Helsinki University of Technology
Dissertations in Computer and Information Science
Espoo 2003 Report D4

ADVANCES IN INDEPENDENT COMPONENT ANALYSIS WITH APPLICATIONS TO DATA MINING

Ella Bingham

Dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Department of Computer Science and Engineering for public examination and debate in Auditorium T2 at Helsinki University of Technology (Espoo, Finland) on the 12th of December, 2003, at 12 o'clock noon.

Helsinki University of Technology
Department of Computer Science and Engineering
Laboratory of Computer and Information Science
P.O.Box 5400
FIN-02015 HUT
FINLAND

Distribution:
Helsinki University of Technology
Laboratory of Computer and Information Science
P.O.Box 5400
FIN-02015 HUT
FINLAND
Tel. +358-9-451 3272
Fax +358-9-451 3277
http://www.cis.hut.fi

Available in pdf format at http://lib.hut.fi/Diss/2003/isbn9512268205/

ISBN 951-22-6819-1 (printed version)
ISBN 951-22-6820-5 (electronic version)
ISSN 1459-7020

Otamedia Oy
Espoo 2003

Bingham, E. (2003): **Advances in independent component analysis with applications to data mining**. Doctoral thesis, Helsinki University of Technology, Dissertations in Computer and Information Science, Report D4, Espoo, Finland.

Keywords: independent component analysis, latent variable models, dimensionality reduction, data mining, complex valued signals, random projection, regression, topic identification, 0-1 data.

ABSTRACT

This thesis considers the problem of finding latent structure in high dimensional data. It is assumed that the observed data are generated by unknown latent variables and their interactions. The task is to find these latent variables and the way they interact, given the observed data only. It is assumed that the latent variables do not depend on each other but act independently.

A popular method for solving the above problem is independent component analysis (ICA). It is a statistical method for expressing a set of multidimensional observations as a combination of unknown latent variables that are statistically independent of each other. Starting from ICA, several methods of estimating the latent structure in different problem settings are derived and presented in this thesis. An ICA algorithm for analyzing complex valued signals is given; a way of using ICA in the context of regression is discussed; and an ICA-type algorithm is used for analyzing the topics in dynamically changing text data. In addition to ICA-type methods, two algorithms are given for estimating the latent structure in binary valued data. Experimental results are given on all of the presented methods.

Another, partially overlapping problem considered in this thesis is dimensionality reduction. Empirical validation is given on a computationally simple method called random projection: it does not introduce severe distortions in the data. It is also proposed that random projection could be used as a preprocessing method prior to ICA, and experimental results are shown to support this claim.

This thesis also contains several literature surveys on various aspects of finding the latent structure in high dimensional data.

Preface

This work has been carried out at the Laboratory of Computer and Information Science (CIS) at Helsinki University of Technology during the years 1999–2003. The work has been funded by Helsinki Graduate School in Computer Science and Engineering (HeCSE) and the CIS laboratory. In addition, I have received grants from Ella and Georg Ehrnrooth Foundation, Finnish Cultural Foundation, Foundation of Technology, KAUTE Foundation and Nokia Foundation, all of which I am grateful for.

Professor Erkki Oja, the supervisor of my thesis, has been a fatherly and trustworthy figure for me. Docent Aapo Hyvärinen, the first of my two instructors, has introduced me to the fascinating world of ICA and scientific research in general. Professor Heikki Mannila, my second instructor, has fed my appetite with new problem domains, and has been of great support whenever needed. I feel obliged to all of these three gentlemen.

It has been a great honor for me to have two such distinguished pre-examiners: Professor Thomas Hofmann and Professor Helena Ahonen-Myka, whom I wish to express my gratitude. In addition, it will surely be a pleasant experience to defend my thesis against Dr Mark Plumbley, whom I thank for agreeing to act as my opponent.

