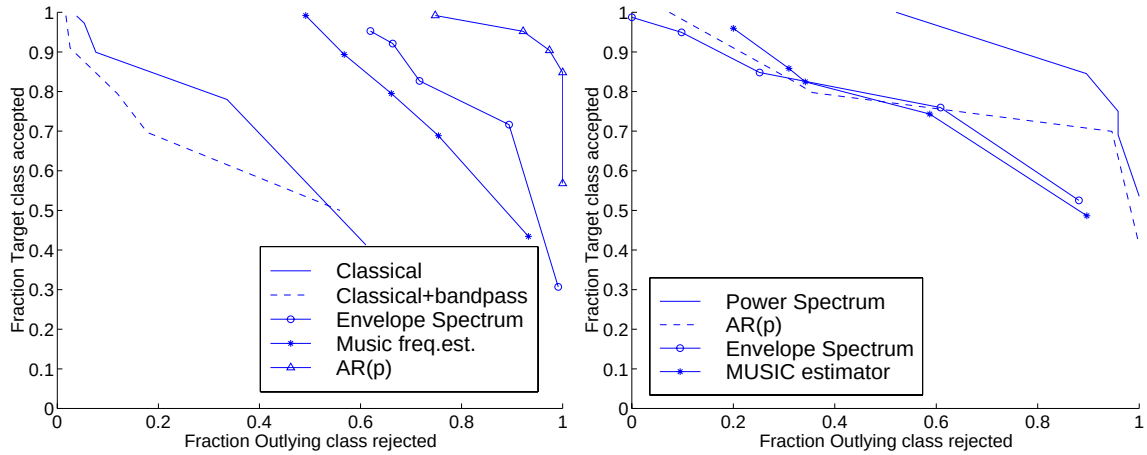


Figure 6.13: Acceptance/rejection performance of SVDD: a. using different features for sensor combination C2 in gearbox monitoring setup; b. using various features with all sensor measurements collected, in submersible pump monitoring setup

We now look at the detection performance using the first and the second combination of sensors. In figure 6.12(b) the performance of the SVDD is shown on all feature sets applied on sensor combination C1. Here the classical features again perform poorly. The envelope spectrum works reasonably well, but both the MUSIC and the AR-model features perform perfectly. The data from sensor combination C1 appears to be clustered better than the data from sensor combination C3. We can observe the same trend in figure 6.13(a), where the performances are plotted for sensor combination C2. The MUSIC estimator and the AR model again outperform the other types of features, but here the error rate is higher: total performance is worse than that of sensor combination C1 and C3. We see that measurements from machines 2 and 3 can be separated by using a domain approximation approach, without the need to deliberately choose the measurement channels to be incorporated in the dataset.

6.4.2 Sensor fusion for submersible pump monitoring

We now investigate the effect of combining several sensors *prior* to feature extraction using spatiotemporal combination methods PCA and ICA, see chapter 4 and [YTD99, Ypm99a].

The objective of this experiment was to study different ways to incorporate multiple sensors in a dataset. The resulting dataset is then used for novelty detection. We postulate that combining the information of several sensors into a few "virtual sensors" allows for smaller data dimensionality and less dependence on the position of the ensemble of sensors, whereas the relevant diagnostic information may still be retained. In other words, distributing an ensemble of sensors over a machine casing would then allow for extraction of the diagnostic information of the relevant (fault) source, regardless the exact position of each sensor. Measurements were obtained from the submersible pump described in chapter 1 and feature vectors were determined from this set of measurements. We used features based on autoregressive modelling (order 64) and (MUSIC) spectrum estimation (64 spectral bins) for 'coding' the vibration signals. We compared this to feature vectors obtained with a 64-bins envelope spectrum, i.e. the spectrum of an envelope detected time signal. No bandpass filtering was employed here (the effect of bandpass filtering on the ROC-curve proved to be small in the previous section). As a reference, a high-resolution (512 spectral bins) normalized power spectrum was obtained from the measurements.

Rationale

From the first block (segment of consecutive samples for a certain channel) in a multichannel measurement at a certain health- and operating mode, the demixing vector (row in the demixing matrix) was determined using the *fast-ICA* algorithm [HO97]. We used the deflation approach with cubic nonlinearity. If a component could not be found (the algorithm did not converge in 100 cycles), the multichannel time series at hand was disregarded in feature extraction. This did not result in a significant decrease in the sample sizes, however. Inclusion of more independent components was performed by iterative deflation of several components. From each virtual sensor component, a feature vector was extracted and added to the dataset as a new sample. This means that the dimensionality of the dataset was equal to the dimensionality of the chosen feature vector. For all blocks following the first block, the same demixing coefficients were used (i.e. we assumed stationary mixing throughout a single measurement). For each new operating mode or health state, a new set of demixing coefficients was obtained. We compared the ICA combination approach with taking linear combinations based on variance maximization (Principal Component Analysis). Alternatively, we used the SVDD with datasets that were generated with feature extraction from measurement signals *without* applying ICA or PCA. Here, features from all measurement channels were included into one dataset. Each channel resulted in a new sample in the dataset. The procedure for sensor combination is illustrated in figure 6.14.

The SVDD is applied to several datasets, differing in sensor combination method and feature extraction method. Since test objects from the outlier class are available (i.e. the fault class comprising imbalance, loose foundation and bearing failure), the rejection performance on the outlier set can also be measured.

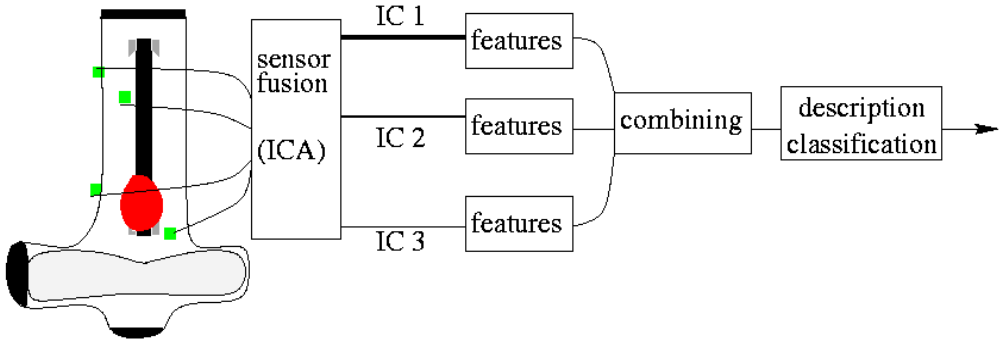


Figure 6.14: ICA-based sensor fusion and fault detection with submersible pump

Novelty detection results

In all experiments we have used the SVDD with a Gaussian kernel. For each of the feature sets we have optimized the width parameter s in the SVDD such that $1\%, 5\%, 10\%, 25\%$ and 50% of the target objects will be rejected, so for each dataset and each target error another width parameter s is obtained. For each feature set this gives an ROC-curve for the target and the outlier class.

In figure 6.13(b) the performance on MUSIC, AR and envelope features are compared to the 512 bins power spectrum, where easurements from all channels were collected into a dataset with no ICA applied. The high resolution power spectrum clearly outperforms all other methods: when all target class objects are accepted, over 50% of the outlier class is rejected; when about 10% of the target class is rejected, almost 80% of the outlier class can be rejected. This shows that a large fraction of the abnormal patterns can be distinguished from the normal class, but some overlap exists. Performance on the AR, envelope spectrum and MUSIC features are worse. Especially when large target acceptance rates are required, large fractions of the outlier class are accepted. Only the AR-model dataset approximates the performance of the 512 bins spectrum for smaller target acceptance rates.

In figure 6.15(a) the SVDD is applied to MUSIC features on which an ICA is performed. Results vary with the number of independent components that are included into the dataset. Performance is optimal when just one component is used, using more than one component results in (comparable) worsened performance for all situations. In figure 6.15(b) we observe the same trend for the envelope features. Again, the first independent component yields most information for fault detection, including multiple components gives significantly worse results. Comparing these results to the results obtained when collecting measurements from all channels in one dataset without 'fusion', figure 6.13(b), we see that ICA-fusion has the effect of tilting the ROC-curve such that lower false alarm rates can be obtained at 'reasonable' fault detection rates (i.e. in the order of 70 % correctly detected fault patterns).

In figure 6.16(a) the performance is shown for the AR-model features. Again the first component gives best performance, but for smaller target class acceptance rates, inclusion

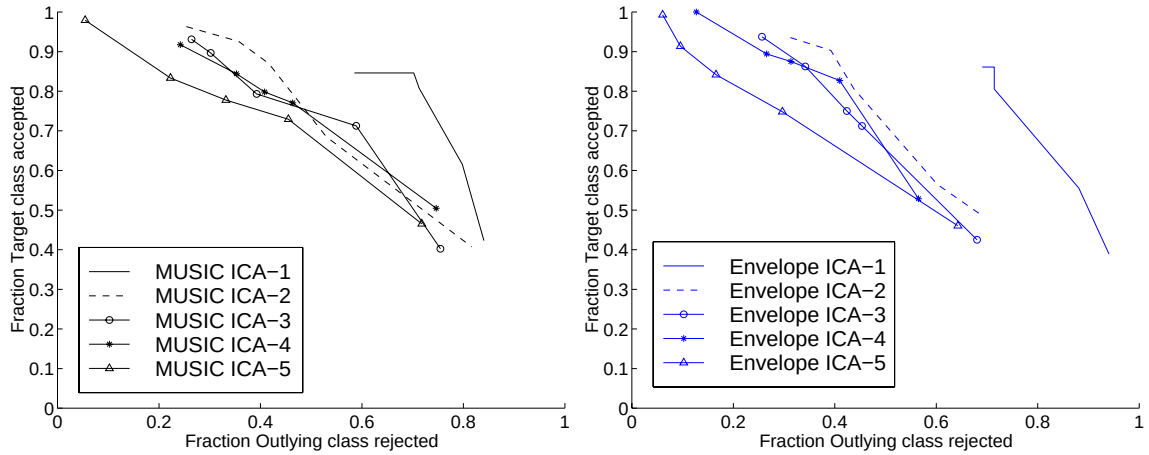


Figure 6.15: Acceptance/rejection performance with ICA-based sensor combination: a. using MUSIC features with various number of components; b. using envelope features with various number of components

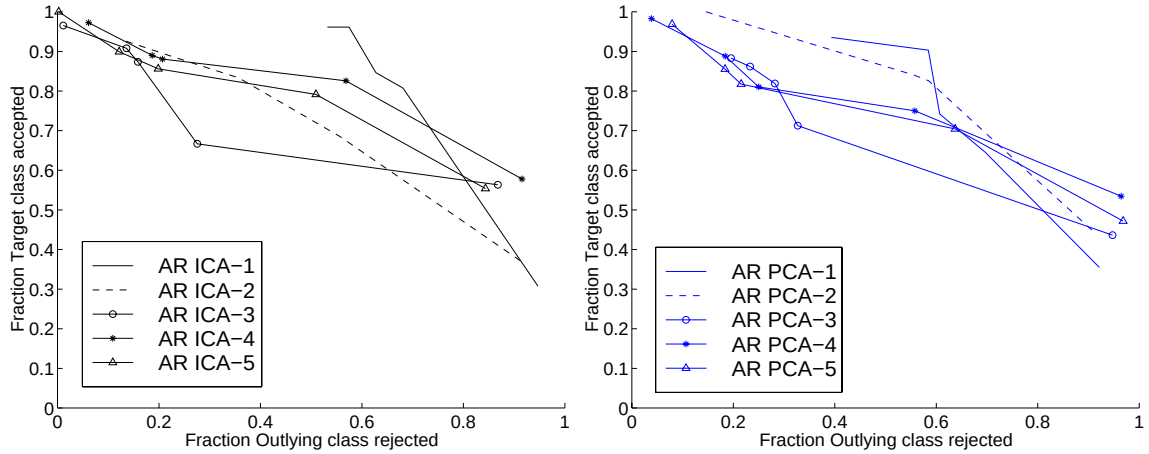


Figure 6.16: Acceptance/rejection performance on the envelope features with various number of components. a. Results of ICA-based combination; b. results of PCA-based combination

of multiple components is useful. In this case it is less clear how much useful information is retained when taking only the first component into account. Although in the non-ICA case the AR model features allow better to distinguish between target and outlier class (see figure 6.13) than envelope spectrum features, in the ICA case the AR features only allow for better performance when more than one component is taken into account, compare figures 6.15(b) and 6.16(a). This may indicate e.g. that a second independent component represents a different mode, that happens to distinghuish better between normal and fault condition when described with an AR model. Note that there is in general no ordering in a set of independent components, although in some ICA algorithms most nongaussian components are deflated consistently first. It is a priori not clear whether the most nongaussian component is also the

mode that allows for best diagnostics.

Finally AR-model features were computed on data with PCA preprocessing, figure 6.16(b). Taking just the first principal component is not enough, only for very large target acceptance rates it outperforms the sets that use more components. When larger fractions of the outlier class should be rejected, information from more components should be used. Note that PCA fusion into one component again leads to a tilt in the ROC-curve, cf. figure 6.13(b), that allows for a smaller false alarm rate at reasonable fault detection rates. With inclusion of the first three or four PCA/ICA fused components, performance is comparable to the all-sensors non-fusion case.

Analysis of ICA-based sensor fusion

We notice that the first ICA-component contains most information to distinguish between normal and abnormal behaviour. This can be understood from the fact that the number of underlying sources in the multichannel submersible pump measurements turned out to be one, i.e. the first eigenvalue of the covariance matrix of the 5-channel measurements (see chapter 5) was significantly larger than the other eigenvalues. We suggest that deflation of one independent component in a multichannel measurement from the submersible pump might have the effect that an omnipresent fault source (a set of fault-related harmonics) is being separated from (sensor position dependent) modal contributions to the spectrum. It is however unlikely that this will happen in our experiments, since we neither bandpass-filtered the data (frequency range was 0-20 kHz) nor used a convolutive demixing algorithm. A possibility is that an (instantaneous) ICA demixing will separate a low-frequency mode that exhibits almost perfect coherence among all sensors in the array because of reverberations (figure 1.3). Extraction of a mode that is enhanced because of broadband excitation due to a fault may give diagnostic information. However, the scaling indeterminacy that occurs in ICA should be resolved in this case, which requires prior knowledge about the underlying sources.

The results for 'fusion' into one PCA-component are slightly worse than for fusion into one ICA-component. However, no significant improvement from ICA-fusion for fault detection over PCA-fusion could be observed. This was partly due to the setup of our experiment: instantaneous ICA-demixing of our submersible pump measurements can at best separate the contribution due to a particular fault-enhanced machine mode. If no 'matching' with other deflated components is done, there is danger that different operating modes at the same health condition give rise to different modal components, deflated in different order. However, since in our experiments the results for deflation of a single component were consistently better than for inclusion of several components in a dataset, we suggest that similar first components were deflated over several operating modes. A possible explanation is that the signal subspace contained mainly one dominant component. Since the *fast-ICA* algorithm uses a prewhitening step, diagnostic information in the independent components deflated in the second and in later steps may be almost negligible. We conclude that feature extraction based on a high-resolution power spectrum allows for a satisfactory trade-off between target acceptance rate and fault detection rate in the submersible pump monitoring problem. How-

ever, using much lower-dimensional feature vectors will increase the hope that results will generalize unto new machine health measurements. The AR(64) model coefficients allowed for still acceptable results; PCA or ICA fusion into a single component may then be used to decrease the false alarm rate of the detection system to values below 5% at a fault detection rate of 60%. If the aim is to maximize the fault detection rate regardless the false alarm rate, inclusion of several channels into a dataset (whether with PCA, ICA or no fusion) is the best choice.

6.4.3 Leak detection with Self-Organizing Maps

Automatic detection of gas leaks in pipelines is of prime importance. Hydrophones can be used to monitor the ambient noise in the neighbourhood of a pipeline. In this case, a noise measurement signifies a normal situation whereas a leak measurement is an abnormal situation. However, measurements of a leak condition are expected to be much better reproducible than measurements of noise: all kinds of circumstances (weather conditions, ships passing by, etc.) may cause very deviating noise patterns to occur. Hence, instead of describing data representative of the 'normal behaviour', we aim at describing the 'reproducible behaviour' of the system. In other words, we describe a set of (simulated) gas leak patterns and distinghuish between leak and noise by judging the similarity of a new pattern to the leak description, see [YD97a]. More information on the background of this research project can be found in section 1.5.3.

Initial measurements at the 'IJ'

This experiment was performed with hydrophone measurement data from a series of leak simulations at the "IJ" harbour at Amsterdam. This simulation involved a blow-down procedure, similar to the procedure described in section 1.5.3. Here, a training set of 100 18-dimensional power spectrum-based feature vectors was formed out of a representative subset of the available data. The purpose of the experiment was the detection of leaks in pipelines. A detection system should be able to distinguish between background noise (water, wind, noise from ships passing by) and leak activity.

Hydrophone measurements were preprocessed by computing the logarithm of the power spectrum and normalizing with respect to mean and standard-deviation. Leaks of various remoteness were simulated by superimposing appropriately attenuated data (acquired near a leak) with background noise. A test set from different measurements with similar characteristics as the training set was used to track the SOM interpolation capabilities, a validation set with different characteristics was used to evaluate the extrapolation possibilities. To verify the SOM capability to distinguish between leak and noise, a representative equal sized set of background noise vectors was constructed, originating from (both nearby and remote) ship noise and normal background noise. Out of each set of leak vectors, two more sets were constructed by attenuation with 10 and 20 dB before superposition with a representative set of noise vectors.

Leak detection at varying distance

A Self-Organizing Map was trained on the training set for 10000 cycles. The MSE was determined for the leak training-, interpolation- and extrapolation sets, and for the set of noise vectors. Next, the goodness of the fit (GOF) between the trained map and the data set under investigation was determined for all sets. This process was repeated for the attenuated versions of the leak sets. Since training of a SOM amounts to a stochastic optimization procedure, every experiment was repeated 20 times. Scatter plots of the MSE and the GOF figures resulting for the (progressively attenuated) leak and noise vectors are shown in figures 6.17 and 6.18. Here, '+'-signs denote the MSE and 'o'-signs the GOF figures. Figure 6.17 shows the outcomes for train leak and noise sets, figure 6.18 for the interpolation and extrapolation sets. It can be observed that the leak sets used to train the SOM can well be discriminated from the noise set: the distance to the straight line $error(noise) = error(leak)$ is fairly large, even for a large amount of attenuation. The same holds for the interpolation data, figure 6.18(a). Extrapolation to data with very different characteristics leads to worse discrimination for large attenuation, figure 6.18(b): attenuation with 20 dB results in an error which is larger for leak than for noise data. Both error measures (MSE and GOF) have sim-

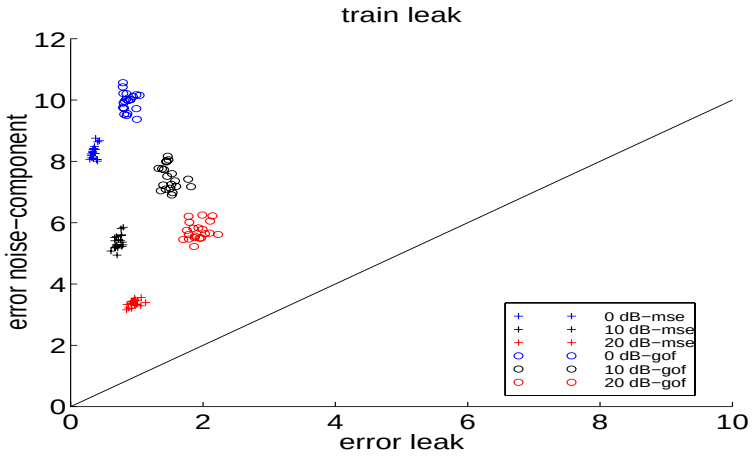


Figure 6.17: Scatter plots of errors on *noise* and *leak* sets. The attenuation increases from 0 dB (topmost two clusters), via 10 dB (middle two clusters) to 20 dB (lowermost two clusters). The difference between the MSE and GOF-clusters is due to the topology-distortion based penalty term in the latter error measure

ilar discrimination capacity: the clusters of goodness-figures are slightly more diffuse and translated from the origin with respect to the straight line, indicating a nearly constant offset with respect to the quantisation error for both leak and noise data. This can be understood by noting that the goodness measure computed over a batch of samples consists of the average quantisation error over the batch plus an "orderliness" term that is a mainly a feature of the map. Differences in input data are mainly reflected in the MSE-part, orderliness penalties will be comparable for incompatible data. For data used to train the map, the situation is different: the map has become ordered with respect to this dataset, so the orderliness penalty will be somewhat smaller. This can be seen in figure 6.17 as the somewhat vertical offset

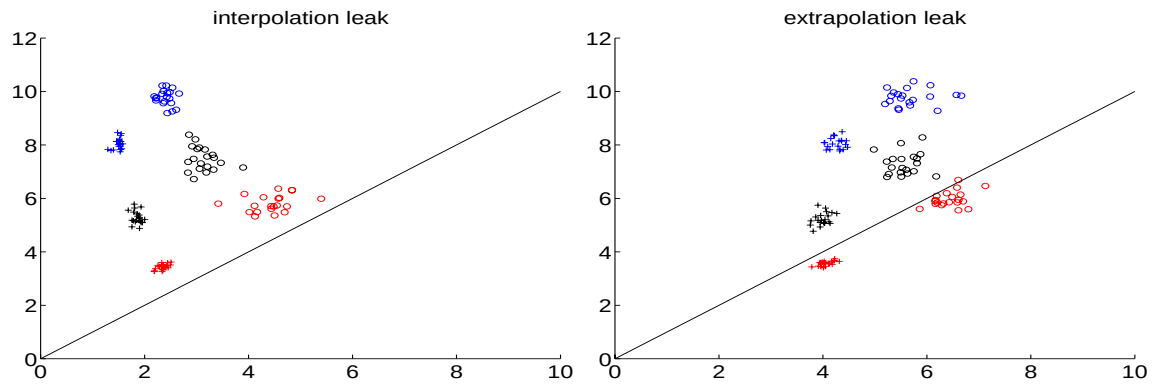


Figure 6.18: Scatter plots of errors on *interpolation* (a) and *extrapolation* (b) sets

of the GOF-clusters from the MSE-clusters. With the training set, map nodes will occur as starting points for the computation of the orderliness term according to the input distribution, opposed to incompatible datasets that will select border nodes more often than inner nodes. The border effect that occurs in SOM training (*the map is less well ordered near the borders of the map* [Koh95]) can lead to higher orderliness penalties in the latter case. Moreover, the probability that an input vector will have one- and two-nearest neighbours in the input space that are not neighbours on the map is expected to become larger as the distance of the vector to the map increases. This justifies the use of map regularity with respect to a certain dataset for determination of the compatibility between a SOM and a dataset.

Description of typical leak measurements in Norway dataset

In the following experiments, we investigate a SOM for leak detection using data that was measured under real conditions (i.e. at the North Sea near Norway, see section 1.5.3). We assume that the contributions of pure noise and leak are not coherent, so in the power spectrum of their sum all cross-terms will be zero. Hence, we can look upon the power spectrum of a measured leak signal (always consisting of pure leak and a noise floor) as a sum of the spectra of the pure leak and noise contributions. After subtraction of the local noise floor spectrum (occurring just before the blowdown), we expect to retain (an approximation to) the clean leak spectrum. We call such a dataset *corrected data*. We trained SOMs of different size (ranging from 5x5 to 14x14) and stiffness (final neighbourhood ranging from 1 to 3) using a subset of the available leaks that had positive residual after subtraction of the local noise. This subset consisted of leak measurements that were confidently classified as leak using a leak detection feedforward neural network. The maps were evaluated with five kinds of data: the set used to train the map with (corrected selected leak) (**set 1**), an independent test set from the same distribution (**set 2**), a set containing corrected leaks from the subset of noise-like leaks (the complement of the prominent leaks with significant noise-residual) (**set 3**), a set of average noise (not corrected, since this would result in negative residuals) (**set 4**) and a set of leak-like noise (**set 5**). Note that the noise sets did not contain the noise-spectra that were used in correction of the leak spectra. The procedure was repeated 25 times, and

mean, minimum and maximum values of the quantisation error (distance from a set to the trained map) was determined for each dataset. In all maps, a result comparable to table 6.2 (corresponding to a 6x6 SOM with final neighbourhood width 3, final learning rate 0.05, and training for 30000 cycles) could be observed. It is clear that the set with difficult noise

Table 6.2: Threshold determination for SOM-based leak detection. Mean AQE for five datasets, based on 25 repetitions of training a 6x6 SOM on corrected leak data

dataset id.	min error	mean error	max error
set 1	0.012	0.013	0.014
set 2	0.017	0.017	0.017
set 3	0.031	0.036	0.038
set 4	0.075	0.095	0.092
set 5	0.031	0.036	0.040

(**set 5**) cannot be separated from the leak data based on the mean-squared quantisation error. However, by putting a threshold at 0.07, noise with average behaviour (approximately 75 % of the available noises NOT used for correction) will be rejected confidently.

Detection of ship noise

The former procedure was repeated with the *uncorrected data*, i.e. using 11-dimensional feature vectors, without subtraction of local noise. The set of leak vectors for map training and testing was chosen from the total set by retaining the vectors representing prominent leaks. In this case, the most difficult leaks (i.e. the leaks most resembling noise) were discarded, along with several other samples due to 'difficult' measurement conditions. The remaining leak vectors were now attenuated with factors $0, 2, 5, 10, 15, 20, 30$ dB and *superimposed* on the (corresponding) local noise. Hence, the remoteness of difficult leaks still detectable by a SOM can be determined. A set of difficult leaks was constructed as well. The sets of normal and difficult noise were the same as in the previous procedure.

A Self-Organizing Map was trained on a leak set, where the attenuation level was zero. Next, the SOM was evaluated with all sets for all attenuation levels. This procedure was repeated 10 times, and mean, minimum and maximum quantisation errors were determined for all sets (figure 6.19). It turned out that a combination of moderate maps size (6x6, 8x8) and significant final neighbourhood (having a value of 3) enabled separation of leak from normal noise up to a certain attenuation level. In figure 6.19, at each evaluated attenuation level, the minimum (lower whisker), maximum (upper whisker) and mean (crossing point vertical and horizontal line, approximately halfway the whiskers) quantisation error are plotted for all five data sets. Setting the threshold at 0.015 enables the rejection of normal noise (the uppermost line around 0.017) from train (solid line starting around 0.011), test (dash-dotted line) and difficult leak (solid line around 0.014), up to 10 dB attenuation of the leaks. The leak-like noise (lowermost line) is clearly not separable from leaks using the SOM. One needs a separate classifier to distinguish between these difficult noises and leaks. Note that the noise set

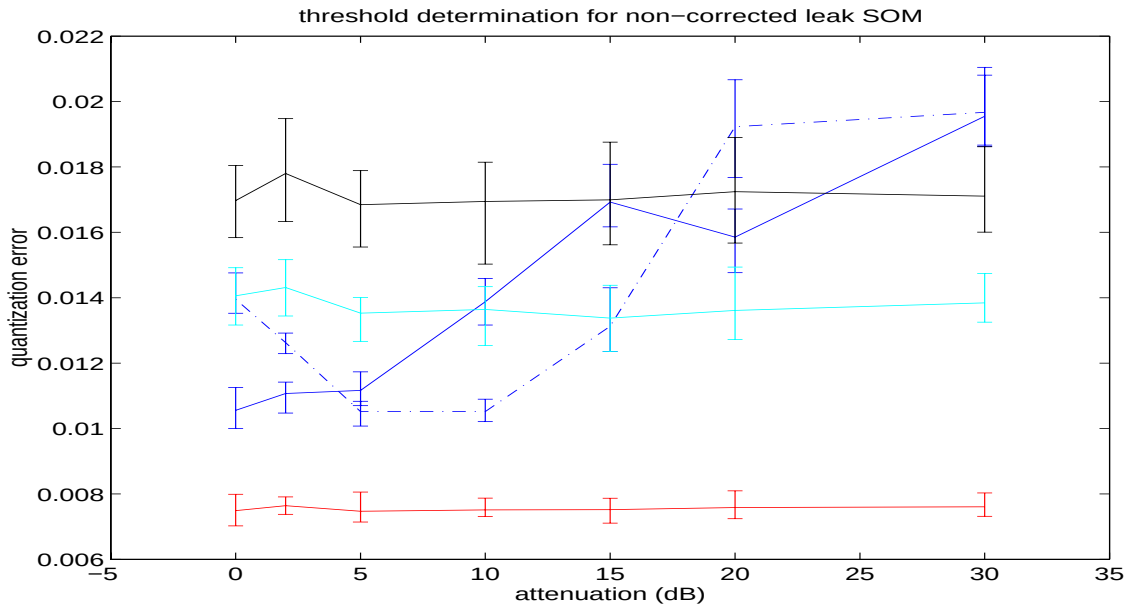


Figure 6.19: Threshold determination with SOM. The figure shows the mean AQE for five datasets (the five different graphs) with various attenuation levels (horizontal axis), based on 10 repetitions of training a 6x6 SOM on non-corrected leak data

does not change with attenuation level. However, the map is retrained on the global training set (that does not change with attenuation level either) for each new attenuation level, which causes that the "level" of both noise sets is somewhat variable. The results above indicate, that the standard goodness measure (MSE) already enables discrimination between normal noise and all leaks, up to 10 dB attenuation.

The previous experiment was repeated with a 8x8 SOM (final neighbourhood width 3, 30000 cycles training), and now a set with different ship noise (acquired at the 'IJ') was used for map evaluation as well. Every run (train the SOM on the leak set, evaluate all *six* data sets on the SOM for all attenuation levels) was repeated for 25 times. The results are shown in figure 6.20. The explanation of the graphs is the same as in the previous figure. Additionally, the errors for ship noise are plotted as the uppermost dash-dotted line. From this figure it is clear that putting a threshold at 0.011 will cause a rejection of "normal" noise and ship noise, at the cost of rejecting a few difficult leak samples. Remark that when a leak persists, it will eventually become closer to the map, which may cause acceptance of these leaks after suitable postprocessing. For attenuation levels up to 15 dB the average mean-squared-error over a batch of samples from a dataset (over 25 repetitions) is the largest for the ship noise dataset.

