Approaches to analyse and interpret biological profile data

Dissertation

zur Erlangung des akademischen Grades
doctor rerum naturalium
– Dr. rer. nat. –

eingereicht an der
Mathematisch-Naturwissenschaftlichen Fakultät
der Universität Potsdam

von

Matthias Scholz

Arbeitsgruppe Bioinformatik
Max-Planck-Institut für Molekulare Pflanzenphysiologie

Potsdam, im Januar 2006

Approaches to analyse and interpret biological profile data

Matthias Scholz
POTSDAM UNIVERSITY
January 2006 · Potsdam · Germany

Summary

This thesis deals with the analysis of large-scale molecular data. It is focused on the identification of biologically meaningful components and explains the potentials of such analyses to gain deeper insight into biological issues. Many aspects are discussed including component search criteria to obtain the major information in the data and interpretation of components.
The first chapter provides an introduction to the concepts of component extraction and beyond. Starting with a biological motivation for component extraction and the problems to identify ideal ones, it introduces many of the central ideas, such as criteria to find highly informative components and the benefit of component analysis to discover relations among molecules and the impact of experimental factors, which will be discussed at greater length in later chapters of this work.
Chapter two deals with the problem of normalisation and its importance to large-scale data from molecular biology.
Classical *principal component analysis (PCA)* is reviewed in chapter three. It is described how PCA is applicable to large-scale data and the impact of prior data normalisation is discussed. This chapter also gives an overview of the most important algorithms for PCA, and discusses their benefits and drawbacks. Both chapter two and chapter three are based on Scholz and Selbig (2006).
Chapter four introduces *independent component analysis (ICA)*. Although non-correlation in PCA is to some extent reasonable, it is shown that the independence condition of ICA is more suitable for the purpose of analysing molecular data. This is particularly important for the problem of multiple distinct factors that impact the observed data. A specific procedure for ICA is proposed, which is applicable to large-scale molecular data, and was successfully applied to real experimental data in Scholz et al. (2004a,b).
Chapter five provides a comprehensive treatment of the nonlinear generalisation of PCA. It considers essentially nonlinear dynamics in time experiments which require more complex nonlinear components. The potentials of such nonlinear PCA (NLPCA) for identifying and analysing nonlinear molecular behaviour are demonstrated by a cold stress experiment of the model plant *Arabidopsis thaliana*. For that purpose, new approaches to validation and missing data handling are proposed. Nonlinear PCA is adapted to be applicable to incomplete data. This also provides the ability to estimate missing values, a valuable property for validating the model complexity. The chapter contains material of Scholz and Vigário (2002) and Scholz et al. (2005).

The final chapter is based on the idea of visualising molecular dynamics by integrating functional dependencies into molecular network representations. A new network model, denoted as *functional network*, is proposed. It provides a framework to integrate results of component analysis as similarity or distance information in molecular networks. The advantage over classical network analysis which traditionally is based on pair-wise similarity measures and static relations, is discussed extensively. The potentials of functional networks to reveal dynamics in molecular systems are demonstrated by generating a network that visualises the adaptation of *Arabidopsis thaliana* to cold stress.

Key words: bioinformatics, molecular data analysis, PCA, ICA, nonlinear PCA, missing data, auto-associative neural networks, validation, inverse problems, molecular networks

Acknowledgements

I wish to express my considerable gratitude to the many people who have helped me with the work presented in this thesis. First and foremost I would like to thank Professor Joachim Selbig for his guidance and advice.
The work of this dissertation has been done at the Max Planck Institute of Molecular Plant Physiology, Potsdam, in collaboration with the University of Potsdam.
Among the many people at those institutes, I would particularly like to thank Joachim Kopka for providing valuable insight into the biological and technical issues behind molecular experiments. I wish to thank Wolfram Weckwerth and Oliver Fiehn for stimulating discussions which have particularly influenced the direction of my work. Furthermore, the comments and ideas of Mark Stitt were of valuable help. I very much appreciated the discussions with Ralf Steuer on the nature of biophysics.
I wish to thank all current and former members and guests of our Bioinformatics group for many helpful discussions and support including Petra Birth, Sven Borngräber, Roman Brunnemann, Carsten Daub, Susanne Grell, Jan Hannemann, Stefanie Hartmann, Peter Humburg, Jan Hummel, Peter Krüger, Jan Lisec, Henning Redestig, Dirk Repsilber, Joachim Selbig, Wolfram Stacklies, Matthias Steinfath, Danny Tomuschat, Dirk Walther, and Daniel Weicht.
Notably, I would like to thank my colleagues for carefully reading parts of this work: Gareth Catchpole, John Lunn, Joachim Selbig, and Dirk Walther. Of cause, all errors and misinterpretations still remain to me.
Several other people contributed to this work in one way or another. In particular, I thank the co-authors of my publications, Oliver Fiehn, Stephan Gatzek, Yves Gibon, Charles L. Guy, Fatma Kaplan, Joachim Kopka, Katja Morgenthal, Joachim Selbig, Alistair Sterling, Mark Stitt, and Wolfram Weckwerth for fruitful collaborations.
Finally, I would like to thank the Max Planck Society and the University of Potsdam for their support.

Matthias Scholz

Contents

1 Introduction

Advances in high-throughput technologies have led to a rapidly increased number of simultaneously measured expression levels of various genes as well as concentration levels of metabolites or proteins. Thus, there is a great demand for methods for analysing such profile data of many variables.
A research field where such high-dimensional data are well-known is the field of machine-learning. Here, many modern techniques were developed for image or signal analysis. It is therefore not surprising that many of the machine-learning techniques became important in bioinformatics. Major emphasis is given on the task of an integrative functional analysis driven by the availability of large-scale experimental data sets of activation or concentration values from various molecular levels: the transcriptome, the metabolome and the proteome.

1.1 Biological motivation

The primary motivation for studying biological systems, the cell or the entire organism, is to get a better understanding of complex molecular processes. Up to now, there is only little knowledge about the functions of genes, their protein products or gene-metabolite relations. To gain more information about the interplay between molecules of a cell, to determine the molecular network, and to learn more about molecular responses to changed environments is, therefore, one of the great challenges. Plants have a large ability to adapt to adverse environmental circumstances. The sensory mechanisms used to cope with stress conditions as well as survival strategies for adaptations can therefore be well investigated on the model plant *Arabidopsis thaliana* (The Arabidopsis Genome Initiative, 2000).
Many experiments were designed to investigate molecular responses under different environmental conditions such as day and night, cold stress, and temporal courses, or variations of different genotypes. The output of these experiments are usually large data matrices of many measured variables for different biological samples. In the beginning, research was mainly focused on expression levels of genes usually measured by microarrays (DNA chips). Though, due to advances in chromatographic mass spectrometry, concentration levels of metabolites or proteins also became available. This makes it possible for an integrative analysis of the different molecular levels to provide a comprehensive insight into the molecular system. Such an integrative analysis is important, since – as the molecules of a cell usually interact strongly between molecular levels – a single level view would be very restrictive. To analyse the responses at several molecular levels simultaneously is, therefore, very advantageous.

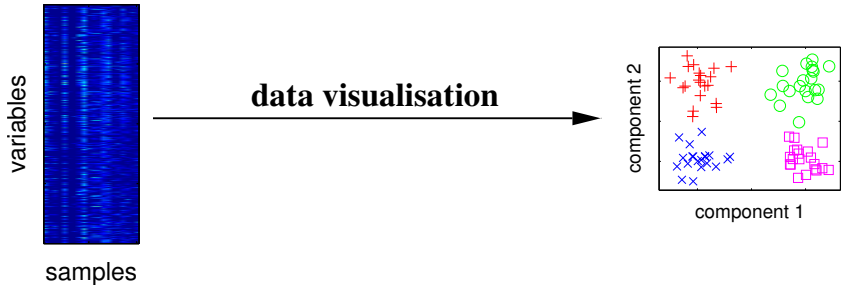


Figure 1.1: Visualising the major characteristics of high-dimensional data is helpful to understand how molecular data reflect the investigated experimental conditions. The large number of variables is given by genes, metabolites or proteins measured for different biological samples. On the right, a visualisation of samples from different experimental conditions is illustrated.

The attempt is made to answer several biological questions by the analysis of such data. It goes from simply confirming the expected molecular response up to identifying potential candidates involved in a biological process with the final objective of identifying potential molecular interactions.
Even though this work is mainly focused on the model plant *Arabidopsis thaliana*, the proposed algorithms can be applied to molecular data from any other organism as well.

Explorative analysis

First, we would generally like to know whether the experimental conditions are reflected in the data. Is there a biological response and can we measure it with today's technologies? Furthermore, we would like to know how well the experimental conditions are reflected and whether there is other unexpected information, biological or technical artifacts or simply undirected noise. Basically, it is the simple but nevertheless essential question of what we really measure. The purpose of such primary analysis is to identify and present all information contained in the data set. This kind of investigative or explorative analysis can be well achieved by data visualisation methods in an unsupervised manner. For example, as illustrated in Figure 1.1, by reducing the dimensionality to two dimensions and plotting them against each other. *Unsupervised* means that the experimental knowledge (e.g., group categories of samples) is not involved in the analysis, and so the information provided by unsupervised methods is extracted independently from this knowledge. Unsupervised analysis can confirm our expectations or even discover unexpected biological or technical factors (Scholz et al., 2004a). It can also lead to refined definitions of categories (Golub et al., 1999).
In contrast, supervised methods are less suitable for such exploratory analysis. Supervised analysis is targeted on group categories (class labels), the main purpose is to predict

the unknown category of a new sample with high accuracy. This can be very useful in final applications such as routine diagnostic tasks. In primarily investigative analyses, however, unsupervised methods often match better our interests in both discovering new biological knowledge and improving the experimental design, e.g., by detecting and eliminating technical artifacts.

Identifying candidates

Once we have detected a data response to an investigated physiological function, the next step would be to identify the respective genes, metabolites or proteins involved in this biological factor. The aim is therefore to identify the most likely candidates and to present them in a list, ranked by their plausibility. The top candidates of such a list are supposed to be highly responsible for the considered physiological function and thus might be functionally similar or even interact with each other. This can be first validated with known functional annotations and known metabolic pathways from the literature. Candidates that are not characterised in literature, however, can be investigated by a next round of experiments to validate the influence on a specific physiological function and the interaction with other molecules in the cell. This can be done, for example, by knocking out a specific candidate gene and observing the effects on the molecular system.

1.2 Component identification

How are biological or technical factors represented in the data? And how can we detect and explain them? A well suited approach to this question is to decompose the data into meaningful components (sometimes also termed features, factors or sources). Our emphasis here is on unsupervised approaches often referred to as *blind decomposition*.
A *component* describes a new variable generated by a weighted combination of all original variables (e.g., genes). Such a component explains a straight line or a curve embedded in the data space, as illustrated in Figure 1.2. The shape and orientation of this line or curve mainly depend on both the data and the optimised criterion, e.g., variance or an information criterion. Components are useful to visualise the characteristics of the usually high-dimensional data.
Ideally, components represent important factors responsible for the variation in the data. In this work the term *factor* refers to any influence that results in a changed molecular composition. This includes internal biological processes as well as external changes in environmental conditions or technical artifacts; for example, the circadian rhythm, differences in ecotypes, and changes in temperature or light.
Since a component is a weighted combination of all variables, we can identify the most important variables (genes) by ranking the variables by their corresponding weights.

These variables contribute strongest to the component and should therefore be most important for the respective biological factor represented by this component. If a component explains a biological process, for example, a stress response, then the identified most important genes or metabolites are supposed to play a major role in this process, they might even belong to the same biochemical pathway. Such multivariate analysis is therefore appropriate for the task of identifying gene or metabolite combinations (*molecular signatures*) instead of individual ones.

The data generation model

In component extraction applications we consider the data as being generated from a usually small number of influence factors (sometimes termed hidden factors or sources). We assume that with given information of all potential factors s (external conditions as well as internal states) and the transformation $\Phi_{gen} : \mathcal{S} \to \mathcal{X}$, the molecular data x can be reconstructed. This is referred to as the generative model $x = \Phi_{gen}(s)$.
The aim is to decompose the data x into components z which approximate the original factors s. This requires to find the extraction transformation $\Phi_{extr} : \mathcal{X} \to \mathcal{Z}$ which is inverse to the usually unknown generation transformation Φ_{gen} such that $s \approx z = \Phi_{extr}(x)$.
The model can be *linear* or *nonlinear*. Linear models can be expressed as a (weighted) sum of their individual parts (factors or genes). Nonlinear models, by contrast, cannot simply be expressed by a sum. More precisely, the linear transformation Φ_{gen} of a linear model is given by a linear function. A function $f(x)$ is termed *linear* when it satisfies both properties: additivity $f(x + y) = f(x) + f(y)$ and homogeneity $f(\alpha x) = \alpha f(x)$, otherwise it is a more complex *nonlinear* function.
A linear model can be represented by a matrix multiplication as commonly done in *principal component analysis (PCA)* or *independent component analysis (ICA)*, for example. The resulting components explain straight lines which together form a hyperplane or linear subspace embedded in the original data space.
Artificial neural networks are frequently used as nonlinear models to perform nonlinear transformations such as in *nonlinear PCA* where the resulting nonlinear components explain curves that describe a curved subspace of the original data space, as illustrated in Figure 1.2.

PCA — principal component analysis

Principal component analysis (PCA) is the standard linear approach for extracting components. The main application is to reduce the dimensionality (the number of variables) of the data. The objective is to find a low dimensional representation of the data which captures most of the variance.
The components are hierarchically ordered. The first component, PC 1, explains the

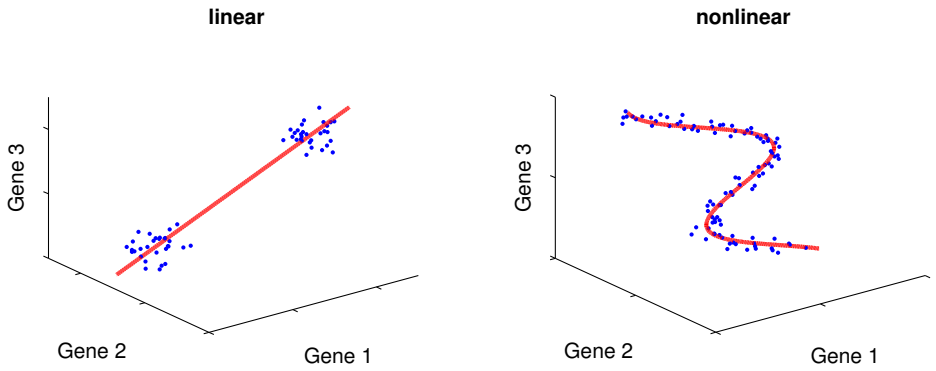


Figure 1.2: Schematic illustration of a linear and a nonlinear component in a data space. The axes represent the variables (e.g., genes) and the data (dots '·') stand for individual samples from an experiment. A component explains the structure of the data by a straight line in the linear case, or by a curve in the nonlinear case. Linear components are helpful for discriminating between groups of samples, e.g., mutant and wild-type. However, in the case of continuously observed factors such as time series, the data show usually a nonlinear behaviour and hence can be better explained by a curve.

highest variance in the data. The second component, PC 2, explains the highest of the remaining variance orthogonal to the first component. This can be naturally extended to obtain a desired number of components. Commonly, the first and second component are plotted against each other to obtain a two-dimensional visualisation that explains the highest variance of the data.
However, by applying PCA, we strictly assume that the desired information is exactly provided by variance. But this assumption might not always hold for molecular data analysis due to the following reasons:

- Often we cannot control the experiments perfectly. Confounding factors (genetic, environmental, technical) might have a large impact on the variance of the data as well.

- Experimental data are often distributed in a non-Gaussian fashion; and thus components obtained by optimising higher than second order (variance) might be more reliable, especially when more than two conditions are examined at a time.

- Some genes or metabolites show only a small variance but might nevertheless have a large impact on the investigated biological process.

An analysis which is not based on variance alone might be more appropriate.

ICA — independent component analysis

A modern alternative to PCA is *independent component analysis (ICA)*. ICA attempts to extract statistically independent components. Statistical independence is a stronger condition than non-correlation in PCA. Informally, it means that the values of one component provide no information about the values of another component. This suggests that ICA is more suitable than PCA for providing individual components of distinct meaning.
However, variance is still an important criterion, but might not be the only one. The investigated biological factors are usually explained by a large amount of variance in the data but not necessarily by the biggest. This can be taken into account by a PCA preprocessing step before ICA is applied. All significantly large variances are maintained in the PCA step, only small variances are removed. In the next step, the variance criterion can be ignored. The condition of statistical independence is then used to separate the contained factors. The aim is to find an optimal balance between different criteria: correlation and variance as considered by covariance in PCA and information theoretic criteria such as mutual information in ICA.

Nonlinear PCA (NLPCA)

Linear methods are sufficient as long as no nonlinear data structure can be expected. Many experiments consider two or a small number of conditions such as mutant and wild-type or disease and control. The samples (replica) within one condition are assumed to be uniform and hence should be located close to each other in a single cluster. A linear component would then be sufficient to discriminate between two conditions. Nonlinearities, by contrast, can be expected when continuously changed factors are investigated. Typically this occurs when samples are measured over time, but any other continuously changed environmental parameter or physiological function may also result in a nonlinear data response. Such data are better explained and analysed by a nonlinear (curved) component as illustrated in Figure 1.2.
This generalisation of components from straight lines to curves, as a nonlinear extension of classical PCA, is referred to as *nonlinear PCA*. Even though a nonlinear extension of ICA would be of greater interest, it is much more challenging or sometimes even intractable. We therefore focus on nonlinear PCA and show that nonlinear PCA is already able to identify desired time components.

Missing data

One of the main problems in analysing molecular data is the frequent absence of some values in the observed sample vectors due to measurement failures. There are many missing data estimation methods. Each of them is based on different assumptions and

objectives that are included in the estimation process. This may, e.g., concern the data structure as well as the measure for an optimal estimation. These assumptions and objectives may be different or even incompatible to those in the subsequent analysis. Instead of estimating the values in a separate step, it would therefore be more reasonable to adapt the analysis methods to be applicable to incomplete data. The attempt is thus to ignore missing values, not *a priori* to estimate them. Such adapted analysis, though, can often itself be used to estimate the missing values.
Concerning component analysis, the objective would be to detect components directly from incomplete data which is even more challenging for nonlinear components. This can be achieved by using *blind inverse models* where the input and the model are optimised to match a given output. Whereas the input-vector of a model in general has to be complete, the output-vector does not necessarily have to be so. Therefore, the inverse model enables us to use the partly incomplete sample vectors as desired outputs to which component values are estimated as suitable inputs. It thereby naturally models the generative process from a set of components (factors) to the observed data.
Such an inverse model can easily be extended to a diagnostic model which gives the most appropriate prediction to a given partially incomplete sample profile. A component, for example, may explain an interesting physiological function. The model can then be used to predict the physiological state from new, even incomplete, samples. A prediction based on metabolite concentrations would then still be possible, even if some metabolites could not be measured or were removed due to high inaccuracy.

Component versus cluster analysis

Although assigning variables (genes) to different components is to some degree similar to separating variables into clusters, there are some major differences. First, in component analysis we do not assume a distinct separation of genes to one or the other component. The variables can contribute to different components. This naturally accounts for the possibility that specific genes or metabolites may be involved in different biological processes. Furthermore, cluster algorithms usually attempt to group all measured variables, although not all of them respond with the investigated experimental conditions. In component analysis, we usually extract only a limited set of relevant components capturing most of the variation in the data. The variables with the largest contribution to those components, i.e. the most significant variables, can then be identified. Consequently, there might be a large number of residual variables which are not associated with any component. This agrees well with the reasonable objective to group only these variables into categories that show a strong behaviour in a specific experiment. Many variables may show no response to the considered experimental conditions, and therefore, can not be categorised.
Component analysis is also advantageous, as the relevant components are most often interpretable. Usually, we select and consider only components with a biological meaning. Such a relation can be detected in the primary visualisation step (Figure 1.1).

1.3 Molecular networks

The primary goal of functional analysis is to identify the molecules that are responsible for a physiological function. However, there is a large interest on a more general view of molecular regulations. The grand objective is no less than to represent the whole molecular interplay within a biological system (cells, organs, or entire organisms) by a molecular network model. Such a comprehensive view to a biological system is a major issue in *Systems Biology*.
Differences between distinct organisms (e.g., evolutionary) or between distinct physiological states can then be investigated by characterising the topology of the corresponding molecular networks. More pragmatic issues involve the use of such networks to discover new biochemical pathways or to identify key molecules.
The molecular network is primarily based on pair-wise interaction knowledge (e.g., gene-gene or metabolite-gene interactions). To gain this information from experimental data, it is assumed that molecules which interact with each other show a similar behaviour. The chosen similarity measure is therefore very important for the usefulness of such reconstructed networks. The similarity measure is used fundamentally as distance in the network and hence influences the quality of the molecular network.
A widely used measure is the pair-wise correlation. Although it can be shown (Weckwerth, 2003; Weckwerth et al., 2004) that correlation networks can show biologically relevant information under specific circumstances (perfectly controlled stationary and uniform conditions) there are some strong limitations. On the one hand many biological relations cannot be detected as they show only low correlation coefficients (Steuer et al., 2003). The reason is not only the high inaccuracy of the data. It is often caused by simultaneously varying biological or technical factors which usually cannot be controlled perfectly. This leads to partial correlations which are difficult to handle in large-scale data sets due to combinatorial complexity. On the other hand, in large-scale data sets many gene pairs show a high correlation value by chance. As this is simply caused by noise in the data, these connections are biologically unreliable.
Even by using other pair-wise similarity measures such as mutual information, we cannot solve that problem because it is mainly caused by the pair-wise consideration of multivariate data sets. Another interesting way would be to use multivariate techniques such as ICA to detect similarities between genes or metabolites. For example, metabolites that highly contribute to the same component (same biological process) are all supposed to play a major role in this process and hence might be functionally similar. A component describes a new (meta-)variable generated by a weighted combination of all original variables (metabolites). Components can therefore be used as additional nodes in the network. The weights (loadings) can then be used as similarity measure between metabolites and components. As biological functionality is included in the resulting network by using the components, such networks can be regarded as *functional networks*.

1.4 Curse of dimensionality

One of the major problems in analysing molecular profile data is the very large number of variables (e.g., genes) compared to a very low number of samples (observations). These variables form a very high-dimensional data space with known positions for the few samples only. In such nearly empty data space it is difficult, for example, in classification tasks, to define reliable decision boundaries that are generalisable for new samples at any position. Usually, a poor performance is obtained in areas of low sample density. The size of such low density areas increases very rapidly with the number of variables. This reduced accuracy of predictive models for data sets with many variables is known as *curse of dimensionality* (Bellman, 1961; Stone, 1980). It states that the number of samples has to increase exponentially with the number of variables to maintain a required level of accuracy.

Dimensionality reduction

Therefore, an important aspect is to reduce the dimensionality. This can either be done by *feature selection* or *feature extraction* techniques. Feature selection simply means that a small subset of important variables (sometimes referred to as features) is selected from the original variable set by using a specific criterion as done, for example, by Hochreiter and Obermayer (2002) for gene expression data. We focus on feature extraction techniques which, by contrast, means that each extracted new variable is a specific combination of all original variables. These new variables are usually referred to as components or features. Even though the main emphasis of feature extraction is to obtain meaningful components (features), it is often used in conjunction with dimensionality reduction. *Independent component analysis (ICA)* is an example of feature extraction, as it aims to extract meaningful components with a high amount of information. By selecting a subset of relevant components, it can also be used to reduce the dimensionality. On the other hand, techniques with a main emphasis on dimensionality reduction, such as the classical *principal component analysis (PCA)*, can also be regarded as feature extraction techniques, since the extracted components may often have some meaning.

Supervised and unsupervised methods

The considered methods belong to *unsupervised* techniques, meaning that the potentially known group identifier (class labels) are not taken into account by the algorithm. As the main objective is often to find a component that discriminates the investigated experimental conditions, it might be useful to extract components in a supervised manner by using the group identifiers, e.g., with classical Fischer's linear discriminant analysis (Fisher, 1936). However, there is a very high risk of over-fitting when applied to a large number of variables and only few samples — the curse of dimensionality. Over-fitting means that the result is driven by the noise in the data and not by the underlying biological process. One solution would be to first reduce the dimensionality by unsupervised

methods (e.g., by PCA). After such pre-processing step supervised techniques can be successfully applied as shown by Goodacre et al. (2003) and Johnson et al. (2003) for metabolite data.
Supervised methods are target orientated, they are useful to analyse a specific known experimental factor. They cannot be used to cover the investigative or exploratory aspect to identify the major experimental factors reflected in the data.
Unsupervised methods, by contrast, have the previously mentioned advantage that the extracted components explain the major or global information or the most important characteristics of the data, independently from the experimental knowledge which is unknown to the algorithm. This can tell us whether the investigated experimental conditions are well reflected by the data as expected, or whether there are stronger artifacts due to badly controlled technical or environmental factors, or whether there are even unexpected biological characteristics.
Sometimes we cannot absolutely trust the labelling of samples. In time series, for example, the response time and developmental state of individual organisms in any experiment differs from the exact physical time of measurement. An unsupervised model will therefore be superior in accommodating the unavoidable individual variability of biological samples.

2 Normalisation

The main objective of this work is to analyse molecular data by identifying meaningful components. Since normalisation has a significant impact on the performance of the final analysis, the effects of normalisation with regard to large-scale data from molecular biology are shortly discussed in this chapter (Figure 2.1).

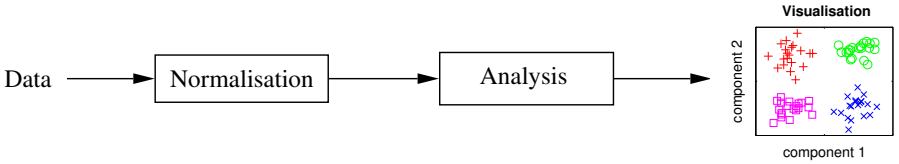


Figure 2.1: Normalisation and analysis techniques are both important for optimal visualisation or component extraction. The plot on the right illustrates samples from different experimental conditions.

The choice of normalisation is in general as crucial as the choice of the analysis technique itself. The final analysis technique is usually chosen to achieve a specific purpose of analysis with respect to the information we are searching for. The optimal normalisation, by contrast, depends more on the characteristics of the data. In general, due to technical and biological reasons, different variables (e.g., different genes) or different samples are not on the same scale and have to be rescaled in a suitable way. Normalisation, in principle, means to make different variables and different samples comparable to integrate them in a joint analysis.
Another important aspect of normalisation is to represent the relevant information in an optimal way and to remove non-biological contributions. Normalisation is a convenient way to include prior knowledge such as additional technical or biological knowledge and to take into account assumptions about the characteristics of the data.
Rescaling the data is identical to using a new metric in the data space. The aim is to find a metric that optimally represents the desired information. This is closely related to finding the optimal distance or similarity measure as used in cluster or network analysis. It is to some extent also related to finding the optimal kernel in a kernel technique such as support vector machine (SVM) (Vapnik, 1995), where a kernel can be interpreted as a similarity measure. Therefore, a good normalisation or metric can reduce the effort in subsequent analysis techniques.
For the purpose of this discussion, platform specific data processing such as background corrections is not considered here. Neither the particular microarray nor mass spectronomy platform, that is used to obtain gene expression or metabolite and protein concentrations, is explicitly considered here. The assumption is to have a data matrix of high-quality measurements representing intensities or concentrations.

Consider a set of n experimental samples, each characterised by d measurements (one for each variable), e.g., d different genes. The data can be arranged in a d x n matrix where rows represent variables (e.g., genes) and columns correspond to different samples. The data matrix can be normalised row-wise or column-wise, such that either the variables or the samples are normalised to make them more comparable. The variables can be rescaled or the total intensity amount of sample vectors can be set constant. One advantage of a sample normalisation is that in case of additional samples the new samples can be normalised individually, no renormalisation of all samples is required, which is important, for example, for diagnostic tasks. Variable normalisation, however, is important when the exact contribution of each variable (gene) to a component or visualisation result is of interest. Both *Log fold change* as variable and *unit vector norm* as sample normalisation are convenient normalisation techniques for molecular data. Nevertheless, it is often useful to preselect the usually very large number of variables in advance, e.g., by variance or intensity. Although a small intensity or variance might have a large biological impact, small observed values are usually strongly corrupted by the relatively large amount of background noise and, therefore, are normally of no use. A comprehensive discussion of data matrix normalisation in respect of microarrays can be also found in Quackenbush (2002).