I am severely indebted to my scientific collaborators. Especially I would like to thank Dr Ata Kabán for patiently introducing me to the secrets of probabilistic modeling, besides being a great friend. All of my co-authors — Docent Aapo Hyvärinen, Professor Heikki Mannila, Dr Ata Kabán, Professor Mark Girolami and Mr Jouni Seppänen — are experts in their fields, which has made my collaboration with each of them a true pleasure.

I could not imagine a better place to conduct research than the CIS laboratory. We have been blessed with excellent leaders: Professor Erkki Oja and Professor Olli Simula, who trust us enough to let us work freely, and who do their best to ensure us excellent material facilities. The atmosphere in the laboratory is active, yet pleasant and youthful. For this I wish to thank the whole of the personnel, especially Anne, Heli, Johan and Karthikesh.

My parents have supported me in many ways during the writing of this thesis, which I truly appreciate. My dear husband Kenrick, the foremost source of happiness in my life, has been of great help. Thank you so much.

Espoo, November 24, 2003

Ella Bingham

Contents

Notation and abbreviations

a	scalar constant
c	scalar constant
d	dimensionality of $\mathbf{x}$ before dimensionality reduction
f	scalar-valued function of a scalar variable
g	scalar-valued function of a scalar variable
i	index of x_i
j	index of s_j or $\mathbf{w}_j$
k	dimensionality of $\mathbf{x}$ after random projection
m	dimensionality of $\mathbf{x}$ after dimensionality reduction
n	number of latent components; dimensionality of $\mathbf{s}$ or $\mathbf{y}$
p	probability density function
s	independent latent component
t	observation index, time index
x	component of observed vector $\mathbf{x}$
y	latent component
ε	scalar constant
π	probability of a component
θ	parameter; set of parameters
E	expectation operator
N	number of observed vectors $\mathbf{x}$
P	probability
$\mathbf{f}$	vector-valued function of a vector variable
$\mathbf{g}$	vector-valued function of a vector variable
$\mathbf{p}$	column vector of probabilities
$\mathbf{s}$	column vector of independent latent components
$\mathbf{w}$	column vector of a projection direction
$\mathbf{x}$	observed column vector
$\mathbf{y}$	column vector of latent components
$\mathbf{x}^T$	vector $\mathbf{x}$ transposed (applicable to any vector)
$\mathbf{x}^H$	vector $\mathbf{x}$ transposed and complex conjugated (applicable to any vector)

A mixing matrix; topic matrix
B binary noise matrix
D matrix of eigenvalues
E matrix of eigenvectors
P matrix of probabilities
R random projection matrix
S matrix of independent latent components **s**
V random matrix in random sampling and quantization
W unmixing matrix
X matrix of observed vectors **x**
Y matrix of latent components

BSS	blind signal separation
EM	expectation-maximization
GTM	generative topographic mapping
ICA	independent component analysis
IR	information retrieval
LDA	latent Dirichlet allocation
LSA	latent semantic analysis
MAP	maximum a posteriori
ML	maximum likelihood
MLP	multilayer perceptron
MPCA	multinomial principal component analysis
MSE	mean squared error
NMF	nonnegative matrix factorization
PCA	principal component analysis
PLSA	probabilistic latent semantic analysis
RP	random projection
SNLP	statistical natural language processing
SOM	self-organizing map
SSE	sum of squared errors
SVD	singular value decomposition

Chapter 1

About the thesis

1.1 Scope of the thesis

This doctoral thesis considers the problem of finding latent structure in high dimensional data. Here the term *latent* means hidden, unknown or unobserved; the term *structure* refers to some regularities in the data; *high dimensional* may be tens or tens of thousands of dimensions, depending on the situation; and *data* is any information that can be transformed into numerical values, most often represented as a matrix of multidimensional observations where each dimension corresponds to a variable whose value we can somehow measure. The aims in this thesis are to answer the question "What is there in the data?", to form a simple representation of a large data set that is difficult to analyze as such, and to present the data in a form that is understandable to a human observer.