6.5 Discussion

In this chapter we investigated learning methods for novelty detection with small sample sizes. The assumption is that periodic measurements for the purpose of health monitoring

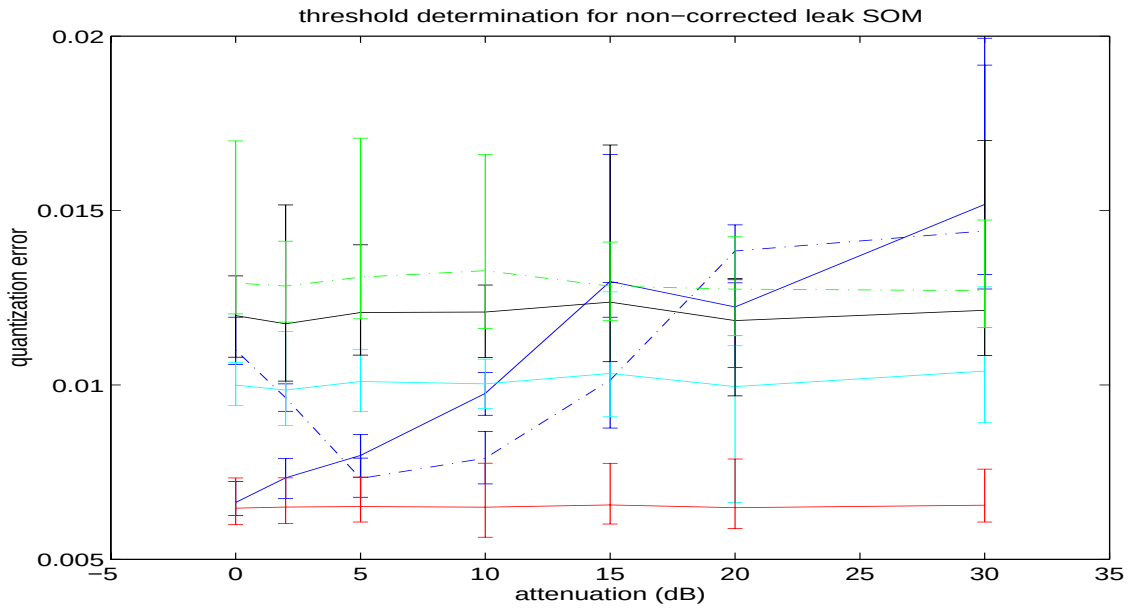


Figure 6.20: Evaluating the novelty of ship noise. The figure shows the mean AQE for six datasets (the six different graphs) with various attenuation levels (horizontal axis), based on 25 repetitions of training a 8x8 SOM on non-corrected leak data

can be expensive and that the number of monitored items (in the case of machine monitoring) may be small. Moreover, a high-dimensional feature vector may be required to describe the health state adequately.

The *wavelet network* is a constructive neural network method, that allows for incorporation of prior knowledge, since position, scale and width of the wavelet basis functions may be initialized using a wavelet decomposition or by using general heuristics about the domain of the dataset. However, succesful training of a wavelet network is probably very dependent on a proper initialization. Moreover, the number of data samples that is needed with high-dimensional feature vectors would soon become prohibitively large. Hence, the network was not considered suitable for novelty detection in a practical monitoring setting.

The *Self-Organizing Map* is trained to quantize a dataset as good as possible, under the constraint of preservation of the topology of the input data. If the accuracy of the topology preservation is not tracked during training, this leads to minimization of the average quantization error while the topology of the dataset may not be preserved properly. This can have the effect that samples that are close in the input space may not always be mapped to adjacent (or close) nodes on the map. A goodness-of-fit measure can be used to track this effect. It was observed experimentally that the neighbourhood width in the SOM fine-tuning phase determines the 'stiffness' of the map: less stiff maps can fit the local data characteristics more easily, which may lead to overtraining. In pipeline leak detection experiments, a threshold could be found such that measurements resembling (only) the training class were accepted, whereas dissimilar (novel) measurements were rejected. This indicates that SOM-based novelty detection is a feasible approach for this particular fault detection problem.

In the *Support Vector Data Description* the volume of a sphere is minimized, where the sphere comprises a *mapped* version of the dataset. This mapping is performed implicitly by replacing the inner products in the description formula with a kernel function. The kernel *induces* a mapping of the dataset into a high-dimensional *feature space* by computing 'kernel similarities' between a data sample and a support vector in the original (input) space. Analogous to the support vector method for classification and regression, slack variables can be introduced that allow for a trade-off between errors made on the target set and the volume of the sphere. This enables a tight description of datasets with arbitrarily shaped domains. When choosing Gaussian kernels in the method, the trade-off is controlled by the spread parameter. This can be set on the basis of the requested *target acceptance rate*. The main drawback of the method is its computational complexity, which can become impractical for large sample sizes. We showed experimentally that in two machine monitoring applications an SVDD-representation of the measurements could be made such that a satisfying trade-off between novelty detection and acceptance of measurements from a healthy machine is obtained. The use of AR or spectral features produced the best results; ICA or PCA fusion into one virtual sensor channel allowed for a decreased false alarm rate at reasonable novelty detection rates (in the case of AR features). The appropriate number of virtual channels appears to be determined by the number of underlying sources in a multichannel measurement.

We introduced the *k-centers* method for approximation of the domain of a dataset. It selects a subset of the data as *centers* in a set of spheres. The overall radius is minimized for a predefined number of centers; each data sample is included in at least one of the spheres. For strongly clustered data the number of spheres appropriate for the dataset can be chosen by a linear search over the number of clusters. If the data is strongly clustered, the proper number of clusters can be estimated with a minimal spanning tree that is fitted to the data distance matrix. The computational complexity of the *k-centers* method is only dependent on the data sample size (not on the feature dimensionality), since only a distance matrix of the data is used to determine the data description. The method is less suitable for datasets containing occasional outliers or datasets that are in a subspace, since (a) the overall radius will be determined by the sample with maximum distance to its nearest center, and (b) the use of multidimensional spheres will lead to a description that includes samples in possibly singular data dimensions. For high-dimensional datasets that represent system measurements this may require a prior mapping into the underlying data subspace, for example with PCA. The method may be used for novelty detection in rotating machines if the analyst is certain that all samples in the set are proper instances of the normal behaviour. However, application of a cross-validation method to determine the number of spheres using an a priori determined fraction of accepted samples in a (normal) test set may give some level of tolerance to outliers.

We conclude that novelty detection proved to be a feasible approach for system monitoring in three real-world case studies. Here, domain approximation using samples from *one system state only* allowed for detection of (most) anomalies. The trade-off between acceptance of normal samples versus rejection of novel samples was satisfactory in the case of submersible pump monitoring, provided the measurements are represented with a proper feature vector, like AR-coefficients or spectral amplitudes.

Chapter 7

Recognition of dynamic patterns

7.1 Introduction

During the lifetime of a machine, components will wear out and this process may give rise to fault development. For an automatic health monitoring method this implies that the initial description becomes obsolete after some time. The question that we address in this chapter is: how to take this changing system behaviour into account in a fault recognition method? This corresponds to the subtasks “dynamic pattern recognition” and “trending” in the scheme of figure 2.4.

Dynamic pattern recognition

Many techniques in pattern recognition deal with static environments: the class distributions are considered relatively constant as a function of the time in which feature vectors are acquired. Time often only plays a secondary role: it should be incorporated in the feature extraction procedure. For practical recognition tasks, the assumption of stationarity of the class distributions may not hold. Alternatively, information in sequences of feature vectors may be used for recognition. We will call both groups of problems *dynamic pattern recognition* problems. A *dynamic pattern* is a multidimensional pattern that evolves as a function of time. With dynamic pattern recognition we mean clustering or classifying an ensemble of dynamic patterns using either dynamical characteristics of the data or adaptivity of a recognition method to changing data characteristics. Suitable methodology can be found in multi-dimensional curve fitting and clustering, nonlinear time series analysis and probabilistic dynamic methods (like Kalman filters and hidden Markov models).

In the previous chapter we described three methods for the description of the domain of a dataset. We showed that such a description can be used for novelty detection in real-world applications. However, if the monitored system has nonstationary characteristics on the large time-scale, the initial description will become obsolete after a while. A retraining to incorporate novel patterns seems necessary. The concept of extending a description of admissible system behaviour as time proceeds is shown in figure 7.1. The dots represent feature vectors, which are described initially using one of the methods from the previous chapter (the k-centers method is displayed in the figure). As time proceeds (the arrows in

the figure), novel patterns (displayed in the figure as deviating grey values) will be observed that reside outside the initial 'normal' domain. Either a transition to a known fault area or an unseen area may occur.

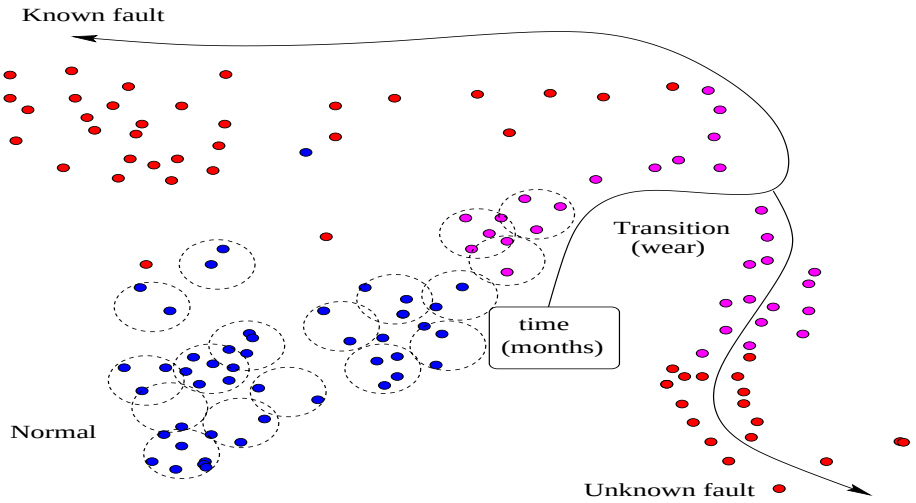


Figure 7.1: Adaptive description of the admissible system domain

The use of context

If a typical wear trajectory exists for a machine or a component, learning this trajectory may be helpful for fault prognostics. However, in cases where typical fault evolutions are not available, the most we can do is describe the behaviour of that machine at a certain stage, detect novelty and adapt the description if necessary. The latter action may prove necessary if system behaviour is changing, but not designated as faulty behaviour. These two approaches are indicative of a deeper issue: how to represent the temporal aspect of the measurements? Fault development may show up as a clear trend in an ordered set of (multi-dimensional) health indicators. If this trend is to be modelled, the ordering in the feature vectors is one of the main characteristics of the dataset. This corresponds to the former (context-sensitive) approach.

Alternatively, a set of feature vectors can be looked upon as the result of independent draws from a multi-dimensional distribution (the latter, context-insensitive, approach). All temporal information should now be present in each feature vector. Fault detection in rotating machines may then be based on the dissimilarity of a set of newly measured feature vectors with respect to a set of known templates. The trajectory of the machine in the feature ('health') space is not utilized explicitly. Changing system behaviour can only be learned by adapting the current description to novel patterns. Once a fault development has been fully monitored (and learned on-the-fly), the description may be used for tracking a new failure in the system.

7.2 Context-insensitive recognition

7.2.1 Adaptive classification

An important aspect of the nonstationary nature of long-term rotating machine measurements is the invalidity of an initial description of the normal behaviour, since new samples due to later measurements will exhibit deviating characteristics. A similar problem is encountered in *speaker identification in a changing environment*. In the process of speaking, persons will utter diverse sounds that will be similar to the corresponding sounds they make at other instances. Hence, a speaker can be represented by a set of (representations of) 'typical sounds', which allows for an (adaptive) nearest neighbour classification approach. In [Spe99] the identification of speakers using a simple nonlinear classifier, a 1-nearest neighbour classifier, is investigated. It is assumed that neighbouring prototype frames are *independent* of each other. A prototype frame is indicative of a sound that is uttered, e.g. the most energetic frame in a sequence of 10 frames that represents the same sound. Applying a majority vote to all individual classifications obtained with a set of prototype frames will lead to a more reliable recognition method. Under several (reasonable) assumptions, the reliability of a speaker identification experiment can be estimated beforehand. The proposed method can be extended to *growing* speaker databases, so that reliability as a function of the number of speakers can be estimated beforehand as well [Spe99, YSD01].

Work on adaptive description of datasets can be found in [GLW99], where an algorithm was proposed for the adaptive clustering of data through time. Furthermore, the support vector data description from chapter 6 can be made adaptive to changing system behaviour as well. In [Mel00a] this was applied successfully for the adaptive description of measurements from a progressively loose foundation induced to the submersible pump from chapter 1.

7.2.2 System tracking with Self-Organizing Maps

The topology preservation property of the Self-Organizing Map makes it suitable for tracking of system behaviour. Applications can be found in speech recognition [Kan94], evolution of weights during neural network training [Hoe99] and system monitoring [Koh95]. However, since the SOM mapping is not *ordering preserving*, mapping of the system trajectory may give rise to jumps between relatively remote areas of the map. This behaviour is more likely to occur if map dimensionality is much smaller than the intrinsic dimensionality of the dataset. This is not troublesome, however, if map areas indicating health states can be discerned. In cases with data that is labeled according to health state (e.g. OK - opmode 1, OK - opmode 2, small wear, larger wear, fault, etc.) one can *calibrate* the map after training. The training process is an unsupervised learning process in which the labels are not used. After training the labeled data can be projected onto the map; a map node is then labeled according to the majority of the samples that 'hit' this particular node[1]. In system monitoring, a trained map can be used to explore health-related structure in data. Typical SOM-based analyses are:

[1] A data sample 'hits' a map node if that node matches the sample best, e.g. in terms of the Euclidean distance between node and sample

- monitoring the operating point of the system; a novel measurement (coded as a feature vector) is projected onto the map. If the distance to the map is within a novelty threshold, the node where the sample 'hits' the map may then be interpreted as the operating point of the system in the 'health space'

- clustering structure of the map nodes; data samples are represented by map nodes, that will ideally show a clustering that is similar to the data samples. The distance matrix of the map nodes can be determined and visualized in turn. Examples of this approach can be found in the unified distance matrix (U-matrix) visualization method by Ultsch [Ult93] or the visualization method proposed by Kraaijveld et al. [KMK95]. For clustered nodes this will give rise to images where ridges are separating different clusters of nodes. Alternatively, the map nodes can be clustered by considering the nodes as data samples, applying a standard clustering method (like *k-means* clustering) to this dataset and colouring the nodes according to the obtained clustering

- inspection of nodes; if a feature vector is interpretable for the user (e.g. a discretized spectrum), inspection of the learned prototype feature vectors can be helpful for labeling map areas

- component plane analysis; a component plane consists of positions of map nodes in one particular direction (component) of the feature space. For 2-D maps, this leads to images where colours or grey-values of the pixels can be interpreted as amplitudes of a particular feature (e.g. energy in a subband, when a spectrum is used a feature vector). Hence, a map node cluster may be identified by extracting its 'meaning' from the corresponding cluster on a component plane

- hitmap analysis; positions of the 'hits' of a new set of (preprocessed) measurements can be used to identify the characteristics of new measurements

- novelty analysis; the percentage of samples in a dataset that exceeds a novelty threshold (after projection of the set) can be used for determining the similarity of this dataset to earlier observations

- trajectory analysis; if a series of patterns is presented to the map in an ordered manner, a trajectory that represents the degradation process may be observed

The SOM-based system tracking and analysis process is illustrated in section 8.5.

7.3 Context-sensitive recognition: hidden Markov models

7.3.1 Dynamic system modelling

A general class of models for dynamical systems is the class of linear Gaussian models (LGM) [RG99]. This class incorporates both Kalman filters and hidden Markov models. Here, a state vector $\mathbf{x}_t$ is introduced that describes the dynamics of a system. Observables $\mathbf{y}_t$

are noisy functions of those state vectors. If the state vector takes its values in a continuous range, a linear state space model describing the system is:

$$\begin{aligned} \mathbf{x}_{t+1} &= A\mathbf{x}_t + \mathbf{w}_t, \qquad \mathbf{w}_t \sim \mathcal{N}(0,Q) \\ \mathbf{y}_t &= B\mathbf{x}_t + \mathbf{n}_t, \qquad \mathbf{n}_t \sim \mathcal{N}(0,R) \end{aligned} \tag{7.1}$$

The terms $\mathbf{w}_t$ and $\mathbf{n}_t$ describe the state noise and the observation noise respectively. An example of a continuous-state model is a *Kalman filter*.

Hidden Markov models

When the state vector takes only discrete values, we have the state space model:

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathrm{WTA}[A\mathbf{x}_t + \mathbf{w}_t] \\ \mathbf{y}_t &= B\mathbf{x}_t + \mathbf{n}_t \end{aligned} \tag{7.2}$$

Here, the WTA$[.]$ function is the *winner-takes-all* function, i.e. a quantization of the continuous state value. The criterion for finding the winner determines the nature of the quantization. A hidden Markov model is an example of a discrete state Gaussian model. It can describe a finite set of discrete states. A prior distribution Π over the set of states specifies the beliefs about starting in a certain state. A transition matrix A specifies the state transition probabilities. Sequences of observables are assumed to be drawn from a finite-size alphabet of symbols[2]. This corresponds to observations in an alphabet of symbols. The probability of emitting a certain symbol when being in a certain state is given by the matrix B.

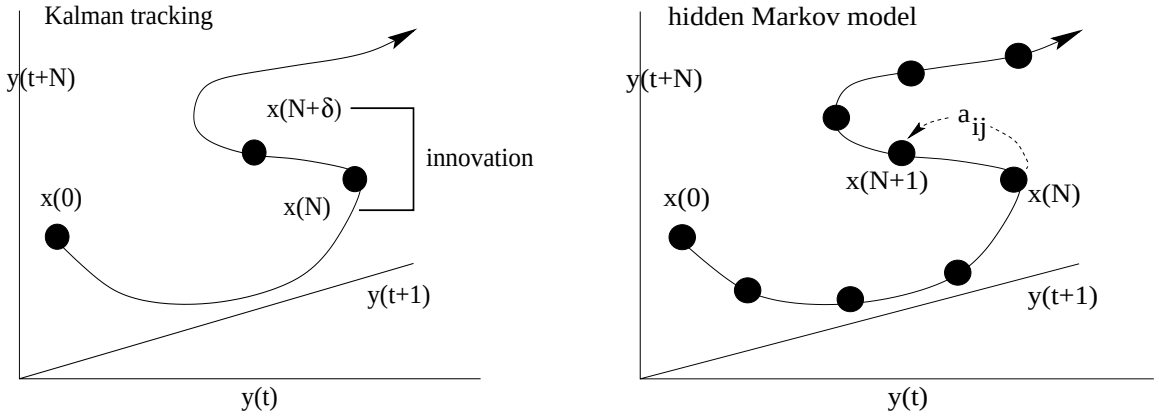


Figure 7.2: Difference between Kalman filter (a) and hidden Markov model (b)

Both Kalman filters and HMMs can be used to model the dynamics of a system. If the system trajectory is continuous, a Kalman filter should be used, figure 7.2(a). If a discrete set of states

[2]Also continuous-observation models exist, but in the experiments with machine signals reported later, we chose to quantize observations into a number of discrete levels

(a discretized trajectory) should be modelled, a hidden Markov model is more appropriate, figure 7.2(b). Once the dynamics has been modelled, transition to a new dynamic regime can be detected by monitoring the residual between observations and model predictions.

Example 7.1: detecting a changing system dynamics

In [PR98], a Kalman filter implementation of an autoregressive (AR) signal model is formulated. The AR model was discussed in chapter 3, where a signal amplitude is expressed as a linear combination of previous signal amplitudes:

$$y_t = -\sum_i^p \alpha_i y_{t-i} + n_t \tag{7.3}$$

The model can be made time-dependent by formulating it as:

$$\begin{aligned} \mathbf{q}_{t+1} &= \mathbf{q}_t + \mathbf{w}_t, & \mathbf{w}_t &\sim \mathcal{N}(0, \mathbf{W}_t) \\ y_t &= F_t \mathbf{q}_t + n_t, & n_t &\sim \mathcal{N}(0, \sigma_t^2) \end{aligned} \tag{7.4}$$

where $F_t = -[y_{t-1}, y_{t-2}, \ldots, y_{t-p}]$ and the state variables $\mathbf{q}_t = [\alpha_{1,t}, \alpha_{2,t}, \ldots, \alpha_{p,t}]^T$ are time-varying AR coefficients. When a signal moves from one dynamic regime to another, the state noise will be temporarily high. By monitoring the state noise a segmentation into dynamical regimes can be obtained, figure 7.3. Here, a signal consisting of multiple sinusoids and noise is fitted with an AR model (based on an initial segment of the signal) and the parameters are reestimated on-line. In the state noise plot, figure 7.3(b), the transition to a different dynamic regime is clearly observable at time stamps corresponding to the frequency transitions in figure 7.3(a). □

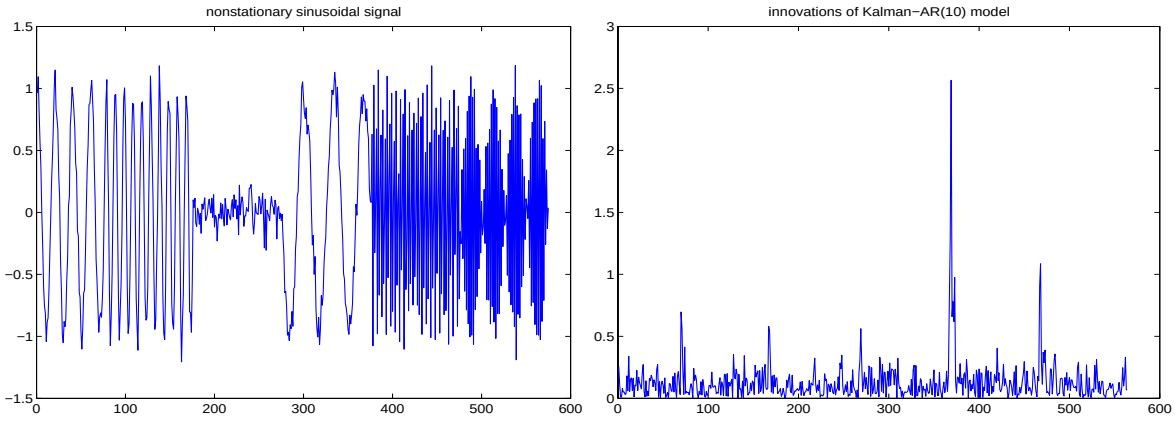


Figure 7.3: Detection of changing dynamics: a. test signal; b. state noise plot

7.3.2 Training a hidden Markov model

The HMM training procedure can be written down as:

1. Inference: a. (*forward-backward* algorithm) the forward-backward algorithm is used to estimate $P(\mathbf{Y}|\lambda)$, the probability that the sequence $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}$ is observed, given the model parameterized by parameter vector $\lambda = (\Pi, A, B)$; **b. state sequence estimation:** (*Viterbi* algorithm) computation of the most probable *state* at each time may not give most probable *sequence*. Hence, the **Viterbi** algorithm is often used for finding the optimal state sequence. It is a dynamic programming method for backtracking all state sequences in order to find the one with least cost

2. Learning: (*Baum-Welch* algorithm) the Baum-Welch algorithm is an EM-algorithm for adjusting the model parameters λ in order to maximize the likelihood $P(\mathbf{Y}|\lambda)$ of the observation sequence $\mathbf{Y}$. The Expectation-Maximization (EM) algorithm is appropriate for modelling of data that is generated by using variables that are unknown (the data has "missing values"). The algorithm consists of repeated executions of two interleaved steps: E = estimating new prediction of the missing values based on the current parameter setting, and M = updating the parameters on the basis of the previous predictions. Under certain conditions, the algorithm is guaranteed to increase the likelihood of the solution after each combined E + M step and end up in a local likelihood maximum of the parameter space.

7.3.3 HMM-based method for time series segmentation

In [Smy94] an approach to fault detection with HMMs has been proposed, see figure 7.4. The assumptions in this approach are that a. state sequence information is useful; b. the (hidden) state sequence is Markovian; c. priors of parameters can be obtained using prior knowledge about the monitored system, for example using heuristics like:

- a_{ij} may be based on mean-time-between-failures (MTBF)
- for proper scaling, upper & lower bounds on feature values are known
- one can estimate a priori how often the system will be in an unknown state.

One problem occurring in the design of an automatic fault detection system is the *determination of thresholds for diagnostics* of the system. Often this is done on the basis of heuristics, see for example [oCI]. We propose the use of hidden Markov models for segmentation of time series (e.g. multiple trend plots indicative of the same underlying wear phenomenon) into common (hidden) health states. Here we use the assumption (chapter 1) that fault development can be modelled as a Markov process and that observed wear indicators give an indirect view of the underlying (hidden) wear process. The aim is to find a set of alarm thresholds more or less 'automatically' by using the dynamic structure in a set of time series that assess machine wear. We omit the *unknown* state in the models; novelty detection has been addressed in the previous chapter. Therefore, it remains to find a way to specify the initial state transitions a_{ij}, and design a suitable HMM architecture. An (approximate)

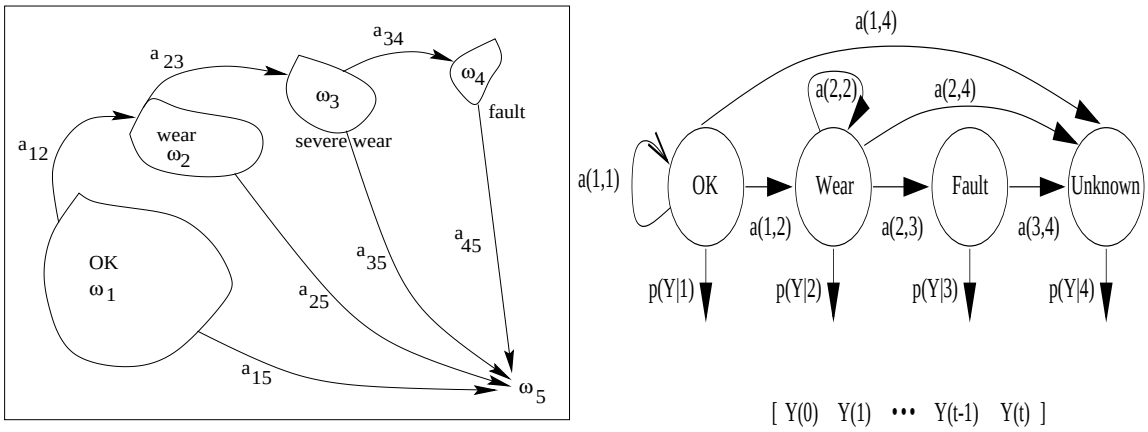


Figure 7.4: Hidden Markov model for fault detection with unknown classes, from [Smy94]

left-right model makes sense in the machine wear segmentation application (i.e. mainly local connections to immediately neighbouring states and self-connections are present in the HMM). In [Hec91], the number of states suitable for describing the tool wear process is said to range from 3 to 6. In a general setup, a designer may add an initial state that accounts for the 'infancy diseases' of the machine (cf. the 'bathtub-curve' from chapter 1). Hence a 4 state left-right model plus a few sub-states may be proposed for modelling machine wear. Bearing wear is considered to traverse four stages, cf. section 1.2.3.

7.3.4 Experiment: segmentation of machine wear patterns

The HMM approach is evaluated in a real-world application with a gradually deteriorating gearbox that was monitored continuously during two months [Ypm00, YSD01]. Assistance in measurements was provided by SKF ERC and Condition Monitoring b.v., The Netherlands. The measurement setup has been described in section 1.5.2.

Feature extraction, alignment and scaling

The machine health state was monitored at 8 positions with 9 different measurement quantities. This results in 72 time series. If each instantaneous measurement at a certain position (which is, in fact, a complete spectrum) is represented by a single overall RMS value, this is a set of 72 1-D time series. A dataset was constructed from these series. From this dataset, a smaller subset was extracted: from the first 4 sensor positions, the first 6 measurement quantities were used. Upon visual inspection, this set of time series seemed more homogeneous than the complete set of time series. This can be related to the fact that the fault developed in the first gear stage (which is monitored with the first 4 sensors). Moreover, the acoustic emission measurements were markedly different from (relatively) low-frequency vibration based quantities.

In order to make all time series the same length we have tried several approaches. First, the samples due to weekend measurements (in fact: artifacts) were determined. We used

the fact that the time series showed a multimodal distribution, where the mode around the smallest mean represented the spurious samples. A threshold could be based on the mean and standard-deviation of this mode, and the spurious samples could be removed. In order to fill up the gaps, (the hidden Markov modelling approach has difficulties with unequally sampled time series) the samples of a time series were looked upon as samples from a random variable and new samples were generated around the existing ones using *k*-nearest neighbour noise injection (chapter 6). Second, the weekend measurements were just left out, since the samples representing weekends were measured at roughly the same instances (for all sensors and all measurement quantities). The main differences in timing were present at the beginning of the measurement period (i.e. end of november/ start of december 1998), so the first measurements were skipped in the time series, leading to equal-length and approximately equal period time series.

The dynamic range of different one-dimensional time series (measurement quantities and sensor positions) is fairly different. Hence, we normalized each individual time series between their minimum and maximum values. No particular outlier removal procedure was used. Moreover, the speed of degradation (steepness of the overall values trend at the end of the monitoring period) was not accounted for in this procedure. After normalization, all time series have values in the range $[0, 1]$.

Since we chose to work with discrete-observation hidden Markov models, we have to represent the values of a time series using some symbol alphabet. Alternatively, each time series value is assigned to a 'bucket' of nearby values, to which is assigned a symbol. For a one-dimensional time series, this means that each time series value is quantized into a level. When assigning letters to each level, going from small to large values means that letters 'high in the alphabet' correspond to 'bad condition', whereas 'early letters' correspond to 'good condition' (in general).