2.1 Log fold change (log ratio)

To apply *log fold change*, it is assumed that the relevant information is the relative change in expression or concentration with respect to the average or to control samples. These ratios should be transformed by a logarithm, to obtain symmetric values of positive and negative changes. This is useful, for example, for considering up and down regulated genes symmetrically. The logarithm to base two (log_2) is frequently used, but any other base can be taken as well. The difference is only given by a global scaling factor c which does not affect the directions of components in PCA or ICA, e.g., $log_2(x) = c * log_{10}(x)$ with $c = 3.32 = log_2(10)$. Thus, to obtain a normalised variable $\tilde{x}_i$, the elements of the variable $x_i = (x_i^1, \ldots, x_i^n)$ are divided by the median of x_i and subsequently transformed by a logarithm.

$$\tilde{x}_i = log\left(\frac{x_i}{median(x_i)}\right)$$

Now, a high variance would point out a high relative change, useful for variance considered analysis techniques such as PCA. It is convenient to use the median, as it is more robust against outliers than the mean. When control samples are available as references (e.g., the wild-type in a mutant experiment or the zero time point in a time series), the samples can be divided by the median of these control samples alone.

$$\tilde{x}_i = log\left(\frac{x_i}{median(x_i^{control})}\right)$$

As the meaning is quite similar, the results are expected to be very close, though a division by control might be easier to interpret.

2.2 Unit vector norm (total intensity)

For applying *unit vector norm* we assume that the total amount of a sample vector $v = (v_1, v_2, \ldots, v_d)$ is nonrelevant, and can therefore be removed by scaling the norm $\| v \|$ (the power or intensity) of this vector to a fixed value, usually one, $\| \tilde{v} \| = 1$. The normalised sample vector $\tilde{v}$ is obtained by

$$\tilde{v} = \frac{v}{\| v \|}$$

Vector normalisation emphasises the ratios between measurements of different variables (e.g., different genes) for one sample. Vector norm is in general referred to as p-norm (Golub and van Loan, 1996). The most important ones are l_1 and l_2 norm.

$$\begin{array}{lrcl} l_1\text{-norm} & \| v \|_1 & = & \sum_i |v_i| \\ l_2\text{-norm} & \| v \|_2 & = & \sqrt{\sum_i |v_i|^2} \\ l_p\text{-norm} & \| v \|_p & = & \sqrt[p]{\sum_i |v_i|^p} \\ l_{infinity}\text{-norm} & \| v \|_\infty & = & \max_i |v_i| \end{array}$$

The l_1 vector norm can be interpreted as transforming data into percentages. The l_2 vector norm explains the *Euclidean length* of a vector. This is geometrically interesting, as it projects the samples onto a unit hypersphere, as shown in Figure 2.2.

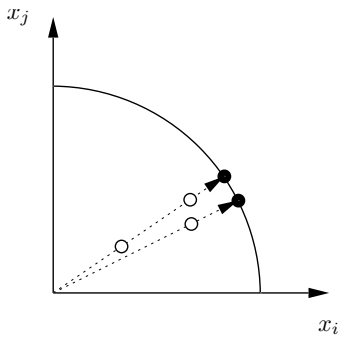


Figure 2.2: Geometrical illustration of l_2 *vector norm*. The samples ('$\circ$') are projected onto a hypersphere. This strongly affects the pairwise distances. Highly correlated samples end up close to each other, even if they had a large distance beforehand. Geometrically, highly correlated samples are located in the same direction from the origin, but with possibly different intensities.

Scaling sample vectors to unit length is closely related to correlation analysis, as highly correlated samples are projected close to each other (small Euclidean distance). With gene expression data it might be convenient to reduce the very large number of genes by a kind of filtering (feature selection). At low intensities close to the background, the observed values may more likely be caused by noise. Genes that fall below a certain threshold of a specific criterion, e.g., the variance or intensity, should therefore not be taken into account. Otherwise, the noise in the potentially large number of experimentally nonrelevant genes may confuse the vector norm.

2.3 Unit variance

By scaling each variable to unit variance $\sigma^2 = 1$ it is assumed that the variance σ_i^2 of each variable x_i, e.g., of each metabolite, has no relevance. As the variance σ_i^2 is the square of the standard deviation σ_i, it is identical to *unit standard deviation*

$$\tilde{x}_i = \frac{x_i}{\sigma_i}$$

Covariance in PCA is then reduced to correlation between variables. The covariance matrix becomes identical to a correlation matrix. The covariance between two variables x_i and x_j $cov(x_i, x_j) = \frac{1}{n-1}(x_i - \bar{x}_i)(x_j - \bar{x}_j)^T$ with mean $\bar{x}_i = \frac{1}{n}\sum_{l=1}^{n} x_i^l$ is equal to the correlation $corr(x_i, x_j)$ when normalised by the standard deviations σ_i and σ_j, $corr(x_i, x_j) = \frac{cov(x_i, x_j)}{\sigma_i \sigma_j}$. Thus covariance is equal to correlation for variables that have variance or standard deviation equal one $\sigma^2 = 1 = \sigma$.
For a variance optimisation technique such as PCA, all variables have the same chance to get a high rank within the important first components, as they all have the same variance. The first component of PCA, which usually reflects variables of high variance and correlation, now solely depends on the largest group of highly correlated variables, which jointly form the direction (component) of highest variance in the data space.
A similar normalisation is termed *z-score* $\tilde{x}_i = \frac{x_i - \bar{x}_i}{\sigma_i}$ where the standard deviation σ_i is also set to one and additionally the mean $\bar{x}$ is set to zero. However, whether the mean is zero or not is usually not important for PCA or ICA, as the algorithms also remove the mean automatically.
Although *unit variance* or *z-score* are standard normalisation methods in many areas, there are strong limitations when applied to molecular data where variance has some importance. This is caused by an important difference in the experimental design. Usually, in many areas variables are observed that are expected to be related to the investigated factor and hence we can assume that a high percentage of these variables gives us useful information. Molecular data, by contrast, are usually obtained by high-throughput screening techniques, where as many variables (e.g., genes) as possible are measured. The goal is mostly to find some relevant candidates within the large number of measured variables. The relevant variables may be emphasised by variance caused by variations at concentration or activity level. Variables which do not respond to our experiment usually have a low variance and hence a low contribution to results of many analysis techniques. The disadvantage of unit variance, however, is that by scaling up these nonrelevant variables, their impact on the analysis result is increased dramatically. Unit variance normalisation should therefore not be used without any pre-selection, especially with the large number of gene expression data. After such pre-selection we can assume that most of the selected genes are experimentally relevant and then unit variance might be reasonable. Caution is also required due to the limited number of samples, where high correlations might occur by chance.

3 PCA — principal component analysis

Variables from gene expression or metabolite data sets are generally correlated in some way. This means that the data points (samples) do not fill out the entire data space. The data usually tend to be restricted to a subspace of lower dimensionality. This leads to the concept of *dimensionality reduction*: to find a low-dimensional data structure hidden in high-dimensional observations (Carreira-Perpiñán, 1997). Principal component analysis (Jolliffe, 1986; Diamantaras and Kung, 1996) reduces the dimensionality, the number of variables of the data, by maintaining as much variance as possible. This is illustrated for three dimensions in Figure 3.1. In applying PCA we necessarily assume that the information we are searching for is exactly provided by the variance in the data. But this assumes that we have perfectly controlled experiments where all variation is only caused by the investigated biological process. However, in most cases we cannot prevent technical artifacts or internal biological variations. Or, we would like to investigate more than one biological process at a time. Variance may then not be the most optimal criterion, but still can be used first to gain an informative impression of the data structure.
PCA transforms a d-dimensional sample vector $x = (x_1, x_2, \ldots, x_d)^T$ into a usually lower dimensional vector $y = (y_1, y_2, \ldots, y_k)^T$, where d is the number of variables (metabolites or genes) and k is the number of selected components. The PCA transformation is given by the k x d matrix V, such that

$$y = Vx$$

Each row-vector of matrix y contains values (scores) of a new variable y_j referred to as principal component (PC). The component PC j, given by the new variable $y_j = (y_{j1}, y_{j2}, \ldots, y_{jn})$, is a linear combination of all original variables $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, weighted by the elements of the corresponding transformation vector $v_j = (v_{j1}, v_{j2}, \ldots, v_{jd})$

$$y_j = \sum_{i=1}^{d} v_{ji} x_i = v_{j1}x_1 + v_{j2}x_2 + \ldots + v_{jd}x_d$$

n is the number of samples and d is the number of original variables. The weights v_{ji} (sometimes referred to as loadings) give us the contribution of all original variables x_i to the jth component. Geometrically, PCA is equivalent to a rotation of the original data space. The new axes are the principal components. The vector v_j gives the direction of the jth principal component (PC j) in the original data space. The first component, PC 1, represented by the variable y_1, is in the direction of highest variance. The second component, PC 2, is the direction that maximises the remaining variance

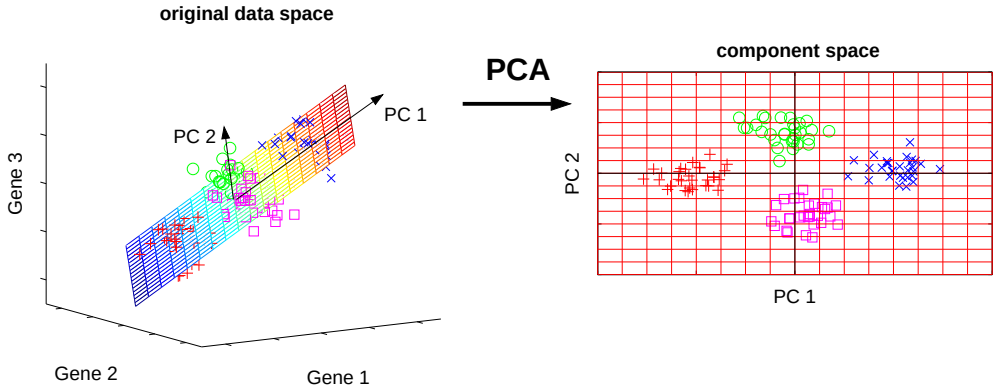


Figure 3.1: **Principal component analysis (PCA).** Illustrated is the transformation of PCA which reduces a large number of variables (genes) to a lower number of new variables termed principal components (PCs). Three-dimensional gene expression samples are projected onto a two dimensional component space that maintains the largest variance in the data. This two-dimensional visualisation of the samples allows us to make qualitative conclusions about the separability of our four experimental conditions. The component space explains a linear subspace of the original high-dimensional space, where the data lie on or near by. PCA simply rotates the original data space such that the principal components (PCs) are the axis of a new coordinate system — the component space. This can be mathematically extended to even more than three original dimensions.

in the orthogonal subspace complementary to the first component. The first and second component together explain the two-dimensional plane of highest variance. This can be naturally extended to obtain the k first principal components. A column-vector $y = (y_1, y_2, \ldots, y_k)^T$ contains the k new coordinates in the space of principal components of the corresponding sample x.

3.1 Conventional PCA

The transformation or rotation matrix V can be estimated by different algorithms. The classical way is to calculate the eigenvectors of the d by d covariance matrix between variables, $cov(X) = \frac{1}{n-1}\sum_{l=1}^{n}(x^l - \bar{x})(x^l - \bar{x})^T$ where the vector $\bar{x}$ contains the mean of all variables, n is the number of samples, and x^l is the lth sample vector $x^l = (x_1^l, x_2^l, \ldots, x_d^l)^T$. The eigenvectors are sorted by their corresponding eigenvalues. The matrix V is then given by the first k eigenvectors v_j, $(j = 1, \ldots, k)$, to the largest eigenvalues. Sometimes the correlation matrix is used instead of the standard covariance matrix. However, this is identical to normalising the data to unit variance in advance, see section 2.3. As the possibly large number d of variables in molecular data can be problematic for solving the eigenvalue problem of a d x d covariance matrix, the required principal components can be more easily obtained by *singular value decomposition* SVD, *multidimensional scaling* MDS or an *adaptive PCA* method.

3.2 SVD — singular value decomposition

A different approach for obtaining the same principal components is *singular value decomposition (SVD)*, see, e.g., Golub and van Loan (1996). SVD is more efficient than the PCA covariance approach, especially when there is a large number of variables and a small number of samples, as is typical in molecular data sets.
The *singular value decomposition* of a d x n data matrix X is

$$X^T = USV^T$$

The columns u_j of U are termed *left singular vectors*, the columns v_j of V are termed *right singular vectors*, and the diagonal elements s_j of the diagonal matrix S are the *singular values*.
If we consider a centred data set X, where the rows, the variables x_i, have zero mean, then the principal components (the scores) y_j are given by the columns of the matrix multiplication $Y = US$. The columns v_j of V are equivalent to the eigenvectors of the covariance matrix.
A comprehensive description of SVD in relation to PCA with respect to gene expressions is given by Wall et al. (2003). Other applications of SVD to gene expressions can be found in (Alter et al., 2000; Holter et al., 2000; Liu et al., 2003).

3.3 MDS — multidimensional scaling

Another convenient way to obtain the principal components is to use a classical approach of *multidimensional scaling (MDS)* based on eigenvalue decomposition. MDS, see, e.g., Cox and Cox (2001); Buja et al. (1998), gives a projection or visualisation of the data by using a distance matrix D alone. Therefore, it is useful in cases where only relative distances d_{ij} from one sample i to another sample j are available, and not the exact position in a multidimensional space. Such a distance can be, e.g., a similar measure between two sequences. Nevertheless, the distances or similarities can also be derived from a data matrix, e.g., by using Euclidean distance, covariance, correlation, or mutual information. The aim is to project the data into a two or low dimensional space such that the pairwise distances $\parallel y_i - y_j \parallel$ are as similar as possible to the distances d_{ij} given by the distance matrix, thus minimising the function

$$\sqrt{\sum_{i \neq j} (d_{ij} - \parallel y_i - y_j \parallel)^2}$$

There exists a wide variety of methods for performing MDS, where nonlinear projections are usually more efficient. However, to explain the relation to PCA we consider the classical linear MDS by eigenvalue decomposition of the covariance matrix as a distance matrix, $D = cov(X)$. Here, the two eigenvectors to the largest eigenvalues of the distance matrix give the required projections (components). For a data set X where the variables x_i have zero mean, it is shown, e.g., by Burges (2004), that by using the n by n covariance matrix between samples (not between variables as in PCA), the eigenvectors of this

covariance matrix are the desired principal component scores y. This is advantageous in the case of small numbers of samples where only a relatively small sample covariance matrix is required.

3.4 Adaptive algorithms

For small numbers of samples, the estimation of the principal components can be efficiently calculated using SVD or MDS, even with a very large number of variables (genes). However, with a decrease in measurement costs, the number of samples will increase rapidly. Then it becomes impossible to estimate all principal components. We have to use adaptive algorithms instead which extract the principal components in a deflationary (sequential) manner, meaning that the components are extracted one after the other starting from the component of highest variance. Consequently, only the first k desired components need to be extracted instead of all components.
Convenient algorithms for this task include: Sanger's learning rule (Sanger, 1989) linear auto-associative neural networks (Baldi and Homik, 1995), the APEX network by Diamantaras and Kung (1996), and expectation-maximisation (EM) algorithm based PCA (Roweis, 1997).

3.5 Application of PCA

In general, the techniques are applicable to gene expression as well as metabolite or protein profile data. As an example, the techniques in this chapter are applied to metabolite data from a crossing experiment of *Arabidopsis thaliana*.

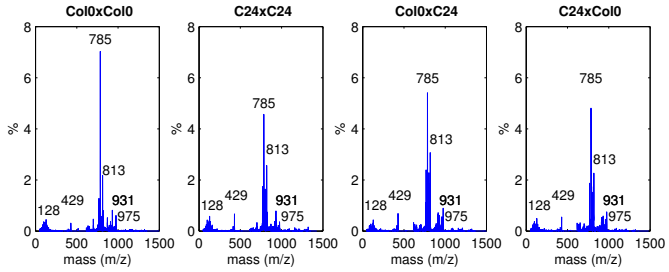


Figure 3.2: Mass spectra of *Arabidopsis thaliana* crosses, analysed to investigate the response at the metabolite level.

Data set: There are four groups, two parental lines Columbia 'Col-0' and 'C24', and two crosses 'Col-0 x C24' and 'C24 x Col-0'. For each group there are 24 samples (observations), hence 96 samples altogether. The samples were analysed by using a direct infusion mass spectrometer without chromatographic separation, thus each spectrum reflects the composition of all metabolites in a given sample, see Figure 3.2.
Each sample is characterised by 763 variables which contain the intensities at 763 different masses (m/z), see Scholz et al. (2004a) for more details about the platform specific

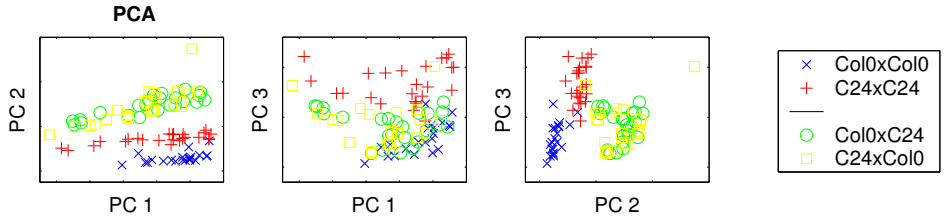


Figure 3.3: PCA applied to metabolite data of *Arabidopsis thaliana* crosses. The best projection is given by the second and third principal components (PC 2, PC 3) and not by the first (PC 1) as expected.

pre-processing. The purpose of the analysis is to investigate how the biological background is reflected in the metabolite data.

Results: Although PCA is often a very useful technique, it fails to give optimal projections when applied to our metabolite data set of *Arabidopsis thaliana* crosses, see Figure 3.3. The first principal component (PC 1), the component of highest variance, contains no information for discriminating the lines or crosses. The components PC 2 and PC 3 give a better result, although they are of smaller variance. Consequently, the major assumption for applying PCA, that the required experimental information have to be related to the highest variance, does not hold for this particular data set. Additionally, the discrimination performance of the components is limited. For example, the best discrimination between the parental lines is not in direction parallel to the axes and hence not perfectly explained by component PC 3.

3.6 Limitations of PCA

Ideally, experiments are designed such that only the relevant factors vary, whereas all other factors are kept as constant as possible. It is assumed that the investigated biological process will then be reflected by highest variance in the observed data and hence PCA would be a good technique to confirm some theoretical assumptions. Often, though, it is not possible to keep all unwanted factors as constant as needed. There are technical artifacts or unwanted biological and environmental variation that also have a large impact on the variance in the data. Even by using normalisation techniques we often cannot reduce these contributions sufficiently. Variance might still be important to some part, but we have to distinguish relevant from nonrelevant contributions. It is therefore necessary to integrate approaches that in addition optimise other criteria than variance. This is, in particular, important when the overall data are non-Gaussian distributed as typical for most molecular data.
In the next chapter it will be shown that *independent component analysis* (ICA) in combination with PCA fits our requirements better than PCA alone. The aim is to identify all factors which have a large impact on the data and to represent them separately by individual components.

4 ICA — independent component analysis

In many fields researchers are interested in reliable methods and techniques enabling the extraction or separation of useful information from superimposed signals corrupted by noise and interferences. The identification of original signals or factors from a given data set is the focus of *blind source separation (BSS)*. The term 'blind', in this context, means that both the original factors (sources) and the mixing process are unknown. With the assumption that the observed data are given by a linear combination of mutually independent factors, we can apply independent component analysis (ICA) to solve this source separation problem.
ICA was first motivated by the so called *cocktail party problem*: with the aim to identify individual speakers or music from a sound mixture given by several microphones. When we consider molecular experiments, the observed variables (genes, metabolites, or proteins) represent the molecular response to specific experimental factors. The observed gene expression value, metabolite or protein concentration depends essentially on the particular value of many external or internal factors such as light, temperature, developmental stage or simply the ecotype. Molecular data can therefore be considered as a mixture of information from different original factors, or simply as a response to a combination (mixture) of several factors as visualised in Figure 4.1. It should be noted that the arrows in Figure 4.1 do not necessarily mean causality in the biological sense. Many internal factors, such as the current state of a circadian rhythm, can only be seen as informative factors driven by the molecular activation or concentration itself. We simply assume that with the given information of all possible factors (external conditions as well as internal states) and the mixing process (the dependencies), the molecular data can be reconstructed. The objective is therefore to explain the data by such a generative model.
ICA can be used to solve this blind source separation problem, when a linear combination can be assumed. The mixing process can then be explained by a matrix A which transforms a vector $s = (s_1, s_2, \ldots, s_k)^T$ of particular values s_i from k different factors (often termed sources) into a d dimensional sample vector $x = (x_1, x_2, \ldots, x_d)^T$, e.g., expression values of the corresponding d genes

$$x = As$$

It is impossible to identify both the factors s and the transformation A as a unique solution from a given data set without further conditions. In ICA, the major assumption is therefore mutual independence of the original factors. The objective is to decompose the data set X into independent components z_i which approximate the original factors s_i.

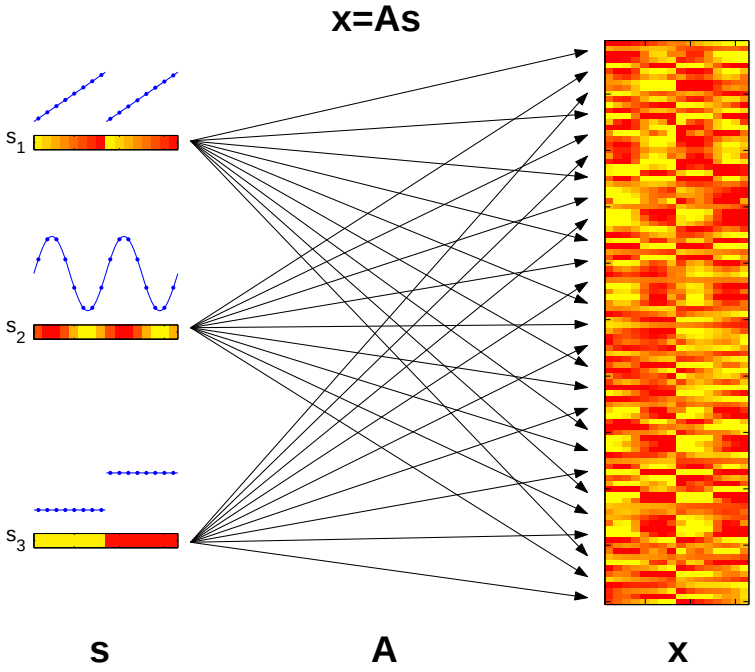


Figure 4.1: The motivation for applying ICA is that the measured molecular data can be considered as derived from a set of experimental factors s. This may include internal biological factors as well as external environmental or technical factors. Each observed variable x (e.g., gene) can therefore be seen as a specific combination of these factors. The illustrated factors may represent an increase of temperature (s_1), an internal circadian rhythm (s_2), and different ecotypes (s_3).
With the sometimes reasonable assumption of linearly combined factors that are independent and non-Gaussian, we can use ICA to identify the original factors s and the dependencies given by the matrix A.

The attempt of ICA is hence to find the reverse transformation given by a matrix W, which is approximately inverse to the unknown matrix A ($W \approx A^{-1}$), such that

$$s \approx z = Wx$$

As in practise it is often impossible to find components that are absolutely independent, the goal is to find a separating matrix W so that the components z_i are as independent as possible.
The meaning of each extracted component can often be interpreted with additional experimental biological as well as technical knowledge. We are therefore able to detect and interpret both expected and unexpected factors. Components related to the examined factors can confirm the expected molecular response, whereas other components may point out unexpected factors caused by interesting unforeseen biological behaviour or simply by technical artifacts. In addition to the component itself, we can also provide

the most important variables (e.g., genes) of highest contribution to a component. Or, from the point of view of a generative model, we can give the most important variables which depend strongest on a specific factor represented by a component.
Applied to molecular data, ICA can outperform the classical PCA as shown in Figure 4.6. This higher informative performance can be achieved by adapting ICA to the characteristics of experimental data in molecular biology. As illustrated in Figure 4.2, this includes a combination with PCA as a pre-processing step and the selection of sub-Gaussian instead of the super-Gaussian components relevant in sound separation.

Figure 4.2: The proposed ICA procedure. First, the data set is reduced by PCA thereby maintaining all of the relevant variances. ICA is applied to this reduced data set and the extracted independent components are ranked by their kurtosis value to obtain components that are sub-Gaussian distributed.

Bibliographic notes. ICA was first introduced by Comon (1994), with subsequent developments by Bell and Sejnowski (1995). Since then a wide variety of ICA algorithms have been developed (Hyvärinen and Oja, 2000; Bell and Sejnowski, 1995; Ziehe and Müller, 1998; Blaschke and Wiskott, 2004; Bach and Jordan, 2002). Comprehensive introductions to ICA can be found in Hyvärinen and Oja (2000), Stone (2002), and in several books published in recent years (Haykin, 2000a,b; Hyvärinen et al., 2001; Cichocki and Amari, 2003; Stone, 2004).
One of the first motivations for the development of ICA was sound signal separation. Now, computational neuroscience is a major field of application of ICA in biomedical science. The aim is to identify artifacts and signals of interest from magnetoencephalograms (MEG) (Vigário et al., 2000; Tang et al., 2002) and from electroencephalograms (EEG) (Jung et al., 2000; Makeig et al., 2002). ICA has also become important in molecular biology. It has been applied by Liebermeister (2002) to analyze gene expression patterns during the yeast cell cycle and in human lymphocytes. Martoglio et al. (2002) applied ICA to ovarian cancer data. In Lee and Batzoglou (2003) different ICA algorithms were compared and applied to yeast cell cycle, *C. elegans*, and human gene expression data. Saidi et al. (2004) showed that clustering on components from ICA give more biologically reasonable groupings than clustering on components from PCA. In Scholz et al. (2004a) ICA was proposed to be used in a particular manner (Figure 4.2) which was successfully applied to metabolite data from crosses of the model plant *Arabidopsis thaliana*. It was found that ICA extracts more meaningful and better interpretable components than PCA, and even an unexpected experimental artifact was discovered. Also, when applied to enzymatic activities (Scholz et al., 2004b), ICA was able to provide components of greater discrimination and with greater meaning than components of PCA.

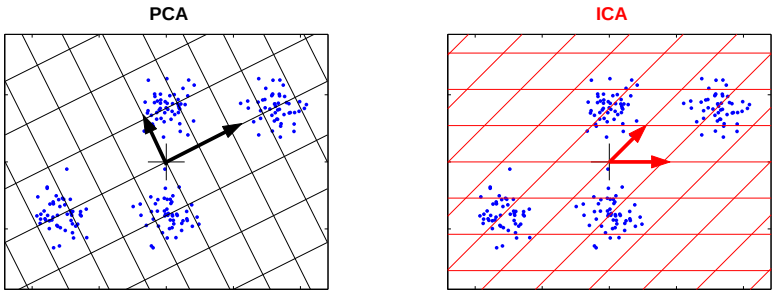


Figure 4.3: PCA and ICA applied to an artificial data set. The grid represents the new coordinate system after PCA or ICA transformation. The identified components are marked by an arrow. The components of ICA are related better to the cluster structure of the data. They have an independent meaning. One component of ICA contains information to separate the clusters above from the clusters below, whereas the other component can be used to discriminate the cluster on the left from the cluster on the right.