Throughout the thesis, it will be assumed that the observed data are generated by interactions between latent variables. The objective is to find out what these latent variables are and how they interact — this is the key to understanding what the data are about. The latent variables will be called *components*, *sources* or *topics*: the data are composed of these latent variables, or the latent variables are the sources of variability in the data, or in particular in text document data the latent variables are the topics of discussion. Depending on the point of view, the "structure" in the data we referred to in the beginning is either due to the values taken by the latent variables or due to the way the latent variables interact. Throughout this thesis, we will assume that there are no inherent dependencies between the latent variables.

In addition to revealing the latent structure in high dimensional data, another aim of this thesis is to present ways of reducing the dimensionality of the data. This aim overlaps partially with the first one: we wish to transform the data into a denser representation and only retain the most important aspects of the data.

Let us present an example of the problem of finding latent structure in the data. A popular example is the so called cocktail party problem: Imagine a room full of people discussing with each other. A few microphones, located at different positions in the room, collect the

sounds of mixed human voices and possible external noises. An outsider listening to the mixtures of sounds recorded by the microphone cannot decipher what was actually discussed in the room. The task now is to decompose the mixtures of sounds back into their original form, that is, human voices and external noises. These original sounds are called the latent sources, as they are "hidden" from the outsider listener. The task is often referred to as source separation. The computational methods discussed later in this thesis are aimed at solving problems similar to this one.

Although the above example is old and frequently cited, it is repeated here because it suits nicely some of the specific contributions of this thesis. Firstly, the sounds arriving at the microphones are more or less delayed and may contain echoes from nearby walls. This poses additional problems in decomposing the mixtures of sounds. One way to overcome these problems will be discussed in Chapter 3 of the thesis. Secondly, imagine that instead of people speaking, we observe their written conversations. A chat room in the Internet is like a big cocktail party where lots of people discuss different topics simultaneously. Again, an outsider cannot at first sight understand what people are discussing, as different discussions get intertwined as they appear on the computer screen. This problem is tackled in Chapters 5 and 6 of the thesis where we present methods for finding out the latent topics of a discussion.

In short, the methods discussed in this thesis estimate the structure in the data by finding latent components whose interactions might have generated the data. We do not know which these latent components might be, neither do we know about the exact way they interact. Nevertheless, we are willing to assume that there are indeed some interactions, so that a typical observation is not generated by one latent component only. To cast more light on this, it may be helpful to contrast our approach with two well-known and different ones.

First, the methods used to analyze the data in this thesis are *unsupervised* in contrast to supervised; that is, there is no teacher telling us whether our decomposition is correct or not. No labeled examples, with known input values and corresponding output values[1] or with a known input-output structure, are given for building a model of the data. Instead, the unsupervised methods try to infer the structure of the data simply by looking at the values taken by the observed variables. Often it is even the case that no "correct" solution or structure exists and we can only try to give a "good enough" characterization of the data. Then the essential question is, how to characterize the goodness in a strict mathematical sense.

Second, a popular way of presenting the structure in high dimensional data is clustering: either the observed data points or the observed variables are organized into groups. We will not study the basic problem of clustering in this thesis. In a clustering problem, it is assumed that each observation (similarly, each observed variable) belongs to exactly one cluster. In contrast, we wish to allow the generation of an observation by several latent variables simultaneously; using the terminology of clustering, we allow an observation (similarly, an observed variable) to belong to several clusters simultaneously. Also, in a basic clustering setting, the focus is either on clustering the observations or the observed variables. In our setting, the latent structure of the data gives rise to both the observations and the observed variables, and in a way we are clustering both of them simultaneously.

[1] To be exact, labeled examples of predictor and predicted variables are used in the regression problem discussed in Chapter 4, as those are an essential element of regression estimation. Nevertheless, the structure in the data is unknown in this case, too.

In this thesis, the main method for analyzing latent structure in the data is *independent component analysis* (ICA), described in Chapter 2. ICA can also be seen as a way of dimensionality reduction, although that is not its primary aim. Several other methods for these two overlapping tasks will be discussed, too.