Segmentation with k-means clustering

We used the *k-means* clustering algorithm to segment 8 of the 72 available one-dimensional time series into three classes (OK, pre-alarm and alarm). It is observed (not shown) that the clustering is done with respect to the amplitudes without taking into account the sequential structure in the time series. Hence, rather artificially looking temporary transitions to a state (cluster) followed by a transition back to the former state occur. The thresholds for cluster transitions can be different for each time series. Validity of the thresholds is quite dependent on the proper scaling of the data and removal of outliers, similar to the HMM approach. However, self-restoring effects [BB96] cannot be accounted for by using this segmentation method.

HMM-based segmentation of 1 selected time series

We selected time series no. 3 that represents acceleration up to 10K Hz (measured at sensor 1, near the fault position) as the trend to be modelled. The time series was quantized into 16 levels and normalized to equal lengths. We trained an HMM on the time series and inspected

the resulting segmentation into 'health states'. The number of states[3] in the model is set to 6, where the connection structure is such that there are two groups of states: states 1-3 describe the normal behaviour, whereas states 4-6 describe the fault development. We initialize the parameters as follows: initial state distribution $\Pi = [0.6\ 0.4\ 0\ 0\ 0\ 0]$, the priors for transition matrix A and state-to-observation matrix B are shown in table 7.3.4. In this table, $\mathbf{1}(p,q)$ denotes a $p \times q$ matrix with unit entries. Note that this is not a left-right model, since it allows for (small-range) backward transitions and two different starting states.

Table 7.1: Prior settings of A and B matrices in first experiment using 1 time series

$A =$

	1	2	3	4	5	6
1	0.8	0.1	0.1	0	0	0
2	0.2	0.7	0.1	0	0	0
3	0	0.2	0.4	0.4	0	0
4	0	0	0.1	0.4	0.4	0.1
5	0	0	0	0.2	0.4	0.4
6	0	0	0	0.1	0.1	0.8

$B =$

	1 - 12	13	14	15	16
1	0.96 * **1**(1,12)/13	0.96 * 1/13	0.04 * 1/3	0.04 * 1/3	0.04 * 1/3
2	0.96 * **1**(1,12)/13	0.96 * 1/13	0.04 * 1/3	0.04 * 1/3	0.04 * 1/3
3	0.96 * **1**(1,12)/13	0.96 * 1/13	0.04 * 1/3	0.04 * 1/3	0.04 * 1/3
4	**1**(1,12)/120	0.3	0.3	0.2	0.1
5	**1**(1,12)/130	1/130	0.2	0.3	0.4
6	**1**(1,12)/130	1/130	0.2	0.3	0.4

Training on time series no. 3 resulted in the parameters: $\Pi = [1\ 0\ 0\ 0\ 0\ 0]$ and A, B are given in table 7.2. Quantization levels 1 to 8 are shown in the first subtable, levels 9 to 16 are shown in the second subtable. The resulting segmentation into health states is shown in figure 7.5. The transition from no wear, to moderate and severe wear is clearly discernible. In the observation matrix B this is visible as a large probability of emitting 'higher symbols' in the states indicating more progressive wear. No unexpected dynamics is present: the transition matrix seems to code each state according to the magnitude of the health index.

In a second setup, the prior over the states was chosen as: $\Pi = [0.8\ 0.2\ 0\ 0\ 0\ 0]$. The transition matrix was initialized as shown in table 7.3. B was chosen identical to the previous setup. The learned state distribution vector was now $\Pi = [0.8\ 0.2\ 0\ 0\ 0\ 0]$. However, the learned A and B matrices were nearly identical to the learned matrices from the first experiment. The resulting optimal state sequence was also identical to the first experiment. This result was observed for several other settings as well. We conclude that for the time series at hand, hidden Markov segmentation leads to results comparable to a simple clustering according to amplitudes.

[3] In [Hec91] the # states suitable for describing tool wear process ranges from 3 to 6

Table 7.2: Learned A and B matrices using 1 time series

$A =$

	1	2	3	4	5	6
1	0.99	0.01	0	0	0	0
2	0.01	0.98	0.01	0	0	0
3	0	0.02	0.93	0.05	0	0
4	0	0	0.03	0.93	0.03	0.02
5	0	0	0	0.09	0.91	0.00
6	0	0	0	0.07	0.07	0.86

$B =$

	1	2	3	4	5	6	7	8
1	0.09	0.06	0.26	0.21	0.14	0.22	0.03	0.02
2	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.47
3	0.01	0	0	0	0	0.00	0.00	0.00
4	0.00	0	0	0	0	0	0	0.00
5	0.00	0	0	0	0	0	0	0
6	0.00	0	0	0	0	0	0	0

9	10	11	12	13	14	15	16
0.00	0.00	0	0	0	0	0	0
0.40	0.00	0.00	0.00	0	0	0	0
0.03	0.40	0.08	0.48	0.00	0.00	0.00	0
0.00	0.00	0.02	0.02	0.94	0.02	0.00	0.00
0	0.00	0.00	0.00	0.10	0.90	0.00	0
0	0	0 0.00	0.26	0.00	0.00	0.61	0.13

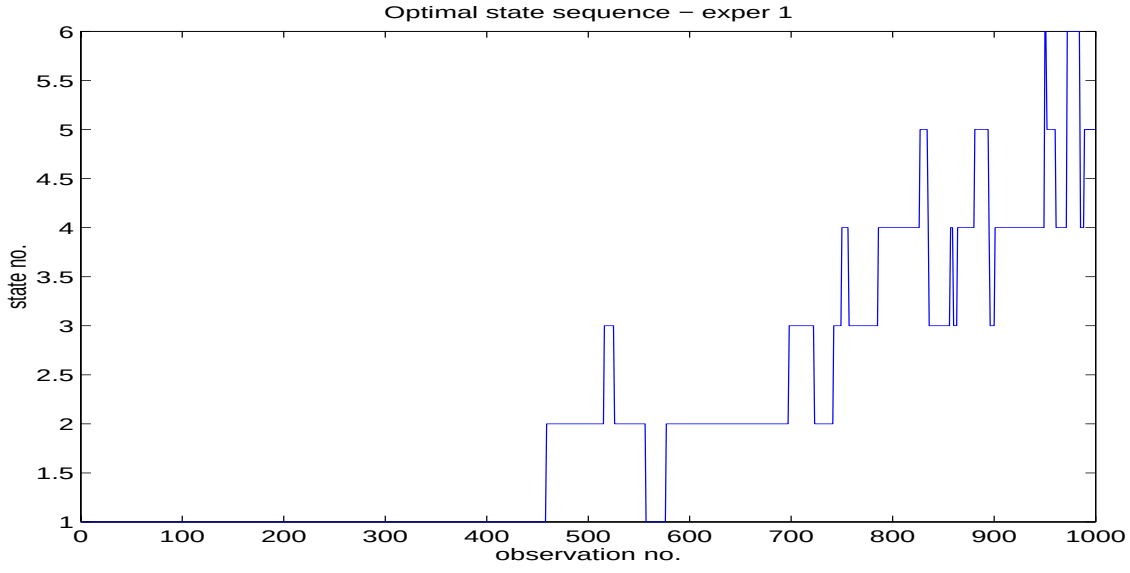


Figure 7.5: Segmentation of time series no. 3 into health states

Table 7.3: Prior settings of A matrix in second experiment using 1 time series

$A =$

	1	2	3	4	5	6
1	0.5	0.4	0.1	0	0	0
2	0.2	0.5	0.3	0	0	0
3	0	0.1	0.4	0.5	0	0
4	0	0	0.1	0.3	0.5	0.1
5	0	0	0	0.2	0.4	0.4
6	0	0	0	0.1	0.1	0.8

HMM-based segmentation of 24 selected time series

We selected the first 6 time series (all measurement quantities that were not related to acoustic emission) for the first 4 sensors. This set was chosen because these time series show fairly comparable degradation behaviour, whereas the acoustic emission related time series are much more deviating. An HMM was initialized as in the second experiment above (where for B the parameters were adjusted since the number of quantization levels was now 20 in stead of 16). The HMM was trained on the ensemble of 24 time series. The learned A matrix is displayed in table 7.4; vector Π equals: $\Pi = [\ 0.8\ 0.2\ 0\ 0\ 0\ 0\ ]$. The B matrix with 20 quantization levels (table 7.5) has been split into two subtables, one with levels 1 to 10 and one with levels 11 to 20.

Table 7.4: Learned A matrix using 24 time series

$A =$

	1	2	3	4	5	6
1	0.98	0.02	0.00	0	0	0
2	0.02	0.96	0.02	0	0	0
3	0.00	0.02	0.97	0.01	0	0
4	0.00	0.00	0.01	0.96	0.02	0.01
5	0	0.00	0.00	0.03	0.95	0.02
6	0	0	0.00	0.00	0.03	0.97

The optimal state sequences for the third and fourth time series in the ensemble of 24 time series are shown in figure 7.6. A drawback of the HMM approach to fault detection now becomes clear: the normalization to maximum values assumes that trends of new fault indicators have the same dynamic range as the trajectories used for training the model. In the figures this is visible as a different moment of transition to a wear state for two different health trajectories, while both time series originate from the same underlying fault mechanism. It appears to be necessary to preceed the HMM modelling with a clustering of time series according to dynamic regime, e.g. the HMM clustering approach in [Smy97], mixtures-of-regressors clustering [GS99] and switching state-space models [Gha97]. Otherwise, an 'averaged' evolution is learned, which leads to alarm states that are invalid for time series with deviating dynamics.

Table 7.5: Learned B matrix using 24 time series

$B =$

	1	2	3	4	5	6	7	8	9	10
1	0.04	0.16	0.79	0.01	0.00	0.00	0.00	0.00	0	0
2	0.01	0.00	0.06	0.91	0.03	0.00	0.00	0.00	0	0
3	0.00	0.00	0.00	0.02	0.62	0.34	0.00	0.00	0.00	0
4	0.00	0.00	0.00	0.02	0.01	0.15	0.78	0.04	0.00	0.00
5	0.00	0.00	0	0	0.00	0.00	0.04	0.50	0.32	0.13
6	0.02	0	0	0	0	0.00	0	0.00	0.01	0.04

11	12	13	14	15	16	17	18	19	20
0.00	0	0	0	0	0	0	0	0	0
0.00	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0.00	0.00	0.00	0.00
0.00	0	0.00	0.00	0	0	0.00	0.00	0.00	0.01
0.00	0.00	0.00	0.00	0	0.00	0.00	0	0.00	0.00
0.20	0.34	0.17	0.08	0.05	0.05	0.01	0.02	0.01	0.02

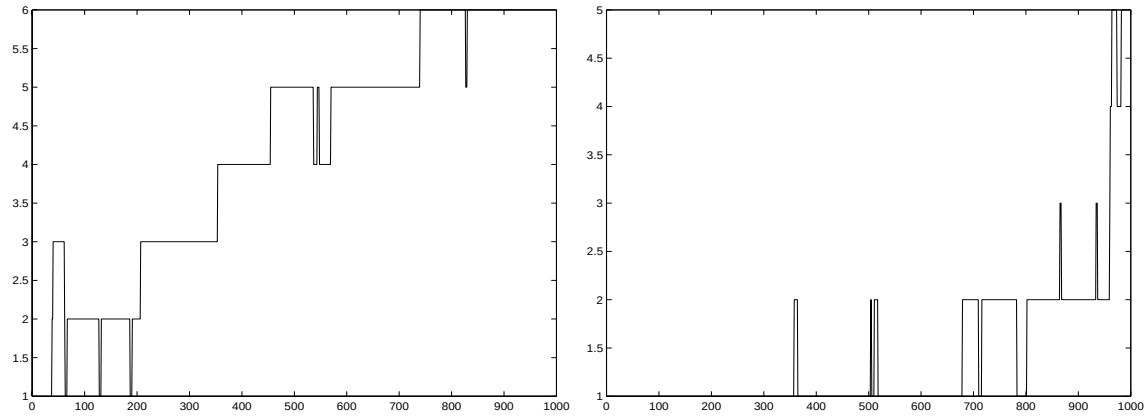


Figure 7.6: Segmentation of 3rd and 4th time series using 24 time series for HMM training

7.4 Discussion

In this chapter we described two approaches to recognition of dynamic patterns: a context-insensitive and a context-sensitive approach. In the former approach, the nonstationary nature of data samples is taken into account (a) by making a recognition algorithm adaptive to new data samples or (b) by choosing algorithms that allow for tracking of changes in the data characteristics. If changing data characteristics are indicative of changing health of a system, such a feature space trajectory may be used for health monitoring. We noticed that *Self-Organizing Maps* have the ability for system tracking. This is illustrated with measurements from a submersible pump that was gradually loosened from its foundation in section 8.5.

The *HMM approach to modelling of fault development* will be meaningful (a) if *durations*

in a certain 'wear' state have a meaning, (b) if there is a sufficient number of equidistantly sampled time series available and (c) if an HMM is trained with time series corresponding to the same dynamic regime. Moreover, since fault development is considered a random growth process [NC97], a segmentation according to the magnitude of the health indicator seems likely to occur. Hence, we expect that HMMs are mainly useful for segmentation of health patterns in cases where a *self-restoring* takes place (a temporary decrease in fault severity, e.g. when small cracks are smoothed by continued motion of rotating elements). In these cases a straighforward thresholding may lead to an alarm, where in fact just a 'warning' state has been reached.

Chapter 8

Health monitoring in practice

In chapter 1 we introduced a framework for machine health monitoring with learning methods. In this chapter we will describe four real-world monitoring problems that involve subproblems that were mentioned in this framework. We will show how the learning methods that have been investigated in this thesis can be used beneficially for practical health monitoring. We start by investigating the *feasibility of submersible pump monitoring* when only one sensor can be used and only a limited number of calibration measurements can be done (since cost-effectiveness is an important issue for commercial application). We will use several conventional classifiers (like feedforward neural networks) in these experiments and focus on the (two-class) fault detection problem. Moreover, two different feature extraction methods are used. Then we describe a *system for gas leak detection* in underwater pipelines based on a combination of supervised and unsupervised learning techniques. A novelty detection 'module' is a necessary extension to a conventional classifier to increase the robustness of the system to unexpected events. *Medical health monitoring* exhibits many similarities to the machine monitoring problem. We then address two medical case studies, where novelty detection (detection of eyeblinks), source separation and classification (detection of Alzheimer's disease) allow for promising detection results. We address two *gearbox monitoring cases* in a real-world setting. In both cases, the transmission system of an operational pump in a pumping station is monitored using vibration measurements on the machine casing. Self-Organizing Maps of preprocessed measurements can be made that show a clustering of measurements that is in accordance with the user's expectations. We conclude this chapter with the description of *MONISOM, a SOM-based system for machine health monitoring* that has been developed in conjunction with two industrial companies. We illustrate the use of Self-Organizing Maps for system tracking by applying MONISOM to a set of measurements from a slowly degrading pump.

8.1 Fault detection in a submersible pump

Practical application of the methods proposed in this thesis to monitoring of a small submersible pump requires that the cost of the monitoring method should be low. Hence we studied to what extent detection of faults in this pump is possible using only one sensor.

Moreover, the usefulness of a trained recognizer will depend on the extent to which repeated measurements are similar to measurements that are used in the training procedure.

8.1.1 Pump monitoring with one sensor

In [SYD00], it was shown that small error percentages are possible for both fault detection and diagnosis, even if a (particular) selection of the available operating modes are included in the training set. More specifically, loads can be interpolated by using noise injection techniques; however, variability due to speed changes appeared much harder to be simulated. This means that all expected running speeds of the submersible pump should be present in a proper calibration procedure (where the normal machine behaviour is determined). We included measurements from all possible operating modes in the datasets, i.e. the machine driving frequency extended from 46 to 54 Hz (in steps of 2 Hz) and the machine loads extended from 2.5 and 2.9 KW to 3.3 KW. The sensors were mounted as in figure 3.1(a). The shaker that is visible in the figure was not used in this case. In the experiments, we used 5 measurement channels: 3 from the triaxial sensor C and one from each of the sensors A and B. The correspondence between channel numbers and the above figure is: B: radial; Cx: radial & parallel; Cy: axial; Cz: radial & perpendicular; A: axial, on the vane. Here, perpendicular stands for: the direction “into the heart of the machine”, and parallel stands for: the direction across the machine. The following classifiers (using the PRTOOLS toolbox for Matlab [Dui00]) were used in the experiments: normal-densities based quadratic discriminant classifier (QDC), nearest-mean classifier (NMC), feedforward artificial neural network with 5 hidden units (ANN-5), feedforward ANN with 10 hidden units (ANN-10), 1-nearest neighbour classifier (1-NN), 5-nearest neighbour classifier (5-NN), linear support vector classifier (SV-1), quadratic support vector classifier (SV-2). All experiments are repeated 3 times. In each repetition the total dataset is split randomly into a training and a test set. We report the mean test errors for each classifier in the tables below.

Experimental setup

Each measurement segment of 4096 samples is used for the computation of one feature vector. Prior to feature extraction, the mean of each segment is subtracted. We compute two features: power spectrum (dataset 1) and autoregressive model coefficients (dataset 2). Each power spectrum is normalised to unit power. The AR parameters are not normalised afterwards. The resulting datasets are reduced in dimensionality using PCA. The dimensionalities investigated are: 2, 5, 10 and 16. In the collection of datasets, we included all faults induced to the machine. The fault induced were: imbalance and a bearing failure. the bearing failure was both small (inner race ditch of approximately 1.0 mm) and (relatively) large (1.5 mm ditch in inner race). All fault patterns were labeled as ’fault class’, whereas the measurements from normal machine vibration were labeled as ’normal class’. Hence, we address a two-class problem.

Detection results

In *dataset 1*, each measurement segment is represented by a power spectrum with 256 bins. Each spectrum is scaled to zero mean and unit power. In the dataset there are 225 samples of the normal class and 539 samples of the fault class. The mean test error over three repetitions of each experiment is shown in table 8.1. The dimensionality of each dataset is varied (vertical axis in the table), and different classifiers are trained on each dataset (the horizontal axis). In *dataset 2*, each segment was represented by the coefficients of an autoregressive model

Table 8.1: Results of pump fault detection with one sensor, using spectral features

	dimension	QDC	NMC	ANN-5	ANN-10	1-NN	5-NN	SV-1	SV-2
channel 1	16	0.15	0.25	0.09	0.09	0.19	0.17	0.13	0.11
	10	0.20	0.25	0.15	0.11	0.17	0.17	0.16	0.11
	5	0.23	0.25	0.19	0.15	0.18	0.20	0.25	0.15
	2	0.30	0.43	0.26	0.25	0.25	0.25	0.29	0.29
channel 2	16	0.06	0.22	0.01	0.01	0.05	0.09	0.01	0.01
	10	0.06	0.23	0.01	0.01	0.05	0.08	0.06	0.01
	5	0.30	0.28	0.13	0.08	0.13	0.17	0.28	0.13
	2	0.35	0.31	0.27	0.22	0.23	0.23	0.31	0.29
channel 3	16	0.01	0.13	0.02	0.02	0.05	0.01	0.02	0.02
	10	0.05	0.14	0.04	0.03	0.05	0.02	0.05	0.05
	5	0.11	0.14	0.05	0.05	0.07	0.06	0.08	0.06
	2	0.18	0.20	0.13	0.12	0.15	0.17	0.15	0.13
channel 4	16	0.05	0.28	0.02	0.02	0.10	0.09	0.03	0.01
	10	0.19	0.28	0.07	0.05	0.12	0.13	0.16	0.06
	5	0.21	0.30	0.08	0.08	0.10	0.14	0.21	0.08
	2	0.34	0.42	0.26	0.21	0.22	0.23	0.30	0.30
channel 5	16	0.27	0.30	0.30	0.29	0.29	0.29	0.25	0.31
	10	0.29	0.29	0.32	0.31	0.30	0.31	0.27	0.30
	5	0.30	0.29	0.29	0.29	0.30	0.30	0.28	0.28
	2	0.29	0.30	0.27	0.27	0.30	0.29	0.28	0.27

of order 16, which is roughly comparable to the model orders that are used in speech recognition. The number of normal and fault samples is the same as in dataset 1. The detection results are shown in table 8.2.

Analysis of results

From the classification results it becomes clear that it is well possible to perform fault detection on the Landustrie submersible pump with one sensor, if the sensor is mounted on the position that is denoted as 'C' in figure 3.1(a). The radial measurement directions allow for the best fault detection results. However, axial measurements done on this central position

Table 8.2: Results of pump fault detection with one sensor, using autoregressive features

	dimension	QDC	NMC	ANN-5	ANN-10	1-NN	5-NN	SV-1	SV-2
channel 1	16	0.05	0.29	0.02	0.01	0.08	0.06	0.03	0.03
	10	0.08	0.29	0.01	0.01	0.08	0.06	0.03	0.03
	5	0.10	0.29	0.03	0.02	0.07	0.06	0.04	0.03
	2	0.27	0.29	0.17	0.13	0.18	0.18	0.24	0.22
channel 2	16	0	0.35	0.00	0.00	0.05	0.02	0.15	0.09
	10	0.01	0.35	0.01	0.01	0.05	0.03	0.16	0.09
	5	0.12	0.35	0.03	0.03	0.06	0.04	0.22	0.21
	2	0.27	0.38	0.24	0.17	0.19	0.17	0.28	0.30
channel 3	16	0	0.20	0.01	0.00	0.01	0.00	0.02	0.01
	10	0.01	0.20	0.00	0.01	0.01	0.00	0.02	0.01
	5	0.03	0.20	0.02	0.01	0.03	0.02	0.03	0.03
	2	0.19	0.25	0.10	0.06	0.08	0.08	0.16	0.15
channel 4	16	0	0.29	0.00	0	0	0	0	0
	10	0	0.29	0.00	0	0	0	0	0
	5	0	0.29	0.00	0	0	0	0	0
	2	0.10	0.29	0.05	0.03	0.04	0.03	0.30	0.06
channel 5	16	0.20	0.39	0.17	0.19	0.28	0.23	0.29	0.29
	10	0.27	0.40	0.21	0.23	0.27	0.24	0.29	0.29
	5	0.34	0.40	0.26	0.26	0.29	0.28	0.29	0.29
	2	0.37	0.40	0.28	0.26	0.32	0.29	0.29	0.29

on the machine (channel 3) still allow for nearly perfect fault detection (if dimensionalities and classifiers are chosen suitably). This choice of dimensionality and classifier is most important for the spectrum features; if we use AR-features to represent the measurements, error rates below 1 % can be obtained, regardless the classifier (except for the nearest-mean classifier, which is almost always the worst) for small dimensionalities. The best performance for dataset 1 can be obtained by using sensor channel 2 or 3. We show a scatter plot of the first two principal components of the dataset obtained with sensor channel 2 in figure 8.1(a). Although the classes may be overlapping in two dimensions, the data is (almost) separable in a higher-dimensional space. The best results for dataset 2 can be obtained by using either sensor channel 2 or 4. The scatter plot for sensor channel 4 is shown in figure 8.1(b). Again, a number of subclusters can be observed in the data, probably due to the inclusion of different operating modes in the data. The normal class measurements can be separated well from the faults, although we may need more than two dimensions in the feature vectors to achieve this. Concluding, a 10-hidden units feedforward neural network seems a good choice for fault detection using only one sensor channel, provided this channel is measured near the defect bearings (i.e. near position 'C'). The measurement direction is not very important. However, when measuring more remote on the machine (positions 'A' and 'B') it is more beneficial to

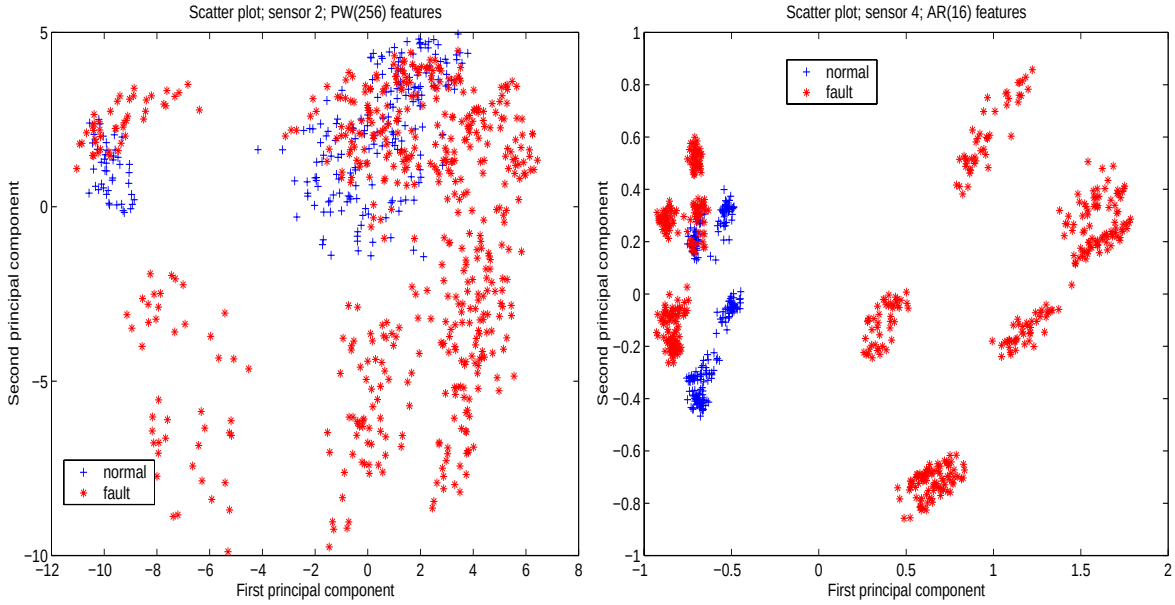


Figure 8.1: Scatter plots Landustrie 1-sensor data. a: PW(256)-sensor 2; b. AR(16)-sensor 4

mount on the same substructure as the origin of the bearing fault (machine casing in stead of the vane).

Data visualization with Self-Organizing maps

The data can be visualized with a Self-Organizing Map as well; now, the mapping is nonlinear and may better preserve the underlying structure in the dataset. A 30×30 SOM was fitted to the autoregressive datasets using the SOM-toolbox from Helsinki University of Technology [AHPV99]. The resulting U-matrices (section 7.2.2) for measurement channel 4 is shown in figure 8.2. In the figure, we used a majority voting rule to calibrate the map. The map calibration procedure was described in section 7.2.2. The original fault labels are used: 1 = normal, 2 = small bearing failure, 3 = large bearing failure and 4 = imbalance. From the U-matrix visualizations of the learned maps we see that bad choice of measurement channel (e.g. channel 5, not shown) leads to overlapping areas for normal behaviour and imbalance (labels 1 and 4). Moreover, bearing failure data is usually quite dissimilar from other data. When choosing channel 2 (not shown), the spread due to operating mode leads to health related areas that are distributed over the map. Such a map is still useful for monitoring, but extensive calibration and analysis of the map is now needed. Finally, channel 4 allows for disjunct health-related map areas. A wrap-around effect (two separate border areas, both corresponding to admissible system behaviour) may be detected by proper map calibration of clustering of map prototypes.

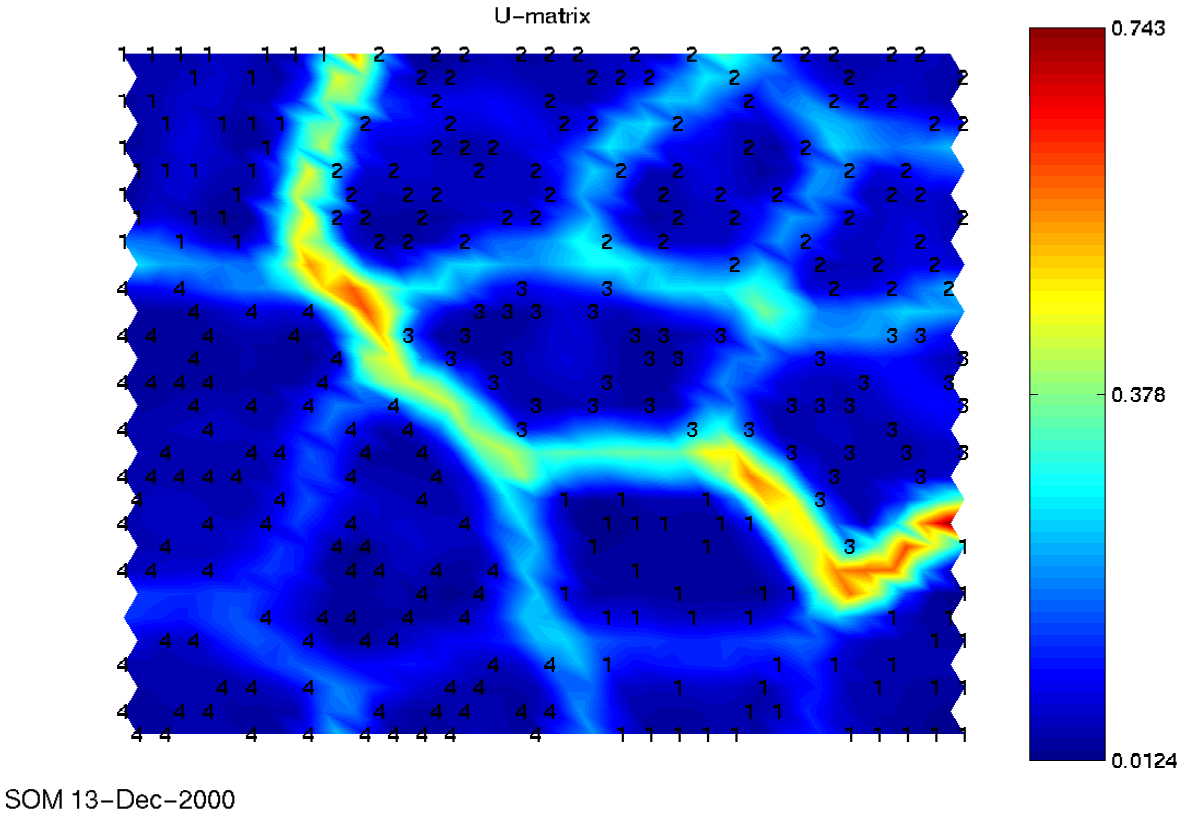


Figure 8.2: U-matrix visualization of 30×30 SOM applied to AR datasets, channel 4

8.1.2 Robustness to repeated measurements

In the next experiment we studied the robustness of a trained classifier to repeated measurements on the same machine.