4.1 Statistical independence

The major assumption for applying ICA is the mutual independence of unknown original factors which determine the observed data. The objective of ICA is therefore to identify these factors by searching for components which are as statistically independent as possible, not only uncorrelated. Independence means that the values of one component provide no information about the values of other components. This is a stronger condition than the pure non-correlation condition in PCA, where the values of one component can still provide information about the values of another component in case of non-Gaussian distributions. In addition, the components of ICA are not restricted to being orthogonal as shown in Figure 4.3.
Mathematically, *statistical independence* is defined in terms of probability densities. Two variables, z_1 and z_2, are independent if, and only if, their joint probability $p(z_1, z_2)$ is equal to the product of the probabilities of z_1 and z_2.

$$p(z_1, z_2) = p(z_1)p(z_2)$$

Using *Bayes Theorem*, the conditional probability $p(z_2|z_1)$ of a variable z_2 given variable z_1 is

$$p(z_2|z_1) = \frac{p(z_1, z_2)}{p(z_2)}$$

Thus, if both variables are independent the conditional probability is equal to the unconditional probability

$$p(z_2|z_1) = p(z_2) \qquad \text{and} \qquad p(z_1|z_2) = p(z_1)$$

This simply means that knowing something about one variable tells us nothing about the other.

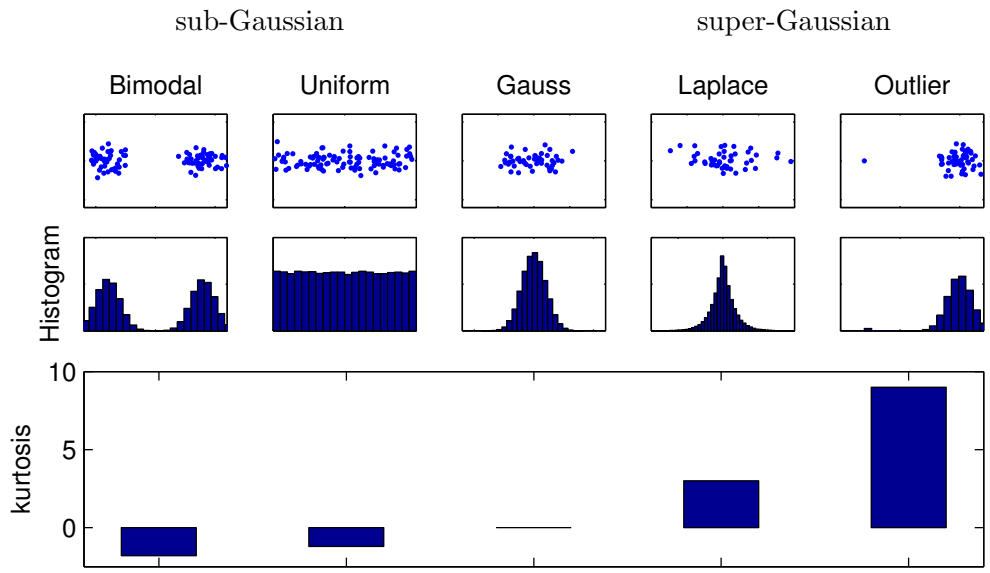


Figure 4.4: Kurtosis is used to measure the deviation of a particular component distribution from a Gaussian distribution. The kurtosis of a Gaussian distribution is zero (middle), of super-Gaussian distributions positive (right), and of sub-Gaussian distributions negative (left). Sub-Gaussian distributions can point out bimodal structures from different experimental conditions or uniformly distributed factors such as a constant change in temperature. Thus the components of most negative kurtosis provide the most important information in molecular data.

However, the probabilities are usually unknown and often difficult to estimate. This has led to a large number of different approaches to extract independent components. Usually they are based on a contrast function which is used as a measure for independence. This includes information-theoretic algorithms that minimise the mutual information between components as well as using higher order statistics such as the maximisation of kurtosis of each component. The latter is motivated by the idea that the sum (mixture) of any two independent random variables is closer to a Gaussian (normal) distribution than the original variables, which can be derived from the *Central Limit Theorem* of probability theory. As a Gaussian distribution is therefore most likely a mixture, we can search for non-Gaussian distributed components to identify the individual original factors. This assumes of course that the factors themselves are not Gaussian distributed. Hence we need a measure of distance from Gaussianity. Possible measures are all normalised cumulants of order higher than two, since these are zero for a Gaussian distribution. A frequently used measure is the fourth order cumulant — the *kurtosis*.

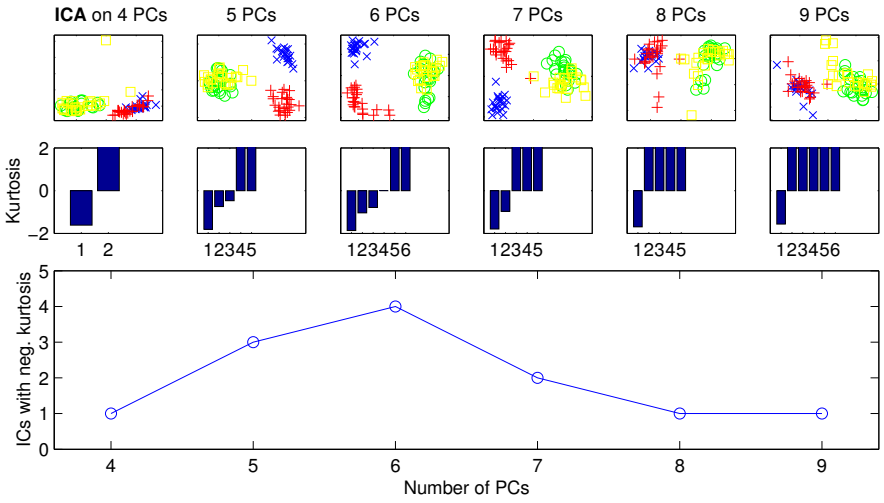


Figure 4.5: ICA is applied to reduced data sets with different numbers of PCs. At six components of PCA, ICA extracts the highest number of relevant independent components (ICs), i.e. ICs with negative kurtosis.

4.2 Component ranking

One difficulty in applying ICA to high-dimensional molecular data is that the number of extracted components equals the number of variables in the data set. The components have no order as in PCA, and hence we need a criterion to rank the components according to our interest. In the classical application of ICA to separate sound signals, we were interested in components that are super-Gaussian distributed, as this is a typical distribution of sound signals.
In molecular data, by contrast, sub-Gaussian distributions are of higher importance. A bimodal distribution can be caused by two clusters of different experimental conditions and a uniform distribution can be caused by uniformly changing experimental factors such as temperature or time.
To identify whether a distribution is sub- or super-Gaussian, we can use the measure of kurtosis, see Figure 4.4. It is a classical measure of non-Gaussianity, it indicates whether the data are peaked (super-Gaussian) or flat (sub-Gaussian) relative to a Gaussian distribution

$$kurtosis(z) = \frac{\sum_{i=1}^{n}(z_i - \mu)^4}{(n-1)\sigma^4} - 3$$

where $z = (z_1, z_2, ..., z_n)$ represents a variable or component with mean μ and standard deviation σ, n is the number of samples. The kurtosis is the fourth auto-cumulant after mean (first), variance (second), and skewness (third).
As a sub-Gaussian distribution has a negative kurtosis value, the components with the most negative kurtosis can give us the most relevant information.

4.3 PCA pre-processing

Originally, ICA was developed to solve a blind source separation problem where we have few variables and many samples, as given by the high sampling rate in sound signals. To be applicable to molecular data, ICA has to be adapted to the opposite situation of few samples in a data space given by many variables. Applying ICA directly to this high-dimensional data set is questionable and the results are usually of no practical relevance. It is therefore essential to reduce the dimensionality in advance which can be well done by PCA. We thereby assume that the relevant information is still related to a significantly high amount of variance but not necessarily to the highest amount. The PCA pre-processing step attempts to preserve all relevant variances and removes only the noise given by small variances. On this reduced data set ICA is then applied to optimise criteria other than variance, namely information theoretic criteria such as mutual information (MI) or higher order statistics (kurtosis). The optimal number of PCs or the optimal reduced dimensionality can be found by considering the goal of our analysis to find as many relevant components as possible. As a negative kurtosis indicates relevant components, the optimal dimensionality is then given by the dimensionality where the highest number of independent components with negative kurtosis can be extracted, see Figure 4.5. Alternatively, the square sum over these negative values can be used instead of counting the number of components with negative kurtosis. This might be a more reliable criterion, since a very negative kurtosis counts higher than those close to zero.

The role of PCA pre-processing

Why can ICA not be directly applied to the high-dimensional data? On the one hand the reason is the high level of noise in the data. There are a lot of samples, which are corrupted by noise, such that they can be regarded as outliers. ICA is sensitive to outliers, because of their super-Gaussian distribution. ICA is then more likely to detect components with outliers (which sometimes can be helpful), rather than the experimental factors of interest. PCA is therefore used to remove noise, including the impact of outliers, such that the data become more compact. On the other hand, there are only few samples in an almost empty high-dimensional space. We can find many directions (components) with a strong bimodal distribution (sub-Gaussian), where half of the samples can be projected to one position and the other half to the other position. This might be possible in many permutations of the samples. To avoid this we need a filter, an additional constraint, such that only bimodal distributions of significantly large variance will be extracted. Such a variance filtering is achieved by the PCA pre-processing step.

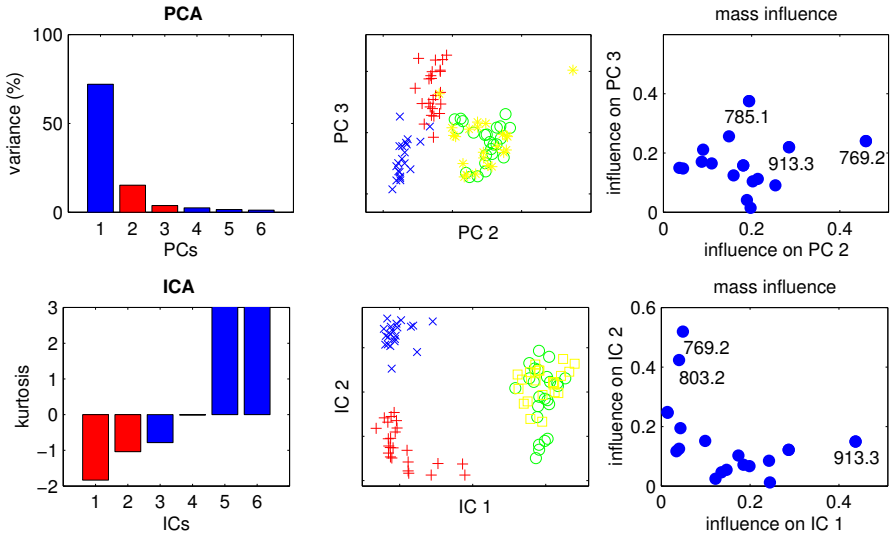


Figure 4.6: **ICA versus PCA.** In the middle, ICA visualisation shows a higher discrimination of the samples than PCA visualisation. Furthermore, in ICA the different discriminations are optimal given by the two axes, the first two independent components (ICs) when ranked by the kurtosis measure. The best PCA result is only given by the second and third principal component (PC) ranked by variance (Figure 3.3). The bar plots on the left show the respective values of the ranking criteria of the first six components for both variance in PCA and kurtosis in ICA. On the right the absolute contributions (loadings) are plotted against each other for the top 20 masses of highest contribution. In PCA the masses are more likely to make a contribution to both components, whereas in ICA the masses are involved differently, contributing to one or the other IC, confirming that different ICs represent independent biological processes where different metabolites are involved.

4.4 Contributions of each variable

As the detected independent components often have a biological interpretation, it would be important to know which variables (genes/metabolites/proteins) contribute most to the components. These contributions are given by the transformation matrices of PCA and ICA and are referred to as *loadings* or *weights*.
PCA transforms a d-dimensional sample vector $x = (x_1, x_2, \ldots, x_d)^T$ into a new vector $y = (y_1, y_2, \ldots, y_k)^T$ of usually lower dimensionality k. Thus, PCA reduces the number d of original variables to a number k of selected components. The PCA transformation is given by the eigenvector matrix V, $y = Vx$. Similarly, ICA transforms this vector y into the desired vector $z = (z_1, z_2, \ldots, z_k)^T$, containing the independent values z_i for each IC i. For that a de-mixing matrix W is estimated by ICA, $z = Wy$. V gives the contributions of each variable to each of the PCs, whereas W gives the contributions of each PC to each of the ICs. We can combine both matrices $U = W * V$ into a direct transformation $z = Ux$, where U gives vector-wise the contributions of each variable to each of the ICs.

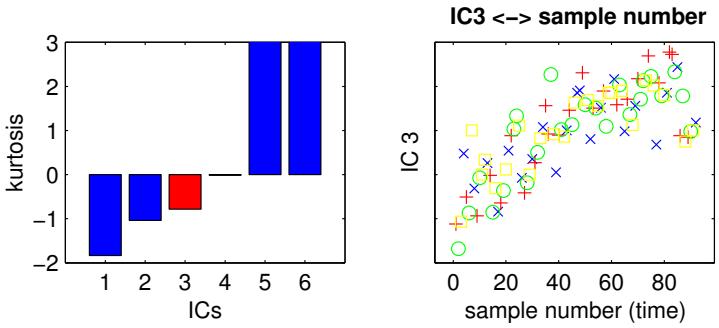


Figure 4.7: Three components with clearly negative kurtosis are detected. The third component (IC 3), an almost uniformly distributed factor, could be interpreted as an experimental artifact, related to the sequence in which the samples were measured.

4.5 Application

ICA, applied to our test case of *Arabidopsis thaliana* crosses, identified three relevant independent components (ICs), i.e. three ICs with a significantly negative kurtosis value. A prior reduction of dimensionality to 5 or 6 principal components was necessary (Figure 4.5). The extracted independent components could be interpreted biologically. The first component, IC 1, can be used to discriminate between the crosses from the background parental lines. The second component, IC 2, contains information to discriminate the two parental lines (Figure 4.6). The third component, IC 3, is not related to the biological experiment. However, there is a relation to the identifier of the samples, representing the order over time of measurement in the mass spectrometer (Figure 4.7). Hence, IC 3 is an experimental artifact due to increasing contamination of the QTOF skimmer along the analytical sequence. This technical factor could not have been discovered by PCA.

4.6 ICA versus clustering

In molecular biology, cluster algorithms are frequently used to divide the full set of measured variables (genes, metabolites or proteins) into distinct subsets which describe clusters or groups of molecules (Eisen et al., 1998; Golub et al., 1999). The purpose is to find a separation where in each single cluster the molecules are functionally similar. Standard cluster techniques are k-means clustering, hierarchical clustering, or Gaussian mixture models (Bishop, 1995; Hastie et al., 2001). Although clustering is useful and intuitive, it suffers from the drawback that it aims to divide and partition variables. This is not entirely biologically plausible, since we know that a particular gene or metabolite may be involved in more than one biological process.

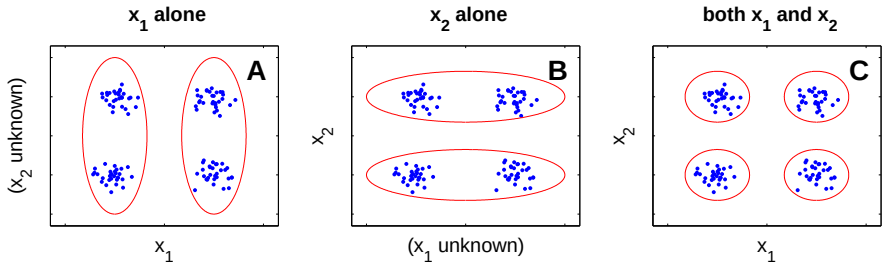


Figure 4.8: Cluster results strongly depend on the selected (observed) variables and on how they are weighted. If only the variable x_1 is known (or very strongly weighted) as in (**A**), the samples (dots '·') are clustered completely different to the result with the only known variable x_2 (**B**). If both variables are known (**C**), we obtain more and smaller clusters. Extended to more distinct variables, each sample will belong to its own cluster, as each sample will become very distinct from any other.
An increasing number of arbitrarily chosen and equally weighted variables leads to an increase of distinct information. This results in the inability to cluster samples, as each individual sample will then be equally distinct or similar to any other.

Even though ICA is no cluster algorithm, it is often more suitable for assigning molecules to different biological functions. In contrast to cluster algorithms, the objective of ICA is neither to separate the variables nor the samples. ICA attempts to find new coordinates (components) that are statistically independent. These components can often be interpreted biologically. As different sets of variables might contribute significantly to different components, there is some relation to clustering the variables. A component, given by a new variable y, is a weighted sum over all d variables x_i thus $y = w_1x_1 + w_2x_2 + \ldots + w_dx_d$. The largest absolute weight value w_i (sometimes referred to as loading) identifies the variable x_i (e.g., gene i) with the largest contribution to this specific component. Each weight value w_i can therefore be seen as a similarity measure between a variable and a component. To compare ICA with cluster results, a component can be seen as a cluster centre and the weight w_i can be considered as distance of a variable to this cluster centre. The main difference, however, is that in ICA the components are not optimised to separate variables into single groups. Theoretically, the variables could even contribute equally well to all components. The result whether a single variable (e.g., gene) contributes to one component (one biological function) or to several components, is purely driven by the biological data itself. There is no assumption or optimised criterion to enforce distinct and compact subsets of variables, as it is done in cluster algorithms.

Similarity in high dimensions

Another interesting problem occurs when we try to cluster samples and not variables as previously done. The result is often based on a very large number of variables as, for example, in gene expression data. The problem in measuring as many genes as possible is that the large number of variables may contain information on many distinct pro-

cesses. When we consider (weight) each variable equally, e.g., by normalising them to unit variance, different subsets of genes will result in different cluster solutions, where in each case different samples are grouped together. When all genes are used, the outcome simply depends on the genes we were able to measure. If we could measure other genes, we would obtain different results, see Figure 4.8.
A growing number of arbitrarily chosen variables leads to an increase of distinct information in the data. This results in an equal distinction or similarity of each individual sample to any other. This is known as Watanabe's *Ugly Duckling Theorem:* with no prior knowledge, the ugly duckling is as similar to a swan as one swan to another (Watanabe, 1985). With a bad choice of variables the ugly duckling can even be made more similar. The theorem states that if we do not weight some variables more strongly than others, everything would be equally similar to everything else. The theorem shows that it is impossible to have a universal notion of similarity. Any such notion must encode some assumptions by weighting some variables more than others.
In gene expression data, however, the variables are already weighted by variance (when not normalised to unit variance). Although variance is not always the optimal criterion, it can be a reasonable weight, especially for ratio values where a high ratio variance points to potentially important variables due to a large change in expression values.
However, to find the optimal weight can also be seen as a feature extraction problem. The aim is then to find one or a small set of components representing all information relevant for our research. The samples can then be grouped together according to this information by using the extracted components instead of all variables. The question of the optimal variable weight then becomes the question of the optimal criterion in component extraction — one of the key issues considered in this work.

4.7 Summary

We have shown that ICA can outperform classical PCA which is restricted to pure variance optimisation. The independence condition in ICA, by contrast, leads in general to components of greater discrimination and distinct meaning. Components of ICA are therefore often better related to biological factors than components of PCA.
To obtain optimal results, ICA has to be combined with a suitable pre-processing and a component ranking criterion. Important components in molecular data are distributed in a sub-Gaussian fashion, as opposed to sound signals with super-Gaussian distributions. We therefore ranked the extracted components by their kurtosis. PCA was used as pre-processing to reduce the dimensionality before ICA can be applied. Therefore, the overall component extraction criterion covers both variance in PCA pre-processing and higher order statistics in the subsequent ICA. The number of principal components in PCA defines how both criteria are balanced. The trade off between them can be found by a criterion that we have proposed.
The described approach was made available for public use in *MetaGeneAlyse*, a server-based system for molecular data analysis, accessible via a web-interface at `http://metagenealyse.mpimp-golm.mpg.de` .

Applied to our experiment of *Arabidopsis thaliana* crosses, ICA was able to detect both expected and unexpected factors. Two components were related to biological factors and hence confirmed our experiment, while a third component of clear relevance was discovered and could be interpreted as a technical artifact.
Due to assumed linear dependencies, ICA is a simplified model of reality, but nevertheless sufficient to describe many phenomena, and can provide valuable results. However, sometimes more complex nonlinear models are needed to analyse a new class of experiments, designed to observe variations continuously over time, as shown in the next chapter.

5 NLPCA — nonlinear PCA

So far we have focused on linear methods for molecular data analysis, in particular PCA and ICA. Linearity, in this context, means to search for important directions in the data space. The data can then be linearly transformed such that the positions of the samples along such a direction are explained by new variables referred to as components. As these components are restricted to be linear, we have to assume that the characteristics of the data can be explained by straight lines. This is reasonable as long as we consider experiments with two or a low number of discrete conditions. Typical experiments are those with disease and control or mutant and wild-type samples. Such samples are expected to be organised by clusters and can therefore be sufficiently discriminated by linear components, as illustrated in Figure 1.2 in the introduction chapter.
More complex nonlinear correlations, by contrast, become important with an increasing number of time experiments. That includes day and night rhythmicity as well as time dependent adaptation to changed environments. As individual molecules generally behave differently over time, the observed molecular data typically present a nonlinear (curved) structure. This means that the data are located within a nonlinear subspace, and hence can be better explained by a single or low number of nonlinear (curved) components, as illustrated in Figure 5.1. Such transformation can be regarded as *nonlinear dimensionality reduction*. Ideally, the components are related to the investigated experimental factors, most commonly to time.
Our main objective is hence to visualise and analyse the potential nonlinear structure of molecular data sets by components that are generalised from straight lines to curves. The components are required to explain as much information as possible in a least square error sense. This leads to a nonlinear generalisation of standard linear *principal component analysis* (PCA) — the *nonlinear principal component analysis* (NLPCA). Special emphasis is hereby placed on the challenging problem of identifying components from data sets with missing data.
We focus on an NLPCA based on a neural network — the *auto-associative neural network* (Kramer, 1991; DeMers and Cottrell, 1993; Hecht-Nielsen, 1995; Kirby and Miranda, 1996; Malthouse, 1998). It is successfully applied in the fields of atmospheric and oceanic sciences (Hsieh, 2004; Monahan et al., 2003), in astronomy and even in biomedical research. In Scholz and Vigário (2002) a hierarchically extended version of NLPCA was developed and applied to spectral data from stars and to electromyographic (EMG) recordings for different muscle activities.
Even though the term nonlinear PCA (NLPCA) is commonly referred to the auto-associative approach, there are many other methods which visualise data and extract meaningful components in a nonlinear manner. *Locally linear embedding* (LLE) (Roweis and Saul, 2000; Saul and Roweis, 2004) and *Isomap* (Tenenbaum et al., 2000) were de-

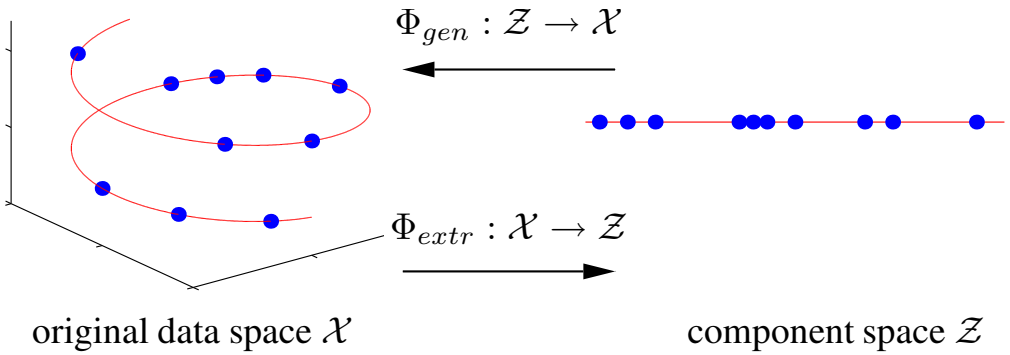


Figure 5.1: **Nonlinear dimensionality reduction.** Illustrated are three-dimensional samples that are located on a one-dimensional subspace, and hence can be described without loss of information by a single variable (the component). The transformation is given by the two functions Φ_{extr} and Φ_{gen}. The extraction function Φ_{extr} maps each three-dimensional sample vector (left) onto a one-dimensional component value (right). The inverse mapping is given by the generation function Φ_{gen} which transforms any scalar component value back into the original data space.
Such circular or helical trajectory over time is not uncommon in molecular data. The horizontal axes may represent metabolites that behave in a circadian rhythm, whereas the vertical axis might represent a metabolite with an increase in concentration due to adaptation to a stress situation.

veloped to visualise high dimensional data by projecting (embedding) them into a two or low-dimensional space. A mapping function as a nonlinear model is not explicitly given. *Principal curves* (Hastie and Stuetzle, 1989) and *self organizing maps* (SOM) (Kohonen, 2001) are useful for detecting nonlinear curves and two-dimensional nonlinear planes. Both methods are limited to the extraction of maximally two components, due to high computational costs. *Kernel PCA* (Schölkopf et al., 1998) is advantageous in noise reduction (Mika et al., 1999) or, when used as pre-processing, to improve classification results.
In the previous chapter, we have shown that ICA in general meets better our requirements than PCA. A nonlinear generalisation of ICA would therefore be of great interest. However, the nonlinear extension of ICA is not only very challenging but also intractable or non-unique in the absence of any a priori knowledge of the nonlinear mixing process. Therefore, special nonlinear ICA models simplify the problem to particular applications in which some information about the mixing system and the factors (source signals) is available, e.g., by using sequence information (Harmeling et al., 2003). A discussion of nonlinear approaches to ICA can be found in Jutten and Karhunen (2003); Cichocki and Amari (2003).
We restrict our attention to the more manageable task of a nonlinear PCA, motivated by the idea that nonlinear PCA, if performed perfectly, should in principle be able to remove all nonlinearities in the data such that a standard linear ICA can be applied

subsequently to achieve an overall nonlinear ICA. Even though this is more theoretical, practically, we will show that nonlinear PCA already generates the desired time component when the data are of adequate quality in the sense that there are no additional artifacts and only a small amount of almost Gaussian noise.
Our challenge is to model the nonlinear process even when the data have *missing values*. In addition, it should be possible to interpret the nonlinear molecular behaviour, for which we explicitly need the nonlinear mapping functions. This can be provided by the neural network based nonlinear PCA. We will show that it can be applied to incomplete data sets by modelling only the second part of the auto-associative neural network, the reconstruction or generation part. The difficulty herein is to estimate both the model weights and the inputs which are now the required components. This is sometimes referred to as a *blind inverse problem*.
All methods discussed in this chapter are again *unsupervised techniques*. They are based entirely on the observed molecular data itself, without reference to the corresponding experimental target data such as the time information. Thus, the risk of over-fitting is much lower than in supervised regression models. Furthermore, the response time and developmental state of plant individuals in any experiment differs from the exact physical time measurement. Hence we cannot absolutely trust the physical experimental time for the description of biological experiments. An unsupervised model will be superior in accommodating the unavoidable individual variability of biological specimens such as plants.

Data generation and component extraction

To extract components, linear as well as nonlinear, we assume that the data are driven by a number of factors and hence can be considered as being generated from them. Since the number of varied factors are usually smaller than the number of observed variables the data are located on a subspace of the given data space. The aim is to describe these factors by components, which are embedded in the data space and thereby explain this subspace. Nonlinear PCA is not being limited to linear components and hence the subspace can be curved, as illustrated in Figure 5.1.
We have a data space $\mathcal{X}$ given by the observed variables and a component space $\mathcal{Z}$ which is a subspace of $\mathcal{X}$. Nonlinear PCA aims to find both the subspace $\mathcal{Z}$ and the mapping between $\mathcal{X}$ and $\mathcal{Z}$. The mapping is given by nonlinear functions Φ_{extr} and Φ_{gen}. The *extraction* function $\Phi_{extr} : \mathcal{X} \to \mathcal{Z}$ transforms the sample coordinates $x = (x_1, x_2, \ldots, x_d)^T$ of the d-dimensional data space $\mathcal{X}$ into the corresponding coordinates $z = (z_1, z_2, \ldots, z_k)^T$ of the component space $\mathcal{Z}$ of usually lower dimensionality k. The *generation* function $\Phi_{gen} : \mathcal{Z} \to \hat{\mathcal{X}}$ is the inverse mapping which reconstructs the original sample vector x from their lower-dimensional component representation z. Thus, Φ_{gen} approximates the assumed data generation process.
While the auto-associative network provides a model for both the extraction and the generation process, we will show that modelling the generation process alone offers a lot of advantages concerning missing data and inverse problems.