The original methods and intuitions behind ICA will be extended in various directions: into different kinds of data (complex valued in contrast to real valued, and binary valued in contrast to continuous valued) and into different problem settings (regression problems and information retrieval).

In this thesis, the point of view taken is often that of *data mining*. Data mining is a name used for a variety of computational methods and techniques for analyzing large data sets. The aim in data mining is to describe the data either in a global or a local level. Global descriptions include clustering, joint probability density estimation, or visualization of the data; local descriptions might be repeating or exceptional patterns in the data, or statistical dependencies between the variables. Although data mining is closely related to traditional statistical data analysis, it has a couple of distinguishing characteristics: the data are not originally aimed for a particular study and so the analyst cannot affect the process of data collection; the data set is often so large that its storage and retrieval must be carefully designed; the emphasis is on local aspects in addition to global behavior in the data. An introduction to data mining is given in [56] and data mining and statistics are compared in [46].

This thesis consists of an introductory part and six publications, listed in Section 1.3. Throughout the introductory part of the thesis, the reader is referred to the publications. They contain most of the contributions of this work and are self-explanatory. The derivations, results and discussions of the publications are seldom repeated in this introductory part. It is assumed that the reader is familiar with the basics of linear algebra, probability and statistics.

1.2 Contributions of the thesis

The scientific contributions of this thesis include the following.

- Experimental results are given on using random projection as a method of dimensionality reduction. In particular, experimental results on the use of a sparse random matrix have not been presented elsewhere.
- The use of random projection as a data preprocessing method for independent component analysis (ICA) is suggested. Empirical validation is presented in the cases of ICA of image data, complex valued signals and text document data.
- A fast fixed-point ICA algorithm for separating linearly mixed complex valued source signals is presented and the local consistency of the estimator given by the algorithm is proved.
- Empirical validation of using ICA as a preprocessing method in nonlinear regression is given.

- It is shown that an ICA-type algorithm can successfully extract the topics of discussion in dynamically evolving natural language text.
- Two algorithms for the estimation of latent structure in binary valued data are given, together with empirical results.
- Literature surveys are given on each topic addressed in the thesis: latent variable decompositions, separation of complex valued signals, ICA-type methods in regression and in the analysis of text documents, and latent variable models of binary valued data.

1.3 Publications of the thesis

Publication 1. Ella Bingham and Aapo Hyvärinen. A fast fixed-point algorithm for independent component analysis of complex valued signals. *International Journal of Neural Systems*, 10(1):1–8, February 2000.

Publication 2. Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In Foster Provost and Ramakrishnan Srikant, editors, *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 245–250, San Francisco, CA, USA, August 2001.

Publication 3. Aapo Hyvärinen and Ella Bingham. Connection between multilayer perceptrons and regression using independent component analysis. *Neurocomputing*, 50(C):211–222, January 2003.

Publication 4. Ella Bingham, Ata Kabán, and Mark Girolami. Topic identification in dynamical text by complexity pursuit. *Neural Processing Letters*, 17(1):69–83, 2003.

Publication 5. Ella Bingham, Heikki Mannila, and Jouni K. Seppänen. Topics in 0-1 data. In David Hand, Daniel Keim, and Raymond Ng, editors, *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 450–455, Edmonton, Alberta, Canada, July 2002.

Publication 6. Jouni K. Seppänen, Ella Bingham, and Heikki Mannila. A simple algorithm for topic identification in 0-1 data. In Nada Lavrač, Dragan Gamberger, Ljupčo Todorovski, and Hendrik Blockeel, editors, *Knowledge Discovery in Databases: PKDD 2003. 7th European Conference on Principles and Practice of Knowledge Discovery in Databases. Cavtat-Dubrovnik, Croatia, September 2003, Proceedings*, number 2838 in Lecture Notes in Artificial Intelligence, pages 423–434. Springer, 2003.