Measurement setup

Three measurement sessions were performed: in each session, the pump was monitored in a range of operating modes i.e. the machine driving frequency extended from 46 to 54 Hz (in steps of 4 Hz) and the machine load had values 2.9 and 3.3 KW. The pump was measured in normal condition and with a loose foundation, an imbalance and a bearing failure (1 mm ditch in the outer ring of the uppermost bearing). After each measurement session, the pump was lifted from the basin and then again put in place. Then the next measurement session was performed. The sensors were mounted differently than in the previous experiment: **channel 1**: axial (Z direction), near lower bearing; **channel 2**: radial (X), near upper bearing; **channel 3**: axial (Z), upper; **channel 4**: radial (Y), upper; **channel 5**: radial (Y'), lower; Here, the measurement direction Y' is inbetween directions X and Y.

Experimental setup

From this set of measurements, each segment of 4096 measurement samples from each sensor channel was used as the basis for a feature vector. Prior to feature extraction, the mean of the segment was subtracted from each segment. We constructed three different datasets using different feature extraction methods: **set I**: power spectrum, 16 bins, each spectrum is normalized to have unit energy; **set II:** AR-model coefficients, model order was chosen to be 32; **set III:** classical features: RMS value of the spectrum, crest factor of each segment and (normalized) kurtosis of each segment.

Each dataset consisted of three subsets: one subset for the first measurement session, one for the second and one for the third session. Each feature vector was labeled according to its health state (OK, imbalance, loose foundation and bearing failure). Then we transformed the problem into a 2-class classification problem by labeling feature vectors from all three faults as one (fault) class. We included 15 consecutive segments of 4096 samples per situation (health state, operating mode, mounting session), leading to subsets consisting of 360 samples. For each dataset (I, II and III), we trained two different classifiers (a 1-nearest neighbour classifier **1-nn** and a 10-hidden units neural network **10-ann**) on the subset corresponding to the first mounting. More specifically, this subset (called the learning set) was projected with PCA to a lower-dimensional subspace, for several feature dimensionalities. The resulting projected learning set was randomly split into a training and a test set (with 180 samples each). Both subsets corresponding to the second and third mounting (called evaluation set, denoted as "ev-2" and "ev-3" respectively) had 360 samples each and were also projected onto the same subspace. Then the classifiers were learned on the training set and evaluated on the training set, the test set and both evaluation sets. This process was repeated 10 times, and the mean error along with the standard deviation was determined. In the figures below we will only present the results for two choices of measurement channels (other channels showed roughly the same behaviour as these two channels). Moreover, the results on dataset III were always very unsatisfactory. We decided to ignore the results for this dataset in the sequel as well.

Dataset I

The results for dataset I are shown in figure 8.3. It can be observed that evaluation set "ev-2" (corresponding to the second repeated measurement session) can be separated fairly well using a classifier trained on session 1 samples. However, for certain choices of classifier and measurement channel the results for classification of session 3 measurements may decrease significantly (compared to the classification results using session 1) with a few percentage points. Especially with higher-dimensional feature vectors (e.g. the 16-D case), the neural networks become somewhat overtrained; this shows up as decreasing generalization performance to measurements from sessions 2 and 3. We may try to increase robustness of the method by adding a small amount of noise to the training set. In [SYD00] it was shown that noise injection allowed for extrapolation of a trained classifier to (previously unseen) samples representing a different machine load. We constructed a learning set with samples from session 1. Then we added white Gaussian noise (zero mean, 0.001 standard-deviation) to the

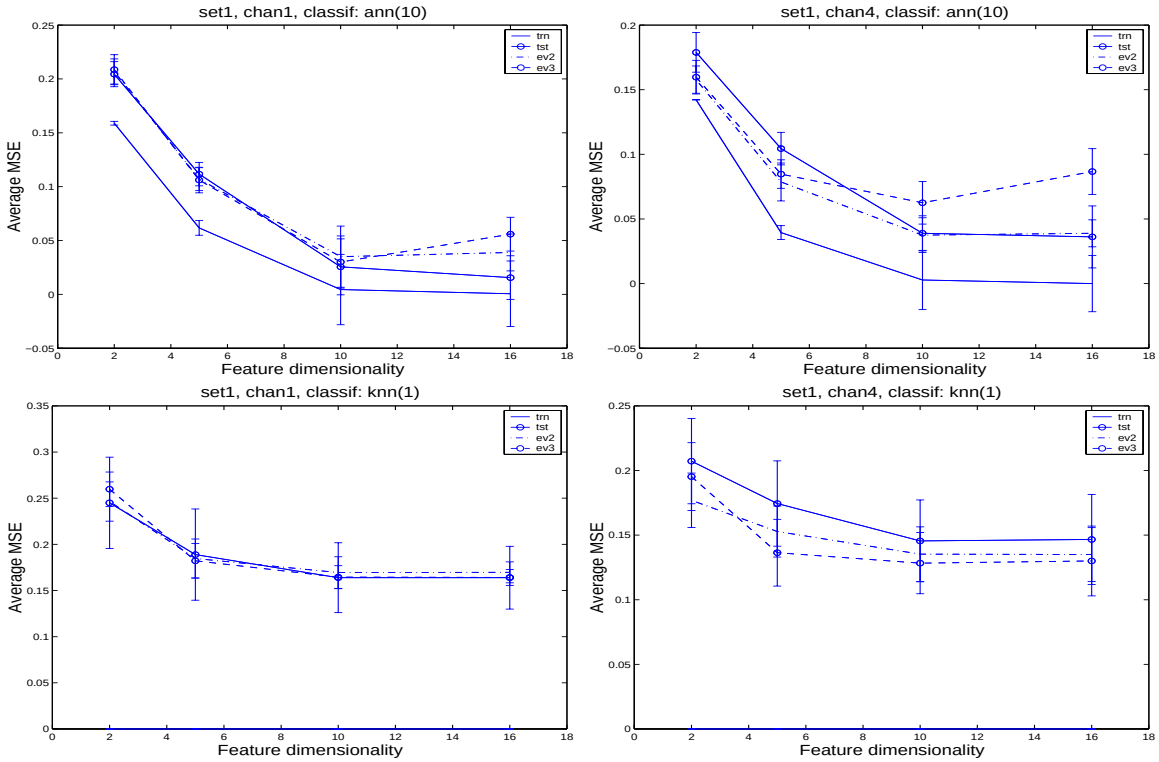


Figure 8.3: Mean errors (+ std) on dataset 1, trained on session 1, evaluated on sessions 1, 2, 3; a. **10-ann**, channel 1, b. **10-ann**, channel 4, c. **1-knn**, channel 1, d. **1-knn**, channel 4

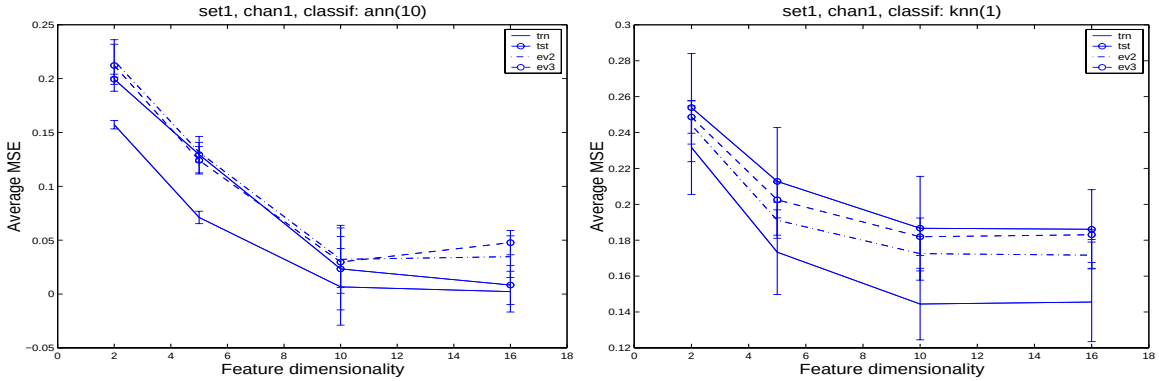


Figure 8.4: Mean errors (+ std) on dataset 1, channel 1; a classifier is trained on noise-enriched samples of session 1 and evaluated on (original) sessions 1, 2, 3; a. **10-ann**, b. **1-knn**

training set (i.e. in the PCA-subspace of the learning set for each choice of feature dimensionality) and trained two different classifiers on the noise-enriched dataset. Testing was done with the part of the learning set that was not used for training (and was not enriched). Evaluation was done with samples from session 2 (“ev-2”) and session 3 (“ev-3”). In figure 8.4 the results are plotted for the first channel of the first dataset. From these figures (and from other

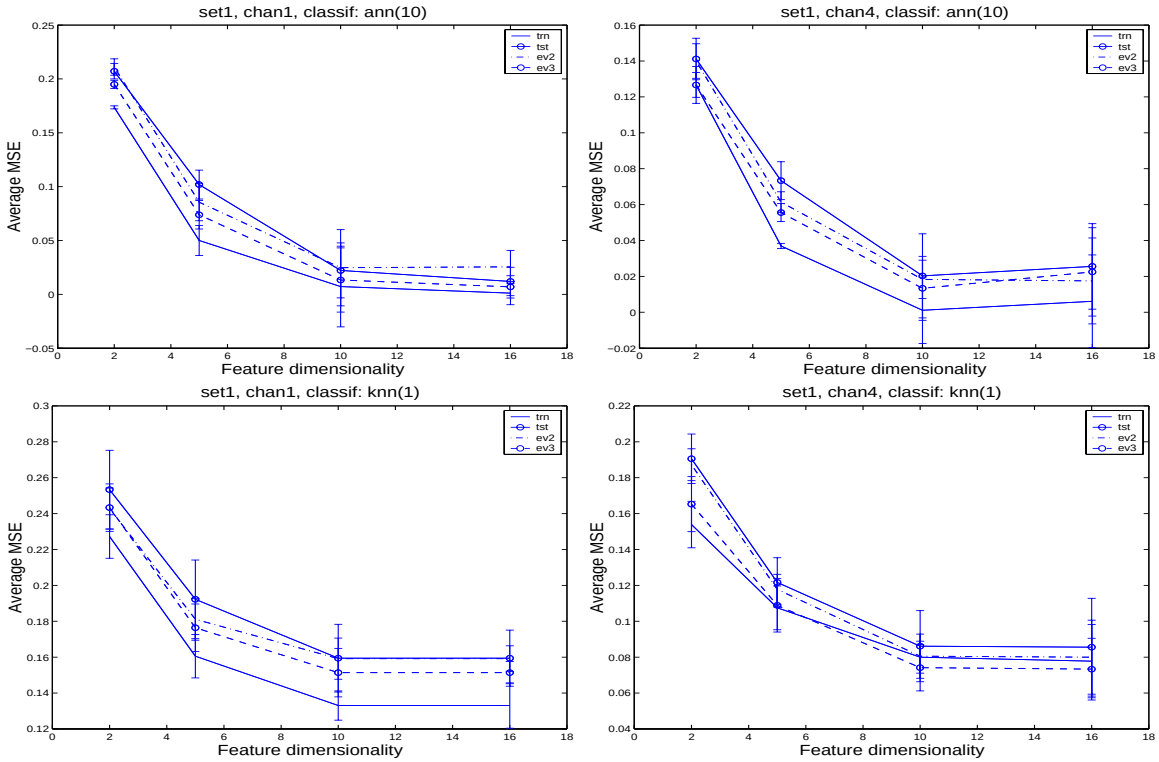


Figure 8.5: Mean errors (+ std) on dataset 1; the classifier is trained on session 1+3 and evaluated on sessions 1, 2, 3; a. **10-ann**, channel 1, b. **10-ann**, channel 4, c. **1-knn**, channel 1, d. **1-knn**, channel 4

results that are not shown here) we see that the effect of noise injection on the robustness is very small for dataset I. We can however reduce the error on an evaluation set by including samples from two repetitions in the learning set and evaluating on the third set. In figure 8.5 the results are plotted for two different classifiers and two (representative) channels, numbers 1 and 4. By comparing this figure to figure 8.3 we can observe that the variability in the results for test and evaluation sets (for the 16-D case) decreases (a,b) or remains relatively small (c,d). This indicates that the neural networks become less overtrained in the 16-D case. The 1-NN results are still unsatisfactory, but showed significant improvement.

Dataset II

The previous experiments were repeated for dataset II. The results for the case where the classifiers were trained on session 1 and tested on all sessions are shown (for channels 1 and 4) in figure 8.6. Again it can be observed that the patterns from session 3 are dissimilar from the session 1 patterns (depending on dimensionality, classifier and channel), leading to decreased performances up to 5 %. Enrichment of the training set with noise led to virtually identical results in the case of **10-ann** with channel 1 and 4 data (not shown). However, for the 1-nearest neighbour classifier, an improvement could be obtained (figure 8.7). Generalization to an unseen measurement session is possible for channel 1. With channel 4, evaluation on

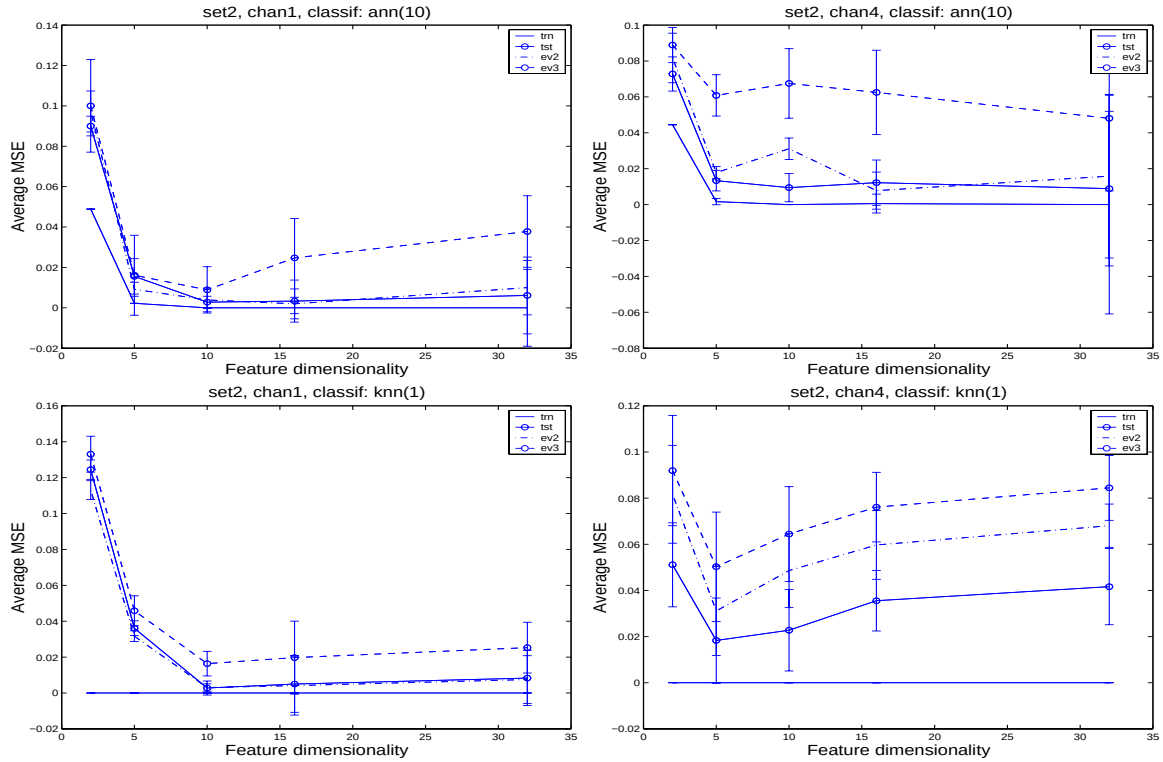


Figure 8.6: Mean errors (+ std) on dataset 2, trained on session 1, evaluated on sessions 1, 2, 3; a. **10-ann**, channel 1, b. **10-ann**, channel 4, c. **1-knn**, channel 1, d. **1-knn**, channel 4

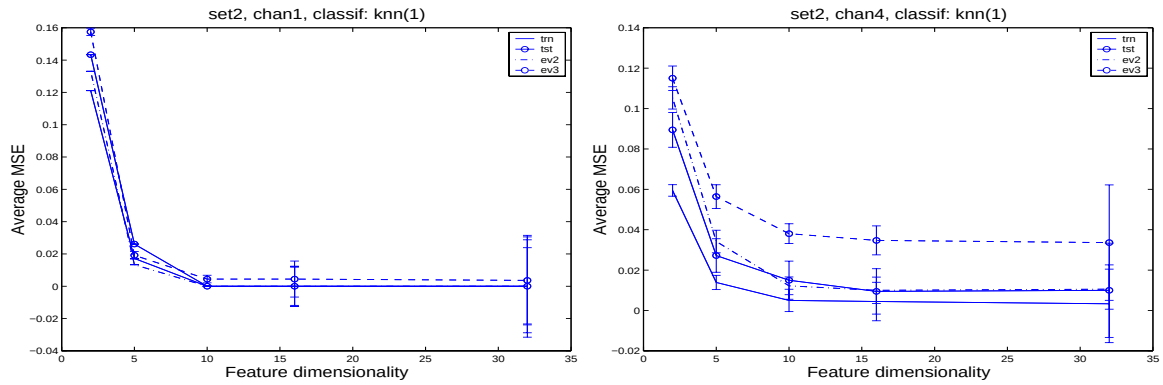


Figure 8.7: Mean errors (+ std) on dataset 2, trained on noise-enriched session 1 and evaluated on (original) sessions 1,2,3: a. **1-knn**, channel 1, b. **1-knn**, channel 4

session 3 involves an additional error of about 3 %. Training on patterns from sessions 1 and 3 leads to very satisfactory results for both channels, see figure 8.8. The overtraining is visible as a high variance only (in the outcomes for larger dimensionalities). However, on average a classifier is obtained that generalizes well to both an independent test set and an evaluation set of measurements from an unseen measurement session. On the whole, we

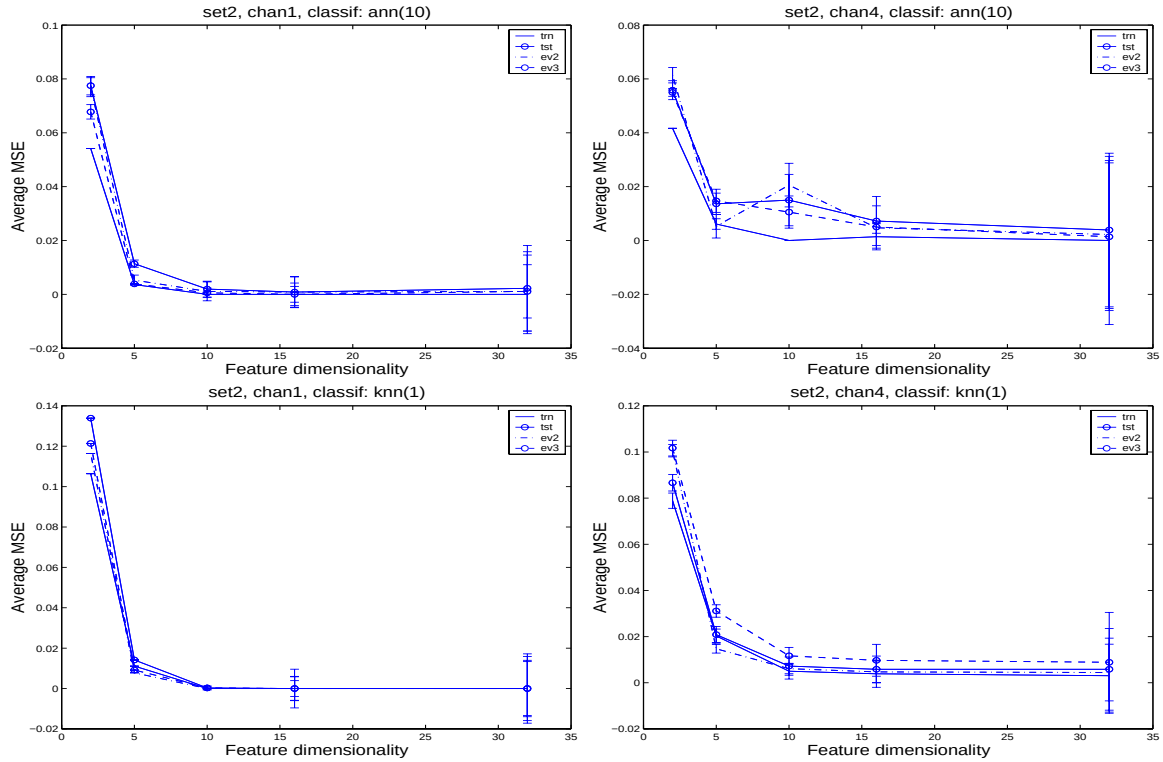


Figure 8.8: Mean errors (+ std) on dataset 2, training on session 1+3, evaluation on sessions 1,2,3; a. **10-ann**, channel 1, b. **10-ann**, channel 4, c. **1-knn**, channel 1, d. **1-knn**, channel 4

see that there is some variability in the patterns due to a different measurement setup. This variability causes evaluation errors that are usually rather small (maximally 5 % larger than the error on the test set; for some choices of classifier, feature dimensionality and channel the difference is almost zero percent). In cases with larger inter-measurement variance, a remedy may be to add a small amount of noise to the training set. However, the effect on the overall performance of the classifier should be monitored closely. Finally, including samples from a number of repeated measurement sessions into the learning set appears to be the most effective remedy to the variability problem.

8.2 Gas pipeline monitoring

The system described in this section is aimed at detection of leaks in underwater gas pipelines. The measurement setup has been described in section 1.5.3. Previous research indicated the feasibility of neural networks for this problem [Kir96]. We will only give an outline of the monitoring method and the detection results here. The system was developed in conjunction with Tax and Duin for Shell Expro UK. A more extensive treatment has been written down in [TYD98].

8.2.1 Fault detection system

The data is measured during 21 blowdowns, each with different settings. Settings differ in gas pressure in the gas cylinder, depth of the gas leak (that is, the depth of the nozzle) and the distance to the hydrophone. All original recordings were measured at a sampling frequency of 100 kHz. From each blowdown 3 samples of noise signals ('noise vectors') are extracted and 10 samples of leak signals ('leak vectors') at different stages of the blowdown. The noise samples are taken before instead of after the actual blowdown occurred, to prevent that residuals of leak signals disturb the measurement. Each of the selected time samples consists of 2000 data points. From each measurement segment, a power spectrum was obtained (with a resolution of 128 bins) and an automatic feature selection procedure was applied to the resulting dataset. A subset of best features was selected for inclusion into the final dataset.

To classify the final spectra, two types of networks are used. The first type is a feedforward neural network. A well-trained neural network can discriminate between two classes, but is less suited to detect if the input can be reliably classified. For that purpose, a SOM is used. All incoming data samples are first passed through this novelty detector. Dissimilar data (like data samples representing noise from ships passing by) may now be rejected (chapter 6), which decreases the false alarm rate. The feedforward neural network (ANN) is then used to distinghuish between leak and noise. The method is illustrated in figure 8.9. Two different situations are now distinguished, depending on the fact if feature values in the

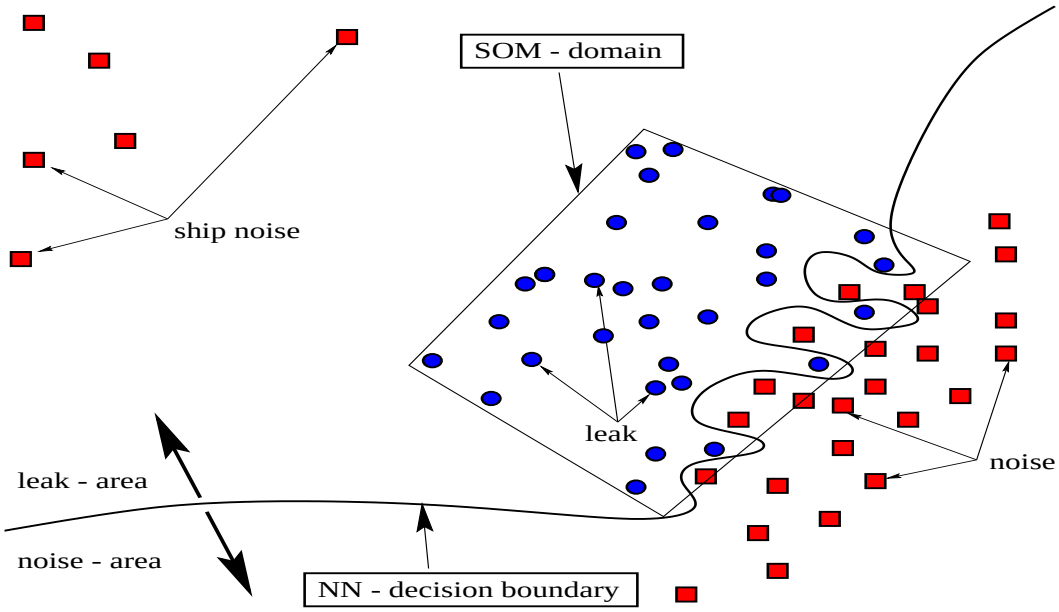


Figure 8.9: Hybrid supervised/unsupervised learning system for pipeline monitoring

preprocessed spectrum remain positive after removing the representative background noise spectrum (see also the experiments in chapter 6):

- all feature values are larger than zero. This may indicate a sudden change in the environment. In this case we can assume that the total detected spectrum is a sum of the

normal background noise and some extra sounds. These extra sounds may be caused by the presence of a leak in a pipeline, but can also be caused by e.g. a ship sailing along. By removing the normal background noise from the measured spectrum, it is expected that a less polluted leak signal is obtained that can more easily be distinguished from e.g. ship noise

- some feature values are smaller than zero. In this case a subtle change in background noise occurs or a very small leak is developing. Now it is not possible to remove the representative background noise: the more contaminated signal has to be used and classified.

We first determine the suitable regime: the sudden change regime (all features > 0) or the slowly developing regime (some features < 0). For each of the regimes, a dedicated SOM and ANN is trained. To minimize the number of false alarms, we use a combination of SOM and ANN for the final detection: only when both SOM and ANN agree that there is a leak present (the SOM indicates that the signal resembles the leak-map and the neural network detects a leak), an alarm is raised. When in the second situation (also containing difficult leak signals) both SOM and ANN agree that the detected signal is noise (the SOM detects a novelty, the signal does not resemble the leak-map; the network detects noise), this noise will be used in the representative background noise set.

8.2.2 Results

The objective of zero false alarms can be met for the sudden change regime by choosing a proper classification threshold: measurements are corrected for local noise (if possible), removing a lot of ambiguity. Moreover, only 5 % of all leaks are misclassified. For situations with gradual change the problem becomes more difficult. However, choosing a proper threshold yields only two false alarms (that originated from the same measurement run). In this case, some 10 % of the available leaks are misclassified. These difficult leak samples mainly originate from measurements of very remote leaks and from samples that are measured at the end of a blowdown procedure. Furthermore, remoteness of leaks is simulated by artificially attenuating prominent leak measurements. For blowdowns with fairly large nozzle size and small distance between hydrophone and nozzle, an attenuation level of about 15 dB is possible (still enabling correct discrimination between leak and noise). When the distance becomes larger, the admissible attenuation level is in the order of 10 dB. The feasibility of novelty detection with Self-Organizing Maps was demonstrated (section 6.4.3) using the data gathered at the “IJ” harbour. Using the Norway data, it was shown (section 6.4.3) that choosing a suitable threshold enables discrimination between leak and most of the noise, under the sudden change regime. A few leak-like noises cannot be separated from leaks, but this is not conflicting with the aim of SOM-modelling. Under the gradual change regime the same applies: remote leaks (up to 10 dB attenuation) can be discriminated from noise using a suitable rejection threshold. As postprocessing one can smooth the classification output by averaging over several consecutive classifier outputs. A sudden leak detection that is isolated in time (i.e. followed by non-leak decisions immediately afterwards) is suppressed in this manner.

Hence, the temporal structure of the problem is taken into account as a postprocessing to a context-insensitive monitoring method. This may serve as an alternative to the HMM-based temporal smoothing presented in chapter 7.

8.3 Medical health monitoring

We think that our framework is suitable for several medical monitoring problems as well and illustrate this in two particular cases: automatic quantification of Tourette's syndrom [YBJMD01] and automatic detection of Alzheimer's disease [MYFS02]. The former work was performed in conjunction with mr. Baunbæk-Jensen from TU Denmark. Measurements were collected by dr. Groeneveld from the Medical department of Erasmus University in Rotterdam. The latter work was performed in conjunction with mr. Melissant [Mel00b] and dr. Frietman from TU Delft. Measurements were collected by dr. Stam from the Medical department of the Free University of Amsterdam.

8.3.1 Detection of eyeblinks from Tourette's syndrom patients

An important indicator of the severity of Tourette's syndrom[1] is the eyeblink activity of the subjects under examination. Eyeblinks and -movements can be registered automatically using ElectroOculoGraphy (EOG). An eyeblink detection system should be able to discriminate between eyeblink-events and all other events. Since the eyeblinks of a subject are assumed to be fairly repetitive, a *novelty detection* approach is taken. A Self-Organizing Map was used to describe the set of eyeblinks. Note that the 'normal set' that is usually modelled in novelty detection should now be interpreted as: 'reproducible set', analogous to the leak detection problem. An eyeblink in the EOG signal consists of a negative peak, followed by a positive peak, figure 8.11. The negative peak is fairly consistent in shape and duration. The positive peak shows more variability and can be characterized by its peak size. A fixed-length window is slided over the 1-D time series in order to obtain segments that are processed with the SOMs.