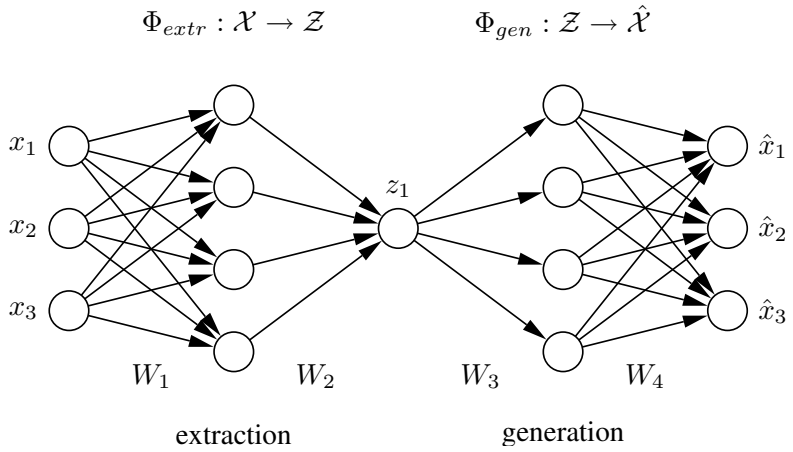


Figure 5.2: **Standard auto-associative neural network.** The network output $\hat{x}$ is required to be equal to the input x. Illustrated is a [3-4-1-4-3] network architecture. Biases have been omitted for clarity. Three-dimensional samples x are compressed (projected) to one component z in the middle by the extraction part. The inverse generation part reconstructs $\hat{x}$ from z. The sample $\hat{x}$ is usually a noise-reduced representation of x. The second and fourth hidden layer, with four *nonlinear* units each, enable the network to perform nonlinear mappings. The network can be extended to extract more than one component by using additional nodes in the component layer in the middle.

5.1 Standard auto-associative neural network

The nonlinear PCA (NLPCA), proposed by Kramer (1991), is based on a multi-layer perceptron (MLP) with an auto-associative topology, also known as an autoencoder, replicator network, bottleneck or sandglass type network. A good introduction to multi-layer perceptrons can be found in Bishop (1995) and Haykin (1998).
The auto-associative network performs the identity mapping. The output $\hat{x}$ is enforced to equal the input x with high accuracy. This is achieved by minimising the square error $\| x - \hat{x} \|^2$.
This is no trivial task, as there is a 'bottleneck' in the middle, a layer of fewer nodes than at the input or output, where the data have to be projected or compressed into a lower dimensional space Z.
The network can be considered as two parts: the first part represents the extraction function $\Phi_{extr} : \mathcal{X} \to \mathcal{Z}$, whereas the second part represents the inverse function, the generation or reconstruction function $\Phi_{gen} : \mathcal{Z} \to \hat{\mathcal{X}}$. A hidden layer in each part enables the network to perform nonlinear mapping functions.
In the following we describe the applied network topology by the notation [l_1-l_2-...- l_n] where l_i is the number of units in layer i: the input, hidden, component, or output layer. For example, [3-4-1-4-3] specifies a network with three units in the input and output layer, four units in both hidden layers, and one unit in the component layer, as illustrated in Figure 5.2.

5.2 Hierarchical nonlinear PCA (h-NLPCA)

In order to decompose data in a PCA related way, linearly or nonlinearly, it is important to distinguish applications where a pure *dimensionality reduction* is required from applications where the identification and discrimination of unique and meaningful components is of primary interest, usually referred to as *feature extraction*. In applications of pure dimensionality reduction, with clear emphasis on noise reduction and data compression, only a subspace with high descriptive capacity is sought. How the individual components form this subspace is not particularly constrained and hence does not need to be unique. The only requirement is that the subspace explains, in the mean square error (MSE) sense, as much information as possible. Since the individual components which jointly explain this subspace, are treated equally by the algorithm without any particular order or differential weighting, this is referred to as symmetric type of learning. This also includes the nonlinear PCA performed by the standard auto-associative neural network which is therefore referred to as s-NLPCA in the following.
By contrast, *hierarchical nonlinear PCA (h-NLPCA)*, as proposed by Scholz and Vigário (2002), provides not only the nonlinear subspace spanned by the optimal set of components, it also enforces the nonlinear components to have the same hierarchical order as the linear components in standard PCA. The h-NLPCA can therefore be seen as a true and natural nonlinear extension to PCA.
Hierarchy, in this context, is explained by two important properties: scalability and stability. Scalability means that the first n components explain as much as possible of the variance in a n-dimensional subspace of the data. Stability means that the i-th component of an n component solution is identical to the i-th component of an m component solution ($m \neq n$).
A hierarchical order essentially yields uncorrelated components. Nonlinearly, this even means that h-NLPCA is able to remove complex nonlinear correlations between components. This can already yield useful and meaningful components as will be shown in section 5.6 when applied to molecular data. Additionally, by scaling the nonlinear uncorrelated components to unit variance, we obtain a complex nonlinear whitening (sphering) transformation as shown in Scholz and Vigário (2002). This is a useful preprocessing step for applications such as regression, classification, or blind separation of sources. Since a nonlinear whitening removes the nonlinearities in the data, the subsequently applied methods can then still be linear. This is particularly important for ICA which can be extended to a nonlinear approach by using a nonlinear whitening.
How can we achieve such a hierarchical order? The naive approach to simply sort the symmetrically treated components by variance does not yield the required hierarchical order, neither linearly nor nonlinearly. In the simple linear case, we can achieve hierarchically ordered components by a sequential (deflationary) approach in which the components are successively extracted, one after the other, on the remaining variance (error) of the previous ones. However, this does not work sufficiently well in the nonlinear case. The remaining variance cannot be considered regardless of the nonlinear mapping.
However, there are two strongly related ways to introduce hierarchy constraints to the

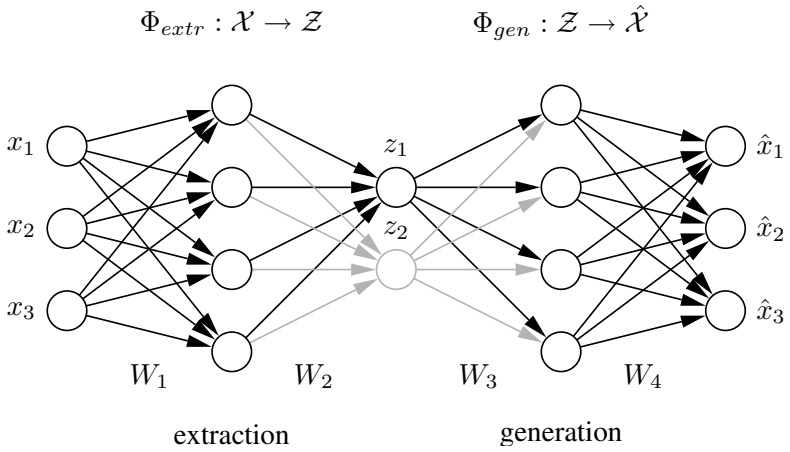


Figure 5.3: **Hierarchical auto-associative neural network**. The standard auto-associative network is hierarchically extended to perform the hierarchical NLPCA (h-NLPCA). In addition to the whole [3-4-**2**-4-3] network (grey+black), there is a [3-4-**1**-4-3] subnetwork (black) explicitly considered. The component layer in the middle has either one or two nodes which represent the first and second components respectively. In each iteration the error E_1 of the subnetwork with one component and the error of the total network with two components are estimated separately. The network weights are then adapted jointly with regard to the total hierarchic error $E = E_1 + E_{1,2}$.

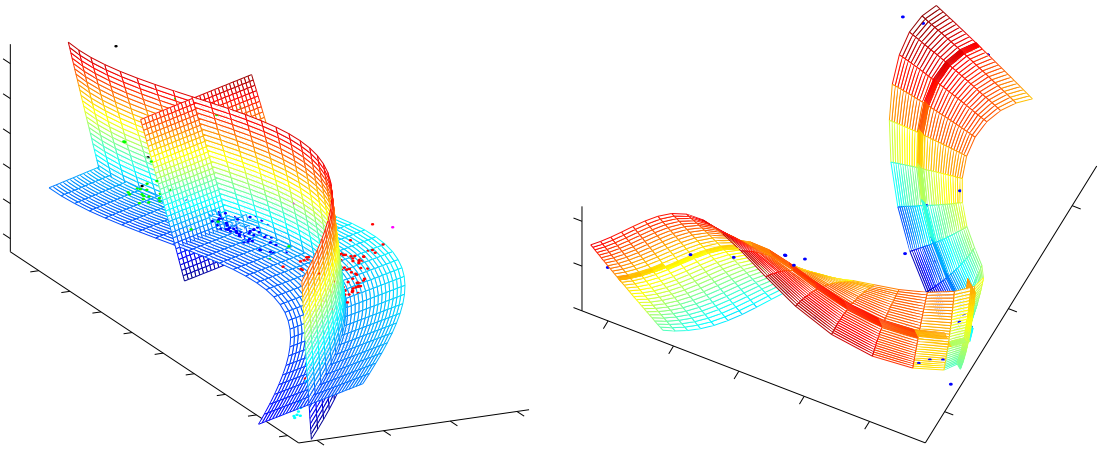
component space. In much the same way as in linear PCA, one is to force the i-th component to account for the i-th highest variance projection. Another strategy would be to search in the original data space for the smallest mean squared reconstruction error while using the first i components. The former may be harder or even impossible to solve than the latter, due to boundary conditions. Hence, we will present a learning strategy that focuses on the reconstruction mean square error (MSE), $E = \frac{1}{dN} \sum_n^N \sum_k^d (x_k^n - \hat{x}_k^n)^2$, where x and $\hat{x}$ are, respectively, the original and the reconstructed data. N is the number of samples, d is the dimensionality. For simplicity, we first restrict our discussion to the case of a two-dimensional component space. All conclusions can be generalised to any other dimensionality.

The hierarchical error function

E_1 and $E_{1,2}$ are the mean reconstruction errors when using only the first or both the first and the second component respectively. In order to perform the h-NLPCA, we have to impose not only a small $E_{1,2}$ (as in s-NLPCA), but also a small E_1. This can be done by minimising the hierarchical error:

$$E_H = E_1 + E_{1,2}$$

To find the optimal network weights for a minimal error in the h-NLPCA as well as in the standard symmetric approach, the *conjugate gradient descent* algorithm (Hestenes



Star data EMG data

Figure 5.4: Hierarchical nonlinear PCA (h-NLPCA) applied to a star spectral data set and to electromyographic (EMG) recordings. Both data sets show a clear nonlinear behaviour. The first three nonlinear components are visualised in the space of the first three PCA components. The grids represent the new coordinate system of the component space. Each grid is spanned by two of the three components while the third is set to zero.

and Stiefel, 1952; Press et al., 1992) is used. At each iteration, the single error terms E_1 and $E_{1,2}$ have to be calculated separately. This is performed in the standard s-NLPCA way by a network either with one or with two units in the component layer. Here, one network is the subnetwork of the other, as illustrated in Figure 5.3. The gradient ∇E_H is the sum of the individual gradients $\nabla E_H = \nabla E_1 + \nabla E_{1,2}$. If a weight w_i does not exist in the subnetwork, $\frac{\partial E_1}{\partial w_i}$ is set to zero.
To regularise the network, a *weight decay* term is added $E = E_H + \nu \sum_i w_i^2$. In most experiments, $\nu = 0.001$ was a reasonable choice. Furthermore, to achieve more robust results, the weights of the nonlinear layer were initialised such that the sigmoidal nonlinearities worked in the linear range. It corresponds to start the h-NLPCA network with the simple linear PCA solution.
The hierarchical error function can be easily extended to k components ($k \leq d$):

$$E_H = E_1 + E_{1,2} + E_{1,2,3} + \ldots + E_{1,2,3,\ldots,k}$$

The h-NLPCA given by E_H can then be interpreted as follows: we search for a k-dimensional subspace of minimal mean square error (MSE) under the constraint that the $(k-1)$-dimensional subspace is also of minimal MSE. This is successively extended such that all $1, \ldots, k$ dimensional subspaces are of minimal MSE. Hence, each subspace represents the data with regard to its dimensionality best.

Application of h-NLPCA

To illustrate the performance of the hierarchical approach, we applied h-NLPCA to two separate data sets (Scholz and Vigário, 2002). The first consists of 19-dimensional spectral information, gathered from 487 stars, see Stock and Stock (1999) for more details on this data set. The second data set is based on electromyographic (EMG) recordings for different muscle activities (labelled as 0, 10, 30, 50 and 70% of maximal personal strength). The one-dimensional EMG signal is then embedded into a d-dimensional space and analysed as a recurrence plot (Webber Jr. and Zbilut, 1994). The final data set then consists of 10 recurrence qualification analysis (RQA) variables for 35 samples (the 5 force levels for each of the 7 subjects). For more details on this data set, see Mewett et al. (2001).
The nonlinear components are extracted by minimising the hierarchical error function $E_H = E_1 + E_{1,2} + E_{1,2,3}$. The auto-associative mappings are based on a [19-30-10-30-19] network for the star spectral data and a [10-7-3-7-10] network for the EMG data.
Figure 5.4 shows that both data sets have clear nonlinear characteristics. While in the star data set the nonlinearities seem moderate, this is clearly not the case for the EMG data. Furthermore, in the EMG plotting, it seems that most of the variance is explained by the first two components. The principal curvature given by the first nonlinear component was found to have a clear relation to the force level, see Scholz and Vigário (2002). The second component is not related to the force. Since the force information seems to be completely explained by the first component, the second component might be related to another, so far unknown physiological factor.

5.3 Inverse model of nonlinear PCA

In this section we propose nonlinear PCA as an inverse problem. While the classical forward problem consists of predicting the output from a given input, the inverse problem involves estimating the input which matches best a given output. When the model or data generating process is not known, this is referred to as a *blind inverse problem*.
In the simple linear case PCA can be considered equally well either as a forward or an inverse problem depending on whether the desired components are predicted as outputs or estimated as inputs by the respective algorithm. Sanger's learning rule (Sanger, 1989) is an example for a forward model of PCA. The auto-associative network models both the forward and the inverse model simultaneously. The forward model is given by the first part, the extraction function $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$. The inverse model is given by the second part, the generation function $\Phi_{gen} : \mathcal{Z} \rightarrow \hat{\mathcal{X}}$. Even though a forward model is appropriate for linear PCA, it is less suitable for nonlinear PCA. Nonlinear PCA is not always a one-to-one mapping. Two identical samples x^n may correspond to distinct component values z^n as illustrated for the self-intersection in Figure 5.6, left graph. The two identical samples might, for example, belong to different times, 24h and 48h, which represent both the same physiological state in a diurnal rhythm of a day and night experiment.

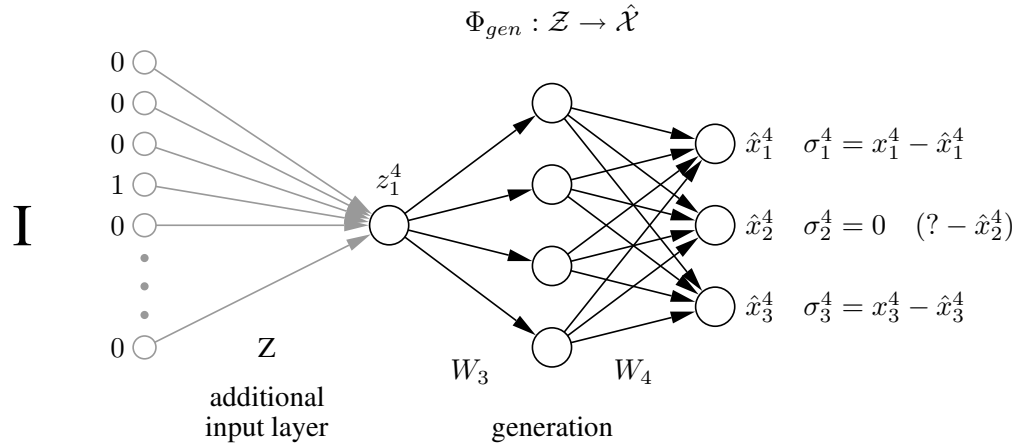


Figure 5.5: **The proposed inverse NLPCA model.** Only the second part of the auto-associative network (Figure 5.2) is needed, as illustrated by a [1-4-3] network (black). This generation part represents the inverse mapping Φ_{gen} which generates or reconstructs higher-dimensional samples x from their lower dimensional component representations z. These component values z are now unknown inputs that can be estimated by propagating the partial errors σ back to the input layer z. This is equivalent to the illustrated prefixed input layer (grey), where the weights are representing the component values z. The input is then a (sample x sample) identity matrix I. For the 4th sample (n=4), as illustrated, all inputs are zero except the 4th, which is one. On the right, the second element x_2^4 of the 4th sample x^4 is missing. Therefore, the partial error σ_2^4 is set to zero, identical to ignoring or non-back-propagating. The parameter of the model can thus be estimated even when there is missing data.

To model nonlinear PCA as a forward extraction process, $\mathcal{X} \rightarrow \mathcal{Z}$, is therefore difficult and sometimes even impossible. Even for more moderate nonlinearities, it is easier to model the inverse mapping Φ_{gen} from components z to data x, since it matches better the assumed generative model. This means, as explained in the ICA chapter, that given the values of all internal or external factors (e.g., the exact time), the corresponding molecular response (e.g., all metabolite concentrations) can be functionally derived. The opposite extraction mapping $\mathcal{X} \rightarrow \mathcal{Z}$ might be functionally very complex or may even exist only as one-to-many mapping which cannot be modelled by a single function.
Consequently, modelling the inverse mapping $\Phi_{gen} : \mathcal{Z} \rightarrow \hat{\mathcal{X}}$ alone, provides a number of advantages: we only need to optimise the second part of the auto-associative network, which is more efficient than using both parts. Also, we model the natural process, which has generated the observed samples, hence we can be sure that such a function exists, which is not always the case for the extraction model. And, most importantly, we can extend such an inverse NLPCA model to be applicable to incomplete data sets. This is possible, since the sample data are only used to determine the error of the model output, which can be estimated even with missing values in the data set. By contrast, using the data set as input, as done in a standard forward model, a complete data matrix would usually be necessary.

The challenge is that the desired components now are unknown inputs. Now, the *blind inverse problem* is to estimate both the inputs and the parameters of the model by only given outputs. This makes sense only with the additional constraint of a lower dimensional input.
For this approach Hassoun and Sudjianto (1997) optimised the weights and the inputs in two alternate steps by minimising an error function which is equivalent to the maximum likelihood. A similar approach was also used by Oh and Seung (1998). As the inputs can be represented by weights, we propose to optimise the inputs and weights simultaneously. The same network architecture is also used by Valpola for a nonlinear factor analysis (NFA) and a nonlinear independent factor analysis (NIFA) (Lappalainen and Honkela, 2000; Honkela and Valpola, 2005).
In the proposed inverse NLPCA approach, one single error function is used to optimise both the model weights w and the components as inputs z simultaneously. The model is applicable to incomplete data sets and can be used in a hierarchical mode.

The inverse network model

Inverse NLPCA is given by the mapping function Φ_{gen}, which is represented by a multi-layer perceptron (MLP), as illustrated in Figure 5.5. The output $\hat{x}$ depends on the input z and the network weights $w \in W_3, W_4$.

$$\hat{x} = \Phi_{gen}(w, z) = W_4 g(W_3 z)$$

The nonlinear activation function g (e.g., *tanh*) is applied element-wise. Biases are not explicitly considered, however, they can be included by introducing an extra unit, or input, with activation set to one.
The aim is to find a function Φ_{gen} which generates data $\hat{x}$ that approximate the observed sample data x by a minimal square error $\parallel x - \hat{x} \parallel^2$. Hence, we search for a minimal error depending on w and z: $\min_{w,z} \parallel x - \Phi_{gen}(w, z) \parallel^2$. Both the lower dimensional component representation z and the model parameter w are unknown and can be estimated by minimising the mean square reconstruction error:

$$E(w, z) = \frac{1}{dN} \sum_{n}^{N} \sum_{i}^{d} \left[x_i^n - \sum_{j}^{h} w_{ij} g \left(\sum_{i}^{m} w_{jk} z_k^n \right) \right]^2 .$$

The dimensionality d is given by the number of metabolites, N is the number of samples. The error can be minimised by a gradient optimisation algorithm, e.g., *conjugate gradient descent* (Hestenes and Stiefel, 1952; Press et al., 1992). The gradients are obtained by propagating the partial errors σ_i^n back to the input layer. For the input gradients it is simply one step further than usual. The gradients of the weights $w_{ij} \in W_4$, $w_{jk} \in W_3$

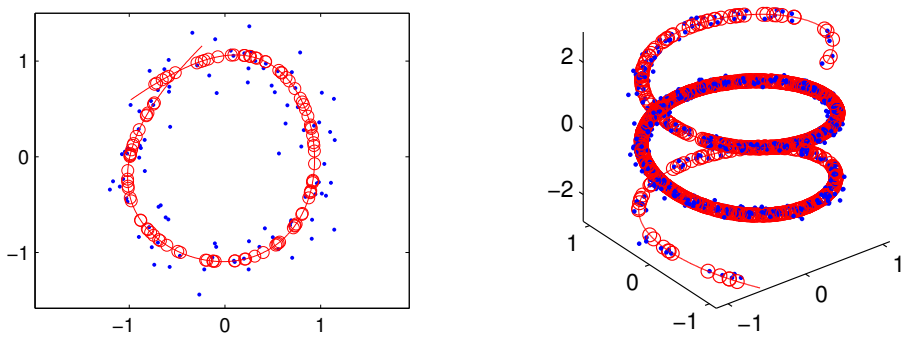


Figure 5.6: Approximation of a circular (left) and a helical (right) structure by the proposed inverse NLPCA model. The noisy data x (dots '$\cdot$') are projected onto a one-dimensional nonlinear component (line). The projection or de-noised reconstruction $\hat{x}$ is marked by a circle '$\circ$'. Note that an inverse model is able to extract self-intersecting components (left).

and inputs z_k^n are the partial derivatives:

$$\begin{array}{lcll} \frac{\partial E}{\partial w_{ij}} & = & \sum_n \sigma_i^n g(a_j^n) & ; \quad \sigma_i^n = \hat{x}_i^n - x_i^n \\ \frac{\partial E}{\partial w_{jk}} & = & \sum_n \sigma_j^n z_k^n & ; \quad \sigma_j^n = g'(a_j^n) \sum_i w_{ij} \sigma_i^n \\ \frac{\partial E}{\partial z_k^n} & = & \sigma_k^n & ; \quad \sigma_k^n = \sum_j w_{kj} \sigma_j^n \end{array}$$

For the *bias*, additional weights w_{i0} and w_{j0} can be used, with associated constants $z_0 = 1$ and $g(a_0) = 1$.
The weights w and the inputs z can be optimised simultaneously by considering (w, z) as one vector to optimise with given gradients. This would be equivalent to an approach where an additional input layer is representing the components z as weights, and new inputs are given by a (sample x sample) identity matrix, as illustrated in Figure 5.5. However, this layer is not needed for implementation. The purpose of the additional input layer is only to explain that the inverse NLPCA model can be converted to a conventionally trained multi-layer perceptron, with known inputs and simultaneously optimised weights, including the weights z, representing the desired components. Hence, an alternating approach as done by Hassoun and Sudjianto (1997) is not necessary. Besides a more efficient optimisation, it also avoids the risk of oscillations during training in an alternating approach.
A disadvantage of such an inverse approach is that there is no mapping function $\mathcal{X} \rightarrow \mathcal{Z}$ required for new data x. However, we can approximate the mapping by searching for an optimal input z to a given new sample x. For that, the network weights w have to be set constant and the input z has to be optimised to minimise the square error $\| x - \hat{x}(z) \|^2$. This is only a line search (in case of one component) or low dimensional optimisation with given gradients, efficiently done by a gradient optimisation algorithm.

The inverse NLPCA is able to extract components of higher nonlinear complexity than the standard NLPCA, even self-intersecting components can be modelled. This is shown in Figure 5.6 for a circular structure in two dimensions, generated from a uniformly distributed factor t (the angle) and a helical structure embedded in three dimensions, generated from a Gaussian distributed factor t. For the uniformly distributed 100 circular data points (plus noise), a [1-3-2] network is trained in 3,000 iterations. The noisy helical structure of 1,000 Gaussian distributed data points is modelled with a [1-8-3] network in 10,000 iterations.
The inverse NLPCA is not restricted to one component. It can be extended to m components by increasing the number of units in the input layer, the component layer z, to m. By using the hierarchical error function, proposed in section 5.2, the nonlinear components $1, \ldots, m$ can be extracted hierarchically in order to achieve a hierarchical nonlinear PCA as an inverse model as well.

5.4 Missing value estimation

A common problem in molecular data analysis is the absence of numerous values in the data set. The reason might either be that the particular value could not be measured or had been discarded due to high inaccuracy or low reliability. Since usually the number of samples is small and the proportion of affected samples is high, we cannot simply discard those incomplete samples from the data set. Instead, we have to find a way to estimate the missing values or to adapt our analysis method to be applicable to incomplete data. A common approach would be to replace each missing value by the mean or median over the available values of the corresponding variable. However, such an approach considers each variable separately and therefore can lead to poor results (Scholz et al., 2005). More successful approaches attempt to use all information available from the incomplete data set. This includes essentially the relations or dependencies among variables. There are many methods for estimating missing values (Little and Rubin, 2002). Some good approaches are based on maximum likelihood in conjunction with an expectation-maximisation (EM) algorithm (Ghahramani and Jordan, 1994). To analyse incomplete data, it is common to estimate the missing values first. The completed data set is then used in a subsequent analysis step.
Such separation of missing value estimation from the final analysis, however, can lead to problems when distinct or even incompatible assumptions are used in the two steps. Again, this concerns essentially questions about the optimal distance measure and importance in the data structures. It is crucially important in unsupervised techniques where the emphasis in general is to investigate the data structure under some specific criteria. One missing data approach might, for example, be optimised to estimate missing values in the mean square error sense, whereas in the final approach relative changes or correlation coefficients might be of interest. Or, concerning our approach, estimating missing values by a linear technique assumes a linear data structure. The final nonlinear analysis, however, is based on the opposed assumption of nonlinearly structured data. In missing value estimation, we modify the data structure by adding new values. Since

these values depend critically on our assumptions, our assumptions may become real in the sense that the data move to a linear structure, by applying a linear technique, even if they were originally nonlinear.
Estimating missing data can therefore not always be seen independently from the subsequent analysis. The best missing data technique is that which estimates the missing values with respect to the final purpose of the analysis. Our strategy is therefore, instead of estimating missing values first, to adapt the analysis technique such that it is applicable to incomplete data sets. Thus, we focus on detecting nonlinear components from incomplete data sets, so in our approach missing values are not estimated a priori. However, once the nonlinear mapping is effectively modelled, the missing values can then be estimated as well. This is shown for an artificial data set and for experimental data in the following sections. Estimation results were compared with results of state-of-the-art estimation techniques. There are two PCA based linear techniques: the recently published Bayesian missing value estimation method for gene expressions (Oba et al., 2003) which is based on *Bayesian principal component analysis* (BPCA) (Bishop, 1999) and *probabilistic PCA* (PPCA) (Verbeek et al., 2002) based on Roweis et al. (2002). Furthermore, there are the k-nearest neighbour based approach *KNNimpute* (Troyanskaya et al., 2001) and a nonlinear estimation by *self organizing maps* (SOM).