Contents of the publications and the contributions of Ella Bingham

In Publication 1, an ICA algorithm for separating linear mixtures of complex valued source signals is presented. The fixed-point algorithm is somewhat similar to the FastICA algorithm [76, 70] which had been developed for real valued signals. The local consistency of the estimator given by the algorithm is proved, too. Ella Bingham was responsible for deriving the fixed-point algorithm, proving the theorem of the local consistency, planning and con-

ducting the experiments reported in the paper, studying the relation to subspace methods, and mainly writing the manuscript.

In Publication 2, the use of random projection as a tool of dimensionality reduction is discussed. Extensive experiments on text document data and both noisy and noiseless images are presented. Also, experimental results on using a sparsely populated random matrix as presented by Achlioptas [1] are given — to the knowledge of the authors of the paper and Achlioptas, these are the first experiments on using sparse random projection. Ella Bingham planned and carried out all the experiments in the paper and wrote most of the manuscript.

Publication 3 discusses the use of independent component analysis in regression. When only a subset of the variables are observed, ICA can be used to predict the values of missing variables. It is shown that this kind of regression is closely related to regression by a multilayer perceptron (MLP) network. Ella Bingham was responsible for the experimental results in the paper.

In Publication 4, an ICA-type algorithm is applied to estimating the dynamically changing topics of discussion in textual data. The algorithm, complexity pursuit [71], decomposes a multidimensional time series into components whose probability distributions have low coding complexity. The textual data in the paper is chat line discussion, and meaningful topics of discussion are found. Ella Bingham wrote most of the paper and planned and carried out all the experiments.

Publication 5 presents methods for analyzing the latent structure of binary-valued data. Ordinary ICA methods have problems in the case of binary or nonnegative sources, and new methods are proposed. Ella Bingham participated in defining the data model and algorithms presented in the paper. She designed and conducted most of the experiments, and participated in writing of the paper.

Publication 6 continues along the lines of Publication 5 in analyzing latent structure in binary valued data. One of the algorithms given in Publication 5 is now enhanced. Ella Bingham showed that the lift statistic can be described in matrix form and derived the corresponding algorithm for estimating the topic structure and topic-attribute probabilities. She carried out and analyzed most of the experiments presented in the paper. She also participated in defining the data model, planning the experiments, and writing the paper.

1.4 Structure of the thesis

This thesis describes several ways of analyzing latent structure in data. The main method for doing this is independent component analysis (ICA), which is extended in various different ways in the original publications of the thesis. These extensions are fairly independent of each other and thus each of them will be discussed separately in this introductory part, always keeping in mind the connection to original ICA.

Chapter 2 of this introductory part describes the main method of analyzing latent structure of data in this thesis, namely ICA. An overview of different ICA algorithms is given. Data preprocessing is also discussed as that is the topic of Publication 2 of the thesis; the pub-

lication is briefly reviewed. Other methods of estimating the latent structure of data are discussed in the end of Chapter 2.

In Chapter 3, a new ICA algorithm for the case of complex valued signals and sources is presented. The problem is discussed along the lines of Publication 1, together with new experimental results.

Chapter 4 and Publication 3 present a way of using ICA in regression problems and discuss its connections to regression by neural networks.

Chapter 5 discusses how ICA can be used in text mining. First, some general ideas of statistical natural language processing are discussed. Then a review is given of the approach taken in Publication 4, namely using an ICA-type algorithm for finding the latent topics of discussion in dynamically evolving text data. Also, some new experimental results are shown.

Chapter 6 considers the problem of analyzing binary valued data where extra constraints are given on the form of the latent structure being sought for. Basic linear ICA cannot be used under such constraints. This chapter reviews and extends Publications 5 and 6.

Finally, Chapter 7 gives a brief conclusion.