We used two different SOMs in the detection method: one SOM (NPSOM) detects the negative peak using examples of negative peaks only; the other SOM (FVSOM) uses information about negative and positive peak size (i.e. it is trained on feature vectors describing the signal segments) to take also the positive peak into account, see figure 8.10(a). The SOM describes the domain of the set of eyeblinks in the feature space. If it is used for novelty detection, a rejection threshold has to be set. We do this by cross-validation using a training and a testset, see figure 8.10(b). The raw data is segmented manually by the medical expert. When a sufficient number of eyeblinks has been segmented (say 200), they are devided into a training set and a test set, and furthermore a set of events that are not eyeblinks is also segmented from the signal. The negative peaks are segmented out and the features are extracted for each blink in the "true-eyeblink class". The NPSOM is then trained with negative peak data, and the FVSOM is trained with the feature data. After training, the detection system

[1] Patients suffer from involuntary muscle contractions and verbal activity

from figure 8.10(a) is then applied to the training set and a variety of threshold combinations for both SOMs are evaluated on the test set, in order to find the optimal combination.

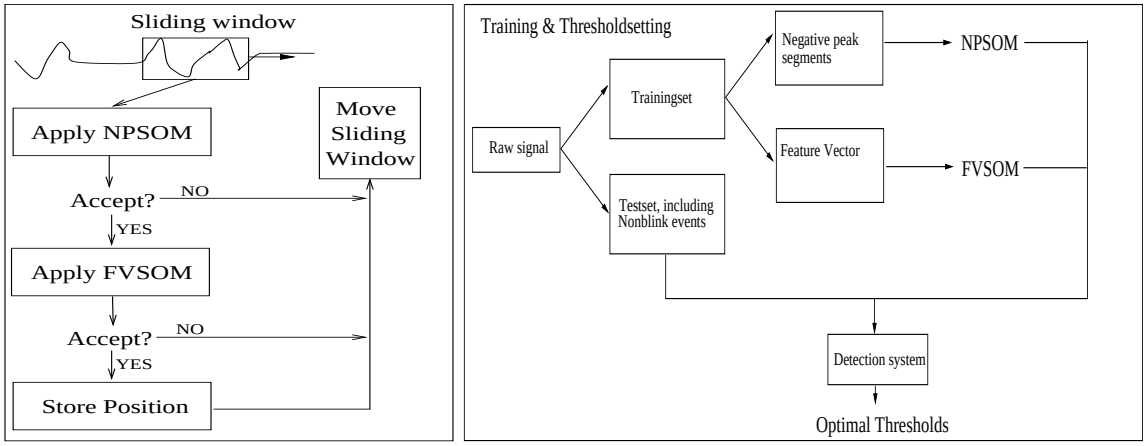


Figure 8.10: Eyeblink detection system (a) and setting of thresholds (b)

Around 200 blinks were segmented from the protocol measurements[2] from *subject 5*. 150 blinks are used for training and 50 blinks are used for threshold setting; 45 non-blink events were segmented as well. The detection system has several adjustable parameters: NPSOM map size, FVSOM map size, size of negative peak window and thresholds for both maps. Experiments suggested that the size of the negative peak used in the NPSOM does not have great influence on the detection results. The eyeblink detection rate for map sizes NPSOM = 8x8, FVSOM = 3x3 was 94% at a *false detection rate*[3] of 7%. Modifying parameters allowed for a slight improvement. The detection result on a unseen segment of non-protocol measurements from subject 5 using previously mentioned parameters is shown in figure 8.11(a). For the protocol-measurements of a second subject (*subject 6*), we used a negative peak window size of 20, NPSOM size of 8x8 and FVSOM size 6x6. We varied the thresholds for NPSOM and FVSOM and inspected the results. Setting the thresholds to 1.0 and 0.09 respectively, allowed for 90% detection rate at the expense of 11% false alarms. The detection result of the system on an unseen set of non-protocol measurements from subject 6 using the above set of parameters is shown in figure 8.11(b). Comparing the labeling of events by a medical expert with our automatic detection we achieved a performance of 97% correct detection at the expense of 3% false alarms. In the measurements at hand, there are almost exclusively blink-events present, which can explain these good results. The overall detection results for both subjects are comparable to the results in [KW98].

8.3.2 Early detection of Alzheimer's disease using EEG

Development of Alzheimer's disease can be detected from ElectroEncephaloGraphic (EEG) measurements of a subject. Automatic analysis of EEG measurements is a nontrivial prob-

[2]During protocol measurements the subject was asked to perform tasks like eye blinking, -closing and -moving. Non-protocol measurements refer to involuntary activity during conversation and movie watching

[3]Ratio of number of detected events/samples in non-blink set to total number of samples in non-blink set

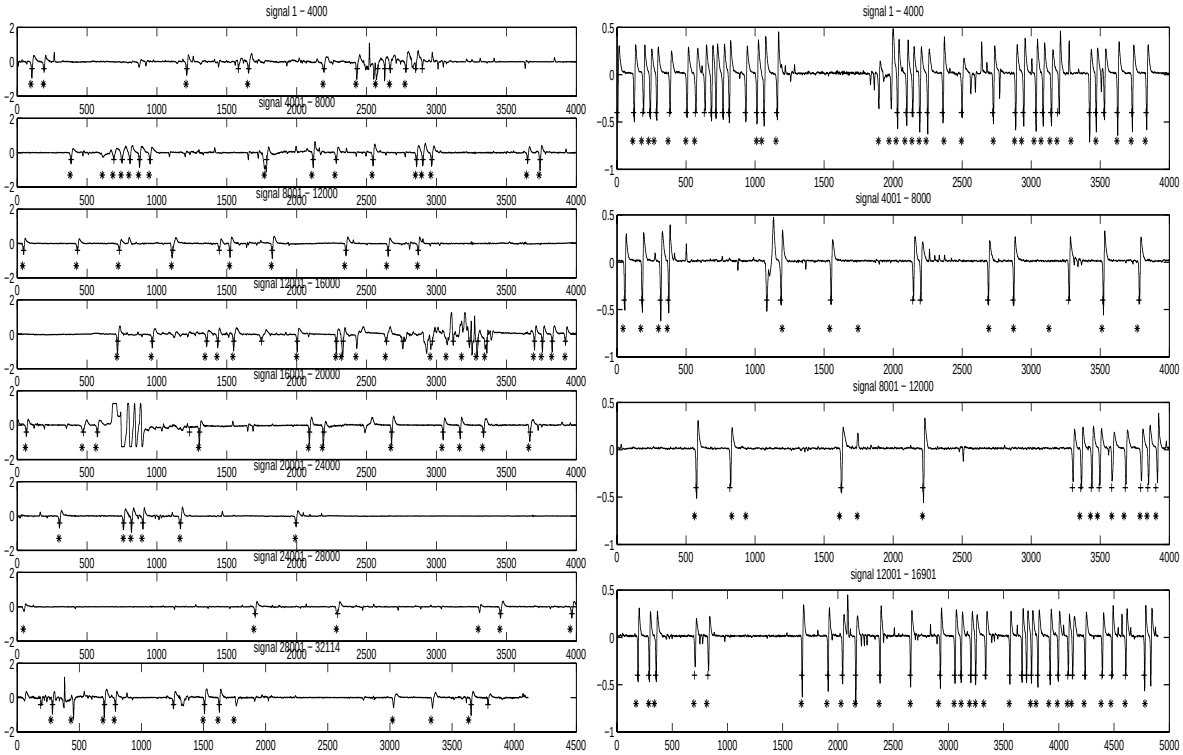


Figure 8.11: Automatic detection of eyeblinks in subject 5 (a) and 6 (b); manually labeled eyeblinks labeled are denoted with *, the detected eyeblinks with $^+$

lem. The activity of the underlying sources is often distorted by noise and interferences; the number of sources is usually unknown. Moreover, diagnostic results may depend on the choice of the sensor(s) that are incorporated in the analysis. Inclusion of multiple sensors will introduce redundant information or lead to high-dimensional feature spaces, which may have a negative effect on the generalization performance of a diagnostical system. In several papers [IT00, JMH$^+$00, KO98, Vig97] it was demonstrated that Independent Component Analysis is an effective method to remove artifacts from an ensemble of EEG measurements. The methods assumes that the electric dipoles in the cortex can be modelled as independent sources and that the combined activities of all dipoles are measured on the scalp as a linear instantaneous mixture. *Saccadic movements* of the eyes introduce electric activity that distorts the EEG measurements. For quantification of Alzheimer's disease this is an unwanted distortion, since it is not related to the electrical fields in the cortex that are indicative of the disease. As an illustration, EEG was measured on a subject that was making eye movements. We processed this EEG with an ICA-algorithm, and one of the resulting components corresponded to saccadic activity. The contribution of this component to the 21 measurement channels is shown in figure 8.12, where the correlation between the component and each of the sensors has been shown as a greyvalue. It can be observed that the saccade causes an electric field over the scalp that is a dipole, which is due to the change of the polarity caused by the movement of the eyeball.

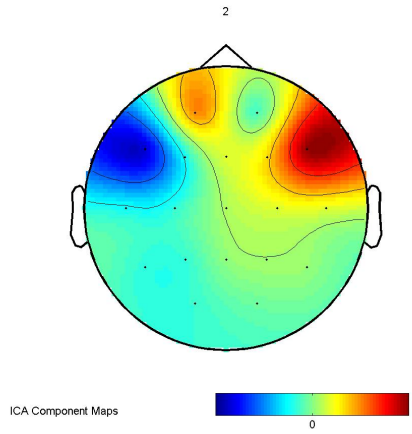


Figure 8.12: Mapping of ICA-extracted saccadic activity component on the scalp

Influence of ICA-enhancement on detection results

In a clinical experiment, two datasets were obtained[4]. Set #1 consists of a control group and a patient group. The control group contains 21 healthy volunteers (12 female, 9 male) of mean age 63 and the patient group consists of 15 severely demented patients (6 female, 9 male), also of mean age 63. All healthy volunteers were in self-reported good health and no relevant abnormalities were found during neurological examination. Set #2 consists of a control group containing 10 subjects (5 male, 5 female) with age associated memory impairment and a patient group consisting of 28 subjects diagnosed as having early Alzheimer's disease. Both groups have mean age 73. The acquired EEG signals were low-pass filtered with cut-off frequency 70 Hz and time constant 1 second at a sampling frequency of 500 Hz. The EEGs were recorded during a "no-task" condition with eyes closed. To test the influence of *correcting* an EEG recording with ICA, a classification experiment is done using features extracted from the original EEGs (from now on referred to as "raw EEGs") and ICA-corrected EEGs (referred to as "ICA EEGs"). The removal of artifacts using ICA-correction is performed in three steps: **1.** computing the ICA components, using the wavelet ICA algorithm by Koehler [KO98]; **2.** manually selecting the component(s) that represent an artifact; **3.** removing the selected components from the original recording. The selection of components that represent an artifact is not trivial: whether a deduced ICA-component represents an artifact, an EEG source (or both) is deduced from the morphology. The kurtosis of the deduced ICA component can be used for detecting eye-blinks. The spectrum can give an indication whether an ICA component is noise, interference or an EEG source. Classification experiments are performed with set #1 and set #2. For set #1 the leave-one-out method is used[5]. Set #2 has been tested on classifiers that are trained on al measurements in set #1. We trained classifiers to distinguish severe Alzheimer patients from healthy people (i.e. on set #1) and

[4]The 10-20 configuration was used in the EEG measurement setup

[5]A classifier is trained on all samples but one; the left-out sample is used for testing. Each sample in the dataset is left out once. The overall test error is the sum of all misclassified samples

Table 8.3: Detection of early-stage Alzheimer's disease with and without interference removal; the format of the figures in the table is: *total classification performance (sensitivity / specificity) (%)*; relative power in θ band is chosen as feature

sensor combination	combination method	LDC	NNC	ANNC
3 channels (C3)	raw	66 (54/100)	63 (50/100)	65 (53/100)
	ICA	71 (61/100)	71 (61/100)	71 (60/99)
17 channels (C17)	raw	55 (39/100)	63 (50/100)	65 (55/93)
	ICA	55 (43/90)	71 (61/100)	70 (60/99)
21 channels (C21)	raw	55 (43/90)	66 (54/100)	71 (62/97)
	ICA	61 (54/80)	74 (64/100)	69 (59/100)

tested the classifier to distinghuish early Alzheimer patients from health subjects (i.e. set #2). We trained a linear discriminant classifier (LDC), a nearest neighbour classifier (NNC) and a feedforward neural network classifier (ANNC) with 5 hidden neurons. Results presented from ANNC are averaged results over several runs, to diminish the influence of initial settings of the neural network. Each preprocessed segment is then represented by as a feature vector. Both correlation-based (the spectrum) and nonlinear dynamics based [Sea96, SPP98] features were examined; the relative spectral power in the so-called θ frequency band (4-8 Hz) proved to be a suitable discriminating feature[6]. The results in table 8.3 indicate that ICA-correction improves the detection of early Alzheimer's disease; the choice of the classifier seems to have no significant effect. The results with channel combination C17 can be compared with the results of [And94]. On average our results are slightly worse, which can be explained by the *different approach for marking a patient* as being in an early stage of Alzheimer's disease. Preprocessing with ICA generally improved the detection of early Alzheimer's disease. In the case of severe Alzheimer's disease (set #1) less improvement was obtained with ICA preprocessing. This is presumably caused by the larger influence of interfering sources in the early stages of the disease.

8.4 Gearbox monitoring in a pumping station

The "Noord-Oost Polder" is a polder in the northern part of The Netherlands. The amount of water in the polder is controlled by two pumping stations, station "Buma" at Lemmer and station "Vissering" at Urk. The water level in the Noord-Oost Polder is approximately 5.5 meters below the water level in the nearby "IJsselmeer" lake, which acts as a drain basin for the polder. Monitoring of the gearboxes involved in the transmission of the electromotor (Lemmer) or gas motor (Urk) power to pumping motion is important to ensure pumping capacity. In recent years, prolonged rainfall has led to floodings of polders at several places in The Netherlands. Measurements at both Lemmer and Urk pumping stations were collected by TechnoFysica Barendrecht b.v. in conjunction with the water management department

[6] It is known that Alzheimer's disease leads to a 'slowing' of the EEG signal

Waterschap Zuiderzeeland. The “Buma” pumping station has been described in section 1.5.1.

8.4.1 Lemmer pumping station

Measurements from several channels were combined in the following manner: from each channel a separate feature vector was extracted and added as a new sample to the dataset. Hence, inclusion of more than one channel gives rise to several data samples, each giving some information on the current measurement setting (as opposed to one sample if only one channel would be selected). If faults are adequately measurable by all sensors, we expect the amount of class overlap in data from a certain sensor to be roughly the same for all sensors. However, since the machine under investigation is quite large and measurement directions are not always the same, this assumption may not hold in practice. Incorporation of multiple channels in the above manner might improve robustness (less dependence on particular sensor to be selected), but on the other hand might introduce class overlap, because uninformative channels are treated equally important as informative channels. An alternative representation of the measurements can be made by concatenating all feature vectors from all channels into one large feature vector. However, to achieve proper generalization of a learning system, this calls for large sample sizes (depending on the size of the concatenated feature vector). The pumps were measured periodically during a year, period autumn 1998 until autumn 1999. Segments of measurements under different water levels (simulated by gradually lowering the sliding door) are processed using a feature extraction procedure (see below). The resulting data samples are labeled in the following manner (table 8.4). If an entry in the table has a symbol “-”, this means that no measurements were used from this machine at that time. Note that we focussed on two machines only: the machine with a severe damage and the healthy machine. We generated a different dataset for each measurement channel.

Table 8.4: Explanation of labels used in SOM visualization of Lemmer dataset

measurement period	pump 2 data labels	pump 3 data labels
week 40 (1998)	2	3
week 47 (1998)	5	6
week 52 (1998)	8	-
week 13 (1999)	11	-
week 27 (1999)	14	-
week 41 (1999)	17	-

Moreover, two different features were chosen: autoregressive model coefficients and power spectral amplitudes. We fitted 30×30 Self-Organizing Maps to each dataset and the resulting U-matrices for the AR dataset using channel 4 is shown in figure 8.13. From this figure, the measurements from week 40 (labels 2 and 3) in 1998 are clearly very dissimilar. Upon closer inspection of the original measurements, the measurement procedure turned out to be fairly different than in all other cases. Settings of amplifiers and pre-filters was quite different and also the mounting positions may have been different from later measurements. Hence,

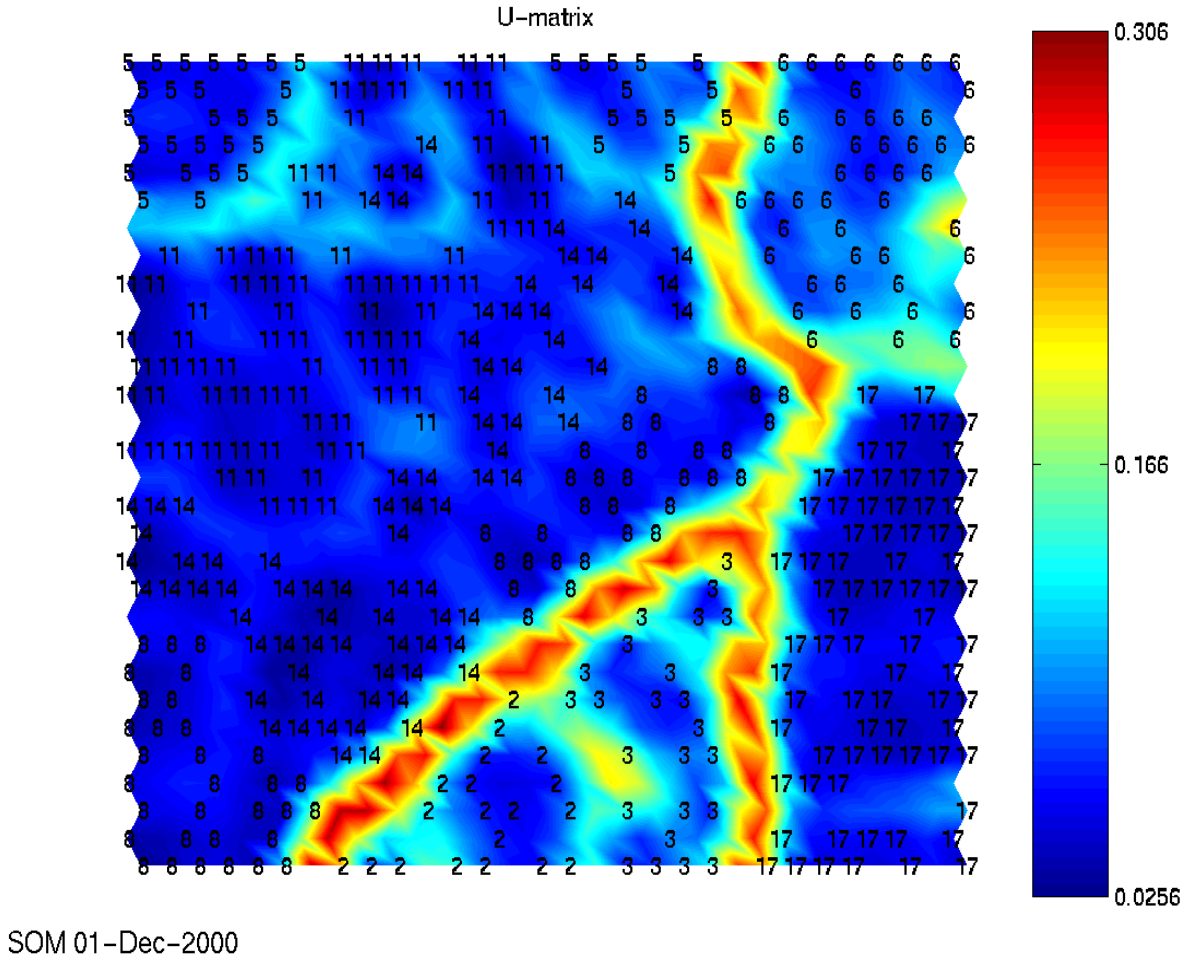


Figure 8.13: U-matrix of 30×30 SOM applied to Buma-AR dataset, channel 1

these samples cannot be compared to samples from later measurements. We can observe that the variability due to each machine results in feature vectors that are mapped onto disjunct areas: the area for pump 3 (labeled with '6') is always disjunct from areas corresponding to a measurements on repaired pump 2 (labels '11 and '14'). This was observed for several channels and both feature sets (not shown). The inter-machine variability seems to be of the same order as the the variability due to fault development (the distances between the areas labeled as 5 and 6 are comparable to the distances between the areas labeled as 5 and 8, 11 and 14). The most progressive wear in pumpset 2 is expressed by areas corresponding to label 5. This results in a clearly discernible fault area on maps from all different sensors. Data samples obtained after replacement of the damaged gear are mapped on areas that are clearly separated from the severe wear areas (which is relevant for fault detection). Finally, the dissimilar patterns with label 17 may indicate the development of new wear or failure.

8.4.2 Urk pumping station

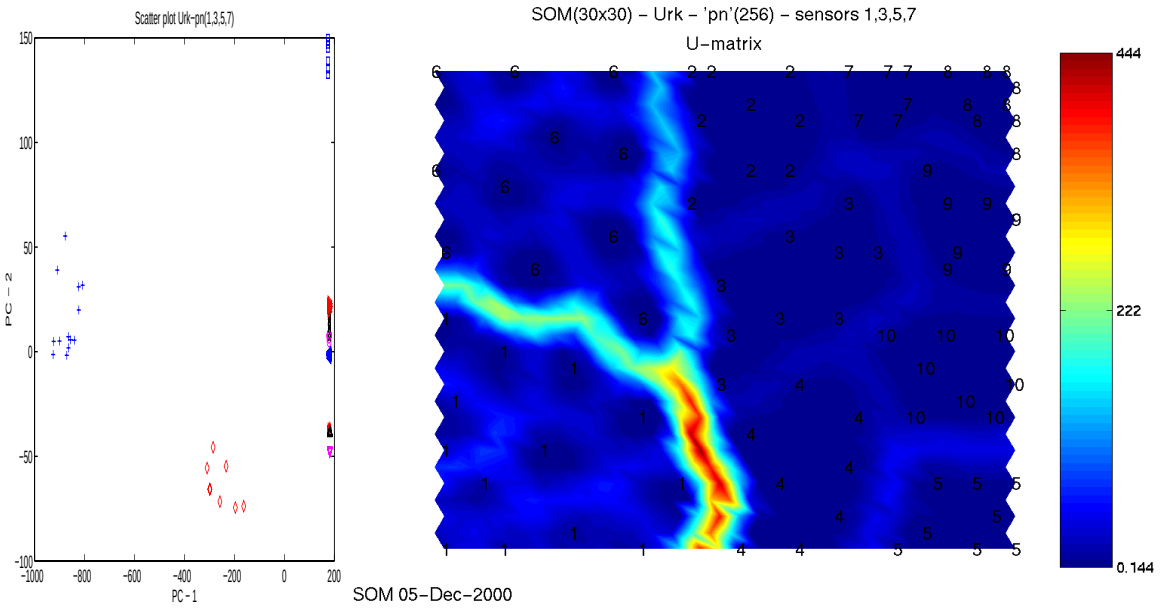


Figure 8.14: a. Linear PCA mapping and b. nonlinear SOM mapping of spectra; both plots indicate a developing fault at Urk pumping station since the patterns corresponding to first and second moment of measurement during pumping mode are clearly distinct ('+' vs. '◇' and '1' vs. '6' symbols, respectively). In generator mode, however, no significant differences are observed in the projected patterns. A manual comparison of the corresponding spectra by a vibration analyst corroborated this conjecture

In a pumping station at Urk, The Netherlands, a gas motor drives either a pump or a generator. At two different moments, vibration measurements on several positions of the machinery were performed at different operating modes (pumping; generator mode at four different power generation levels). Power spectra with 256 bins were estimated on segments of 4096 samples. The dataset was reduced in dimensionality with PCA to 100 features, figure 8.14(a). Then a 30x30 SOM was trained on the data [YK01]. The resulting map (visualized with the U-matrix method) is shown in figure 8.14(b). Labels 1 - 5 denote measurements in March 2000 (1 = pumping mode, 2 - 5 = generator modes). Labels 6 - 10 denote measurements in May 2000 (6 = pumping mode, 7 - 10 = generator modes). Patterns with label 1 are mapped in the lower-left part of the map, patterns with label 6 are mapped in the upper-left part and all generator mode patterns are mapped close to each other in the right part of the map. During pumping, a gearbox failure has developed in the second measurement series; it is absent in the first measurement series. However, in generator mode no significant differences in condition between both measurements are observed by the human expert. This is visible in the map as small distances between the 'generator mode neurons' (dark greyvalues), whereas the 'pumping mode neurons' are separated with large distance (the light ridge between the '1' and '6' areas).

8.5 MONISOM: practical health monitoring using the SOM

In conjunction with two industrial companies (TechnoFysica Barendrecht b.v., Landustrie Sneek b.v.) and the Dutch Technology Foundation STW, a monitoring tool called MONISOM was developed [YKVD01]. The tool is based on Self-Organizing Maps, and offers several analysis and monitoring facilities. The kernel is written in Visual C++ and uses the C library for statistical pattern recognition and artificial neural networks SPRANNLib [HKd$^+$] (which was developed in the Pattern Analysis Group of TU Delft). A dedicated visualization front-end has been developed in Visual Basic by R. J. Kleiman of TechnoFysica b.v. Additional support is obtained from V. van Eijk of TechnoFysica b.v. (WINAPI), J. Valk of Landustrie b.v., R. Hartman of Croon b.v. and R. Hogervorst of TechnoFysica b.v. (advice on utilization). The software runs under the Windows98 operating system (and later versions). It consists of the components **1. dataset manager, 2. map training and analysis, 3. novelty detection tools, 4. evaluation tools**. The dataset manager contains tools to read and save datasets, select features, scale features and split a dataset into a training and a test set. Map training can be done by customizing map width, neighbourhood and learning parameters and initialization. Moreover, map goodness-of-fit can be tracked. Either a new map is made or an existing map can be read and retrained. Training can be stopped by the user at will. The map is visualized by displaying the median-distance matrix of the map nodes. The map can be interpreted by the monitoring expert by inspecting the prototypes of the map (max. 4 simultaneous displays), showing the component planes (max. simultaneous displays) and clustering the SOM nodes. A 'wrap-around' effect (apparently distinct 'border-clusters' that are indicative of the same underlying cluster) can be detected with the latter analysis. Thresholds for novelty detection can be set manually, e.g. on the basis of the novelty of a separate evaluation set. Moreover, an ROC-curve can be plotted that allows the user to judge the validity of the thresholds setting for practical usage. The labeling of areas is facilitated by a hitmap (histogram of evaluation set hits on the map) and a trajectory plot. Finally, the user can 'grow' areas on the U-matrix, and designate an area as a separate map cluster. An interpretable labelling (which is stored along with all other 'sattellite information' obtained in a data analysis session) completes the analysis. We will illustrate the map creation, analysis and monitoring process in the experiment below.