5.4.1 Modified inverse model

The inverse NLPCA model can be extended to be applicable to incomplete data sets in the following way (Scholz et al., 2005). If the ith element x_i^n of the nth sample vector x^n is missing, the partial error σ_i^n is set to zero before back-propagating, hence this error is ignored, it has no contribution to the gradients. Thus, the nonlinear components are extracted by using all available observations. With these components the original data can be reconstructed, including the missing values. The network output $\hat{x}_i^n$ gives the estimation of the missing element x_i^n.
The same approach can be used to weight each measured value differently. This might be of interest when for each value an additional probability value (p-value) is available. Each partial error σ_i^n can then be weighted $\hat{\sigma}_i^n = p * \sigma_i^n$ before back-propagating. The contribution to the gradients can thereby be decreased. However, even though an individual weighting may be important, e.g., for gene expression data, our emphasis has been on the missing data approach so far.

5.4.2 Missing data: artificial data

Even though an artificial data set does not reflect the whole complexity of real biological data, it is useful to illustrate the problem of missing data and hence can give a better understanding of how the data are handled by various methods.
The inverse NLPCA approach was therefore first applied to an artificial data set and the results were compared with other missing value estimation techniques, the linear

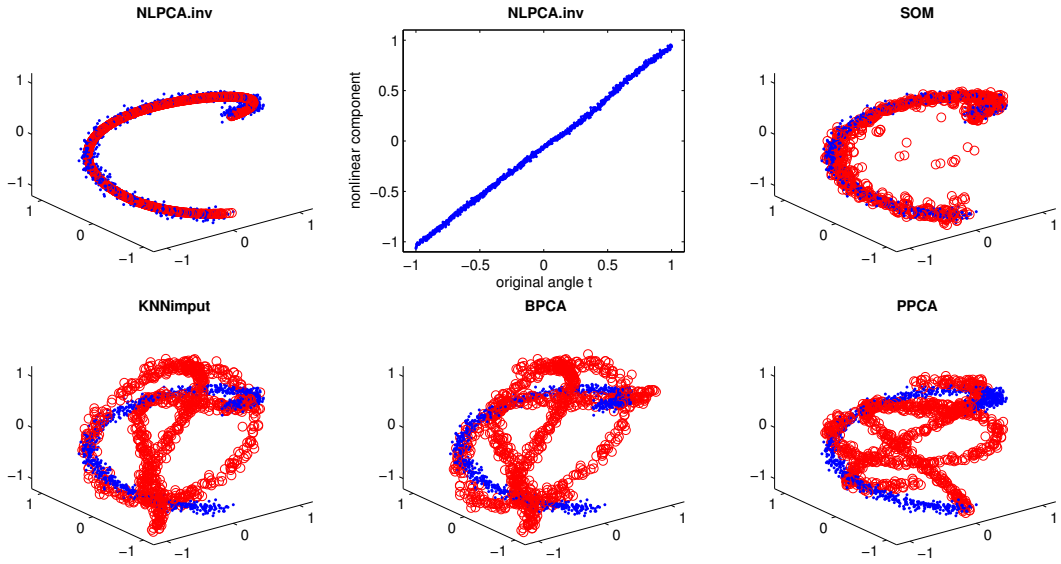


Figure 5.7: Artificial data were generated to test different missing value algorithms. The samples form a helical loop. From each of the three-dimensional samples, one value is removed and then estimated by each missing value algorithm. The known complete samples are plotted as dots '·' and the estimated values as circle '○'. Above: the inverse NLPCA is able to extract the nonlinear component from this highly incomplete data set, and hence it can give a very good estimation of the missing values. SOM also gives a reasonably good estimation, but the linear approaches BPCA and PPCA, as well as the k-nearest neighbour based approach KNNimpute, fail in this particular nonlinear case, see also Table 5.1.

techniques BPCA[1] and PPCA[2], the k-nearest neighbour based approach KNNimpute[3], and the nonlinear SOM[4]. The data x lie on a one-dimensional manifold (a helical loop) embedded in three dimensions, plus Gaussian noise η of standard deviation $\sigma = 0.05$, see Figure 5.7. 1,000 samples x were generated from a uniformly distributed factor t over the range [-1,1], t represents the angle:

$$\begin{aligned} x_1 &= \sin(\pi t) + \eta \\ x_2 &= \cos(\pi t) + \eta \\ x_3 &= t + \eta \end{aligned}$$

From each three-dimensional sample, one value is randomly removed and regarded as missing. This gives a high missing value rate of 33.3 percent. However, if the nonlinear component (the helix) is known, the estimation of a missing value is exactly given by the two other coordinates, except at the first and last position of the helix loop, where in

[1] http://hawaii.aist-nara.ac.jp/~shige-o/tools/
[2] http://carol.science.uva.nl/~jverbeek/software/
[3] http://smi-web.stanford.edu/projects/helix/pubs/impute/
[4] http://www.cis.hut.fi/projects/somtoolbox/

MSE of missing value estimation

	noise	noise-free
NLPCA.inv	0.0021	0.0013
SOM	0.0405	0.0384
KNNimpute	0.4435	0.4429
BPCA	0.4191	0.4186
PPCA (k=3)	0.4354	0.4347
mean	0.4429	0.4422

Table 5.1: Mean square error (MSE) of different missing value estimation techniques, applied to the helical data (Figure 5.7). The inverse NLPCA model provides a very good estimation of the missing values. Although the model was trained with noisy data, the noise-free data were better represented than the noisy data, confirming the de-noising ability of the model.
Also SOM gives a good estimation. The linear techniques BPCA and PPCA as well as KNNimpute fail to achieve good results. Their results are similar to the results of naive substitution by the mean over the residuals of one variable.

the case of missing vertical coordinate x_3, the sample can be assigned either to the first or to the last position. Consequently, there are two possible optimal solutions. Missing value estimation is not always unique in the nonlinear case.
In Figure 5.7 and Table 5.1 it is shown that even if the data sets are incomplete for all samples, the inverse NLPCA model is able to detect the nonlinear component and provides a very accurate missing value estimation. The SOM also achieves a reasonably good estimation, but the linear approaches BPCA and PPCA as well as the k-nearest neighbour based approach KNNimpute fail in this nonlinear data set case.

5.4.3 Missing data: metabolite data

The performance of the missing value estimation techniques was also assessed by using a real experimental data set. For that we used a completely available set of 140 metabolites from a cold stress experiment, see section 5.6 for more details. Different percentages of values were randomly removed and regarded as missing. A good overall missing value estimation is obtained for up to 50 percent of the missing values. This unexpectedly high tolerance might be caused by the high redundancy in the data, possibly due to high connectivity or dependency among the metabolites. By comparing the different techniques, we first found that BPCA gives the best average over all 140 metabolites, see Figure 5.8. But rather than in returning a good average we are interested in a good estimation for the most important metabolites. As our data values are ratios, see section 5.6.1, a high variance indicates an important metabolite. Therefore, we compared the performance for the first n metabolites of highest variance which mostly also show

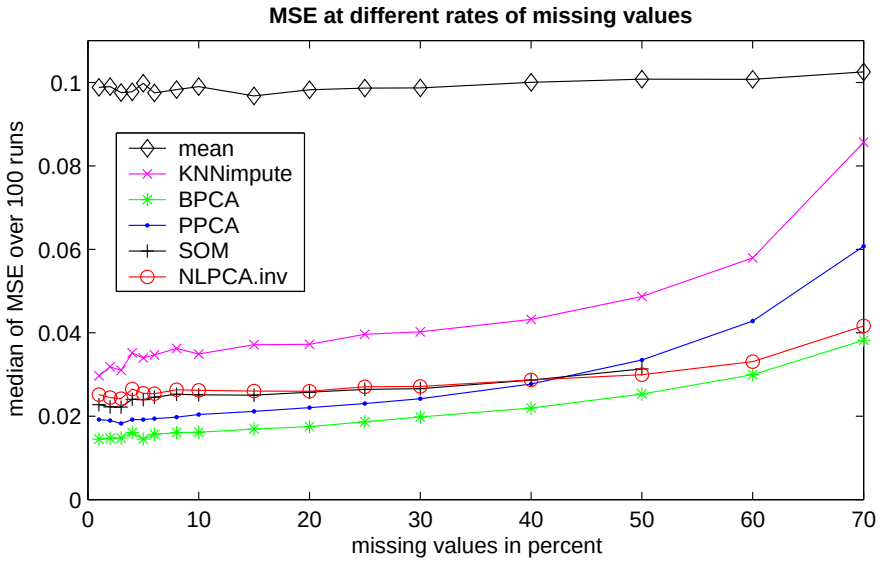


Figure 5.8: From an experimental data set of completely available 140 metabolites, different percentages of values were removed randomly and estimated by different missing value algorithms and repeated 100 times. The median of the mean square error (MSE) over all runs is plotted. An estimation by mean over the residual values produces the worst result. It is used as a base line. BPCA performs best. However, this is only the case when all 140 metabolites are considered, including the large number of non-relevant metabolites with small relative variances.

a strong nonlinear behaviour. Now the results are different, see Figure 5.9. The inverse NLPCA and SOM, which perform almost equally well, provide the best result for the first five most important metabolites, and perform almost equally well as PPCA for the remaining metabolites.

5.4.4 Missing data: gene expression data

In order to obtain a fair and comprehensive comparison, we also tested the performance of the missing data estimation by using a larger set of gene expression data obtained from the same cold stress experiment. The data were again transformed to $\log_2$ ratios, relative to the median of control samples at time zero. In total, 16,996 genes were reduced to 1,000 of highest log ratio variance. These genes are expected to be most important, as they show the largest relative expression change. Twenty-one samples were measured at seven different time points.
Again, instead of a good averaged missing value estimation over all genes, we are interested in a good estimation of the most important genes, those of highest relative variance. Therefore, the cumulative mean square error (MSE) for the first 30 genes of highest ratio variance is shown (Figure 5.10). The results differ from those on the metabolite data

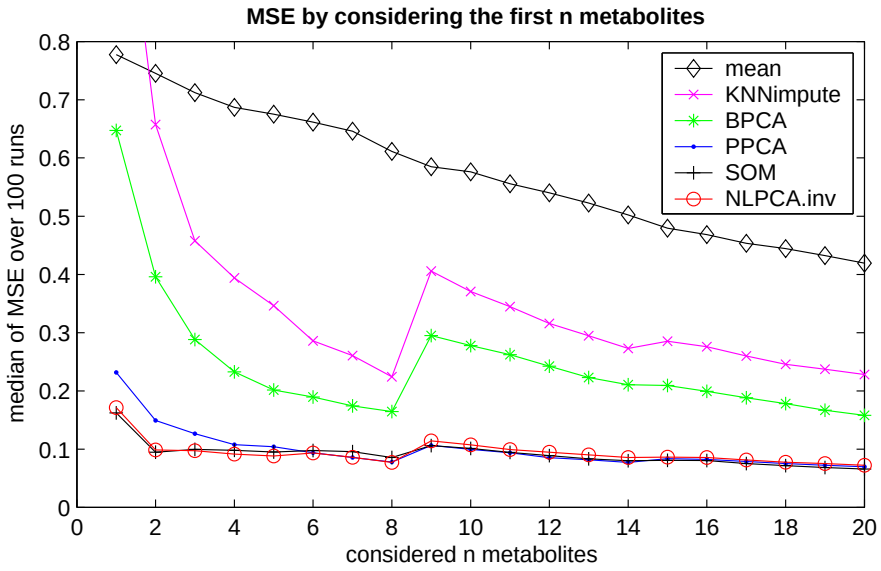


Figure 5.9: In contrast to Figure 5.8 we have considered only the top n metabolites of highest variance, n=1,...,20, at a fixed missing value rate of 10 %. As the data set contains ratios, metabolites of high variance are assumed to be important. The results differ from those in Figure 5.8. Here, BPCA does not perform satisfactorily, but still better than KNNimpute (k=10 neighbours). The best result of PPCA was given with k=5 components. However, at the first five metabolites, this result could still be outperformed by the nonlinear techniques, the inverse NLPCA and SOM, which perform almost equally well. All techniques show an abrupt rise at the 9th metabolite (citramalic acid), caused by badly distributed data.

set in Figure 5.9. All methods give quite similar but significantly better results than naive substitution by the mean of the remaining values of each gene. However, BPCA which was developed for this kind of high-dimensional data sets, gave the best result for both the averaged estimation (not shown) and the estimation for the first n genes as shown in Figure 5.10. BPCA is successful because it uses principal components in the lower dimensional data space given by the small number of samples and not by the genes. Similar results can therefore also be obtained by the similar technique of PPCA when applied to the transposed data set. However, the advantage of BPCA is that no parameter k, the number of used components, has to be chosen as it is necessary in PPCA. The results of NLPCA were also improved when applied to the transposed matrix, and by using more than one nonlinear component (k=4). However, there might be no advantage of a nonlinear technique applied to the transposed data set, as a nonlinear data structure in gene data space does not necessarily lead to a nonlinear structure in sample space (where genes are data points).
Consequently, for estimating missing values in large gene expression data sets, BPCA is a good choice. In data sets with a smaller number of variables, as is typical for metabolite

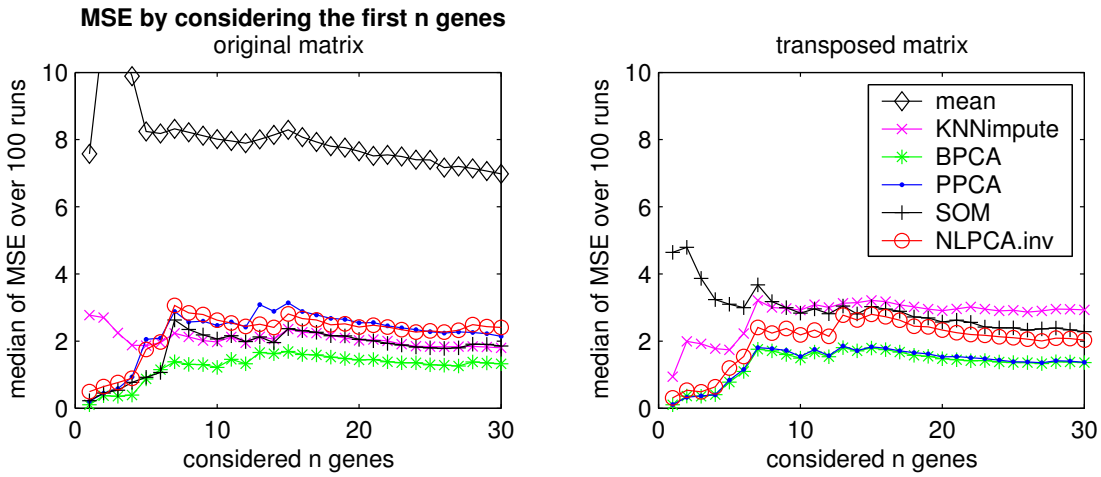


Figure 5.10: Missing data algorithms applied to gene expression data of 1,000 genes with 10 % randomly removed values. The results differ from those on metabolite data in Figure 5.9. Again, we consider the most important genes of highest ratio variance. The cumulative mean square error (MSE) is given for the first 30 genes of highest ratio variance. All algorithms achieve significantly better results than the naive substitution by the mean. The best result, though, is given by BPCA. Right: the results of most methods can be improved when applied to the transposed matrix. PPCA with k=5 components is then almost as good as BPCA which was applied alone without transposition, since it already has an internal transposition.

or protein data sets, other methods are more suitable. These include nonlinear techniques, such as NLPCA or SOM, when the data are nonlinearly distributed. Both gene expression and metabolite data are available at `http://nlpca.mpimp-golm.mpg.de`. However, our main objective is to detect nonlinear components in incomplete data sets. As these components should explain the experimental factors in the data space given by genes (where samples are data points), a transposed matrix is of no use.

5.5 Validation

In order to obtain reliable components, we have to validate the complexity of our model. This is even of much greater importance when we search for nonlinear components in molecular data with many dimensions (metabolites) and few samples. When the model has too little flexibility, it cannot fit the full complexity of the real process, e.g., nonlinear processes cannot be modelled sufficiently by linear methods. A model of too much flexibility, on the other hand, can even fit the nonrelevant noise in the data and hence gives a poor approximation of the original process. This is referred to as *overfitting* problem, illustrated in Figure 5.11. The aim is to find a model whose complexity is neither too small nor too large.

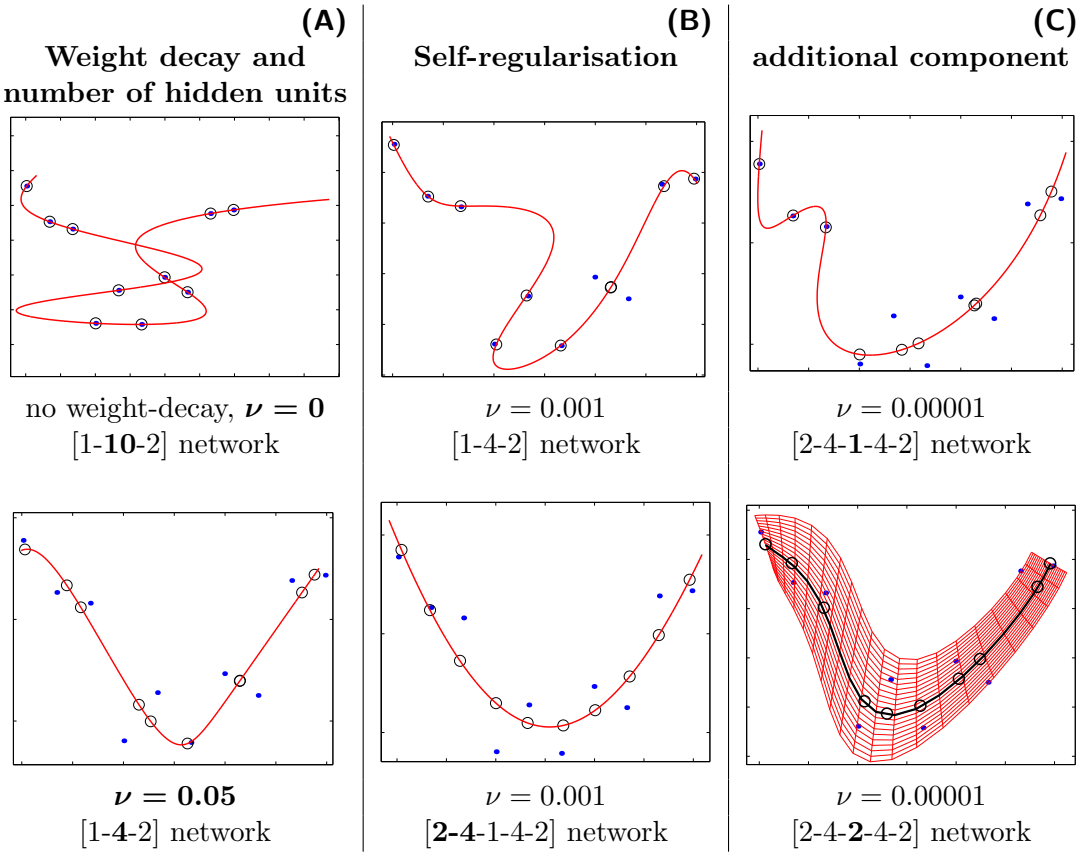


Figure 5.11: **Network-regularisation.** Column-wise illustrated is the effect of different complexity parameters. Shown is always a poor regularisation (above) and a good regularisation (below) with respect to the particular parameter. (**A**) The number of hidden units is decreased from ten to four and a weight-decay term is added to the error function with a strong influence of $\nu = 0.05$. (**B**) The inverse part alone (above) is more flexible than the entire auto-associative network (below) by the use of identical parameters. (**C**) The first component is regularised by hierarchically extracting an additional component.
The data samples x '·' are generated by a square function and additive Gaussian noise with standard deviation $\sigma = 0.4$. The projection $\hat{x}$ onto the first nonlinear component is marked by a circle '○'.

5.5.1 Model complexity

To control the complexity of the auto-associative network, different model parameters can be adjusted, as illustrated in Figure 5.11. This includes standard methods for neural networks such as the increase or decrease of the number of hidden units and the use of weight-decay (Hinton, 1987) as an additional regularisation term in the error function. Additionally, auto-associative neural networks have a kind of self-regularisation, caused

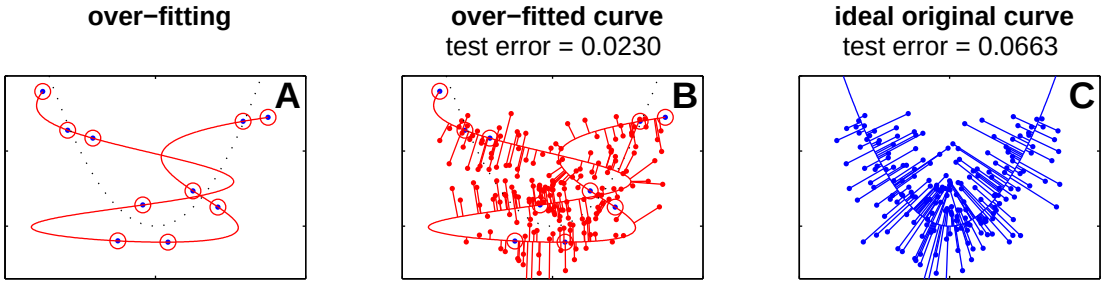


Figure 5.12: Illustrated is the test error of an over-fitted (**B**) and a well-fitted (**C**) model. (**A**) Ten samples were originally generated from a quadratic function (dotted line) plus noise. When we use a nonlinear PCA model of high complexity, a clearly over-fitted nonlinear component is obtained (solid line). However, when we validate the over-fitted model with an independent test data set (**B**), it gives a better (smaller) test error than by using the ideal model, the original model from which the data were generated (**C**). Thus, in contrast to supervised learning, such unsupervised models cannot be validated by using a test set, as in cross-validation. With increasing complexity, an unsupervised model is able to provide a curved component with increasing data space coverage, such that even test data can be projected onto the curve by a decreased distance (error).

by the fact that for each mapping function, the inverse function has to be estimated as well. A complex function usually has a much more complex inverse function or such an inverse function does not even exist. The auto-associative network is therefore constrained to keep the functions as simple as possible. When the inverse part is optimised alone, regularisation is then of greater importance. A similar effect is observed when extracting nonlinear components in a hierarchical order, where subsequent components are extracted in respect to the previous components. A complex first component would strongly increase the complexity of the second or later components. Thus, the network is constrained to generate very smooth first components.

5.5.2 The test set validation problem

A common approach to test the generalisation performance of a model is to use an independent test set, either by using a completely new data set if available, or when the number of samples is limited, by performing cross-validation by repeatedly splitting the original data into a training and test set. The motivation for this is that only the model, which represents the underlying biological process best, can provide optimal results on new, for the model previously unknown, data.
While test set validation is a standard approach in supervised applications, it suffers from the lack of a known target in unsupervised techniques. Unsupervised models can therefore not be validated by using a test set. Higher complex models, that over-fit the

original training data, can in principle be able to meet the specific criteria on test data better than it would be possible by the true or original model. This is illustrated in Figure 5.12 for 10 training and 200 test samples generated from a quadratic function plus Gaussian noise of standard deviation $\sigma = 0.4$. The test error refers to the mean square reconstruction error of the test data set on both an over-fitted and a well-fitted ideal model. Geometrically, the error is given by the squared distance between the data points and their projection onto the curve. Given the same test data, the ideal model results in an error almost three times larger than that of the overly complex model that over-fits the data.

To understand this contradiction, we have to distinguish between an error in supervised learning and the fulfilment of specific criteria in unsupervised learning. Test set validation works well as long as we measure the error as a distance or difference to a required target (e.g., class labels) which is the typical task in supervised learning. By contrast, in unsupervised learning the target (e.g., the correct component) is unknown, instead we search for a model that satisfies a specific criterion. Sometimes we even use supervised models to perform unsupervised analysis such as the multi-layer perceptron in nonlinear PCA. The error then means, in this unsupervised context, simply how well our criterion is achieved. Higher complex models can usually better satisfy this criterion, even on test data, as shown in Figure 5.12. When the only criterion is, for example, to project the data by the shortest way onto a curve, models of large flexibility can achieve a good performance on both the training and the test data. In addition to the criterion, we therefore have to restrict the complexity or flexibility of the model by a suitable regularisation. Hence, we have to find a way to determine the optimal model complexity, but we cannot use a test data set for the purpose of unsupervised validation.

For unsupervised techniques, instead of a test error, we can provide confidence in the results by determining the stability or robustness under slightly different variants of the original data set. A common method is to use the resampling technique *bootstrap* (Efron and Tibshirani, 1994). The idea behind this is that the result should not vary significantly when one or a few samples are discarded or if others count twice or more. Bootstrap can, for example, be used to estimate the reliability of independent components of ICA as proposed and successfully applied by Meinecke et al. (2002). Another validation approach for ICA, proposed by Harmeling et al. (2004), is to corrupt the data by a small amount of Gaussian noise. The motivation is that reliable components should be robust and stable against small random modification of the data.

Stability of linear components can then be determined by measuring the variation in the direction of specific components when applied to different resampled data sets or different noise-injections. Comparing nonlinear components, by contrast, is no trivial task. Curves can be explained by many properties concerning curvature, position, or the parameter of the nonlinear model that generates the curve. However, instead of a problematic comparison of these properties, we propose a new validation approach which does not concern stability and robustness of different results. Instead, we validate our model by measuring the performance of missing data estimation on a test set. It is motivated by the idea that only the model which best explains the real biological process, is able to give a good missing value estimation on new test data.

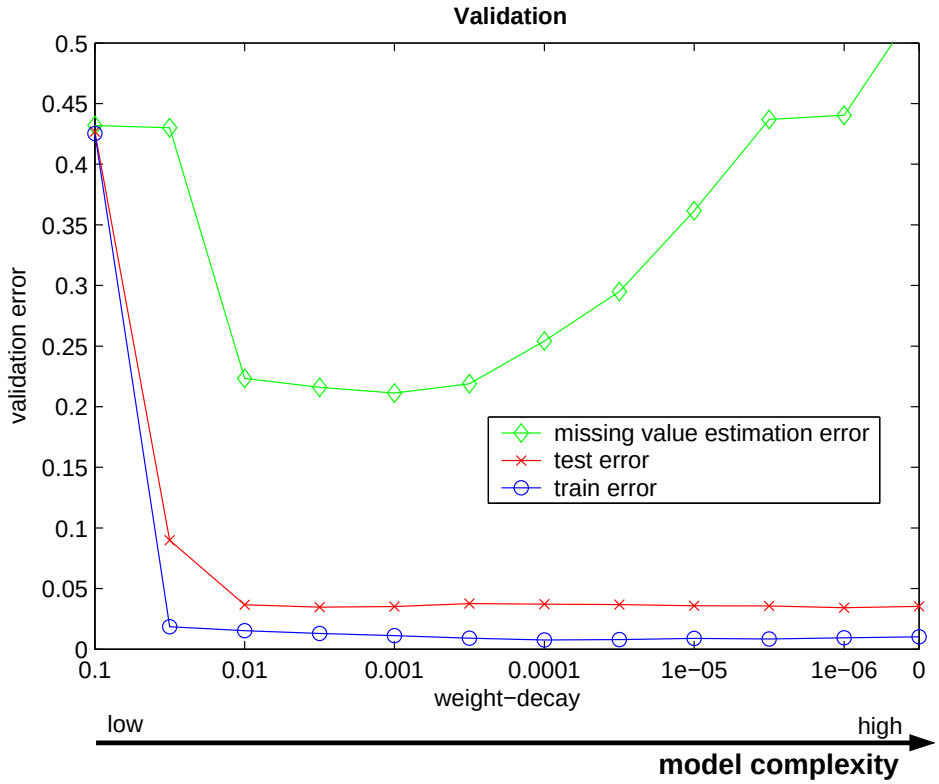


Figure 5.13: **Validation.** An artificial data set was used to determine the optimal complexity of the nonlinear PCA model. The complexity of the network is varied by using different weight-decay coefficients ν, while all other model parameters are kept constant. A network of low complexity which is almost linear (left) results in a high error as expected for both the training and the test data. However, when the model becomes increasingly complex, there is no increase in test error, even a slight decrease instead. This is contradictory to our knowledge from supervised learning, where the test error becomes worse when the model over-fits. However, when the validation is based on the missing data estimation performance, we obtain the intuitively expected result that the error becomes worse again with increasing model complexity. The optimal complexity is given by a clear minimum in the middle of the performance curve.