8.5.1 Experiment: monitoring a progressively loose foundation

A loose foundation was induced to the submersible pump described in chapter 1. Two of the three screws that fastened the pump to the base of the concrete basin were loosened in 25 stages: after each turn of the screws a set of measurements was done in which the pump was running at 50 Hz driving frequency at 3.3 KW load. The measurements were labelled accordingly afterwards, and an autoregressive model of order 20 was applied to segments of 4096 samples of a measurement channel. We chose the first measurement channel for construction of the final dataset. The resulting dataset was linearly projected with PCA to a 2-D subspace and is plotted in figure 8.15(a). The increasing looseness is visible by tracking the labels of the feature vectors in the dataset. By going from bottom-right in the scatter plot

(labels '+') , we follow the arrows until top-middle. Then the feature vectors continue with a jump (a new vibration regime) to bottom-middle and then again following the arrows to the left part of the plot. A minimal spanning tree was fitted to the Euclidean distance matrix of

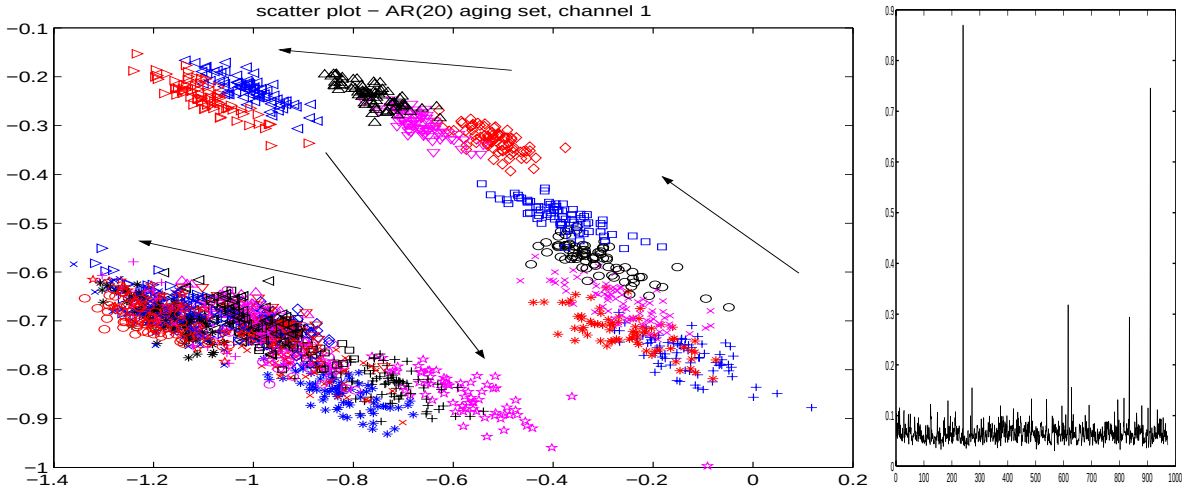


Figure 8.15: Progressively loose foundation in submersible pump: PCA projection (left) and medial axis of MST applied to the data distance matrix

the dataset. The distances on the medial axis are plotted in figure 8.15(b). The hierarchical clustering structure of the dataset is clearly visible. Separate clusters correspond to progressive (simulated) wear. From the above figure it is clear that the measurements show a gradual shift in the feature space with progressing (simulated) wear. A Self-Organizing Map is fitted to this dataset, and increasing wear is now observed as a trajectory from a 'normal map area', via several 'intermediate wear areas' to a 'severe failure area'. Map training and analysis was performed using the MoniSom package for SOM-based system monitoring. All feature components were scaled into the interval $[0, 1]$. The MoniSom console window is shown in the bottom-middle subfigure. A 50x50 map was constructed by random uniform weight initialization and training until the train error was relatively constant (and small enough). Along with the average quantization error, the goodness-of-fit measure by [KL96] (see chapter 6) can be tracked to detect topology distortions. The resulting U-matrix visualization of the map is shown in figure 8.16 (top-middle subfigure). Approximately 5 to 8 dominant clusters can be observed in the data (represented by the SOM). This is also shown in the top-right subfigure, where the nodes in the map have been clustered using *k-means clustering*. The prototypes in the map can be inspected by clicking them (subfigure immediately below the U-matrix window). Moreover, the amplitudes of the first component of each prototype vector across the map (i.e. the first *component plane*) is shown in the bottom-left subfigure. From this it can be conjectured that the left-bottom cluster in the clustered map corresponds to samples with relatively high amplitudes in the first component (in this case: high first autoregressive coefficient). A new dataset can be projected onto the trained map. This dataset is called the *evaluation set*; it should be compatible with the training set (and the map). This means that the same dimensionality (e.g. the same feature selection) and scaling should be

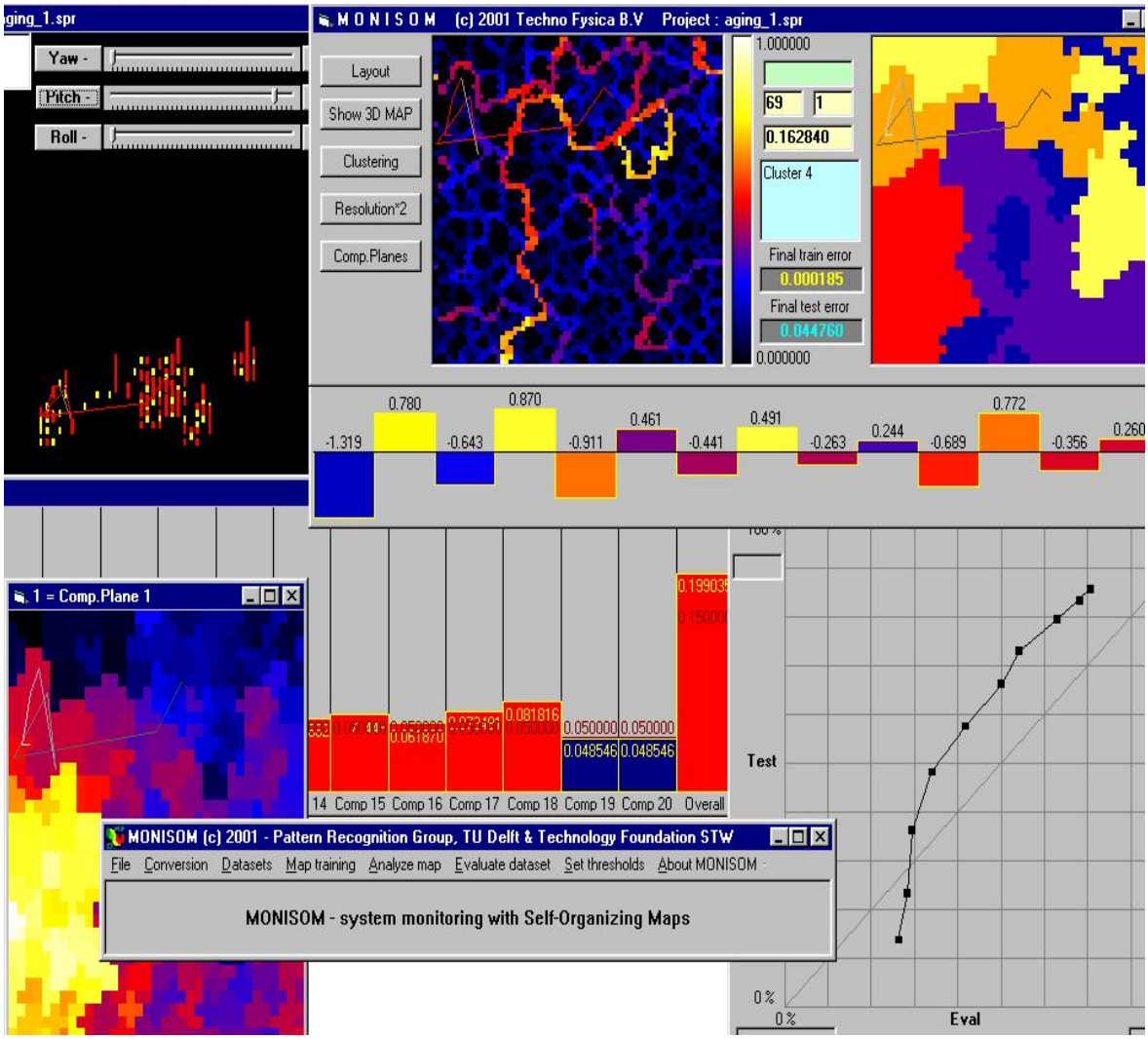


Figure 8.16: MONISOM: a SOM-based system for machine health monitoring, applied to a dataset representing a progressively loose foundation

applied to train, test and evaluation sets. After projecting, the distribution of the 'hits' on the map can be displayed (top-left subfigure). Hits with distance larger than the a priori defined novelty threshold are colored red, admissible hits are colored yellow (in the figure they can be distinghuished by differing grey values). The average distance (novelty) of the evaluation set is determined *per feature* as well and compared to the per-feature threshold; this is shown in the middle subfigure. Per-feature distances larger that the per-feature threshold are colored red; admissible distances are colored blue (again visible as differing grey values in the figure). The effect of a single deviating feature can hence be tracked. The overall novelty is compared to the overall novelty threshold in the right-most bar (which is colored red in this case). This signifies an overall novelty.

The similarity of testing set and evaluation set in terms of their novelty can be analyzed

using the ROC-curve profile (bottom-right subplot). The effect of modification of the overall novelty threshold on the acceptance rate of test and evaluation sets is computed and plotted. A 'dissimilar' evaluation set will show a profile significantly above the $y = x$ line (approximately 100% acceptance of test samples at the expense of accepting only a small percentage of evaluation patterns). Similar datasets will give rise to a graph that is in the vicinity of the $y = x$ line (which is the case in our example). Finally, the samples in the evaluation set can be projected sequentially on the map. The trajectory of the operating point on the map (and its derivative 'views' like component plane, clustered map and hitmap) can now be shown (see the sliding tail in the clustered map, top-right window). This is also shown in a 3-D plot, where the Z-axis corresponds to the novelty of each sample. If the novelty is higher than the novelty threshold, the tail is colored red; the overall novelty threshold is visible in the plot as a green grid that surmounts the U-matrix of the map. During tracking, the label of each sample is shown along with the index of the sample in the dataset. This facilitates detection of the onset of wear or failure on-the-fly.

8.6 Discussion

In this chapter we applied the learning approach to several real-world monitoring problems. It is possible to find a meaningful description of machine health in three pump monitoring problems, on which an unsupervised (SOM) or supervised (classifier) recognition method can be trained. In the *submersible pump* problem, a healthy machine can be distinghuished from a machine that exhibits either of small or large bearing failure or an imbalance by using *one sensor* only. An autoregressive model gives rise to a proper description of machine health, since the dimensionality of the feature vectors can be kept low whereas the detection results are satisfactory. The variability due to different measurement sessions proved to be fairly small. We suggest that repeated measurements using the same mounting procedure will not cause very different patterns to occur, since the system characteristics will be almost the same (but this has to be verified for each new machine). When larger feature dimensionalities are chosen, however, overtraining may lead to classifiers that do not generalize well to patterns from a repeated measurement. This can be remedied by including patterns from several repeated measurement sessions into the learning set.

A combination of supervised and unsupervised learning techniques was proposed in a *gas leak monitoring* system. This allowed for a small false alarm rate and a high detection rate in an experiment with measured leak activity and backgroud noise.

We then studied the applicability of the learning framework presented in chapter 1 for two *medical applications*. We showed that a Self-Organizing Map can be used to learn the characteristics of an eyeblink, which may be used for quantification of the severity of Tourette's syndrom based on EOG measurements. Moreover, we demonstrated that ICA-denoising of multichannel EEG-measurements prior to classification may allow for improved detection of early Alzheimer's disease. This preliminary result should be verified in larger-scale clinical experiments before the use of ICA-denoising for this problem can be really established.

In two cases where pumps in a *pumping station* were monitored, a description of the measurements with AR- and spectrum features allowed for the construction of interpretable Self-Organizing Maps. Moreover, fault patterns are mapped on 'fault areas' that are distinct from 'normal areas' if the sensor position is chosen properly. Combined with the possibility to assess the novelty of a newly acquired data sample with a SOM (chapter 6), we obtain a monitoring method with practical usability.

Hence, a *SOM-based monitoring system* MONISOM was developed which enables the monitoring expert to make maps that describe a set of (preprocessed) measurements, analyze and calibrate the map and set the novelty thresholds in a principled manner.

Chapter 9

Conclusions

In this thesis we have presented a framework for health monitoring with learning methods. In many monitoring problems, the signals due to the relevant sources are disturbed with interfering sources. Proper feature extraction is an important application-dependent task. Data from a normally functioning system is often much easier to obtain than (a complete set of) fault-related measurements. Moreover, (health-related) observations from a dynamical system will usually not be stationary over time, despite the fact that the system may still behave in a normal (admissible) manner. For each of these subproblems we have identified and investigated a number of different learning methods. Our aim has been to assess the usefulness of these methods in a real-world setting. We now summarize the main results of the research reported in this thesis.

9.1 Main results of this thesis

From the results presented in this thesis we draw the following conclusions:

Feature extraction *correlation-based signal processing methods are suitable as health indicators when using machine vibration for health monitoring.* Deviations in the vibration patterns occurring in rotating machines will often show up as (semi-) periodic events. However, explicitly modelling this periodicity (parametric signal modelling) may require too much prior knowledge and may offer too little robustness to actual measurement conditions (noise; operating-mode dependence; machine-specific deviations; modelling errors). A general correlation-based feature extraction procedure (nonparametric spectrum; enveloping; autoregressive modelling; self-organized extraction of translation-invariant features) allows for robust extraction of health related information from machine vibration signals (chapters 3, 6 and 8). Nonlinear and nonstationary phenomena were not observed in the machines under investigation. However, they may very well occur in other machine setups. The relevant feature extraction procedures for these phenomena (chapter 3) are again correlation-based (cf. the bilinear forms framework in chapter 5 that subsumes both cyclostationarity and time-frequency analysis). However, results may then depend on proper choice of algorithm parameters (time- and frequency resolution; cyclic frequency) and the signal-to-noise ratios

in the measurements.

Linear mixing *machine health monitoring applications often involve linear mixing of vibration sources*. In acoustical settings, a far-field narrowband rotating machine source can be mixed with its environment according to a linear instantaneous mixing model (chapters 4 and 5). For the more general acoustic mixing case and for vibrational mixing in mechanical structures, the convolutive mixing model is appropriate (chapters 1 and 5). The supposed linearity of the model is supported by the much-practiced (local) linear approximation for modelling machine structures (e.g. the FEM-modelling approach in chapter 3). For the submersible pump the validity of linear modelling was verified in chapter 3.

Blind demixing *the dominant sources can be separated 'blindly' when mixtures are registered on a sensor array*. The appropriateness of linear mixing models allows a range of blind source separation methods to be used for reconstruction of the (original machine) sources. The advantage of this approach is that one does not need to identify the frequency response function (chapter 3) of the machine explicitly using an (expensive) i/o modelling procedure. Different mixed machine sources will usually have a different spectrum. Hence, we may use second-order statistics based (chapter 4 and 5) techniques for source separation. In experiments with simulated (mixed) acoustic sources, no significant differences in performance between higher-order and second-order statistics based separation methods could be found, unless one of the supposed source properties (either different nongaussianity or spectrum) is absent. From our experiments in chapter 5 we can conclude that combination of different types of separation criteria in the orthogonal source separation framework may give rise to a more robust separation procedure. This will depend on source characteristics, e.g. the choice for the time-lags in the combined SOBI-SCORE algorithm highly influenced the separation results. We demonstrated that mixed machine sources can be reconstructed blindly in three different setups: acoustical mixing of a simple pump (using a time-frequency separation algorithm), vibrational demixing of two connected pumps in a laboratory setup and reconstruction of an oil pump signature in a real-world monitoring setup (in both latter cases a convolutive demixing algorithm was appropriate).

The learning approach to health monitoring *learning methods can be applied successfully for practical health monitoring*. Several experiments with real-world data reported in this thesis point in this direction:

For the *submersible pump*, we demonstrated that a satisfying trade-off between false positives and false negatives can be obtained using a novelty detection approach (chapter 6), provided a proper description of machine health is being made (e.g. by using a high-resolution power spectrum or a sufficient order autoregressive model). In these experiments, measurements from several operating modes of the machine were incorporated in the datasets, indicating a certain level of robustness. The submersible pump may be monitored (anomalies can be detected) with one sensor only, provided a proper mounting position and feature extraction is chosen. For small feature dimensionality and proper choice of measurement channel, classifier can be constructed that generalize to patterns from unseen measurement sessions.

In general it appears to be advisable to incorporate patterns from several repeated measurement sessions into the learning set, since this increases the generalization capabilities to test and evaluation data from the same machine (chapter 8).

We demonstrated that a *novelty detection approach with Self-Organizing Maps* allowed for a discrimination between (typical) noise and leak signals in a *leak detection problem* (chapter 6). A monitoring system involving both a supervised and an unsupervised learning module (feedforward neural network and SOM, respectively) yielded adequate results: a low false alarm rate and high detection rate can be obtained and dissimilar (ship) noise patterns can be rejected when processing the complete set of measurements, see chapter 8.

We showed that the *framework* for health monitoring with learning methods presented in chapter 1 is applicable within the realm of *medical monitoring problems* as well. Detection of eyeblinks in EOG measurements from Tourette's syndrom patients could be automated fairly well when taking a SOM-based learning approach. Moreover, ICA-denoising of EEG patterns from patients suffering from early Alzheimer's disease prior to classification allowed for improved recognition results; these results were considered satisfactory, given the complexity of the task at hand.

For practical usage, an *interpretable description of machine health* should be offered to the vibration analyst. In two *gearbox monitoring problems* in pumping stations, Self-Organizing Maps could be constructed that were considered interpretable by the monitoring practitioner and revealed a clustering structure that corresponded to the user's expectations (chapter 8). Combined with its possibilities for novelty detection, a SOM was considered a suitable method for practical health monitoring. Ensuingly, a practical tool (MONISOM) has been developed.

9.2 Recommendations for practitioners

An important question that arises from this research is: *what should the practictioner keep in mind?*

Generality each machine vibrates in a different manner. A monitoring method will probably have to be retrained for each machine. However, training on several repeated measurements on several similar machines in several operating modes may allow for a more general monitoring method. Moreover, dedicated features should be determined for each new (type of) machine.

Feature extraction & dimensionality we assumed that a proper feature (selection) has been chosen, such that the feature dimensionality is not too high (since this might hamper generalization properties of the method). If the data lies in a subspace, application of an initial dimensionality reduction may be a good idea.

Source separation in practice the separation of multichannel machine measurements in independent (or temporally uncorrelated) components may allow for better interpretation of vibration spectra and removal of interfering sources. However, the number of components that is to be retrieved and the filter order will influence the results of the sepa-

ration procedure. Knowledge about the machine configuration and experimenting with different algorithm settings are expected to be helpful in avoiding spurious components.

Dynamic modelling and the SOM the SOM is not an ordering-preserving mapping. In the future, one will probably like to monitor a machine using a map that gives an indication of the likelihood that the machine will transition into the 'next' stage of wear. This type of maps may then be used for trending as well. A hidden Markov model showed some disadvantages for modelling jumps to health states (chapter 7). A switching state-space model or a Bayesian approach may be more appropriate. However, we think that a SOM is still appropriate for monitoring, but an extensive map calibration and validation phase has to be performed in order to obtain a suitable monitoring method.

Summarizing, the learning approach to health monitoring proposed in this thesis is a feasible approach for practical system monitoring. It exploits general assumptions about the problem, such as: (a) health information can be extracted from machine vibration; (b) machines vibrate fairly reproducible over time if no significant wear is present; (c) the number of relevant machine operating modes is fairly limited; (d) interfering vibration sources are often mixed linearly into the sensor array. Using our approach, an extensive input-output modelling or a formalization of all possible fault scenarios prior to automatic fault detection may be avoided. Accuracy and robustness of the resulting method will however depend on the machine type and the environment, operating modes, feature extraction procedure and proper calibration/ evaluation of the learning system. The learning approach offers a solid *framework* for system monitoring, not a fully automatic design procedure for obtaining an accurate monitoring method. This should be kept in mind by the practitioner.

Appendix A

Technical details on pump setup

Control of water, load and pressure
To keep the motor compartment water-free a seal is placed around the shaft just above the impeller casing. To ensure correct working of the seal an oil-compartment is located between the casing and the motor. In this compartment a 'water-in-oil' detector is placed measuring the conductivity. When fluid passes the seal and mixes with the oil there will be a noticeable change of conductivity which can be used as a warning mechanism. The pump is lowered in the basin until it is completely under water. To control the pressure of the outlet of the pump a pipe system with a membrane valve is connected to the outlet. Via this pipe the water is redirected to a separated part in the basin to calm down, see figure A.1(a).

Impeller and motor
When the impeller rotates, an increase of momentum is applied to the fluid in the casing. Also a suction effect at the inlet of the pump through a reduction of pressure is derived as a result of this fluid motion. The fluid escapes through the outlet in the impeller casing. With its rigid closed single vane impeller, see figure A.1(b), this pump is used especially for drainage of waste water. The motor is located in the upper half of the pump. Three coils are each connected to one of the three alternating power phases applied to the pump. The shaft is embedded with bars and acts as the rotor. The rotation speed of the shaft can be regulated with the frequency of the phase voltage. The resulting heat dissipates in the fluid in which the pump is submerged. To control the temperature in the motor, PT-100 elements are interwoven with the coils to measure the temperature of the coils. The power consumption is 3 KW and the maximum rotation speed is 1500 RPM. The highest outlet flow capacity is 1.7 M^3/min.

Sensors and mounting
The transducers used for the measurements are (PCB) acceleration underwater sensors. In the initial setup, one ring accelerometer is placed on top of the pump to measure axial vibration, the other ring accelerometer is placed close to the single bearing. Near the double bearing a triaxial accelerometer is used. This triaxial sensor records vibration in three orthogonal directions. At later stages the sensors were placed at different measurement locations. All

sensors are mounted with a screw-stud, which in turn is glued to the surface of the pump with a hard waterresistent epoxy glue.

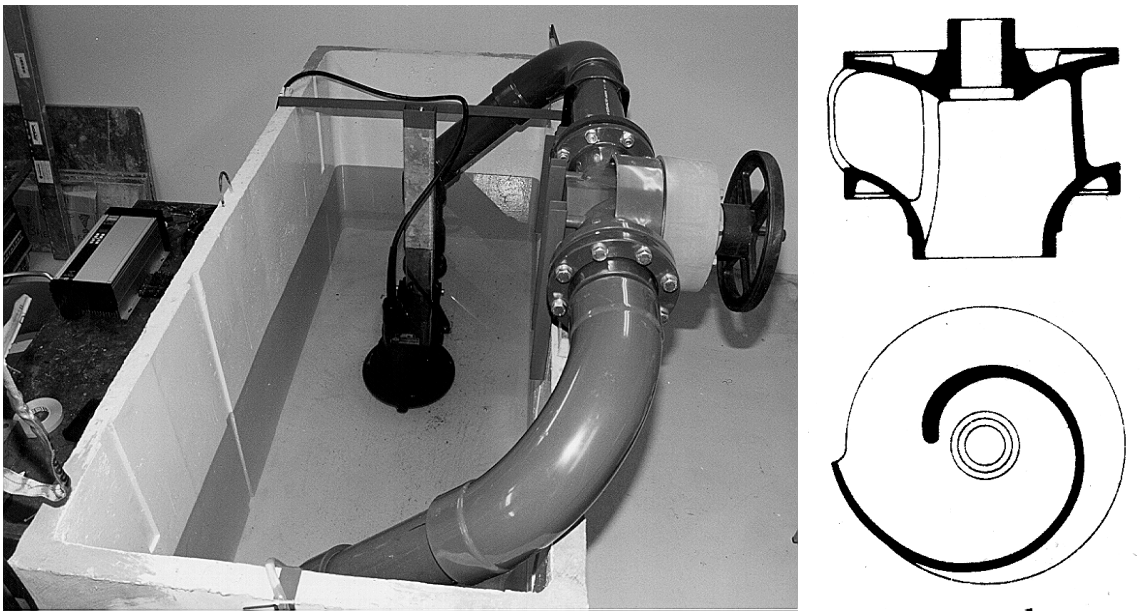


Figure A.1: a. Front-view of our setup. The concrete basin measurements are $2.6 \times 1.2 \times 1.4M$ and is filled with approximately 3000 l water. The pump is guided in place via the metal bar mounted on the mounting feet which is screwed on the basin floor. The outlet pipe system with the valve can be seen on the right side of the basin; b. closed single vane impeller, also known as one-channel impeller. This impeller is used in the DECR 20-8 Landustrie pump for transporting highly polluted fluids thanks to its rigid construction

Speed and frequency range

Machine vibration can be measured via displacement, velocity and acceleration [SL86]. Each parameter has its own field of application. For measuring low frequencies up to 100 Hz displacement sensors can be used. A velocity sensor gives the flattest frequency spectrum and is applicable in the range of 2 Hz to 1 kHz. High frequencies from 20 Hz to 10 kHz can be measured with acceleration transducers, which were chosen in this study. Velocity and displacement can then still be determined through integration. The maximum rotation speed of the pump is 1500 RPM, i.e. 25 Hz, although in practice the machine rotates a little bit slower due to non-nominal loads and slip. Hence, we need to measure within the frequency range of 8 Hz = $\frac{1}{3} \times 25$ [Sv96] to 10 kHz. This is generally accepted as a sufficient bandwidth for detecting defects in relatively slow machines [Bro84, Sv96, Mit93].

Appendix B

Musical instrument recognition

A *sound classification* problem that is related to machine diagnostics is recognition of speakers and musical instruments. *Speaker recognition* bears many similarities to machine diagnostics: one measures time series from a system (the speaker) with a wide range of operating behaviour (the many different utterances of a speaker), and the aim is to distinguish one speaker's characteristics from other speakers. Here one can use the fact that each person has its own characteristic speech pattern. The vocal tract is usually modelled by an autoregressive (AR) or "all-pole" model [RJ93]. From the AR-parameters one can derive so-called mel-scale cepstral coefficients, which are known to be a very powerful feature for speech and speaker recognition [BM93, KB97]. These coefficients correspond to the influence of the human vocal tract on the excitation signal that is being generated by the lungs and vocal chords. The residual with respect to the model (which is an approximation to this excitation signal) contains information that can be used for speaker identification [DT96, KB97, TH95], like information about the natural frequency of the excitation signal and on the (un)voicedness of a phoneme.

Dubnov & Tishby [DT96] represented the timbre of a musical instrument with higher-order moments of the residual distribution. They proposed that sounds from musical instruments and machines have a texture (or timbre) that cannot be characterized with an analysis basis on second-order statistics, like a spectrum analysis. After a temporal whitening step (deconvolution of a segment of machine or instrument sound such that a signal with flat spectrum is obtained), the residual information in the signals is of a higher-order nature. In the feature space spanned by the third- and fourth-order moments of the residuals, different types of musical instruments could be separated (e.g. brass, strings and woodwind sounds). Moreover, smoothly running machines and noisy engines could be distinguished in this space.

We repeated this experiment [Spe99], where the note C4 was acquired from 7 synthesized musical instruments and split into frames with a duration of 10 to 100 ms. A scatter plot of the first principal component against the second principal component (horizontal and vertical axis, respectively) of the dataset in the three-dimensional space spanned by 2nd, 3rd and 4th order moments of the residual empirical distribution is plotted in figure B.1. We see that for these (relatively stationary) sounds the residual method of Dubnov and Tishby extracts useful

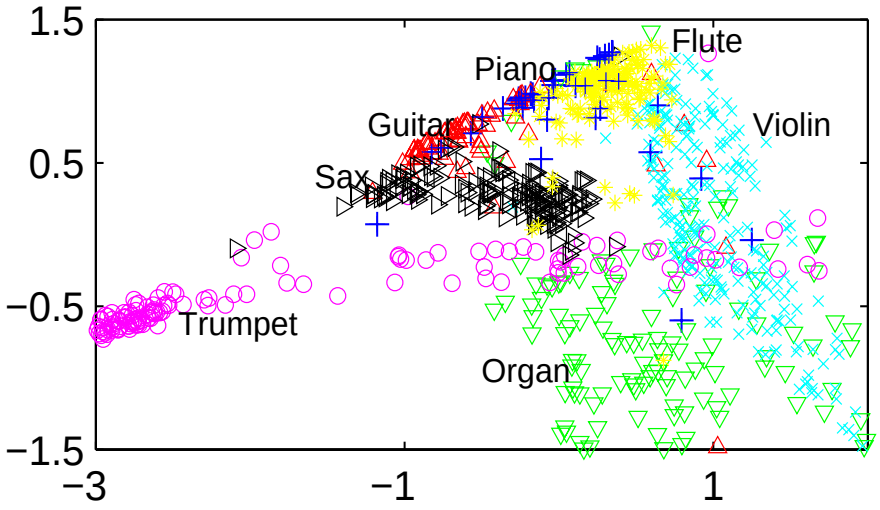


Figure B.1: Identification of musical instruments using higher-order moments of residual

information: the different instruments can be distinguished fairly well using the higher-order features.

Repeating the experiment on speaker utterances showed that the method is less suitable for speaker identification. Scatter plots revealed much overlap. From preliminary experiments, the influence of the frame length appears to be quite large with this method. This can be caused by the fact that speech is usually less stationary on a 23 ms interval than musical sounds (in [AH92] it is stated that the all-pole model is apt for voiced speech for a period of approximately 10 ms, during which the source is considered stationary). The suitability of the approach in a machine monitoring application will be determined by the degree of stationarity of the machine signals: shorter stationary periods cause a larger variation in estimates of high-order statistics which yields poor identification results. For machine noise, however, Dubnov & Tishby achieved promising results.

Appendix C

Delay coordinate embedding

It has been shown that time series produced by low-dimensional chaotic systems can be predicted by reconstructing the attractor using a delay coordinate embedding. Takens' theorem states that an attractor of dimension D can be reconstructed by sliding a window (a delay vector, as in equation (3.15)) of length at least $2D+1$ over the time series from the system, and look upon this as a dataset in the $2D+1$ dimensional space [WG94]. A system with low-dimensional structure (e.g. a deterministic chaotic system) exhibits low intrinsic dimensionality when observations are embedded, whereas a fully stochastic system (without any particular structure) fills up the whole embedding space. We will demonstrate in the following example that embedding of a time series from a *nonlinear system* may reveal (low-dimensional) structure that is not visible from the time series itself.

Example C.1: low-dimensional structure via embedding

A dynamical system can be described [Tak94] with an evolution law $\phi(x)$ that governs the new position in the state space from the current x position[1]. The law can involve a complete probability distribution over the state space, but in case of a deterministic system $\phi(x)$ will consist of one value only. A simple dynamical system with low-dimensional structure is the logistic map. It has an evolution law

$$\phi(x) = \mu x(1-x) \tag{C.1}$$

After setting the μ parameter that controls the chaotic nature of the system to 3.9, the time series in figure C.1(a) can be observed. Figure C.1(b) illustrates the broadband nature of the signal. There is clearly structure present in the time signal, see figure C.1(a), but that is not visible from its spectrum: the time series plot suggests a low-pass characteristic, but in the spectrum higher frequencies are almost equally energetic as lower frequencies. We need a different viewpoint to highlight this structure. After embedding the resulting time series in a two-dimensional delay-space, we can observe this structure in figure C.1(c). After randomizing the order of the samples in the time series (so the temporal structure is destroyed) no structure in the embedding can be observed, see figure C.1(d). □

[1] We assume that state variable x contains all the necessary information to predict the system evolution

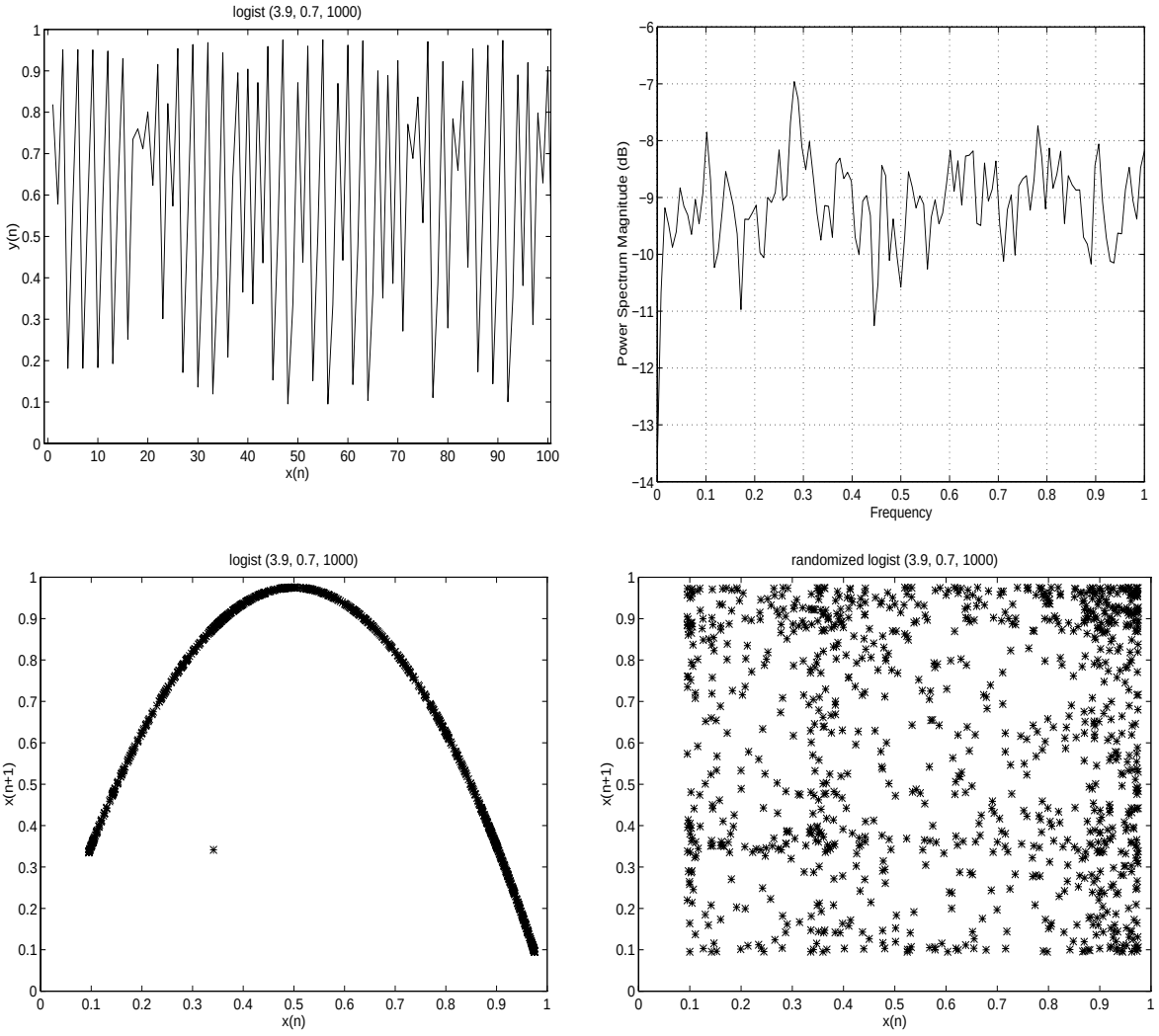


Figure C.1: Logistic time series (a) and its spectrum (b); delay coordinate embedding of logistic time series (c) and randomized logistic time series (d)

Appendix D

Two linear projection methods

We give some details about two linear projection methods on which the blind source separation methods described in chapters 4 and 5 are based.

D.1 Maximizing variance: PCA

Principal Component Analysis (PCA) is a well-known technique in multivariate data analysis, where an N-dimensional (zero-mean) dataset $\mathbf{x}$ is projected on the eigenvectors of its covariance matrix

$$\mathbf{v} = U^T \mathbf{x} \tag{D.1}$$

where U is an orthogonal matrix containing the eigenvectors of the data covariance matrix. PCA can be used for *sphering* the data $\mathbf{x}$, if the PCA components are scaled with respect to the component variances

$$\mathbf{z} = \Lambda^{-\frac{1}{2}} \mathbf{v} \tag{D.2}$$

where Λ is the diagonal matrix of eigenvalues of the original data covariance matrix

$$\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_N), \quad \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N \tag{D.3}$$

The sphered data $\mathbf{z}$ has a covariance matrix that is a $N \times N$ diagonal matrix. Projection onto the eigenvectors decorrelates the dataset (whitening) and scaling with the inverse of the eigenvalues results in a variance normalized dataset (sphering). Truncating the expansion in (D.1) to the most significant eigenvectors allows for a dimension reduction, while preserving most of the information in the data (since the eigenvalues λ_i correspond to the variance σ_i^2 of the data projected onto the corresponding eigenvectors). This truncation is optimal in a least-squares sense [Hay94]. The result of PCA on an artificial dataset is shown in figures 4.5(b) (whitening) and 4.5(c) (sphering).

D.2 Maximizing nongaussianity: projection pursuit

In the late eighties, the well-established technique of *projection pursuit* was applied to general data analysis in [Fri87], now termed *exploratory projection pursuit*. In this technique, one tries to find 'interesting' directions in the data by projecting it onto a low-dimensional subspace while maximizing a certain interestingness criterion. It is proposed that the Gaussian distribution is the least interesting distribution, since the multivariate density is completely characterized by its linear structure (first and second order moments), all of its projections are normal, most views (linear combinations) of possibly interesting components will be Gaussian (Central limit theorem!), and the normal distribution has the least information in terms of entropy or Fisher information (for fixed variance). In order to maximize nongaussianity, one formulates [Fri87] a criterion that measures the departure from the normal distribution. After having sphered the original data to Z, the task is to find a projection direction α such that the probability density $p_\alpha(X)$ of the projected data

$$X = \alpha^T Z \tag{D.4}$$

is highly structured, i.e. as nongaussian as possible. When normalizing the projected data X using the standard normal cumulative density function

$$\Phi(X) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{X} e^{-\frac{1}{2}t^2} dt \tag{D.5}$$

according to

$$R = 2\Phi(X) - 1 \tag{D.6}$$

it follows that R will be uniformly distributed on the interval $-1 \leq R \leq 1$ if X is normally distributed. As a measure of nongaussianity of X one can now take the nonuniformity of R, e.g. by measuring

$$\int_{-1}^{1} p_R^2(R) dR - \frac{1}{2} \tag{D.7}$$

where $p_R(R)$ is the probability density of R and the $\frac{1}{2}$ denotes the uniform distribution over the interval $-1 \leq R \leq 1$. This measure emphasizes nonnormality in the body of the distribution rather than in the tails, which is natural when one is looking for projections that exhibit clustering or other kinds of nonlinear associations. Expansion of $p_R(R)$ in Legendre polynomials of order j (i.e. $P_j(R)$) and taking sample estimates leads to the sample projection index $\hat{I}(\alpha)$ of α at order J

$$\hat{I}(\alpha) = \frac{1}{2} \sum_{j=1}^{J} (2j+1) [\frac{1}{N} \sum_{i=1}^{N} P_j(R)]^2 \tag{D.8}$$

The above index is to be maximized with respect to the q components of α under the constraint $\alpha\alpha^T = 1$. This constraint enforces that all linear combinations of the data Z have unit variance (remain sphered). When approaching the 2D-case, one should also enforce the orthogonality of both projection vectors, i.e. $\alpha^T\beta = 0$. The maximization can be done using standard gradient ascent, since the gradient of (D.8) with respect to the components of α can readily be derived.

Structure removal

With projection pursuit, views need not be orthogonal and there is no reason to expect that a certain view maximizing nonnormality will be the only informative view (the nonnormality will be manifest in several 1-D or 2-D projections). Hence a structure removal procedure was proposed, in which a view is found, the structure induced by this view is removed from the data and then the projection index is remaximized until there is no structure (i.e. interestingness) left in the data. Basically, one applies a transformation

$$X' = \Phi^{-1}(F_\alpha(X)) \tag{D.9}$$

to the projection under investigation $X = \alpha^T Z$ that makes the result X' have a standard normal distribution. Define U as an orthogonal matrix with α as the first row. Applying U to the data leads to $T = UZ$. When we now take a transformation Θ with components $\theta_1 \ldots \theta_q$ that has the property that it normalizes the first view and leaves the other directions unchanged

$$\theta_j(T_j) = \begin{cases} \Phi^{-1}(F_\alpha(T_1)), & j = 1 \\ T_j, & 2 \leq j \leq q \end{cases} \tag{D.10}$$

we have found the necessary transformation after noting that

$$Z' = U^T\Theta(UZ) \tag{D.11}$$

will do the job[1]. Finally, we end up with a hierarchical (1-D or 2-D) description the that should capture the relevant (nongaussian) structure in the data.

Example D.1: projection pursuit with spirals data

Consider a spiral in 5-D space, which is expressed analytically as

$$(x_1, x_2, x_3, x_4, x_5) = (\cos(n), \sin(n), \frac{1}{2}\cos(2n), \frac{1}{2}\sin(2n), \frac{1}{2}\sqrt{2}n) \tag{D.12}$$

and the data length is chosen to be 500. A two-dimensional projection pursuit with a large gradient ascent learning rate (0.9), a moderate number of steps (25) in the premaximization, gradient ascent and gaussianizing leads to the first four views that are shown in figure D.1. From these views, the intrinsic structure (sine-like and 'wavy' in four directions, elongated

[1] Note that since successive projection directions will usually not be orthogonal, Gaussianizing at some stage in the pursuit may affect previously found projections, i.e. restore interestingness in already analyzed (and Gaussianized) components. However, in practice the amount of restored structure is small, according to [Fri87]

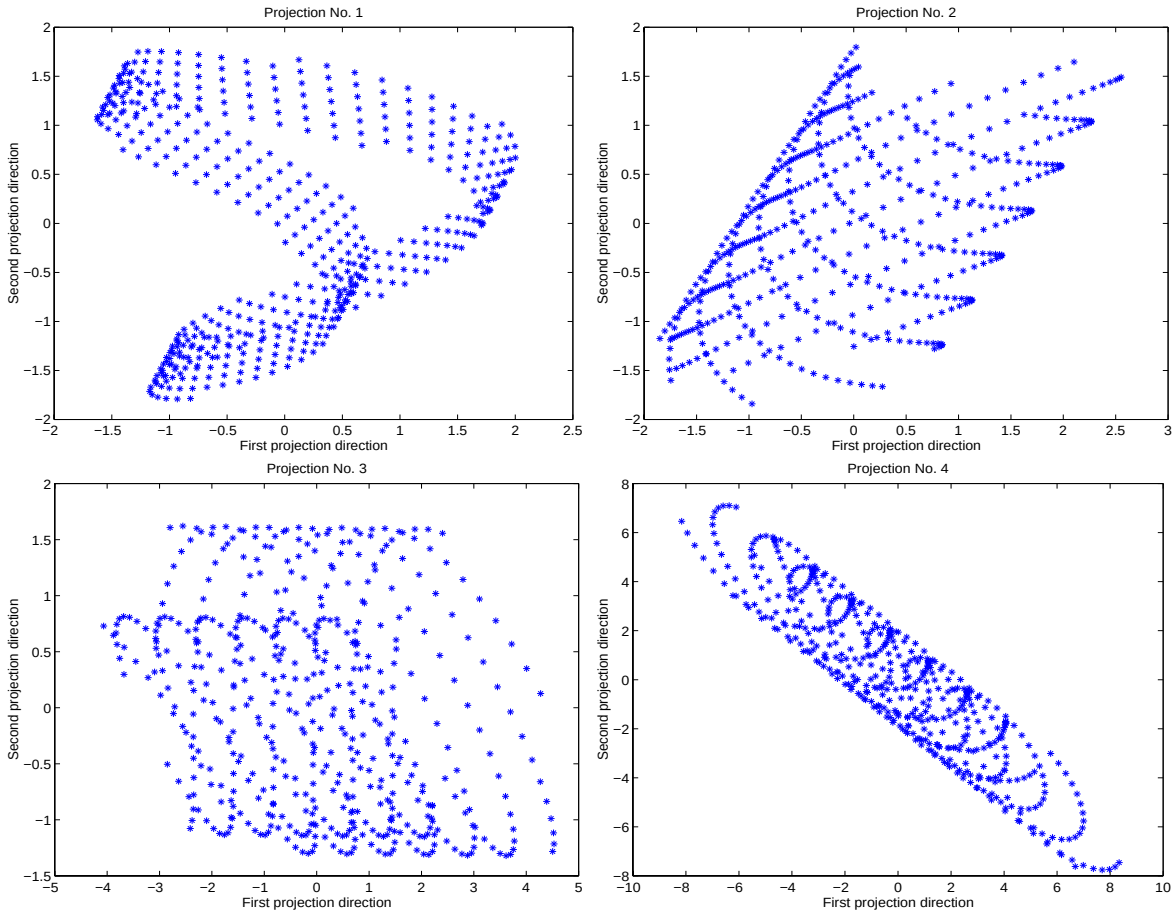


Figure D.1: 2D projection pursuit: first four views

in the fifth direction) becomes clear. Discontinuities in the views are caused by an undersampling of the curve. □

Appendix E

Optimization in MDL-based ICA

Cost function (5.8) that is given in section 5.2 leads to the minimization of *logarithmic* mapping and signal complexity terms, which seems a natural thing to do in the context of algorithmic complexity. Using a logarithmic cost function may however lead to very steep and narrow minima (e.g. when values are becoming small in magnitude, their logarithm tends to $-\infty$), giving rise to a nonstable minimization procedure. One remedy would be to *decrease the stepsize* during the gradient descent minimization (e.g. linearly or exponentially). Another approach could be to introduce a momentum term α into the gradient descent update formula

$$\Delta W(n) = -\mu \frac{\partial J(W(n))}{\partial W(n)} + \alpha \Delta W(n-1) \tag{E.1}$$

in order to avoid rapid oscillations during the minimization procedure. Note that all of these 'stabilizing' procedures are at the expense of slower convergence and increased computing time. It has been observed experimentally that measuring separation performance with the cross-talking error of section 4.2.2 gives rise to nonsmooth minimization behaviour, i.e. the performance measure doesn't decrease smoothly during minimization. This can be due to the maximum operator in the denominators in (4.15), causing sudden jumps in the cross-talking error, whereas the cost function may decrease smoothly.

Gradient descent is known to be prone to problems with complex error surfaces with numerous local minima. The larger the dimensionality of the space in which to search, the less well-behaved the error surface may be. One can search for the set of 'independent' components by a number of consecutive searches for a single component. In order to force the emergence of a useful *new* component, all the structure represented by previously found components is removed or *deflated* from the data during the course of the search. More precisely, one decomposes the 'mapping complexity term' in a number of disjoint parts

$$\log |\det W| = \sum_{i=1}^{N} \log \|(I - P_i)\mathbf{w}_i\| \tag{E.2}$$

where the projection matrix P_i is

$$P_i = \mathbf{W}_i(\mathbf{W}_i^T\mathbf{W}_i)^{-1}\mathbf{W}_i^T \tag{E.3}$$

using the subspace of previously discovered components $\mathbf{w}_1,\ldots,\mathbf{w}_{i-1}$

$$\mathbf{W}_i = [\mathbf{w}_1 \ldots \mathbf{w}_{i-1}] \tag{E.4}$$

Note that P_1 is the empty matrix, so that the mapping term in the cost function for the first component becomes $\|\mathbf{w}_1\|$. Now the components can be extracted one by one, using only the structure not present in the subspace of previously found components, represented by the residual $(I-P_i)\mathbf{w}_i$. This means that the logarithmic cost function (5.8) can be rewritten as

$$\log J = \sum_{i=1}^{N}(\frac{1}{2L}\log\det R_i - \log\|(I-P_i)\mathbf{w}_i\|) \tag{E.5}$$

which can be minimized by minimizing every individual gradient

$$\frac{\partial \log J}{\partial \mathbf{w}_j} = \nabla_{source_j} + \nabla_{map_j} \tag{E.6}$$

$$= \frac{1}{L}\,\mathrm{tr}\,\{R_j^{-1}E(\mathbf{y}_j^T\mathbf{y}_j')\} - \frac{(I-P_j)^T(I-P_j)\mathbf{w}_j}{\|(I-P_j)\mathbf{w}_j\|^2} \tag{E.7}$$

This term can be computed using the currently and previously deflated components only. It should be noticed that it is not guaranteed to find the global minimum in this manner: the joint minimum along the ensemble of directions (weight vectors) need not be the same as the ensemble of the minima along individual directions. This bears some similarity to the problem encountered with greedy search algorithms, where a sequence of locally optimal steps at each stage need not result in the overall optimal sequence [BD89].

Finally, we mention the possibility to optimize the MDL-ICA cost function on the Stiefel manifold [EAS98], analogous to the formulation of a gradient for SOBI with respect to the Stiefel manifold [RR00]. The MDL-ICA cost function can be easily 'plugged into' the algorithm by Rahbar and Reilly. Simulation with similar test signals as used in [RR00] again revealed that a small number of mixed sources may be unmixed using MDL-based ICA (if suitable step sizes, etc. are chosen), but for larger number of sources the results degrade.

Appendix F

Measuring topology preservation

Topology preservation of a mapping from some input into some output space means that points nearby in one space are mapped onto points that are nearby in the other space [BP92]. The SOM update formula (6.10) enforces the topology preservation property of the mapping from map space to input space, when the input space dimension is larger than the output space dimension [MS94a]. The reverse mapping need not be perfectly topology preserving, which may lead to map folding [Kas97]. Danger of overtraining and map folding becomes higher with decreasing final neighbourhood width (see figure 6.5). This effect can be tracked by incorporating a penalty for topology distortions in the SOM mapping [KL96]. This goodness measure is referred to as *map goodness-of-fit* (GOF).

Several measures were proposed in the literature for evaluation of a map's preservation of the input space topology. In [Zre93] it is expected that proper map organization implies that map units that are close to each other on the grid are only activated by vectors that are close in the input space. Hence, the line connecting two neighbouring prototypes should not be intersected too often by another unit's Voronoi region (the set of input vectors that are closest to the unit), since this indicates remoteness in the input space. Note that this method is independent of a certain set of input vectors, but is determined using the map only. Alternatively, one can focus on the continuity of the nonlinear mapping from input space to map grid. In [BP92] the *topographic product* TP was used for this purpose. Basically, the measure consists of two terms that indicate the correspondence between k-nearest neighbours in the original and the projected space. The measure tracks folding in both directions ($input \mapsto map$ and $map \mapsto input$). However, it has been pointed out that this measure cannot distinguish between folding of the map along nonlinearities in the data manifold (which is favourable) and folding *within* a data manifold (leading to discontinuities in the mapping) [VDM94]. Hence, using just the topology preservation property for quantification of map goodness (like in [MS94a]) will not suffice for tracking a map's regularity. Also the fact that a metric in the output space is incorporated makes the measure sensitive to a certain topology.

The *goodness* of a map with respect to a dataset is defined [KL96] as the average quantization error over the data set plus a term that penalizes remoteness of units on the map grid that have prototypes that are close in the input space. More specifically, the goodness C is defined as $C = E[d(x)]$, where

$$d(x) = \|x - m_{p_1(x)}\| + \min_i \sum_{k=0}^{K_{p_2(x),i}-1} \|m_{I_i(k,x)} - m_{I_i(k+1,x)}\| \tag{F.1}$$

In this formula, $I_i(k,x)$ denotes the index of the *k*th unit on a path *i* along the map grid from the unit $I_i(0,x) = p_1(x)$ matching the input vector *x* best, to the second nearest reference vector in the input space $I_i(K_{p_2(x),i},x) = p_2(x)$. The min operator in the above formula expresses that the second term consists of the cost of the *shortest* path [CLR90] from winning node to second-nearest reference vector:
C = AQE + cost[shortest_path(winner, 2-nn)].[1]

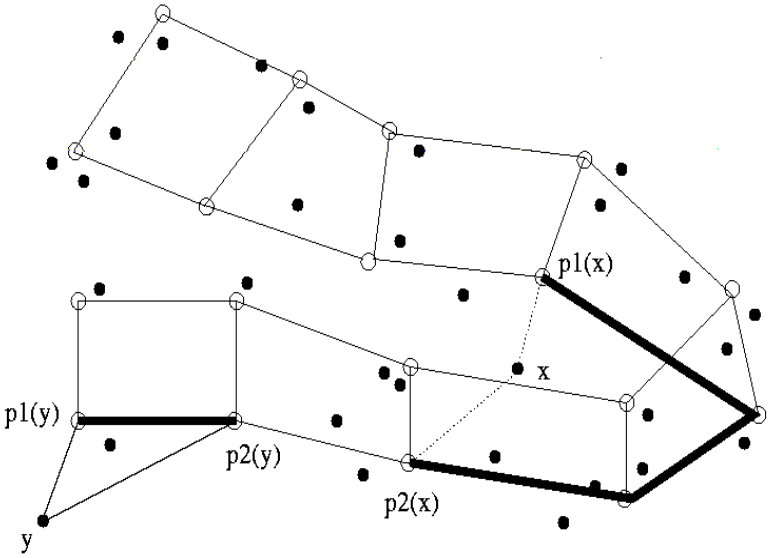


Figure F.1: Self-Organizing Maps: goodness measure for topology preservation

The measure is illustrated in figure F.1. In the vicinity of sample *y* no map folding occurs: the cost of the shortest path from the best-matching unit $p_1(y)$ to the second-nearest unit $p_2(y)$ is relatively small. Near sample *x* there is a map fold present, which is expressed by the relatively large cost of the path from $p_1(x)$ to $p_2(x)$.

[1] Note that by using this measure, only the amount of topology preservation in the mapping from input space to map grid can be tracked. This suffices, since the SOM learning algorithm enforces the preservation of the topology in the *inverse mapping*, provided the input space is of higher dimension than the map dimension [MS94a] (which will almost always be the case in practice)

Curriculum Vitae

Alexander Ypma was born in Ljouwert (Leeuwarden), The Netherlands on 15 september 1971. He received his high school education (VWO) at the RKSG “Titus Brandsma” at Boalsert (Bolsward), from which he graduated in 1989. He received a BSc. in Computer Science from Twente University in 1990. He continued his studies at the Faculty of Mathematics and Natural Sciences of Groningen University at Groningen (Grins), where he received the MSc. degree in Computer Science in 1995. His graduation work entitled ”Neural control of artificial human walking” was performed in the group on neural networks headed by prof. L. Spaanenburg.

In 1996 Ypma started his PhD research at the Pattern Recognition Group of Delft University of Technology under supervision of dr. R.P.W. Duin. The topic of his research was ”Machine diagnostics by neural networks”. In 1998 he has visited the Laboratory for Computer and Information Science of Helsinki University of Technology, headed by Prof. E. Oja. He finished his PhD research in 2001, after a (partly commercial) extension of the project that was devoted to practical monitoring applications. Currently, he is working as a postdoc at the Foundation for Neural Networks SNN at Nijmegen University on the project “Graphical models for data mining”.

Ypma’s scientific interests are in the fields of learning methods, Independent Component Analysis and dynamical systems. He is also interested in applications like (medical and industrial) health monitoring, sound/ speech/ music recognition and recognition of brain signals (EEG).

Acknowledgements

An acknowledgement in a booklet like this often reminds me of an Academy Award ceremony. This one is no exception. I'd like to thank the following persons:

Amir Leshem, Petteri Pajunen, Jan Valk, Rob Kleiman, Roel Hogervorst, Roel Hartman, Gerard Schram, Andre Smulders, Teun van den Dool, Willem Keijzer, for a fruitful and inspiring cooperation

Jean-Pierre Veen (STW), for his drive to put science into practice

SNN Nijmegen, for allowing me to finish up this piece of work 'in their time'

Co, Ole, Onno, Jan, Vincent, for showing that coaching can both be fun and fruitful

Ronald, Ed, Wim, Ad, for their support (both technical and conceptual) in running the machine diagnostics project

Geertdv, Peter, for introducing Chopin's Fantaisie Impromptu to me, which motivated me to take up piano lessons again

Mike, and all other members of PHAIO, for creating an atmosphere in which (critical) discussion and team spirit flourished

Neural team (Bob, Ed, Aarnoud, David, Dick, Ela, Marina, Pavel, Ronald), for being friends and colleagues at the same time

Prof. Van Vliet, Oja and Young, for their efforts to create a stimulating research environment; for their very useful comments on the manuscript; for putting a piano at the lab and supporting a commercial extension of the machine diagnostics project

Bob, for teaching me science and involvement; for giving me both freedom and support in my quest for questions and answers

My friends, for demonstrating that there is more to life than just work

Heit, mem, Rainier, Irene, for laying down the foundation that allowed me to reach my goal

Siepie, for ... all of the above and more. The sum of our parts surpasses the individual constituents. Thanks for all the support and joy you gave me in the past 8 years!

Bibliography

[AG99] J. Anemüller and T. Gramss. On-line blind separation of moving sound sources. In *Proceedings of ICA'99*, Aussois (France), 1999.

[AH92] A. N. Akansu and R. A. Haddad. *Multiresolution signal decomposition*. Academic Press, 1992.

[AHPV99] E. Alhoniemi, J. Himberg, J. Parviainen, and J. Vesanto. Som toolbox 2.0 for matlab. available from http://www.cis.hut.fi/projects/somtoolbox/, 1999.

[AMLL98] M. F. Abdel-Magied, K.A. Loparo, and W. Lin. Fault detection and diagnosis for rotating machinery: a model-based approach. In *Proceedings of American Control Conference*, pages 3291 – 3296, 1998.

[And94] P. Anderer. Discrimination between demented patients and normals based on topographic eeg slow wave activity. *EEG and Clin. Neurophys.*, (91):108 – 117, 1994.

[Ang87] M. Angelo. Vibration monitoring of machines. Technical report, Technical Review No. 1, Brüel & Kjær, Denmark, 1987.

[AS98] H. Attias and C. E. Schreiner. Blind source separation and deconvolution: the dynamic component analysis algorithm. *Neural Computation*, 10:1373–1424, 1998.

[ASG90] B.G. Agee, S. V. Schell, and W. A. Gardner. Spectral self-coherence restoral: a new approach to blind adaptive signal extraction using antenna arrays. *Proceedings of the IEEE*, 78(4):753 – 767, april 1990.

[Att98] H. Attias. Independent factor analysis. *Neural Computation*, 11:803–851, 1998.

[BA97] A. Belouchrani and M. Amin. Blind source separation using time-frequency distributions: algorithm and asymptotic performance. In Proceedings of ICASSP97, volume 5, pages 3469 – 3472, Munchen, 1997.

[BAMCM97] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines. A blind source separation technique using second-order statistics. *IEEE Trans. on signal processing*, 45(2):434 – 444, 1997.

[BB96] A. Barkov and N. Barkova. Condition assessment and life prediction of rolling element bearings - part 1, 2. Technical report, VibroAcoustical Systems and Technologies, St. Petersburg, Russia, 1996.

[BD89] E. Backer and R. P. W. Duin. *Statistische patroonherkenning (in Dutch)*. Delftse Uitgevers Maatschappij, b.v., 1989.

[Ber91] J. E. Berry. How to track rolling element bearing health with vibration signature analysis. *Sound and Vibration*, pages 24–35, november 1991.

[Bis95] C. M. Bishop. *Neural networks for pattern recognition*. Oxford Univ. Press, 1995.

[BM93] F. Bimbot and L. Mathan. Text-free speaker recognition using an arithmetic-harmonic sphericity measure. In *Proceedings of EuroSpeech'93*, 1993.

[Boo87] M.M. Boone. *Design and development of a synthetic acoustic antenna for highly directional sound measurements*. PhD thesis, Delft University of Technology, 1987.

[BP92] H.-U. Bauer and K. R. Pawelzik. Quantifying the neighbourhood preservation of self-organizing feature maps. *IEEE Trans. on neural networks*, Volume 3, No. 4:570–579, 1992.

[Bri00] D. Brie. Modelling of the spalled rolling element bearing vibration signal. *Mechanical systems and signal processing*, 14(3):353 – 369, 2000.

[Bro84] J. T. Broch. *Mechanical vibration and shock measurement*. Brüel & Kjær, 1984.

[BS95] A. Bell and T. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, 7(6):1129 – 1159, 1995.

[BTOR97] D. Brie, M. Tomczak, H. Oehlmann, and A. Richard. Gear crack detection by adaptive amplitude and phase demodulation. *Mechanical systems and signal processing*, 11(1):149 – 167, 1997.

[Car98] J.F. Cardoso. Blind source separation: statistical principles. *Proceedings of the IEEE*, 86(10):2009 – 2025, October 1998.

[Cem91] C. Cempel. *Vibroacoustic condition monitoring*. Ellis Horwood Ltd., 1991.

[CL96] J.-F. Cardoso and B. Laheld. Equivariant adaptive source separation. *IEEE Trans. on signal processing*, 44(12):3017 – 3030, 1996.

[CLR90] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to algorithms*. MIT Press. McGraw-Hill, 1990.

[Coh95] L. Cohen. *Time-frequency analysis*. Prentice-Hall, 1995.

[Com94] P. Comon. Independent component analysis - a new concept? *Signal processing*, 36(3):287 – 314, 1994.

[CPSY99] J. M. Carson, E. J. Powers, R. O. Stearman, and E.-J. Yi. Applications of higher-order statistics to ground vibration testing of aircraft structures. In *Proceedings of IEEE SP Workshop on higher-order statistics*, pages 362 – 366, 1999.

[CS93] J.F. Cardoso and A. Souloumiac. Blind beamforming for non-gaussian signals. *IEE Proceedings F*, 140(6):362 – 370, December 1993.

[CSL96] V. Capdevielle, Ch. Servière, and J.-L. Lacoume. Blind separation of wideband sources: application to rotating machine signals. In *Proc. of EUSIPCO '96*, pages 2085 – 2088, 1996.

[CSL00] C. Capdessus, M. Sidahmed, and J. L. Lacoume. Cyclostationary processes: application in gear faults early diagnosis. *Mechanical systems and signal processing*, 14(3):371 – 385, May 2000.

[Cyb89] G. Cybenko. Approximations by superpositions of a sigmoidal function. *Mathematical control, signals and systems*, (2):303 – 314, 1989.

[DA97] J. M. Danthez and R. Aquilina. Separation of broadband sources concept of the Labrador software. *Mechanical systems and signal processing*, 11(1):91–106, 1997.

[DFH+97] R.P.W. Duin, E.E.E. Frietman, A. Hoekstra, R. Ligteringen, D. De Ridder, M. Skurichina, D.M.J. Tax, and A. A. Ypma. The use of neural network tools in statistical pattern recognition. In *Proceedings of SNN'97*, Amsterdam, 1997.