5.5.3 A missing data approach in model validation

A model that becomes increasingly complex is able to explain a more complicated structure in the data space. Even for new test samples, it is more likely to find a short projecting distance (error) onto a curve which covers the data space in a complete fashion than by a curve of moderate complexity (Figure 5.12). The problem, however, is that we can project the data onto *any* position on the curve. There is no further restriction

in pure test set validation. In missing data estimation, by contrast, the position on the curve is *fixed*, given by the remaining available values of the same sample. When a value is artificially removed and regarded as missing, we get an exact target. Thus, we changed the unsupervised validation problem into a supervised regression validation problem. We now test the predictive performance for an arbitrarily chosen value, given all other values of the same sample. This is done in any combination such that we obtain an averaged predictive error over values from all variables. Only the model which approximates the original process best, is therefore able to estimate missing values at highest accuracy. Even though missing value estimation is sometimes not unique in the sense that multiple distinct but still valid solutions may exist, it is, in general, a negligible problem, since it rarely appears in higher dimensions of large redundancy as typical for molecular data.
Since our nonlinear PCA method is able to estimate missing values, we can now use this property to validate the complexity of our model. The performance was measured by mean square error between a randomly removed value and its estimation by the nonlinear PCA model. This was done 100 times with newly generated data each time. The median over all 100 mean square errors was taken as an ultimate performance measure (Figure 5.13). The data consists of 20 training and 1,000 test samples, artificially generated from the same helical function as in Section 5.4.2 and additive Gaussian noise of standard deviation $\nu = 0.2$. A [1-10-3] network architecture was optimised in 5,000 iterations by using the *conjugate gradient descent* algorithm. The complexity of the model was changed by varying the impact of the weight-decay regularisation term in the error function.
The result, as shown in Figure 5.13, is that the test error does not become worse with increasing complexity, instead, it becomes even slightly better. This confirms again that even very complex and flexible nonlinear PCA models can achieve good performance on test data. Thus, there is no possibility to determine the optimal model complexity. The missing value validation approach, by contrast, provides a nice performance curve where the optimal complexity is shown by a clear minimum.
The true generalisation error in such an unsupervised technique is the missing value estimation error and not the classical test set error. The missing value approach can be seen as an adaptation of the standard test set validation to be applicable in unsupervised learning. It can easily be used in a cross-validation manner, in the case of a limited number of samples.
In contrast to robustness and stability validation approaches in ICA, the missing value approach is able to validate a single model instead of a set of models, provided that we have sufficient additional test data. This may be useful if we want to validate a particular model such as the ultimately chosen.
The approach is not restricted to one-dimensional curves, it can be extended to curved subspaces of higher dimensions, depending on the intrinsic dimension of the data.
The proposed validation approach is applicable only to models that can be additionally used for missing value estimations. It is therefore another favourable aspect for extending the nonlinear PCA to a missing data approach.

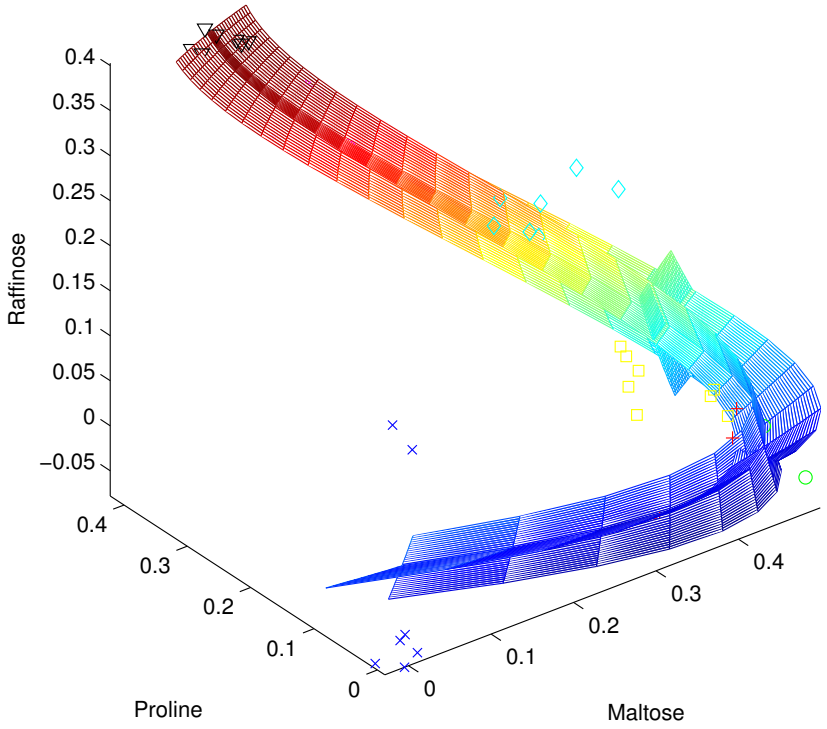


Figure 5.14: The first three extracted nonlinear components are plotted into the data space, given by the top three metabolites of highest variance. The grid represents the new coordinate system after the nonlinear transformation. The principal curvature, the first nonlinear component, shows the trajectory over time in the cold stress experiment. The additional second and third component only represent the noise in the data, but they are useful for controlling the complexity of the first component.

5.6 Application

Cold stress to cells can cause rapid changes in metabolite levels. Here, we have analysed the temporal metabolite response to cold stress at 4 °C in the model plant *Arabidopsis thaliana*, see also Scholz et al. (2005). The proposed inverse NLPCA model was applied to these, partly incomplete, metabolite data (Kaplan et al., 2004). This gives us an approximation of the mapping function from a given time point to the corresponding metabolite response, and hence we obtain a 'noise-free' model of the biological cold stress adaptation. For each time point t_i we are able to identify the metabolites in the order of importance, i.e. the metabolites are ranked by the relative change in their concentration level at this specific time point. This procedure is analogous to a ranked list of metabolites for one particular component in PCA or ICA.

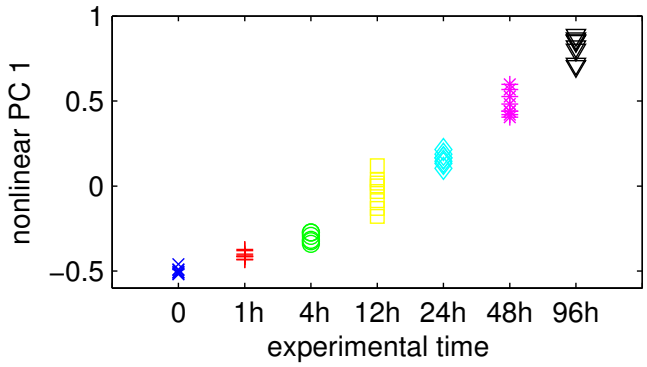


Figure 5.15: The extracted first nonlinear component represents the time factor. This relation is shown by plotting the first component against the observed experimental time.

5.6.1 Data acquisition

We used *gas chromatography / mass spectrometry (GC/MS)* to measure 497 metabolites at seven different time points, at 0, 1, 4, 12, 24, 48, and 96 hours; time point zero represents the control samples. Only 140 metabolites had available measurements for all samples, these metabolites were used in the previous section 5.4.3 to test the different methods for missing value estimation. In this experimental section the inverse NLPCA is applied to all metabolites which have less than 1/3 missing values. After removing 109 metabolites, the final data set contains 388 metabolites (140 complete, 248 incomplete) and 52 samples at seven different time points (7 - 8 samples per time point).
The data are transformed to log fold changes (log ratios). All measurements of each metabolite $x_i = (x_i^1, \ldots, x_i^{52})^T$ are divided by the median of the control samples at time point zero. Consequently, we are analysing ratios of metabolite concentrations with respect to a control time point. The logarithm $\log_2$ is used to get symmetric changes: $x_{normed} = \log_2\left(\frac{x}{median(x^{control})}\right)$.

5.6.2 Model parameters

We used a network with a [3-20-388] architecture as inverse NLPCA model. This means that we extracted three nonlinear components; 20 nonlinear hidden units were used to perform the nonlinear transformation, and 388 metabolites were approximated. The training was done in 300 iterations. To limit the complexity of the model we also added a weight decay term to the error function $E_{total} = E + \nu\left(\sum_i w_i^2 + \sum_j z_j^2\right)$ with $\nu = 0.001$ and we extracted the second and third component in a hierarchical order (Scholz and Vigário, 2002), which stabilises the first component.
The inverse NLPCA model yields a nonlinear transformation from three estimated nonlinear components to a 388 dimensional metabolite data set. This is shown in Figure 5.14 for the top three metabolites of highest variance.

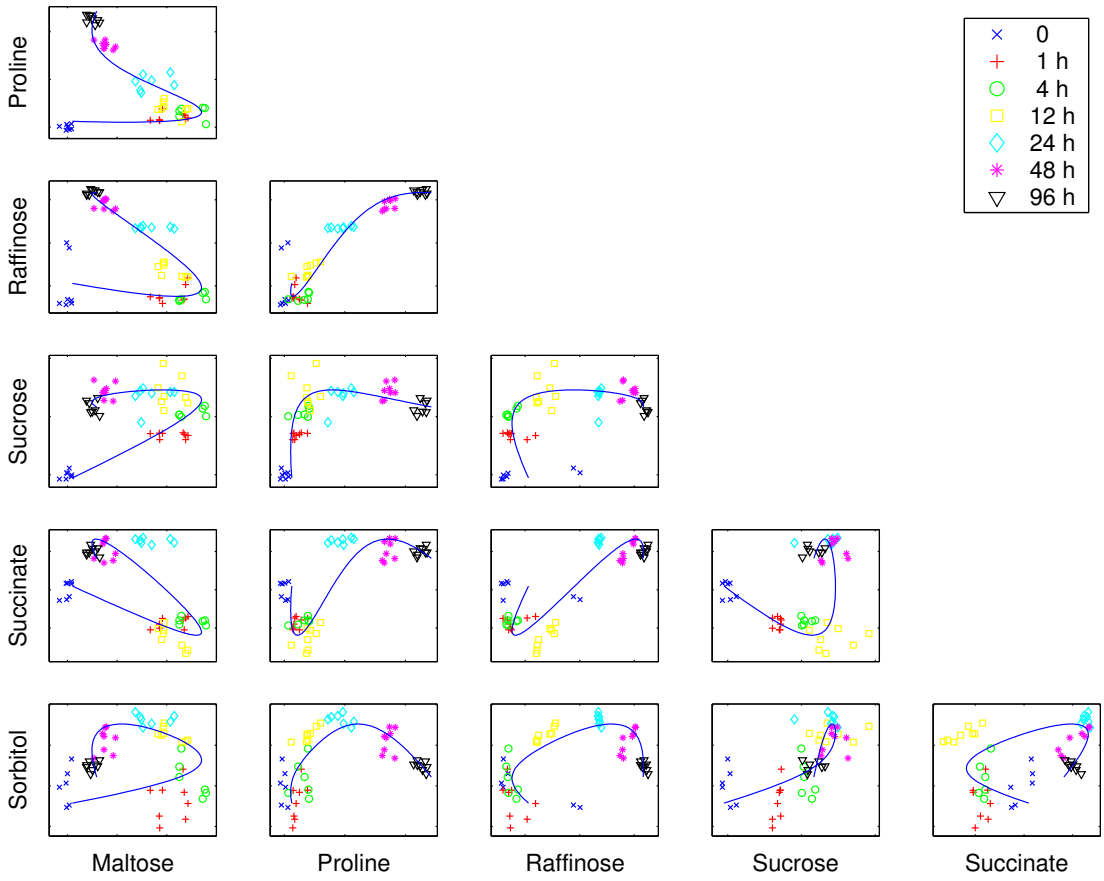


Figure 5.16: **Time trajectory.** Scatter plots of pair-wise metabolite combinations of six selected metabolites of highest relative variance. The extracted time component (nonlinear PC 1) is marked by a curve, which shows a strong nonlinear behaviour.

5.6.3 Results

The extracted first nonlinear component is directly related to the experimental time factor, see Figure 5.15. This means that the global or main information, represented by variance, is the metabolite change over time. This time trajectory clearly shows a nonlinear behaviour. The time is represented by a component which describes a strongly curved line in the original metabolite data space, as shown in Figure 5.16. It can be regarded as a noise reduced representation of the cold stress response. The inverse model gives us a mapping function $\mathcal{R}^1 \rightarrow \mathcal{R}^{388}$ from a time point t to the response x of all considered 388 metabolites $x = (x_1, ..., x_{388})^T$. Thus, we can analyse the approximated response curves for each metabolite, shown in Figure 5.17. The cold stress is reflected in almost all metabolites, however, the response behaviour is quite different. Some metabolites have a very early positive or negative response, e.g., maltose and raffinose, whereas other metabolites only show a moderate increase.

Top 20 metabolites at time points t_1 and t_2

t_1, approx. 0.5 hours		**t_2, approx. 96 hours**	
$\hat{q}$	metabolite	$\hat{q}$	metabolite
0.43	Maltose methoxyamine	0.24	[614; Glutamine]
0.23	[932; Maltose]	-0.20	[890;Dehydroascorbic acid dimer]
0.21	Fructose methoxyamine	0.18	[NA_293]
0.19	[925; Maltose]	0.18	[NA_201]
0.19	Fructose-6-phosphate	0.17	[NA_351]
0.17	Glucose methoxyamine	0.16	[NA_151]
0.17	Glucose-6-phosphate	0.16	L-Arginine
0.16	[674; Glutamine]	0.16	L-Proline
0.16	[NA_1]	-0.14	Sorbitol
0.15	[NA_154]	-0.13	4-Aminobutyric acid
0.14	[NA_341]	0.13	[612; Proline]
0.14	[NA_19]	0.12	[NA_42]
0.14	L-Arginine	-0.11	[NA_118]
0.13	Glycine	-0.11	[NA_37]
0.13	[NA_160]	-0.11	[NA_70]
0.12	[949; Glucopyranose]	0.11	[529; Indole-3-acetic acid]
0.12	[NA_84]	0.10	[NA_210]
-0.12	[890;Dehydroascorbic acid dimer]	0.10	[NA_68]
0.12	[880; Maltose methoxyamine]	-0.10	Galactinol
0.12	L-Glycerol-3-phosphate	-0.10	[NA_117]

Table 5.2: The most important metabolites are given for an early time point t_1 of around 0.5 hours (interpolated) cold stress and a very late time point t_2 of around 96 hours.
The metabolites are ranked by their influences $\hat{q}$ at a specific time point, given by the gradient of the nonlinear time component at this time point. As expected, maltose, fructose and glucose show a strong early response to cold stress, however, even after 96 hours there are still some metabolites with significant changes in their level. Brackets '[...]' denote an unknown metabolite, e.g., [925; Maltose] denotes a metabolite with high mass spectral similarity to maltose.

In classical PCA we can select the metabolites that are most important to a specific component by a rank order of the absolute values from the corresponding eigenvector, also termed loadings or weights. As the components are curves in nonlinear PCA, no global ranking is possible. The rank order is different for different positions on the curved component, hence it is different at different time points in our case. However, we can give a rank order for each individual time point by computing the gradient $q_i = \frac{dx_i}{dt}$ on the nonlinear time curve at this time point. The rank order of the top 20 metabolites is shown in Table 5.2 for an early time point t_1 and a late time point t_2. The influence values $\hat{q}_i$ are the l_2-normalised gradients q_i, $\sum_i (\hat{q}_i)^2 = 1$. The gradient curves over time are shown in Figure 5.17. We found that even at the last time point of the experiment, 96 hours, there are still some metabolites with significant changes in their concentrations.

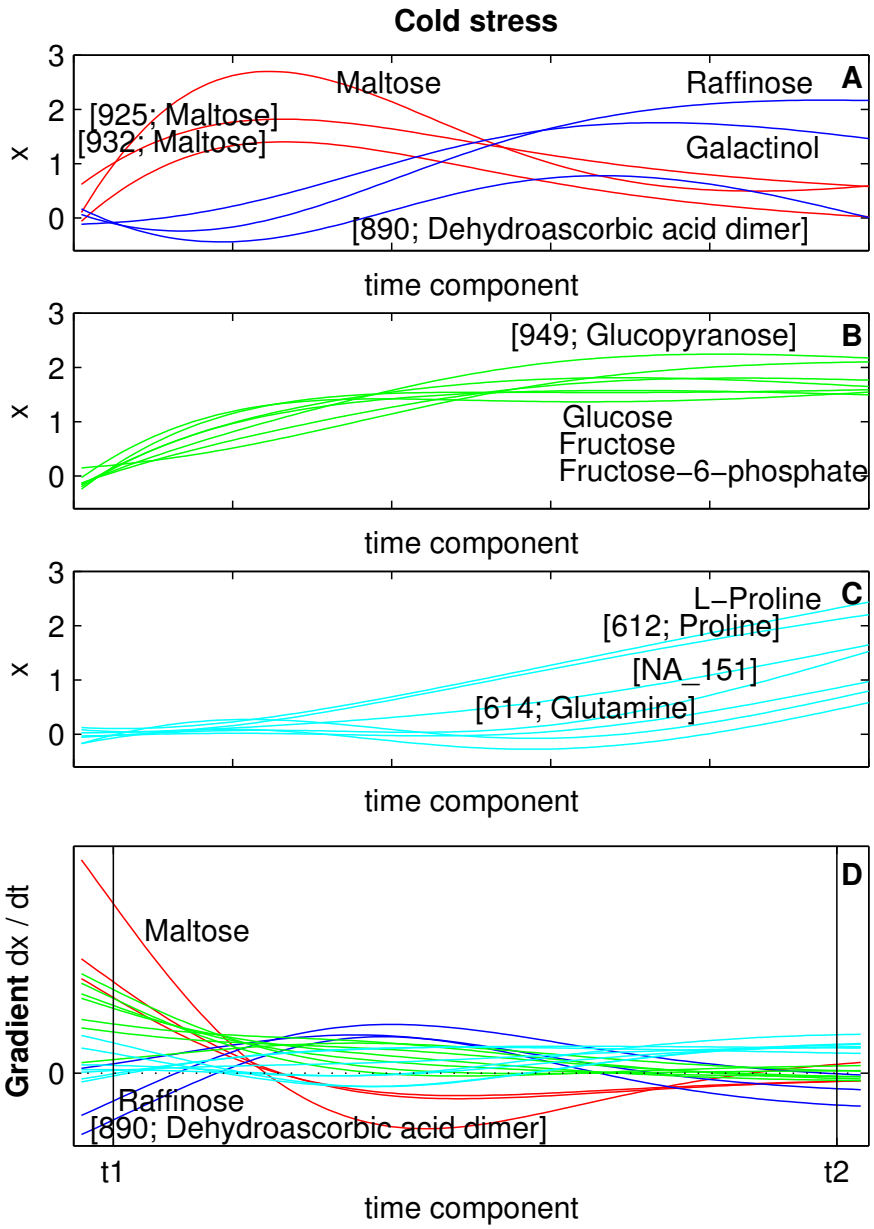


Figure 5.17: The top three graphs show the different shapes of the approximated metabolite response curves over time. (**A**) Early positive or negative transients, (**B**) increasing metabolite concentrations up to a saturation level, or (**C**) a delayed increase, and still increasing at the last time point. (**D**) The gradients represent the influence of all metabolites at any time point, analogous to loading factors in PCA. High positive or high negative gradients at particular times relate to metabolites with strongly changing levels. There is a strong early dynamic, which is quickly moderated, except for some metabolites that are still not stable at the end. The top 20 metabolites with the highest absolute gradients are plotted. The rank order for the marked early time t_1 and late time t_2 is given in Table 5.2.

5.7 Summary

Nonlinear PCA (NLPCA) was proposed as an inverse model to be applicable to incomplete data sets. With this inverse NLPCA we were able to extract nonlinear (curved) components from data sets with a large number of missing values. The idea behind solving the missing data problem is that the model of missing data estimation has to match the model of the final analysis. Our strategy was therefore to adapt nonlinear PCA to be applicable to incomplete data instead of estimating the missing values in a prior separate step.
Since the extracted components can be used, together with the model, to reconstruct the original data, we can even estimate the missing values. The missing data estimation performance is compared to other algorithms: Both nonlinear techniques, the inverse nonlinear PCA and self organising maps (SOM), improved the missing value estimation performance for the most important metabolites of the lower dimensional metabolite data set. In the larger gene expression data set, the best missing data estimations were obtained by BPCA and PPCA.
However, our goal was to identify nonlinear components. A question concerned here is reliability. We have shown that validation by a test data set is of no use in such unsupervised analysis. Driven by the idea that missing data can be best estimated with the model that corresponds best to the real process, we proposed to validate the complexity of a model by its missing data estimation performance.
Nonlinear PCA was applied to a time course of metabolite data from a cold stress experiment on the model plant *Arabidopsis thaliana*. The detected first nonlinear component was directly related to the experimental time factor. Thus, the inverse nonlinear PCA model gives us the continuous metabolite response over the time frame of the experiment. The identified trajectory over time provides greatly improved information for a better understanding of the complex response to cold stress. For each time point, including interpolated time points, we are able to give a ranked list of the most important metabolites, analogous to a ranked list for a particular component in PCA or ICA.
The cold stress response clearly showed a nonlinear behaviour over time at the metabolite level (Kaplan et al., 2004). A similar nonlinear behaviour was also found in gene expression data from the same cold stress experiment (data not shown). This nonlinear analysis can therefore be done in the same way for such data.
Even though time is the most common factor, nonlinearities are not restricted to temporal experiments, they can also be caused by other continuously changed factors, e.g., different temperatures at a fixed time point. Even natural phenotypes often take the form of a continuous range (Fridman et al., 2004) where nonlinear molecular behaviour may occur.

6 Molecular networks

There is a great interest in molecular biology to obtain a comprehensive view of all relations among molecules within a biological system, and to ultimately determine the complete molecular network. Network models are useful to discover regulatory differences between distinct organisms or even between distinct physiological states or developmental stages. They might even be helpful to discover evolutionary characteristics.
One approach is to generate networks from existing biochemical pathway knowledge to analyse and to visualise the usually large amount of information contained in many databases (Jeong et al., 2000). However, this chapter is focused on the much more challenging task of reconstructing molecular networks from experimental observations.
Typically, the nodes of a molecular network (also termed vertices) stand for individual molecules, e.g., metabolites. Molecules that are related in some way are connected by edges. Depending on the source of information, edges can directly represent biochemical reactions from database knowledge; or based on experimental observations, edges can represent similar behaviour of molecules under varied experimental conditions.
Given the network model, global topological measures as well as local network motifs can be applied to describe and analyse the characteristics of a network. The most elementary measure is the degree or connectivity of a network node which is given by the number k of links (edges) to any other node. A node of high degree represents a molecule with many connections to other molecules and hence is supposed to play an important role in the network. However, there exist many other so-called centrality measures (Freeman, 1979).
In the recent years global network topological measures were under strong discussion since Barabasi and Albert (1999) discovered that most natural and social networks cannot be sufficiently described by random graphs — the traditional model in graph theory (Erdös and Rényi, 1960). Instead, the connectivity in most networks is *scale-free* which is characterised by a power-law degree distribution $P(k) \sim k^{-\gamma}$, where $P(k)$ is the probability that a node has k links and γ is the degree exponent. It describes network topologies consisting of many nodes with few connections and few nodes with high connectivity. In random graphs, by contrast, all nodes have about the same number of connections. The connectivity of nodes is Poisson distributed around an average degree $< k >$ which is used to characterise random networks. Scale-free networks, by contrast, cannot be meaningfully characterised by their average connectivity number.
Many studies have shown evidence that molecular networks, metabolic as well as protein networks, are scale-free (Jeong et al., 2000; Barabasi and Oltvai, 2004). This is important because the scale-free property offers a number of advantages. In contrast to random networks, scale-free networks are more robust against random loss of nodes (Al-

bert et al., 2000) and are characterised by a short average minimum distance between two arbitrary chosen nodes, the small-world property (Watts and Strogatz, 1998). A scale-free network topology is typical for networks which are not static. Instead, growth is supposed to be the important factor responsible for this topology. Growth means that scale-free networks expand continuously by the addition of new nodes which preferentially attach to nodes that are already well connected. For molecular networks, growth can be seen evolutionarily.

Another important aspect is to visualise molecular networks to better understand the complexity of biological systems. Graph visualisation algorithm can be used to give a two-dimensional representation either of the total network by using all molecules or of a subnetwork only by selecting the molecules of interest. The nodes are represented by dots that are connected by lines (edges) as done in Figure 6.1. Energy optimisation algorithms are frequently used to obtain a graphical network representation where close distances correspond to strong relations. However, other representations are possible as well, for example, with minimised crossing edges. The objective to present the important information in a very understandable way, can sometimes be better achieved without realistic distances as, for example, in subway maps, see also Krempel (2004). The choice of lay-out algorithm always depends on the information that is to be emphasised in the graphical visualisation.

However, the use of experimental observations for network generation concerns fundamentally the problem of identification and quantification of molecular relations. The key issue, discussed in this chapter, is therefore to define biological reasonable distances or similarity measures. Even though correlation might be reasonable and to some extent successful, there are strong limitations which restrict its utility significantly. This includes the restriction to linear dependencies as well as the problem of pair-wise measures in multivariate data sets. Nonlinear dependencies, for example, can be better handled by the measure of *mutual information* (Steuer et al., 2002). Other pair-wise distance measures are discussed by Cichocki and Amari (2003) on page 544.

Nevertheless, one of the main problems is that pair-wise measures are restricted to relations between two variables only and hence are insufficient for multivariate data sets. Multivariate relations such as partial correlations are usually not taken into account. Additionally, molecular similarity depends strongly on the involved biological factors. A multivariate technique such as independent component analysis, which is able to identify the major factors, may therefore be more suitable for detecting factor dependent similarities. Both aspects can thereby be covered: multivariate relations and factor specificity. For example, metabolites may respond similarly to an external stimulus but not to another. Representing different biological aspects by individual components enables us to reveal dynamic behaviour. The purpose of this chapter is to propose a new network model that includes functional dependencies in order to analyse and interpret molecular dynamics. It is denoted as the *functional network*.

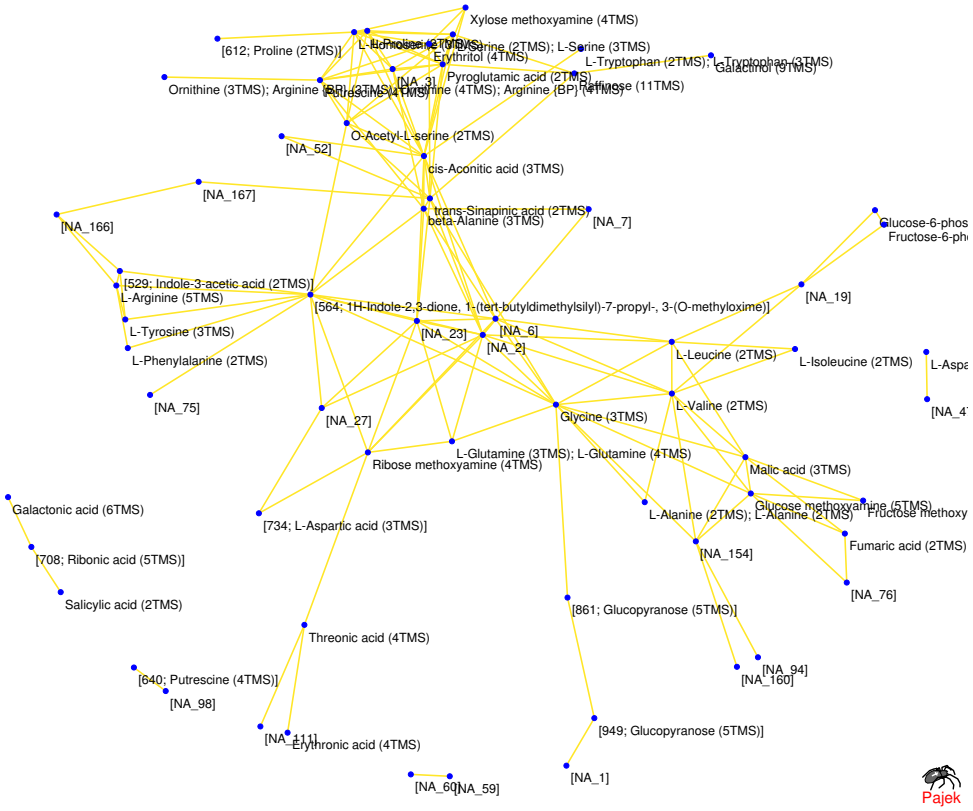
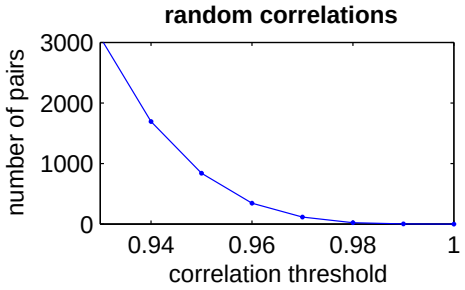


Figure 6.1: **Correlation network.** The graph shows the metabolite relational map of cold stress adaptation of *Arabidopsis thaliana*. The network is based on pair-wise correlation coefficients. The nodes represent metabolites which are connected by an edge if their correlation coefficient exceeds a given threshold. A close distance refers to a high correlation and hence to a similar behaviour during the experiment, which may indicate a similar functionality of or even a biochemical interaction between these metabolites.
Even though a large number of connections shown here are biologically reasonable, in general, networks based on correlation alone should be viewed with caution. A high correlation coefficient may often occur just by chance in large-scale data (Figure 6.2), while strong biological relations, on the other hand, may show only a low correlation coefficient due to the impact of multiple superimposed factors (Figure 6.3).