[DFM91] M.F. Dimentberg, K. V. Frolov, and A. I. Menyailov. *Vibroacoustical diagnostics for machines and structures*. Research studies press Ltd., Taunton (UK), 1991.

[DJB96] B. Delyon, A. Juditsky, and A. Benveniste. Accuracy analysis for wavelet approximations. *IEEE Trans. Neural Networks*, Volume 6, No. 2:332 – 347, 1996.

[DO96] G. Deco and D. Obradovic. *An information-theoretic approach to neural computing*. Springer-Verlag, 1996.

[DT96] S. Dubnov and N. Tishby. Influence of frequency-modulating jitter on higher-order moments of sound residual. In *Proceedings of Int. Computer Music Conference*, 1996.

[Dui98] R. P. W. Duin. Relational discriminant analysis and its large sample size problem. *Proceedings of ICPR'98, Brisbane (Australia)*, 1998.

[Dui00] R.P.W. Duin. Prtools3.0, pattern recognition toolbox for matlab. ftp://ftp.ph.tn.tudelft.nl/pub/bob/prtools/, January 2000.

[EAS98] A. Edelman, T. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. on Matrix Analysis and Applications*, 20:303 – 353, 1998.

[Epp91] I. K. Epps. *An investigation into vibrations excited by discrete faults in rolling element bearings*. PhD thesis, University of Canterbury, New Zealand, 1991.

[FD99] C. R. Farrar and S. W. Doebling. Structural health monitoring at los alamos national lab. In *Proc. of IEE Colloquium on Condition monitoring of machinery, external structures and health, April 22-23*, pages 2/1–2/4, Birmingham (UK), 1999.

[FFD96] G. A. P. Fontaine, E. E. E Frietman, and R. P. W. Duin. Preventive and predictive maintenance

using neural networks. *J. Microelectric systems integration*, Volume 4, No. 2, 1996.

[FKFW91] E. E. E. Frietman, E. J. H. Kerckhoffs, and F. W. F.W. Wedman. Guarding vibration patterns of a mechanical assembly with the help of neural networks. In *Proceedings of ESM'91, Copenhagen (Denmark)*, pages 279–287, 1991.

[Fri87] J. H. Friedman. Exploratory projection pursuit. *Journal of the American Statistical Association*, 82(397):249 – 266, March 1987.

[FS99] P. Fabry and Ch. Servière. Blind separation of noisy harmonic signals using only second-order statistics. In *Proc. of ICECS'99*, Cypress, Sept. 1999.

[FSL98] P. Fabry, Ch. Servière, and J. L. Lacoume. Improving signal subspace estimation and source number detection in the context of spatially correlated noises. In *Proceedings of EUSIPCO'98*, 1998.

[GBD92] S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias-variance dilemma. *Neural computation*, (4):1 – 58, 1992.

[GCD99] G. Gelle, M. Colas, and G. Delaunay. Separation of convolutive mixtures of harmonic signals with a temporal approach. application to rotating machine monitoring. In *Proceedings of ICA'99*, 1999.

[Gel98] G. Gelle. *Les statistiques d'ordre superieur appliquees a la detection et a la separation de sources. Utilisation en analyse vibratoire et acoustique*. PhD thesis, Universite de Reims Champagne-Ardenne, 1998.

[GF97] M. Girolami and C. Fyfe. Extraction of independent signal sources using a deflationary exploratory projection pursuit network with lateral inhibition. *IEE Proceedings on vision, image and signal processing journal*, 14(5):299 – 306, 1997.

[Gha97] Z. Ghahramani. Learning dynamic bayesian networks. In C.L. Giles and M. Gori, editors, *Adaptive processing of temporal information*, 1997.

[GLW99] I. D. Guedala, M. London, and M. Werman. An on-line agglomerative clustering method for nonstationary data. *Neural computation*, 11:521 – 540, 1999.

[Grü98] P. Grünwald. *The minimum description length principle*. PhD thesis, University of Amsterdam, 1998.

[GS99] S. Gaffney and P. Smyth. Trajectory clustering with mixtures of regression models. Technical report, University of California, Irvine, 1999.

[Hay94] S. Haykin. *Neural networks, a comprehensive foundation*. Macmillan College Publishing Company, Inc., New York, NY, USA, 1994.

[Hec91] L. P. Heck. *A subspace approach to the automatic design of pattern recognition systems for mechanical system monitoring*. PhD thesis, Georgia ITech, 1991.

[Hec96] D. Heckerman. A tutorial on learning with bayesian networks. Technical report, Microsoft Research, Redmond, 1996.

[HKd$^+$] A. Hoekstra, M.A. Kraaijveld, de Ridder, D., W.F. Schmidt, and A. Ypma. *The complete SPRlib and ANNlib, version 3.1*. Pattern recognition group, TU Delft.

[HKO01] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley, 2001.

[HO97] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural computation*, 9(7):1483–1492, 1997.

[Hoe99] A. Hoekstra. *Generalization in feedforward neural classifiers*. PhD thesis, Pattern recognition group, TU Delft, 1999.

[HP86] Hewlett-Packard. The fundamentals of modal testing - application note no. 243-3. Technical report, Hewlett-Packard, 1986.

[HSW89] M. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural networks*, (2):359 – 366, 1989.

[Hyv99] A. Hyvärinen. Survey on independent component analysis. *Neural computing surveys*, (2):94 – 128, 1999.

[Hyv00] A. Hyvärinen. Complexity pursuit: Separating interesting components from time-series. In *Proceedings of ICA-2000*, Helsinki, 2000.

[IT00] S. Ikeda and K. Toyama. Independent component analysis for noisy data–meg data analysis. *Neural networks*, 13(10):1063 – 1074, 2000.

[JHB+95] A. Juditsky, H. Hjalmarsson, A. Benveniste, B. Delyon, L. Ljung, J. Sjöberg, and Q. Zhang. Nonlinear black-box models in system identification: mathematical foundations. Technical report, Technical Report IRISA/ INRIA, june 1995.

[JMH+00] T. P. Jung, S. Makeig, C. Humphries, T.-W. Lee, M. J. McKeown, V. Iragui, and T. J. Sejnowski. Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, (37):163 – 178, 2000.

[Kan94] J. Kangas. *On the analysis of pattern sequences by Self-Organizing Maps*. PhD thesis, Helsinki University of Technology, 1994.

[Kas97] S. Kaski. *Data exploration using Self-Organizing Maps*. PhD thesis, Helsinki University of Technology, 1997.

[KB97] M. Kuitert and L. Boves. Speaker verification with gsm coded telephone speech. In *Proc. of EuroSpeech'97*, 1997.

[KDBU94] M. S. Kompella, P. Davies, R. J. Bernhard, and D. A. Ufford. A technique to determine the number of incoherent sources contributing to the response of a system. *Mechanical systems and signal processing*, 8(4):363–380, 1994.

[Kei00] W. Keizer. Responsie analyse van een pomphuis ten behoeve van patroonherkenning. Technical report, RND Mechanical engineering Delft, August 2000.

[Kir96] I. Kirk. Neural networks for pipeline leak detection. Technical report, AEA-Commercial, 1996.

[KKL97] T. Kohonen, S. Kaski, and H. Lappalainen. Self-organized formation of various invariant-feature filters in the adaptive subspace som. *Neural computation*, 1997.

[KL96] S. Kaski and K. Lagus. Comparing self-organizing maps. In *Proc. of ICANN '96*, volume 1112 of *Lecture notes in computer science*, pages 809–814. Springer, 1996.

[KMK95] M. A. Kraaijveld, J. Mao, and Jain A. K. A nonlinear projection method based on Kohonen's topology preserving maps. *IEEE Trans. on neural networks*, Volume 6, No. 3:548–559, May 1995.

[KO98] B.-U. Koehler and R. Orglmeister. ICA of electroencephalographic data using wavelet decomposition. In *Proc. of VIII Mediterranian Conference on MBEC*, Cyprus, 1998.

[KO99] B.-U. Koehler and R. Orglmeister. Ica using autoregressive models. In *Proc. of ICA99*, pages 359 – 363, Aussois (France), 1999.

[KO00] B.-U. Koehler and R. Orglmeister. A blind source separation algorithm using weighted time delays. In *Proc. of ICA2000*, pages 471 – 475, Helsinki, 2000.

[KOBF98] B.-U. Koehler, R. Orglmeister, and B. Brehmeier-Flick. Independent component analysis of eeg data using matched filters. In *Proc. of EMBS98*, Hong Kong, 1998.

[Koh95] T. Kohonen. *Self-Organizing Maps*. Springer, 1995.

[KOW+97] J. Karhunen, E. Oja, L. Wang, R. Vigário, and J. Joutsensalo. A class of neural networks for independent component analysis. *IEEE Trans. on neural networks*, 8(3):486 – 504, May 1997.

[KT97] W. A. Kuperman and G. Turek. Matched field acoustics. *Mechanical systems and signal processing*, 11(1):141 – 148, 1997.

[KW98] X. Kong and G. Wilson. A new eog-based eyeblink detection algorithm. *Behavior Research Methods, Instruments & Computers*, 30(4):713 – 719, 1998.

[Lac99] J.L. Lacoume. A survey of source separation. In *Proc. of ICA'99*, Aussois, 1999.

[LB97] P. J. Loughlin and G. D. Bernard. Cohen-Posch (positive) time-frequency distributions and their application to machine vibration analysis. *Mechanical systems and signal processing*, 11(4):561 – 576, 1997.

[Les99] A. Leshem. Source separation using bilinear forms. In *Proceedings of the 8th Int. Conference on Higher-Order Statistical Signal Processing*, 1999.

[LGS99] T.-W. Lee, M. Girolami, and T. J. Sejnowski. ICA using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural computation*, 11(2):409 – 433, 1999.

[LS97] L. D. Lutes and S. Sarkani. *Stochastic analysis of structural and mechanical vibrations*. Prentice-

Hall, New Jersey, 1997.

[LSL93] K. N. Lou, P. J. Sherman, and D. E. Lyon. System identification and coherence analysis in the presence of a harmonic signal. *Mechanical systems and signal processing*, 7(1):13 – 27, 1993.

[LV93] M. Li and P. Vitányi. *Kolmogorov complexity and its applications*. Springer, 1993.

[LW93] G. Liang and D. M. Wilkes. Arma model order estimation based on the eigenvalues of the covariance matrix. *IEEE Trans. on signal processing*, 41(10):3003 – 3009, 1993.

[LYDF97] R. Ligteringen, A. Ypma, R. P. W. Duin, and E. E. E. Frietman. Fault diagnosis of rotating machinery by neural networks. In Kappen, B. and Gielen, S., editors, *Neural networks: best practice in Europe - proceedings of the SNN Conference 1997, Amsterdam*, volume 1, pages 157 – 160. World scientific, 1997.

[LYFD97] R. Ligteringen, A. Ypma, E. E. E. Frietman, and R. P. W. Duin. Machine diagnostics by neural networks, experimental setup. In Bal, H. E., Corporaal, H., Jonker, P. P., and Tonino, J. F. M., editors, *Proc. of ASCI'97*, volume 1, pages 185 – 190. Advanced School for Computing and Imaging - TWI-TUD, Delft, The Netherlands, 1997.

[Lyo87] R. H. Lyon. *Machinery noise and diagnostics*. Butterworths, Boston, 1987.

[McC98] A. C. McCormick. *Cyclostationarity and higher-order statistical signal processing for machine condition monitoring*. PhD thesis, University of Strathclyde (UK), 1998.

[Mel00a] A. Melis. Adaptive one-class classification. Master's thesis, Pattern recognition group, TU Delft, 2000.

[Mel00b] C. Melissant. Detecting alzheimer's disease using quantitative eeg analysis. Master's thesis, Pattern recognition group, TU Delft, October 2000.

[Men91] J. M. Mendel. Tutorial on higher-order statistics (spectra) in signal processing and system theory: theoretical results and some applications. *Proceedings of the IEEE*, 79(3):278 – 305, March 1991.

[MH99] T. Marwala and H. E. M. Hunt. Fault identification using finite element models and neural networks. *Mechanical systems and signal processing*, 13(3):475 – 490, 1999.

[Mit93] J. S. Mitchell. *An introduction to machinery analysis and monitoring - 2nd ed.* PennWell Publ. Comp., 1993.

[Mit97] T. Mitchell. *Machine learning*. McGraw-Hill, 1997.

[MIZ99] N. Murata, S. Ikeda, and A. Ziehe. An approach to bss based on temporal structure of speech signals. *IEEE Trans. on Signal Processing*, 1999.

[ML95] D. J. McCarthy and R. H. Lyon. Recovery of impact signatures in machine structures. *Mechanical systems and signal processing*, 9(5):465 – 488, 1995.

[ML96] J. Ma and C. J. Li. Gear defect detection through model-based wideband demodulation of vibrations. *Mechanical systems and signal processing*, 10(5), 653 - 665 1996.

[MM92] C. K. Mechefske and J. Mathew. Fault detection and diagnosis in low speed rolling element bearings - part i: the use of parametric spectra. *Mechanical systems and signal processing*, 6(4):297 – 307, 1992.

[MM98] A. Munoz and J. Muruzabal. Self-organizing maps for outlier detection. *Neurocomputing*, pages 33 – 60, 1998.

[MP97] A. Murray and J. Penman. Extracting useful higher order features for condition monitoring using artificial neural networks. *IEEE Trans. on signal processing*, 45(11):2821 – 2828, November 1997.

[MP99] C. Mejuto and J. C. Principe. A second-order method for blind source separation of convolutive mixtures. In *Proceedings of ICA'99*, 1999.

[MPZ99] K.-R. Müller, P. Philips, and A. Ziehe. Jadetd: Combining higher-order statistics and temporal information for blind source separation (with noise). In *Proceedings of ICA'99*, pages 87 – 92, 1999.

[MS84] P. D. McFadden and J. D. Smith. Model for the vibration produced by a single point defect in a rolling element bearing. *Mechanical systems and signal processing*, 96(1):69 – 82, 1984.

[MS94a] T. Martinetz and K. Schulten. Topology representing networks. *Neural networks*, Volume 7, No.

3:507–522, 1994.

[MS94b] L. Molgedey and H. G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Physical review letters*, 72(23), June 1994.

[MYFS02] C. Melissant, A. Ypma, E. E. E. Frietman, and C. J. Stam. A method for detection of Alzheimer's disease using ICA-enhanced EEG measurements. *submitted to Artificial Intelligence in Medicine*, 2002.

[NC97] H. G. Natke and C. Cempel. *Model-aided diagnosis of mechanical system*. Springer - Verlag, Berlin - Heidelberg, 1997.

[NTJ95] H.L. Nguyen Thi and Ch. Jutten. Blind sources separation for convolutive mixtures. *Signal Processing*, 45(2):209 – 229, August 1995.

[oCI] Technical Associates of Charlotte Inc. Criteria for overall condition rating.

[Oeh96] H. Oehlmann. *Analyse temps-fréquence de signaux vibratoires de boîtes de vitesses*. PhD thesis, Université H. Poincaré, C. de Recherche en Automatique, Nancy, 1996.

[OWY83] A. V. Oppenheim, A. S. Willsky, and I. T. Young. *Signals and systems*. Prentice Hall Signal processing series. Prentice Hall International Editions, USA, 1983.

[Paj98a] P. Pajunen. Blind source separation using algorithmic information theory. *Neurocomputing*, 22:35–48, 1998.

[Paj98b] P. Pajunen. *Extensions of linear Independent Component Analysis: neural and information-theoretic methods*. PhD thesis, Laboratory of computer and information science, Helsinki University of Technology, 1998.

[Paj99] P. Pajunen. Blind source separation of natural signals based on approximate complexity minimization. In *Proceedings of ICA'99, Aussois (Fr.)*, 1999.

[PJKS95] P. Pajunen, J. Joutsensalo, J. Karhunen, and K. Saarinen. Estimation of equispaced sinusoids using maximum likelihood method. In *Proc. of the 1995 Finnish Signal Processing Symposium (FINSIG'95), TU Helsinki*, pages 128–132, June 1995.

[PLTU99] R.J. Patton, C. J. Lopez-Toribio, and F. J. Uppal. Ai approaches to fault diagnosis. In *Proc. of IEE Colloquium on Condition monitoring of machinery, external structures and health, April 22 - 23*, pages 5/1– 5/18, Birmingham (UK), 1999.

[PM92] J.G. Proakis and D.G. Manolakis. *Digital signal processing - principles, algorithms and applications, 2nd ed.* MacMillan Publ., New York, 1992.

[PP96] B.A. Pearlmutter and L.C. Parra. A context-sensitive generalization of ICA. In *Proceedings of ICONIP'96*, 1996.

[PR98] W. D. Penny and S. J. Roberts. Dynamic models for nonstationary signal segmentation. Technical report, Imperial College of Science, London, 1998.

[PS99] L. Parra and C. Spence. Convolutive blind separation of nonstationary sources. *IEEE Trans. on Speech and Audio Processing*, 1999.

[PTVF92] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C - 2nd ed.* Cambridge University Press, Cambridge, NY, USA, 1992.

[Ran87] R. B. Randall. *Frequency analysis*. Brüel & Kjær, Denmark, 1987.

[RG99] S. Roweis and Z. Ghahramani. A unifying review of linear gaussian models. *Neural computation*, 11(2):305 – 345, 1999.

[RJ93] Rabiner and Juang. *Fundamentals of speech processing*. Prentice Hall, 1993.

[RR00] K. Rahbar and J. P. Reilly. Geometric optimization methods for blind source separation of signals. In *Proceedings of ICA'2000*, pages 375 – 380, 2000.

[Sch97] B. Schölkopf. *Support vector learning*. PhD thesis, TU Berlin, 1997.

[SCL97] Ch. Servière, V. Capdevielle, and J.L. Lacoume. Separation of sinusoidal sources. In *1997 IEEE Signal Processing Workshop on Higher-Order Statistics*, 1997.

[Sea96] C. J. Stam et al. Use of non-linear eeg measures to characterize eeg changes during mental activity. *Electroencephalography and clinical Neurophysiology*, 99:214–224, 1996.

[Sku01] M. Skurichina. *Stabilizing weak classifiers*. PhD thesis, Pattern recognition group, TU Delft, The Netherlands, 2001.

[SL86] M. Serridge and T.R. Licht. *Piezoelectric Accelerometers*. Brüel & Kjær, 1986.

[Sma98] P. Smaragdis. Blind separation of convolved mixtures in the frequency domain. In *Int. Workshop on Independence & Artificial Neural Networks*, Tenerife, 1998.

[SMN98] A. Swami, J. M. Mendel, and C. L. Nikias. Higher-order spectral analysis toolbox manual. The Mathworks, 1998.

[Smy94] P. Smyth. Markov monitoring with unknown states. *IEEE Journal on Selected Areas in Communications*, December 1994.

[Smy97] P. Smyth. Clustering sequences with hidden markov models. In M.C. Mozer, M.I. Jordan, and T. Petsche, editors, *NIPS 9*, 1997.

[SN96] G. Strang and T. Nguyen. *Wavelets and filter banks*. Wesley-Cambridge Press, Wellesley MA, USA, 1996.

[Spe99] O. Speekenbrink. Signal processing methods for speaker recognition. Master's thesis, Pattern recognition group, TU Delft, March 1999.

[SPP98] C. J. Stam, J. P. M. Pijn, and W. S. Pritchard. Reliable detection of non-linearity in experimental time series with strong periodic components. *Electroencephalography and clinical Neurophysiology*, 112:361 – 380, 1998.

[SSM+99] I. Schiessl, M. Stetter, J. E. W. Mayhew, S. Askew, N. McLoughlin, J. B. Levitt, J. S. Lund, and K. Obermayer. Blind separation of spatial signal patterns from optical imaging records. In *Proceedings of ICA'99*, pages 179 – 184, 1999.

[Sta00] M. Staroswiecki. Quantitative and qualitative models for fault detection and isolation. *Mechanical systems and signal processing*, 14(3):301 – 325, 2000.

[Sv96] H. Smit and Th. van Zanten. Course on machine condition monitoring. Technical report, Brüel & Kjær Condition Monitoring Nederland B.V., The Netherlands, 1996.

[SWT97] W. J. Staszewski, K. Worden, and G. R. Tomlinson. Time-frequency analysis in gearbox fault detection using the wigner-ville distribution and pattern recognition. *Mechanical systems and signal processing*, 11(5):673 – 692, 1997.

[SYD00] M. Skurichina, A. Ypma, and R.P.W. Duin. The role of subclasses in machine diagnostics. In *Proceedings of ICPR-2001*, pages 668–771, 2000.

[Tak94] F. Takens. *Chaos and time series analysis*. Dept. of Mathematics, University of Groningen (NL), 1994.

[Tax01] D. M. J. Tax. *One-class classification*. PhD thesis, Pattern recognition group, TU Delft, The Netherlands, 2001.

[TD99] D.M.J. Tax and R.P.W Duin. Data domain description using support vectors. In Verleysen, M., editor, *Proceedings of the European Symposium on Artificial Neural Networks 1999*, pages 251 – 256, Brussels, April 1999. D-Facto.

[TH95] P. Thevenaz and H. Hugli. Usefulness of the LPC-residue in text-independent speaker verification. *Speech Communication*, 17(1-2), 1995.

[TNTC99] L. Tarassenko, A. Nairac, N. Townsend, and P. Cowley. Novelty detection in jet engines. In *Proc. of IEE Colloquium on Condition monitoring of machinery, external structures and health, April 22 - 23*, pages 4/1– 4/5, Birmingham (UK), 1999.

[Tow91] D. P. Townsend. *Dudley's gear handbook*. McGraw-Hill, Inc., 1991.

[TYD98] D.M.J. Tax, A. Ypma, and R.P.W. Duin. A neural network based system for gas leak detection with hydrophone measurements. Technical report, research project for Shell Expro UK, 1998.

[TYD99a] D. M.J. Tax, A. Ypma, and R. P. W. Duin. Support vector data description applied to machine vibration analysis. Proceedings of ASCI'99, 1999.

[TYD99b] D.M.J. Tax, A. Ypma, and R.P.W. Duin. Analysis and improvement of a neural network based system for gas leak detection. Technical report, research project for Shell Expro UK, 1999.

[TYD99c] D.M.J. Tax, A. Ypma, and R.P.W. Duin. Pump failure detection using support vector data descriptions. In *Proceedings of Third Symposium on Intelligent Data Analysis IDA'99*, Amsterdam, 1999.

[Ult93] A. Ultsch. Self organized feature maps for monitoring and knowledge acquisition of a chemical

process. In S. Gielen and B. Kappen, editors, *Proc. ICANN'93*, pages 864 – 867, London (UK), 1993. Springer.

[Vap95] V. Vapnik. *The nature of statistical learning theory*. Springer, New York, 1995.

[Vap98] V. Vapnik. *Statistical learning theory*. Wiley, New York, 1998.

[VDM94] T. Villmann, R. Der, and T. Martinetz. A new quantitative measure of topology preservation in Kohonen's feature maps. In *ICNN'94 Proceedings*, pages 645–648. IEEE Service Center, Piscataway, NJ, 1994.

[vdV98] A.J. van der Veen. Algebraic methods for deterministic blind beamforming. *Proceedings of the IEEE*, 86(10):1987 – 2008, October 1998.

[vdVP96] A.-J. van der Veen and A. Paulraj. An analytical constant modulus algorithm. *IEEE Trans. Signal Processing*, 44(5):1136–1155, May 1996.

[Vig97] R. N. Vigário. Extraction of ocular artifacts from eeg using independent component analysis. *Electroencephalography and Clinical Neurophysiology*, 103:395 – 404, 1997.

[VL99] P. Vitányi and M. Li. Minimum description length induction, bayesianism and Kolmogorov complexity. *IEEE Trans. on Information theory*, pages 1 – 35, 1999.

[WG94] A. S. Weigend and N. A. Gershenfeld, editors. *Time series prediction*, Santa Fe Institute studies in the sciences of complexity, 1994.

[WK85] M. Wax and T. Kailath. Detection of signals by information theoretic criteria. *IEEE Trans. on acoustics, speech and signal processing*, 33:387 – 392, 1985.

[WM96] W. J. Wang and P. D. McFadden. Application of wavelets to gearbox vibration signals for fault detection. *Sound and Vibration*, Volume 192, No. 5:927–939, 1996.

[YA97] H. H. Yang and S. Amari. Adaptive online learning algorithms for blind separation: max. entropy and min. mutual information. *Neural computation*, 9:1457–1482, 1997.

[YBJMD01] A. Ypma, O. Baunbæk-Jensen, C. Melissant, and R.P.W. Duin. Health monitoring with learning methods. In *Proc. of ICANN'01, Vienna, August 22-24*, 2001.

[YD97a] A. Ypma and R. P. W. Duin. Novelty detection using self-organizing maps. In Kasabov, N., Kozma, R., Ko, K., O'Shea, R., Coghill, G., and Gedeon, T., editors, *Progress in connectionist-based information systems*, volume 2, pages 1322–1325. Springer-Verlag Singapore, 1997.

[YD97b] A. Ypma and R. P. W. Duin. Using the wavenet for function approximation. In Bal, H. E., Corporaal, H., Jonker, P. P., and Tonino, J. F. M., editors, *Proceedings of 3rd annual conference of ASCI, ASCI'97*, volume 1, pages 236 – 240. Advanced School for Computing and Imaging - TWI-TUD, Delft, The Netherlands, 1997.

[YD98] A. Ypma and R.P.W. Duin. Support objects for domain approximation. In *Proceedings of ICANN98*, pages 719 – 724, 1998.

[Yer00] A. Yeredor. Approximate joint diagonalization using non-orthogonal matrices. In *Proc. of ICA2000*, pages 33 – 38, Helsinki, 2000.

[YK01] A. Ypma and H. Koopmans. Automatic fault detection in rotating equipment with neural networks (in Dutch). *Drive and control*, pages 32 – 36, 2001.

[YKVD01] A. Ypma, R. J. Kleiman, J. Valk, and R. P. W. Duin. MONISOM - a system for machine health monitoring with neural networks. In *Proceedings of 13th Belgian-Dutch Conference on Artificial Intelligence BNAIC'01, October 25-26, Amsterdam*, 2001.

[YL00] A. Ypma and A. Leshem. Blind separation of machine vibration with bilinear forms. In *Proceedings of ICA-2000*, pages 405 – 410, Helsinki, June 2000.

[YLD02] A. Ypma, A. Leshem, and R. P. W. Duin. Blind separation of rotating machine sources: bilinear forms and convolutive mixtures. *accepted for Neurocomputing, Special Issue on ICA/BSS*, 2002.

[YLFD97] A. Ypma, R. Ligteringen, E. E. E. Frietman, and R. P. W. Duin. Recognition of bearing failures using wavelets and neural networks. In *Proc. of TFTS'97*, pages 69–72. University of Warwick, Coventry (UK), 1997.

[YP98] A. Ypma and P. Pajunen. Second-order ICA in machine vibration analysis. Technical report, Lab. of Computer and Information Science, TU Helsinki, 1998.

[YP99] A. Ypma and P. Pajunen. Rotating machine vibration analysis with second-order independent

component analysis. In *Proc. of ICA'99, Aussois*, pages 37–42, January 1999.

[YPD98] A. Ypma, E. Pekalska, and R.P.W. Duin. Domain approximation for condition monitoring. In *Proceedings of ASCI'98*, 1998.

[Ypm99a] A. Ypma. Creating virtual sensors for machine monitoring with independent component analysis (in Dutch). *NAG journaal*, (145):3 – 12, March 1999.

[Ypm99b] A. Ypma. Methods for blind source separation applied to airport noise surveillance. Technical report, research project for TNO-TPD, TU Delft, 1999.

[Ypm00] A. Ypma. Threshold determination in gearbox monitoring using hidden markov models. Technical report, research project for SKF Engineering and Research Centre, Nieuwegein, The Netherlands, 2000.

[YSD01] A. Ypma, O. Speekenbrink, and R. P. W. Duin. The use of context in dynamic pattern recognition applications. In *Proceedings of 13th Belgian-Dutch Conference on Artificial Intelligence BNAIC'01, October 25-26, Amsterdam*, 2001.

[YTD97] A. Ypma, D. M. J. Tax, and R. P. W. Duin. Between ship noise and gas leak: reliable gas leak detection. In Kappen, B. and Gielen, S., editors, *Neural networks: best practice in Europe - proceedings of the SNN Conference 1997, Amsterdam*, volume 1, pages 161 – 164. World scientific, 1997.

[YTD99] A. Ypma, D. M. J. Tax, and R. P. W. Duin. Robust machine fault detection with independent component analysis and support vector data description. In *Proceedings of 1999 IEEE Int. Workshop on Neural Networks for Signal Processing*, pages 67 – 76, Madison, Wisconsin (USA), 1999.

[YW96] D. Yellin and E. Weinstein. Multichannel signal separation: methods and analysis. *IEEE Trans. on Signal Processing*, 44(1):106 – 118, 1996.

[Zat98] M Zatman. How narrow is narrowband? *IEE Proceedings F*, 145(2):85 – 91, 1998.

[ZB92] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Trans. Neural Networks*, Volume 3, No. 6:889 – 898, 1992.

[Zha93] Q. Zhang. Regressor selection and wavelet network construction. Technical report, Technical Report No. 709, IRISA/ INRIA, april 1993.

[ZM98] A. Ziehe and K.-R. Müller. TDSEP-an efficient algorithm for bss using time structure. In *Proc. of ICANN98*, volume 2, pages 675–680. Springer, 1998.

[ZNCM00] A. Ziehe, G. Nolte, G. Curio, and K.-R. Müller. Optimal filtering algorithms for source separation. In *Proc. of ICA2000*, pages 127 – 132, Helsinki, 2000.

[Zre93] S. Zrehen. Analyzing Kohonen maps with geometry. In Gielen, S. and Kappen, B., editors, *ICANN '93 Proceedings*, pages 609–612. Springer-Verlag, 1993.