6.1 Correlation networks

Classical correlation (sometimes referred to as Pearson correlation) is a widely used distance measure to explain similarities between genes or metabolites. It is used to reconstruct molecular networks (Stuart et al., 2003), but also common in other distance-based data analysis techniques such as in cluster algorithms (Eisen et al., 1998), and often used for identifying coexpressions across multiple microarray data sets (Lee et al., 2004).



number of false gene-pairs
(data set : 8,000 genes, 10 ... 30 samples)

#	>0.80	>0.85	>0.90	>0.95
10	**174,515**	**58,832**	**12,343**	**822**
15	10,962	1,922	155	2
20	728	66	2	0
25	52	3	0	0
30	4	0	0	0

Figure 6.2: **Random correlations.** Illustrated is the huge number of pair-wise correlations that occurs by chance in large-scale data sets. An artificial data set of typical size (8,000 genes, 10 samples) was generated randomly from a Gaussian distribution. A large number of false gene-pairs was detected to have 'significant' correlations. 822 gene-pairs, for example, were found to have a correlation factor higher than 0.95. In the case of 20,000 genes, the numbers are even more than 6 times larger.
Better results were obtained by using more samples, as shown in the table. By using 20 samples only 2 pairs were of higher correlation than 0.9 and in case of 30 samples only 4 pairs higher than 0.8.

Given two variables x_i and x_j (e.g., two metabolites), the correlation coefficient C_{ij} can then be obtained by $C_{ij} = corr(x_i, x_j) = \frac{cov(x_i,x_j)}{\sigma_i \sigma_j}$ which is given by the covariance $cov(x_i, x_j) = \frac{1}{n-1}(x_i - \bar{x}_i)(x_j - \bar{x}_j)^T$ normalised by standard deviations σ_i and σ_j of both variables x_i and x_j, n is the number of samples. A more detailed discussion related to normalisation and PCA can be found in Chapter 3.
Two metabolites are considered similar when their absolute correlation coefficient $|C_{ij}|$ exceeds a certain threshold C^T, $|C_{ij}| > C^T$. A correlation network can then be constructed by representing metabolites as nodes and connecting them pair-wise with an edge if their correlation coefficient is greater than the threshold C^T chosen in advance. The characteristics of such a network can be analysed with respect to certain graph theoretic criteria, and two-dimensional graphical representations can be generated by various layout algorithms. Note that similarity obtained from correlation should not be confused with causality, since two metabolites showing a similar behaviour do not necessarily belong to the same biochemical reaction.
Figure 3.1 shows a correlation network derived from the cold stress experiment from Section 5.6. The network is based on the pair-wise correlation coefficients between 388 metabolites with 52 samples from seven different time points. A threshold of $C^T = 0.94$ was chosen and all isolated nodes of metabolites, with none of their coefficients higher than C^T, has been discarded. The two-dimensional graphical layout was done with the software package *Pajek*[1] (Batagelj and Mrvar, 1998) by using the *Fruchterman-Reingold algorithm* (Fruchterman and Reingold, 1991).

[1] http://vlado.fmf.uni-lj.si/pub/networks/pajek/

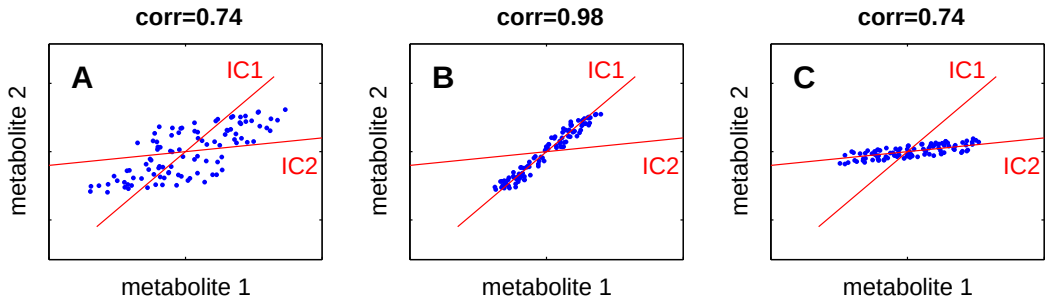


Figure 6.3: **Factor sensitivity of correlation.** Illustrated is an observed pair-wise metabolite response to two varied factors. One factor (IC 1) might be a change in light intensity where both metabolites respond to and the other (IC 2) might be a variation in temperature where metabolite 1 responds to. (**A**) When both factors are varied, the metabolites are partially correlated, and hence a low pair-wise correlation is observed. In (**B**) only the light factor (IC 1) is considered. The temperature is filtered out by scaling down IC 2. Now we get a high correlation value as both metabolites respond equally well to the imaginary light factor. (**C**) When we consider only the temperature factor by scaling down the light variation, we still obtain a low correlation value, as only metabolite 1 responds significantly to temperature.

Choosing the optimal threshold, however, is one of the most crucial issues. The connectivity and hence the entire network structure depends strongly on this threshold. If the threshold is too low, it would lead to a less informative graphical representation with too many connections which even may contain an unacceptably large number of biologically false connections (Figure 6.2). If, on the other hand, the chosen threshold is too high, we lose biologically relevant relations which are not indicated by high correlation coefficients.
In the cold stress experiment (Figure 6.1), a threshold of $C^T = 0.94$ was chosen which was high enough to show a clearly arranged map of the most important correlations, but too high to show other biological similarities such as between maltose and metabolites with high mass spectral similarity to maltose, whose coefficients were below 0.90. The chance of obtaining false similarities, however, is relatively small for this data set, since there is a relatively low number of variables (388) and a large number of samples (52) compared to, e.g., gene expression data sets.
However, beside the question of the optimal threshold, there are some more issues concerning pure correlation as a distance measure and the general problem of pair-wise measures. Interactions in molecular systems proceed dynamically according to biological requirements. The question is therefore to what extent links in correlation networks are driven by the investigated biological process and to what extent by confounding factors (biological or technical artifacts) or simply by background noise.

6.1.1 Drawbacks of pure correlation based distances

Even though pure correlation is a commonly used pair-wise distance or similarity measure, there are some major disadvantages. We exclude, for example, important information provided by other criteria such as intensity, variance, distribution or information theoretic criteria. For using this additional information it might be better to combine different criteria or at least to take them for pre-selection (feature selection) to analyse correlations on a previously reduced variable set.
Variance, for example, can be used as a measure for reliability. Even though variables of small variance may be equally well involved in a biological process as variables of larger variance, they are closer to the variance of background noise and hence more corrupted by measuring inaccuracy. A large correlation between variables of higher variance is therefore more reasonable and should be weighted higher in a subsequent network analysis. Otherwise many false correlations are obtained just by chance, as shown in Figure 6.2 for large-scale data sets where many variable-pairs show by chance a high correlation coefficient caused by background noise.

6.1.2 Partial correlations and the problem of pair-wise measures

Many known biochemical interactions, on the other hand, cannot be identified as relevant because their correlation coefficients appear too low. Typically only a subset of metabolites or enzymes assigned to a given pathway is significantly correlated (Steuer et al., 2003; Ihmels et al., 2003). The reason is not only the high inaccuracy of the data. In Figure 6.3 we demonstrate that such an effect can be caused by the large number of distinct factors contained in molecular data. This includes biological as well as technical factors. Metabolites that depend on several factors are often partially correlated and hence show a poor pair-wise correlation.
The reason is that the correlation measure is factor sensitive. Whether we can observe a high correlation coefficient or not depends essentially on all involved factors: examined factors as well as confounding factors. Different experiments lead to different correlation results. And multiple factors in one experiment, including confounding technical or biological variation, lead to a mixture of influences to individual metabolites and hence to a poor correlation result.
As illustrated in Figure 6.3, meaningful correlation coefficients can be achieved if there is only one single factor involved. Data from real experiments, however, contain many more factors. Even if we investigate only one biological process and might be able to control perfectly all environmental and technical influences, there would still be a biological variation caused by several internal factors. This includes slightly different physiological states of internal regulatory processes.
Consequently, a meaningful distance between two metabolites can only be given with respect to a specific factor, e.g., a biological process or environmental variation. This includes that two metabolites can be correlated with respect to one factor but not with respect to another, which is biologically reasonable, since two metabolites might interact with each other in one particular biological process but not in the other.

One approach to handle multiple influences is the concept of partial correlations. Partial correlations can be calculated between two variables with regard to one or more other variables. However, this is a difficult task in large-scale data sets. In order to exclude the influence of confounding factors we would need metabolites that depend directly and solely on these factors. But usually we neither know these metabolites nor might such metabolites even exist, as all metabolites may respond to several factors simultaneously. Nevertheless, there are studies on the concept of partial correlations in which their application was successfully demonstrated, even when applied to gene expression data (de la Fuente et al., 2004).
The problem of multiple factors arises essentially in pair-wise comparisons of multivariate data. Even if we use other pair-wise distance measures such as mutually information, the problem still remains. Partial correlation analysis can already be seen as an extension to more than two variables. But the real consequence, however, would be to use full multivariate data analysis, ICA, for example.

6.1.3 Necessary assumptions in correlation analysis

We do not state that correlation is insufficient at all, but using it alone requires strong assumptions which do not hold for real molecular data in most cases.
The most important requirement for measuring a meaningful correlation between two metabolites is that both respond to a single factor only. Such a factor might be a variation in a particular environmental condition or an internal regulatory process. All other potential variations must be decreased by controlling them as best as possible.

> The easiest situation would be one single experimental factor of higher influence than all other biological or technical variation, hence one strong single factor is responsible for nearly all variation in the experiment.

All metabolites affected by this factor should give a meaningful pair-wise correlation coefficient. For example, the cold stress experiment, with the cold stress adaptation over time as single factor, shows some biologically reasonable similarities in the correlation network (Figure 6.1). However, the association with specific time points is missing. And, metabolites which are not affected by the stress situation or corrupted by other confounding variations might confuse the correlation measure.
On the other hand, for experiments of more than one factor at a time, correlation analysis is less suitable. This includes experiments such as the crossing experiment of section 3.5 in which at least two factors are under examination, one that separates male from female and another that separates the parent from the next (F1) generation.
However, multiple factors do not lead to poor correlation results in general. Special experimental setups of identical genotypes under identical environmental conditions have led to biologically reasonable correlation networks (Weckwerth et al., 2004). These data without varied experimental factors are sometimes referred to as *observational data*, in contrast to *experimental data* of specific targeted variation enforced by distinct conditions. In observational data all samples are treated equally. Thus, the variation in the data is only caused by biological variability, provided that the technical variation is

sufficiently controlled.
The question is: under which assumptions and to what extent can we achieve reliable correlation results from observational data. The strong requirement that only one single factor varies during the experiment is not necessarily needed at all. It is important that each metabolite responds to only one factor. But this includes that distinct metabolites can also respond to different factors. The problem of partial correlations occurs only when two metabolites depend on both identical and distinct factors as illustrated in Figure 6.3.

> Biologically reasonable correlations can be obtained as long as distinct sets of metabolites respond to distinct factors. The distinct sets may represent metabolite groups of distinct functionality. The weaker and more general requirement would therefore be that multiple factors may occur but must be associated with distinct (groups of) metabolites.

These factors may, for example, refer to distinct internal regulatory processes such as a circadian cycle. A variation in the physiological state within such a process, generally influences only the set of metabolites that belong to this particular process. As long as these metabolites belong to no other process, correlation analysis should be able to identify them as 'functionally similar'.
Although most specific processes involve a limited set of substrates, individual substrates are not restricted to only one process. Many metabolites such as primary carbohydrates (e.g., sucrose, fructose, or glucose) take part in several biochemical pathways and thus respond in parallel to several biological processes. Such responses to a mixture of factors lead to poor correlation results. In addition, badly controlled environmental influences such as light or temperature, generally have an impact on many metabolites and hence may interfere with responses to other factors. Luscombe et al. (2004) have shown that even transcription factors are frequently used across multiple processes. Due to this potentially large number of superimposed responses, the use of the correlation measure alone is very limited. Even though we sometimes obtain reasonable results (Figure 6.1), many relations remain undetected (Steuer et al., 2003).
As a consequence, we have to find a way to consider each factor separately, e.g., by decomposing the data space. Since metabolites may interact with distinct metabolites in alternative processes, we cannot define a universal similarity among them. Similarity in the sense of co-behaviour or co-response can only be seen with respect to a particular biological process or functionality.
This implies that the naive approach of joining distinct experimental data sets for an overall correlation analysis, should be taken with caution. Many data sets, even with identical experimental conditions, typically lead to an increase of distinct factors, in particular confounding technical factors, which often cannot be discarded adequately by normalisation techniques. Instead of integrating raw data, integrating results from individual experiments might be more appropriate, as discussed next.
In principle, a decomposition, as done by ICA, could be convenient for this purpose due to its ability to identify even unexpected and unwanted factors which can then be excluded.

6.1.4 ICA to filter out confounding factors

ICA provides a mathematical framework for identifying and separating all significant information contained in a data set. The objective of ICA is to represent distinct information separately by individual components. ICA could therefore be used as a filter technique which subsequently decreases the influence of confounding factors by identifying and scaling them close to zero as illustrated in Figure 6.3. Metabolites that respond equally well to the same factor (component) will then get an increased correlation coefficient.
This implies, however, that components themselves can directly be used to explain similarities in the sense that metabolites that contribute highly to the same component are functionally similar. The metabolites of highest contribution are those with smallest angles to the direction of the component in the original data space.
In addition, using ICA has the advantage that we consider multiple criteria, including variance in PCA pre-processing, to reduce the influence of the background noise. And, importantly, components are often functionally interpretable. This can be used for getting a functional or dynamic view of molecular relations.

6.2 Functional networks

A flexible organisation of cellular networks enables dynamic responses to changing environmental conditions. Several studies have shown evidence of extensive regulatory mechanisms which activate or deactivate biochemical interactions according to biological requirements (Ihmels et al., 2003; Luscombe et al., 2004; Han et al., 2004). This requires a dynamic view of connectivity or similarity of molecular substances (metabolites, proteins, RNA). Similarity cannot be considered irrespective of biological processes or functionalities. Our objective is therefore to identify dynamics in molecular experiments and to visualise them by a specific network type which we denote as the *functional network*.
The functional network is constructed as a bipartite graph with two types of nodes: functional nodes and molecular nodes. *Functional nodes* denote dynamics of biological processes or functionalities, not states. This includes regulatory and developmental processes as well as variations between disease and control, between mutations and wild-type, or responses to external stimuli: environmental, chemical, physiochemical, or biological (pathogen). Any response to an action which results in a changed molecular composition can be represented by a functional node. *Molecular nodes* represent the individual molecules or genes observed in the experiments.
Each molecular node is connected with each functional node. The distances denote the importance of molecules to functionalities according to molecular responses in experiments. A close distance stands for a strong relative change in concentration or activation of a particular molecule or gene during the experimental variation represented by the functional node. The network is represented by a bipartite graph because neither molecule nodes nor functional nodes are directly connected. The graph is undirected but could be considered as directed graph with a meaning in both directions: either from

molecule nodes to functional nodes with the meaning that molecules are important to specific biological functions, or from functional nodes to molecular nodes denoting the impact of biological processes on individual molecules.
The focus on distances between molecules and biological processes avoids the difficult or even impossible problem of defining a *static* and universally valid similarity measure between molecules which interact *dynamically* according to requirements in different biological processes. A network structure that sets molecules and processes into relation is therefore well suited for visualising and analysing molecular dynamics. Precisely, it provides the information to which stimuli molecules respond in common and to which not. This cannot be done by a static molecule-molecule similarity measure.
The potentials of such a functionally related view are demonstrated by generating a network that visualises the metabolic dynamics over time in cold stress adaptation of the model plant *Arabidopsis thaliana*. We assume that the entire cold stress process consists of functionally distinct sub-processes (e.g., early quick response, permanent cold stress response). Each of these potentially overlapping sub-processes may require an individual change of the metabolic composition. To visualise the potential dynamics, different time points are represented by different functional nodes. Differences in metabolite responses at individual time points will result in individual distances between metabolites and the functional nodes respectively. The functional network therefore provides a view of cold stress adaptation that represents the importance of individual molecules over time.
In a next step, the results of several individual experiments could be integrated into a joint network representation. Such an extended view would provide more detailed and refined information for a better understanding of the dynamic relations within a particular class of experiments, such as those of stress response. The final objective, however, would be to represent and to understand the reorganisation of the whole molecular system according to external stimuli or developmental stages.
Based on individual relations between molecules and biological processes, even the processes are arranged in relation to each other which reveals similarities between them. Some processes might be closer related than others according to similar effects to the molecular system. For example, cold stress might be closely related to heat or even to drought stress, also the salt stress response might be in some way related to the osmotic stress (water or drought stress) response.
Instead of a hierarchical model of functionalities as used by (Ihmels et al., 2003), a network model provides a functional map which is not necessarily hierarchically organised. Biological processes can be considered as being located in a functional space without strictly defined boundaries, as opposed to a hierarchical view of isolated modules. A modification of one process might lead to a new closely related process, represented by a continuous movement onto a new position in the functional map as demonstrated by the time course (Figure 6.5).
Thus, functional networks provide a framework for integrating results from many individual experiments. Functional networks comprehensibly explain the importance of molecules to individual biological processes. Since even the processes or functionalities themselves are set into relation, the resulting map of functionalities may reveal interesting physiological similarities.

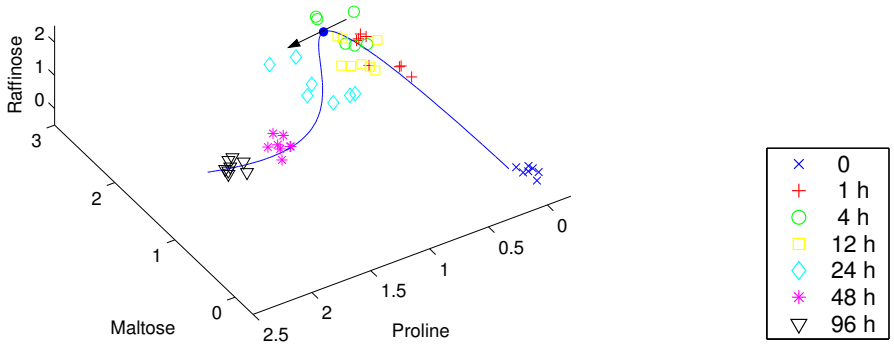


Figure 6.4: The tangent (black arrow) on the curved time component provides the direction of change in molecular composition at the particular time corresponding to the position on the curve. The most important metabolites, those with highest relative change on their concentration levels, are given by the closest angle between any of the axes representing the metabolites and the direction of the tangent.
The whole data space is given by all 388 metabolites. However, for the purpose of illustration, three metabolites, maltose, proline, and raffinose, are exemplarily plotted in a three-dimensional space.

6.2.1 Deriving similarities from large-scale data sets

Given a set of experimental observations, in general, we cannot extract the information whether two substrates interact biochemically directly, instead we can only examine how similar their behaviour is in a specific experiment. Since similarity depends on the involved experimental factors, it means exactly how similar their response is to one or more factors of the experiment. Hence, relations between molecules and individual biological functionalities or processes are the basic information available from the data. The objective is therefore to identify both the factors having a significant impact on the data and the response of molecules to each of these factors. Suitable techniques would be unsupervised decomposition methods such as PCA and ICA as well as supervised discriminant or regression analysis, provided that the results can be validated well. The purpose is always to identify and to represent the factors by linear and nonlinear components.
Biological processes as response to external stimuli usually result in a changed molecular composition. This is reflected by a shift of observations in a particular direction in the molecular data space, the space where axes stand for individual molecules. The objective is to find the direction or curve, often termed linear or nonlinear component, that explains best the trajectory or variation of a particular biological response. The importance of molecules can then be directly determined. The direction of a linear component or the direction of a tangent on a specific position on a nonlinear (curved) component (Figure 6.4) is often explained by weights (loadings) related to the original axes. The absolute weight values are inversely proportional to the angle between the direction and

the axes (molecules). A small angle (large absolute weight) implies a large impact on this direction and hence suggests a high importance of the respective molecule within the response process.
In principle, there are two potential strategies to construct networks from component information. The way, demonstrated here, is to set molecules and biological processes into relation depending on their impact or importance given by the weight values. In general, no other threshold has to be chosen than the number of important molecules to reduce the complexity of the representation.
Alternatively, we could link all molecules to each other that are important for a particular process, assuming that all molecules of high contribution are functionally similar. Hence we would define functionally related clusters of totally connected molecules. The clusters potentially overlap due to molecules involved in more than one cluster. However, this requires a threshold defining the relevance of a molecule to a particular process, either by setting a weight (loading) value corresponding to a certain amount of importance or contribution, or by setting the total number of molecules for each cluster. Nevertheless, both choices would be crucial.
Even though we could easily apply topological measures (Barabasi and Oltvai, 2004) to such molecule-molecule network, links between two molecules should be interpreted with caution since they do not necessarily represent direct biochemical reactions.
However, instead of defining molecule-molecule distances, this chapter focused on the definition of distances between molecules and biological functions or processes. It appropriately reflects the information we can derive from large-scale functional studies.

6.2.2 Application: metabolite cold stress network

Results from nonlinear principal component analysis (nonlinear PCA) in section 5 are used to build a functional network from a cold stress experiment of *Arabidopsis thaliana*. The concentrations of 388 metabolites, observed at several times, are used in nonlinear PCA to model the entire cold stress adaptation. Specifically, normalised data are used representing the concentrations relatively to control time zero. The trajectory over time is approximated by a nonlinear component, a curve lying in the data space given by all metabolites. At any, even interpolated, time, the metabolites can be ranked by importance. A metabolite is supposed to be important, if its relative concentration changes strongly at the considered time point. The amount of change is shown by weights or loading factors (Table 5.2) which specify the direction of a tangent at the particular time position on the curve, as illustrated in Figure 6.4. The weights can therefore be regarded as similarity measure used for defining distances between metabolites and time nodes in the network. A close distance is used for large weights, and hence corresponds to a metabolite of high response.
Individual time points are represented by functional nodes in the network. Potentially distinct biological processes at different time points can therefore result in individual network characteristics. Thus, the resulting network structure represents the relations between distinct cold stress adaptation processes over time according to their metabolite responses.

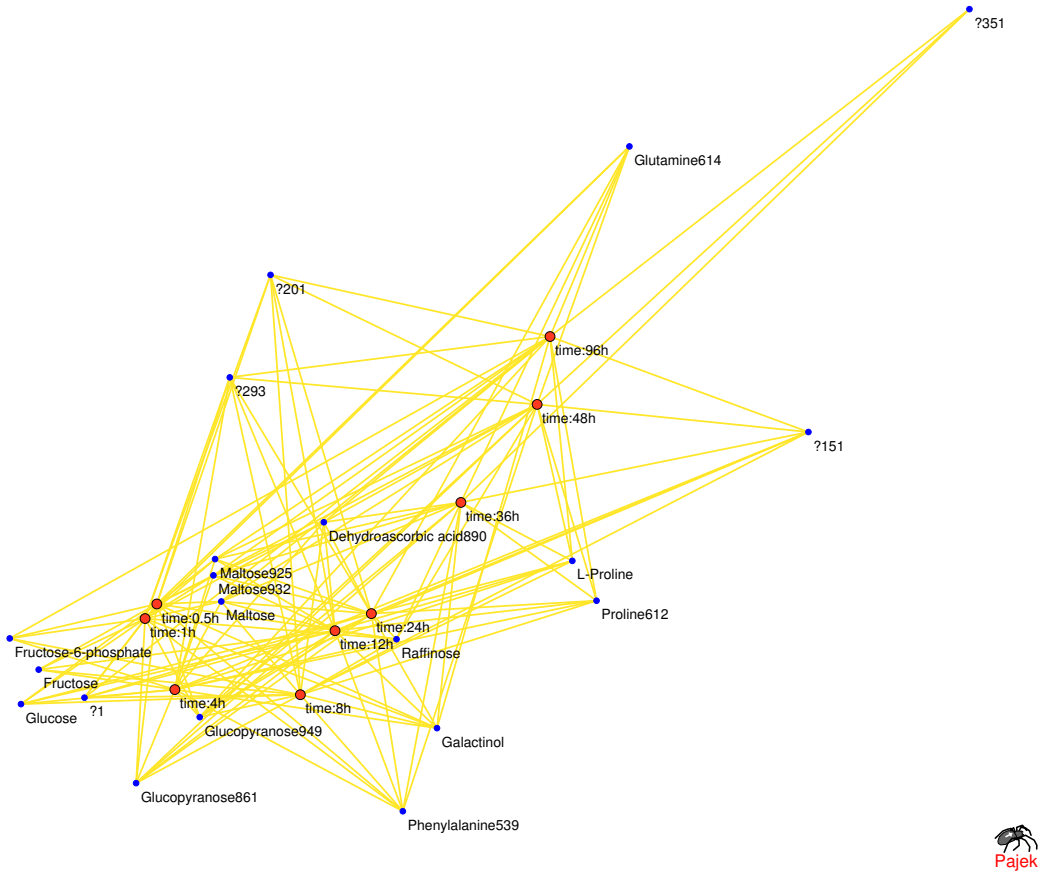


Figure 6.5: **Functional network.** The dynamic in cold stress adaptation of *Arabidopsis thaliana* is visualised. In this bipartite graph, metabolites are connected with functional nodes which represent different times and hence potentially distinct processes. The distances between metabolites and functional nodes are defined by nonlinear PCA according to the importance of metabolites at individual time points. A close distance stands for a strong relative change in metabolite concentration.
The 20 metabolites of highest response at any time are plotted. Even though no distances were defined between time points, they show a well ordered trajectory over time. Also, functionally similar metabolites were mapped close to each other, although no distance was defined between them. Maltose and metabolites with high mass spectral similarity to maltose (e.g., maltose932) were all located in the centre of early time nodes. Thus they are supposed to play an important role in the beginning of cold stress. Similarly, fructose and glucose are functionally important at very early times (0.5, 1, up to 4 hours). Other metabolites such as proline, glutamine, and some unknown metabolites (marked by '?') are closer connected to later time points.

Nine time points are chosen. This includes the original (experimental) times at 1, 4, 12, 24, 48, and 96 hours, as well as three interpolated times at 0.5, 8, and 36 hours used to fill gaps between original times in the graph. In order to emphasise the main aspects in cold stress we restrict the network representation to the top 20 metabolites of importance at any time. Other, less important metabolites have longer distances and hence would be located outside of the current graph.
The resulting network structure in Figure 6.5 shows that cold stress adaptation is a dynamical process with timely distinct metabolite responses. The importance of metabolites varies over time. Maltose and its variants occupied a central position among early functional time nodes. This suggests a great importance in early cold stress response.

6.2.3 Summary

Functional networks provide detailed information for an understanding of the dynamical behaviour in molecular systems. The utility of functional networks was demonstrated by generating a network of the complex nonlinear cold stress adaptation of the model plant *Arabidopsis thaliana*.
The results confirmed our expectations. For example, maltose and its variants are shown to play a central role in early cold stress adaptation. In correlation analysis, by contrast, maltose and its variants could neither be identified to be very similar nor to be important in cold stress.
Although no distances were defined between functional time nodes, they show a reasonable trajectory through the network. Functional networks can therefore also be used to visualise similarities between several processes, including responses to different conditions of distinct experiments. The result of such an integrative analysis would be a comprehensive map of physiological processes surrounded by their most involved molecules.

7 Conclusions

The purpose of this work was to provide approaches to analyse and interpret molecular data from experimental observations. Special emphasis was given to several important issues: the impact of multiple factors, large-scale or high dimensionality, missing data, complex nonlinear behaviour, and dynamical co-behaviour of molecules.
One of the key question is that of the optimal method that provides best the desired information. In addition to the characteristics of the data, it depends strongly on our focus of research. While supervised classification or regression might be most reasonable for diagnostics tasks, unsupervised methods, as considered in this work, are very suitable for more general exploratory research questions such as how experimental conditions are reflected at molecular levels. Without using target information (class labels), unsupervised methods aim to extract all available information from the data and hence enable us to discover even unexpected phenomena. Unsupervised methods are therefore valuable to get a better understanding of molecular processes or to improve molecular technologies by reducing the impact of discovered confounding factors.
However, the crucial issue is still to distinguish relevant from non-relevant information. There are several ways to include additional knowledge for defining our aims. Beside the sometimes convenient way of using appropriate normalisation techniques, the choice of an analysis technique is very important, precisely it is the choice of the analytic criterion which defines our 'interest'.
As long as we can assume that all variation in the data is caused by the examined conditions, variance would be the appropriate criterion to extract the relevant information. Thus, standard principal component analysis (PCA) would provide an optimal visualisation of the data. However, often we cannot control the experiment perfectly, and hence there are confounding factors with a large impact on the variance in the data as well. Thus, we have to separate the multiple factors which, by assuming almost independent factors, leads us directly to the concept of independent component analysis (ICA). ICA aims to extract statistically independent components by using higher order or information theoretic criteria instead of variance. Since variance is still important for reliability, it is essential to apply ICA in conjunction with PCA as pre-processing step to filter out small variances close to background noise. Hence, the combination of criteria is effective, because it can reveal relations that covariance (correlation and variance) in PCA or information criteria such as mutual information in ICA alone cannot. Applied to experimental data, we could show that components of ICA are of higher sensitivity and independence than components of PCA. ICA was able to identify the examined factors more precisely than PCA. Such precise components are essential to identify the corresponding molecules with high accuracy. Additionally, with ICA we even discovered

an unexpected (confounding) factor which could be interpreted as a technical artifact. Thus, ICA provides a sound and flexible framework to separate the multiple factors that have an impact on the observed data.
However, even though ICA could be successfully applied to several data sets, it is a linear transformation and hence limited to discover linear relations. As long as we examine one condition in relation to another, linear methods might be sufficient. However, experiments where we observe the molecular behaviour continuously over time or any other factor, potential nonlinear relation between molecules might lead to more complex data characteristics. Thus, analysing these data requires more complex nonlinear analysis techniques. Here, we focused on a nonlinear extension of PCA which is based on an auto-associative neural network. We have modified this nonlinear PCA algorithm to achieve a hierarchical order of components, to be able to solve inverse problems, and very important for molecular data: to be applicable to data sets with missing values.
With the idea that the model of missing data estimation has to match the model of the final analysis, our strategy was to adapt nonlinear PCA to be applicable to incomplete data instead of estimating the missing values separately in a prior step. Even though our main objective was to extract nonlinear components from incomplete data sets, the algorithm can be used to estimate missing values as well. Additionally, we proposed to use the missing data estimation capability to validate the complexity of the model which was shown to be very difficult for unsupervised nonlinear models. With an artificial data set, we successfully demonstrated the validation potential.
By applying our modified nonlinear PCA algorithm to a cold stress adaptation of *Arabidopsis thaliana* we could confirm that cold stress response is a nonlinear process over time. Nonlinear PCA provides a model of this cold stress adaptation which could be used to identify important molecules at any time point, including at interpolated times. Nonlinear PCA is an unsupervised model and hence very suitable for providing results which are unaffected by the individual variability of samples over time which occurs due to the difference of individual response time from exact physical time of measurement. Although there are only few samples in a high dimensional data space, the results are robust and biologically reasonable. This suggests that the experiment was well controlled. On the other hand, there might be a high redundancy in molecular data, hence a low intrinsic dimension, which makes it possible to handle the large number of variables. Nevertheless, an increasing number of samples together with improved experimental technologies can increase the accuracy of results significantly.
Finally, an attempt was made to link multivariate component analysis with network representations. We proposed a network model, referred to as *functional network*, which is based on molecule-function distances derived from component information. Since it sets molecules in relation to their function or their contribution to biological processes, it offers a method to visualise the dynamics in molecular processes. Conventional molecule-molecule networks, by contrast, cannot reflect the dynamics which occur when molecules behave similarly in one situation but distinct in an other. The potentials of functional networks were demonstrated by generating a network of the cold stress adaptation of *Arabidopsis thaliana*. The resulting cold stress network reveals very well the strong dynamics in the beginning of cold stress and shows the involved molecules.

A next step could be to extend this approach by including more and more biological functions or processes. Thus, we could use functional networks as a framework to integrate data from different experiments in order to generate a comprehensive map of physiological processes and embedded molecules.

Glossary

Bioinformatics It is the field of using computers to get valuable information from large amounts of data produced in biology and medicine. Bioinformatics covers a wide range of topics from data acquisition, storage, and representation up to analysis, visualisation, and interpretation. The two major objectives are to introduce standards and concepts for collecting and organising data in order to make them easily accessible, and to develop mathematical algorithms for analysing these data with the purpose of gaining new insight into biological issues.

Component A component denotes a new variable obtained by a decomposition of a data set. Ideally, components are directly related to experimental factors. Components can therefore be seen as approximations of the original factors.

Factor In this work the term factor refers to specific sources of experimental variability: environmental, genetical, or technical. This includes variations in light or temperature conditions, a genotypical variation or even confounding factors such as technical artifacts or unexpected biological behaviour. The molecular response to these original factors, given by experimental data sets, can be detected and explained mathematically by components.

Model A mathematical model describes real processes in a usually simplified manner that still covers the important aspects. It is commonly built by a function or any other mathematical construct which can be used to make predictions or to draw conclusions. Choosing the optimal model is a crucial issue, since each model has its own drawbacks and benefits.

Nonlinear correlation *Nonlinearly correlated* means that a change in the value of one variable has disproportionate effects on the values of other variables. The relations between the variables cannot be explained by a linear function.

Nonlinear PCA (NLPCA) *Nonlinear principal component analysis* (NLPCA) is generally seen as a nonlinear generalisation of standard linear *principal component analysis* (PCA). The principal components are thereby generalised from straight lines to curves.

Pathway A biochemical pathway is a line of molecules that are connected in a series of biochemical reactions to achieve a specific functionality, commonly the production of a particular metabolic substrate.

Sample Here, a sample represents the molecular profile of an individual specimen taken from a plant or any other organism. Such a profile may contain the concentration levels of all observed metabolites.

Supervised method An algorithm that generates a function which maps inputs to known outputs is termed supervised. This includes classification as well as regression tasks. The inputs are, for example, observed molecular profile data of samples with known class labels (e.g., control or disease) as output. For diagnostic tasks, such function can then be used to predict the state of unknown biological samples.

Unsupervised method Algorithms that explain relationships and characteristics in data sets without using known sample categories (class labels) are referred to as unsupervised. The objective is to provide the major information contained in the data set. This is helpful for an exploratory analysis to confirm expectations or to discover unexpected biological or technical factors.

Variable Here, a variable stands for a particular gene, metabolite, or protein. It is a mathematical quantity that represents the activation or concentration levels over all observations in a particular experiment.

List of publications

Matthias Scholz, Fatma Kaplan, Charles L. Guy, Joachim Kopka, and Joachim Selbig. 2005.
Non-linear PCA: a missing data approach.
Bioinformatics 21(20):3887–3895.

Katja Morgenthal, Stefanie Wienkoop, **Matthias Scholz**, Joachim Selbig, and Wolfram Weckwerth. 2005.
Correlative GC-TOF-MS-based metabolite profiling and LC-MS-based protein profiling reveal time-related systemic regulation of metabolite-protein networks and improve pattern recognition for multiple biomarker selection.
Metabolomics 1(2):109–121.

Matthias Scholz, Yves Gibon, Mark Stitt, and Joachim Selbig. 2004.
Independent component analysis of starch deficient pgm mutants.
In *Proceedings of the German Conference on Bioinformatics*, edited by R. Giegerich and J. Stoye, pages 95–104.

Matthias Scholz, Stephan Gatzek, Alistair Sterling, Oliver Fiehn, and Joachim Selbig. 2004.
Metabolite fingerprinting: detecting biological features by independent component analysis.
Bioinformatics 20(15):2447–2454.

Matthias Scholz and Ricardo Vigário. 2002.
Nonlinear PCA: a new hierarchical approach.
In *Proceedings ESANN*, edited by M. Verleysen.

Sebastian Mika, Bernhard Schölkopf, Alexander J. Smola, Klaus-Robert Müller, **Matthias Scholz**, and Gunnar Rätsch. 1999.
Kernel PCA and de-noising in feature spaces.
In *Advances in Neural Information Processing Systems (NIPS) 11*, edited by M.S. Kearns, S.A. Solla, and D.A. Cohn, pages 536–542. MIT Press.

Book chapters

Matthias Scholz and Joachim Selbig. 2006.
Visualization and analysis of molecular data.
In *Metabolomics: methods and protocols. Methods in Molecular Biology Series*, edited by Wolfram Weckwerth. Humana Press, New York, USA. To appear.

Matthias Steinfath, **Matthias Scholz**, Dirk Walter, Joachim Selbig, and Dirk Repsilber. 2006.
Integrated data analysis for genome wide research.
In *Plant systems biology*, edited by Alisdair Fernie and Sacha Baginski. To appear.

Matthias Steinfath, **Matthias Scholz**, and Joachim Selbig. 2006.
Profile data analysis, dimension reduction, clustering and classification.
In *European Summer School "Plant Genomics & Bioinformatics". I. Expression Micro Arrays and beyond — a Course Book.* To appear.

Bibliography

Albert, R., Jeong, H., Barabási, A. Error and attack tolerance of complex networks. *Nature*, 406:378–382, 2000.

Alter, O., Brown, P., Botstein, D. Singular value decomposition for genome-wide expression data processing and modeling. *PNAS*, 97(18):10101–10106, 2000.

Bach, F.R., Jordan, M.I. Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48, 2002.

Baldi, P.F., Homik, K. Learning in linear neural networks: a survey. *IEEE Trans. on Neural Networks*, 6(4):837–858, 1995.

Barabasi, A., Albert, R. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

Barabasi, A., Oltvai, Z. Network biology: understanding the cell's functional organization. *Nat Rev Genet*, 5(2):101–113, 2004.

Batagelj, V., Mrvar, A. Pajek – a program for large network analysis. *Connections*, 21 (2):47–57, 1998.

Bell, A.J., Sejnowski, T.J. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

Bellman, R. *Adaptive control processes: A guided tour*. New Jersey: Princeton University Press, 1961.

Bishop, C. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

Bishop, C. Variational principal components. In *Proceedings Ninth International Conference on Artificial Neural Networks, ICANN'99*, pages 509–514, 1999.

Blaschke, T., Wiskott, L. CuBICA: Independent component analysis by simultaneous third- and fourth-order cumulant diagonalization. *IEEE Transactions on Signal Processing*, 52(5), 2004.

Buja, A., Swayne, D., Littman, M., Dean, N. XGvis: Interactive data visualization with multidimensional scaling, 1998.

Burges, C.J.C. Geometric methods for feature extraction and dimensional reduction. In Rokach, L., Maimon, O., eds., *Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers*. Kluwer Academic Publishers, 2004. to appear.

Carreira-Perpiñán, M. A review of dimension reduction techniques. Technical Report CS–96–09, Dept. of Computer Science, University of Sheffield, January 1997.

Cichocki, A., Amari, S. *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*. Wiley, 2003.

Comon, P. Independent component analysis, a new concept? *Signal Processing*, 36(3): 287–314, 1994.

Cox, T.F., Cox, M.A.A. *Multidimensional Scaling*. Chapman and Hall, 2001.

de la Fuente, A., Bing, N., Hoeschele, I., Mendes, P. Discovery of meaningful associations in genomic data using partial correlation coefficients. *Bioinformatics*, 20(18):3565–3574, 2004.

DeMers, D., Cottrell, G.W. Nonlinear dimensionality reduction. In Hanson, D., Cowan, J., Giles, L., eds., *Advances in Neural Information Processing Systems 5*, pages 580–587, San Mateo, CA, 1993. Morgan Kaufmann.

Diamantaras, K., Kung, S. *Principal Component Neural Networks*. Wiley, New York, 1996.

Efron, B., Tibshirani, R. *An Introduction to the Bootstrap*. Chapman and Hall, New York, 1994.

Eisen, M., Spellman, P., Brown, P., Botstein, D. Cluster analysis and display of genome-wide expression patterns. *PNAS*, 95(25):14863–14868, 1998.

Erdös, P., Rényi, A. On the evolution of random graphs. *Public Mathematical Institute of Hungary Academy of Sciences*, 5:17–61, 1960.

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

Freeman, L. Centrality in social networks: I. conceptual clarification. *Social Networks*, 1:215–239, 1979.

Fridman, E., Carrari, F., Liu, Y.S., Fernie, A., Zamir, D. Zooming in on a quantitative trait for tomato yield using interspecific introgressions. *Science*, 305(5691):1786–1789, 2004.

Fruchterman, T.M.J., Reingold, E.M. Graph drawing by force-directed placement. *Software - Practice and Experience*, 21(11):1129–1164, 1991.

Ghahramani, Z., Jordan, M. Learning from incomplete data. Technical Report AIM-1509, Massachusetts Institute of Technology, Cambridge, MA, USA, 1994.

Golub, G., van Loan, C. *Matrix Computations*. The Johns Hopkins University Press, 3rd edition, 1996.

Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science*, 286(5439):531–537, 1999.

Goodacre, R., York, E.V., Heald, J.K., Scott, I.M. Chemometric discrimination of unfractionated plant extracts analyzed by electrospray mass spectrometry. *Phytochemistry*, 62(6):859–863, 2003.

Han, J., Bertin, N., Hao, T., Goldberg, D., Berriz, G., Zhang, L., Dupuy, D., Walhout, A., Cusick, M., Roth, F., Vidal, M. Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430:88–93, 2004.

Harmeling, S., Meinecke, F., Müller, K.R. Injecting noise for analysing the stability of ICA components. *Signal Processing*, 84:255–266, 2004.

Harmeling, S., Ziehe, A., Kawanabe, M., Müller, K.R. Kernel-based nonlinear blind source separation. *Neural Computation*, 15:1089–1124, 2003.

Hassoun, M.H., Sudjianto, A. Compression net-free autoencoders. *Workshop on Advances in Autoencoder/Autoassociator-Based Computations at the NIPS 97 Conference*, 1997.

Hastie, T., Stuetzle, W. Principal curves. *Journal of the American Statistical Association*, 84:502–516, 1989.

Hastie, T., Tibshirani, R., Friedman, J. *The Elements of Statistical Learning*. Springer, 2001.

Haykin, S. *Neural Networks - A Comprehensive Foundation*. Prentice Hall, 2nd edition, 1998.

Haykin, S., ed. *Unsupervised Adaptive Filtering, Vol. 1: Blind Source Separation*. Wiley, 2000a.

Haykin, S., ed. *Unsupervised Adaptive Filtering, Vol. 2: Blind Deconvolution*. Wiley, 2000b.

Hecht-Nielsen, R. Replicator neural networks for universal optimal source coding. *Science*, 269:1860–1863, 1995.

Hestenes, M.R., Stiefel, E. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):409–436, 1952.

Hinton, G. Learning translation invariant recognition in massively parallel networks. In *Proceedings of the Conference on Parallel Architectures and Languages Europe (PARLE)*, pages 1–13, 1987.

Hochreiter, S., Obermayer, K. Feature selection and classification on matrix data: From large margins to small covering numbers. In *NIPS*, pages 889–896, 2002.

Holter, N., Mitra, M., Maritan, A., Cieplak, M., Banavar, J., Fedoroff, N. Fundamental patterns underlying gene expression profiles: simplicity from complexity. *PNAS*, 97 (15):8409–8414, 2000.

Honkela, A., Valpola, H. Unsupervised variational bayesian learning of nonlinear models. In *Advances in Neural Information Processing Systems 17*, 2005. to appear.

Hsieh, W.W. Nonlinear multivariate and time series analysis by neural network methods. *Reviews of Geophysics*, 42, 2004. RG1003, doi:10.1029/2002RG000112.

Hyvärinen, A., Karhunen, J., Oja, E. *Independent Component Analysis*. Wiley, 2001.

Hyvärinen, A., Oja, E. Independent component analysis: Algorithms and applications. *Neural Networks*, 4–5(13):411–430, 2000.

Ihmels, J., Levy, R., Barkai, N. Principles of transcriptional control in the metabolic network of *Saccharomyces cerevisiae*. *Nature Biotechnology*, 22:86–92, 2003.

Jeong, H., Tombor, B., Albert, R., Oltvai, Z., Barabási, A.L. The large-scale organization of metabolic networks. *Nature*, 407:651–654, 2000.

Johnson, H.E., Broadhurst, D., Goodacre, R., Smith, A.R. Metabolic fingerprinting of salt-stressed tomatoes. *Phytochemistry*, 62(6):919–928, 2003.

Jolliffe, I.T. *Principal Component Analysis*. Springer-Verlag, New York, 1986.

Jung, T.P., Makeig, S., Lee, T.W., McKeown, M.J., Brown, G., Bell, A., Sejnowski, T. Independent component analysis of biomedical signals. In *Proc. Int. Workshop on Independent Component Analysis and Signal Separation*, pages 633–644, Helsinki, Finland, 2000.

Jutten, C., Karhunen, J. Advances in nonlinear blind source separation. In *Proc. Int. Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 245–256, Nara, Japan, 2003.

Kaplan, F., Kopka, J., Haskell, D., Zhao, W., Schiller, K., Gatzke, N., Sung, D., Guy, C. Exploring the temperature-stress metabolome of arabidopsis. *Plant Physiology*, 136 (4):4159–4168, 2004.

Kirby, M.J., Miranda, R. Circular nodes in neural networks. *Neural Computation*, 8(2): 390–402, 1996.

Kohonen, T. *Self-Organizing Maps*. Springer, 3rd edition, 2001.

Kramer, M.A. Nonlinear principal component analysis using auto-associative neural networks. *AIChE Journal*, 37(2):233–243, 1991.

Krempel, L. *Visualisierung komplexer Strukturen. Grundlagen der Darstellung mehrdimensionaler Netzwerke*. Max Planck Institut für Gesellschaftsforschung, 2004.

Lappalainen, H., Honkela, A. Bayesian nonlinear independent component analysis by multi-layer perceptrons. In Girolami, M., ed., *Advances in Independent Component Analysis*, pages 93–121. Springer-Verlag, 2000.

Lee, H., Hsu, A., Sajdak, J., Qin, J., Pavlidis, P. Coexpression analysis of human genes across many microarray data sets. *Genome Research*, 14:1085–1094, 2004.

Lee, S.I., Batzoglou, S. Application of independent component analysis to microarrays. *Genome Biology*, 4(11):R76, 2003.

Liebermeister, W. Linear modes of gene expression determined by independent component analysis. *Bioinformatics*, 18(1):51–60, 2002.

Little, R.J.A., Rubin, D.B. *Statistical Analysis with Missing Data*. John Wiley & Sons, New York, 2nd edition, 2002.

Liu, L., Hawkins, D., Ghosh, S., Young, S. Robust singular value decomposition analysis of microarray data. *PNAS*, 100(23):13167–13172, 2003.

Luscombe, N., Babu, M., Yu, H., Snyder, M., Teichmann, S., Gerstein, M. Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, 431:308–312, 2004.

Makeig, S., Westerfield, M., Jung, T.P., Enghoff, S., Townsend, J., Courchesne, E., Sejnowski, T.J. Dynamic Brain Sources of Visual Evoked Responses. *Science*, 295 (5555):690–694, 2002.

Malthouse, E.C. Limitations of nonlinear pca as performed with generic neural networks. *IEEE Transactions on Neural Networks*, 9(1):165–173, 1998.

Martoglio, A.M., Miskin, J.W., Smith, S.K., MacKay, D.J.C. A decomposition model to track gene expression signatures: preview on observer-independent classification of ovarian cancer. *Bioinformatics*, 18:1617–1624, 2002.

Meinecke, F., Ziehe, A., Kawanabe, M., Müller, K.R. A resampling approach to estimate the stability of one-dimensional or multidimensional independent components. *IEEE Transactions on Biomedical Engineering*, 49(12):1514–1525, 2002.

Mewett, D.T., Reynolds, K.J., Nazeran, H. Principal components of recurrence quantification analysis of EMG. In *Proceedings of the 23rd Annual IEEE/EMBS Conference*, Istanbul, Turkey, 2001.

Mika, S., Schölkopf, B., Smola, A., Müller, K.R., Scholz, M., Rätsch, G. Kernel PCA and de–noising in feature spaces. In Kearns, M., Solla, S., Cohn, D., eds., *Advances in Neural Information Processing Systems 11*, pages 536–542. MIT Press, 1999.

Monahan, A.H., Fyfe, J.C., Pandolfo, L. The vertical structure of wintertime climate regimes of the northern hemisphere extratropical atmosphere. *J. Climate*, 16:2005–2021, 2003.

Oba, S., Sato, M., Takemasa, I., Monden, M., Matsubara, K., Ishii, S. A bayesian missing value estimation method for gene expression profile data. *Bioinformatics*, 19 (16):2088–2096, 2003.

Oh, J.H., Seung, H. Learning generative models with the up-propagation algorithm. In Jordan, M.I., Kearns, M.J., Solla, S.A., eds., *Advances in Neural Information Processing Systems*, volume 10, pages 605–611. The MIT Press, 1998.

Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2nd edition, 1992.

Quackenbush, J. Microarray data normalization and transformation. *Nature Genetics*, 32:496–501, 2002. Review.

Roweis, S. EM algorithms for PCA and SPCA. In *Neural Information Processing Systems 10 (NIPS'97)*, pages 626–632, 1997.

Roweis, S.T., Saul, L.K. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

Roweis, S.T., Saul, L.K., Hinton, G.E. Global coordination of locally linear models. In Dietterich, T.G., Becker, S., Ghahramani, Z., eds., *Advances in Neural Information Processing Systems 14*, pages 889–896, Cambridge, MA, 2002. MIT Press.

Saidi, S.A., Holland, C.M., Kreil, D.P., MacKay, D.J.C., Charnock-Jones, D.S., Print, C.G., Smith, S.K. Independent component analysis of microarray data in the study of endometrial cancer. *Oncogene*, 23:6677–6683, 2004.

Sanger, T.D. Optimal unsupervised learning in a single layer linear feedforward network. *Neural Networks*, 2:459–473, 1989.

Saul, L.K., Roweis, S.T. Think globally, fit locally: Unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 4(2):119–155, 2004.

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

Scholz, M., Gatzek, S., Sterling, A., Fiehn, O., Selbig, J. Metabolite fingerprinting: Detecting biological features by independent component analysis. *Bioinformatics*, 20 (15):2447–2454, 2004a.

Scholz, M., Gibon, Y., Stitt, M., Selbig, J. Independent component analysis of starch deficient *pgm* mutants. In Giegerich, R., Stoye, J., eds., *Proceedings of the German Conference on Bioinformatics*, pages 95–104, 2004b.

Scholz, M., Kaplan, F., Guy, C., Kopka, J., Selbig, J. Non-linear PCA: a missing data approach. *Bioinformatics*, 21(20):3887–3895, 2005.

Scholz, M., Selbig, J. Visualization and analysis of molecular data. In Weckwerth, W., ed., *Metabolomics: methods and protocols. Methods in Molecular Biology Series*. Humana Press, New York, USA, 2006. to appear.

Scholz, M., Vigário, R. Nonlinear PCA: a new hierarchical approach. In Verleysen, M., ed., *Proceedings ESANN*, pages 439–444, 2002.

Steuer, R., Kurths, J., Daub, C.O., Weise, J., Selbig, J. The mutual information: Detecting and evaluating dependencies between variables. *Bioinformatics*, 18(2):231–240, 2002.

Steuer, R., Kurths, J., Fiehn, O., Weckwerth, W. Observing and interpreting correlations in metabolomic networks. *Bioinformatics*, 19(8):1019–1026, 2003.

Stock, J., Stock, M. Quantitative stellar spectral classification. *Revista Mexicana de Astronomia y Astrofisica*, 34:143–156, 1999.

Stone, C. Optimal rates of convergence for nonparametric estimators. *Annals of Statistics*, 8:1348–1360, 1980.

Stone, J.V. Independent component analysis: An introduction. *Trends in Cognitive Sciences*, 6(2):59–64, 2002.

Stone, J.V. *Independent Component Analysis: A Tutorial Introduction*. MIT Press, 2004.

Stuart, J., Segal, E., Koller, D., Kim, S. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643):249–255, 2003.

Tang, A.C., Pearlmutter, B.A., Malaszenko, N.A., Phung, D.B., Reeb, B.C. Independent components of magnetoencephalography: Localization. *Neural Computation*, 14(8): 1827–1858, 2002.

Tenenbaum, J., de Silva, V., Langford, J. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

The Arabidopsis Genome Initiative. Analysis of the genome sequence of the flowering plant arabidopsis thaliana. *Nature*, 408:796–815, 2000.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B. Missing value estimation methods for DNA microarrays. *Bioinformatics*, 17:520–525, 2001.

Vapnik, V. *The nature of statistical learning theory*. Springer Verlag, New York, 1995.

Verbeek, J., Vlassis, N., Kröse, B. Procrustes analysis to coordinate mixtures of probabilistic principal component analyzers. Technical report, Computer Science Institute, University of Amsterdam, The Netherlands, 2002.

Vigário, R., Särelä, J., Jousmäki, V., Hämäläinen, M., Oja, E. Independent component approach to the analysis of EEG and MEG recordings. *IEEE Trans. Biomedical Engineering*, 47(5):589–593, 2000.

Wall, M., Rechtsteiner, A., Rocha, L. Singular value decomposition and principal component analysis. In Berrar, D., Dubitzky, W., Granzow, M., eds., *A Practical Approach to Microarray Data Analysis*, pages 91–109. Kluwer, Norwell, MA, 2003.

Watanabe, S. *Pattern recognition: human and mechanical*. Wiley, New York, 1985.

Watts, D., Strogatz, S. Collective dynamics of 'small-world' networks. *Nature*, 393: 440–442, 1998.

Webber Jr., C., Zbilut, J. Dynamical assessment of physiological systems and states using recorrence plot strategies. *Journal of Applied Physiology*, 76:965–973, 1994.

Weckwerth, W. Metabolomics in systems biology. *Annu Rev Plant Biol.*, 54:669–689, 2003.

Weckwerth, W., M.E. Loureiro, K.W., Fiehn, O. Differential metabolic networks unravel the effects of silent plant phenotypes. *PNAS*, 101(20):7809–7814, 2004.

Ziehe, A., Müller, K.R. TDSEP - an efficient algorithm for blind separation using time structure. In *Proc. ICANN'98, Int. Conf. on Artificial Neural Networks*, pages 675–680, 1998.

Index