Chapter 2

Independent component analysis

2.1 Introduction

Independent component analysis (ICA) ([29, 84, 72]) is a well-known method of finding latent structure in data. ICA is a statistical method that expresses a set of multidimensional observations as a combination of unknown latent variables. These underlying latent variables are called sources or independent components and they are assumed to be statistically independent of each other. The ICA model is

$$\mathbf{x} = \mathbf{f}(\theta, \mathbf{s}) \tag{2.1}$$

where $\mathbf{x} = (x_1, \ldots, x_m)$ is an observed vector and $\mathbf{f}$ is a general unknown function with parameters θ that operates on statistically independent latent variables listed in the vector $\mathbf{s} = (s_1, \ldots, s_n)$. A special case of (2.1) is obtained when the function is linear, and we can write

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{2.2}$$

where $\mathbf{A}$ is an unknown $m \times n$ mixing matrix. In Formulae (2.1) and (2.2) we consider $\mathbf{x}$ and $\mathbf{s}$ as random vectors. When a sample of observations $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ becomes available, we write $\mathbf{X} = \mathbf{A}\mathbf{S}$ where the matrix $\mathbf{X}$ has observations $\mathbf{x}$ as its columns and similarly the matrix $\mathbf{S}$ has latent variable vectors $\mathbf{s}$ as its columns. The mixing matrix $\mathbf{A}$ is constant for all observations.

Throughout this thesis, matrices are denoted by uppercase boldface letters, vectors by lowercase boldface letters and scalars by lowercase letters. An entry (i, j) of a matrix is denoted as $\mathbf{A}(i, j)$. Sometimes we write $\mathbf{A}_{m \times n}$ to indicate that $\mathbf{A}$ is an $m \times n$ matrix. The entries of a vector are denoted by the same letter as the vector itself as shown after Formula (2.1); generally, y is an element of $\mathbf{y}$ and so on. All vectors are column vectors.

The linear model (2.2) is identifiable under the following fundamental restrictions [29]: at most one of the independent components s_j may be Gaussian, and the matrix $\mathbf{A}$ must be of full column rank. The identifiability of the model is proved in [29] in the case $n = m$ and for

those source densities whose variance is defined. Recently, the identifiability of more general mixing models and source densities has been discussed in [39].[1] Generally, independent components s_j in the linear model (2.2) can be estimated up to a permutation of their order and a scaling of their values.

What is ICA used for? The most well-known applications of ICA are in the field of signal processing: biomedical, speech and telecommunications signals to mention a few. Brain activity is often measured by the electroencephalogram (EEG), magnetoencephalogram (MEG) or functional magnetic resonance imaging (fMRI), which are recordings of electric and magnetic fields on the surface of the head. These signals can be seen as mixtures of different physical activities and external noise sources. ICA has been successfully used to extract different sources in multidimensional measured signals. Similarly, separation of different speech signals, recorded at microphones at different locations, possibly time delayed and noise corrupted, is a problem that can be cast in the ICA framework. In the third application area, telecommunications, a common transmission line has to be divided among several users. The code division multiple access (CDMA) technique is a modern way to accomplish this: each user has an individual code that distinguishes his signal from the others as the signals are mixed during transmission. Other applications of ICA include feature extraction in images and finding hidden factors in financial data. The applications mentioned here are discussed in depth in [72]. Some of the newer application areas will be discussed in this thesis.

There are two schools of thought with respect to what actually is the aim in estimating the independent components in the data. First, one may regard the data being generated by a combination of some existing but unknown independent source signals s_j, and the task is to estimate them. This viewpoint is chosen in the so called blind source separation (BSS) framework — there are some sources which have been mixed, and the mixing process is completely unknown to us (hence the word "blind"). The application areas of ICA listed above mostly fall into the BSS category.

Another point of view is to regard ICA as a method of presenting the data in a more comprehensible way by revealing the hidden structure in the data and often reducing the dimensionality of the representation. According to this latter school of thought, it might well be that there are no "true" source signals generating the data — it still pays to represent the data as a combination of a few latent factors that are statistically as independent as possible. This view can be called a data mining approach of the problem.

This thesis mostly concentrates on the data mining viewpoint of ICA, but the BSS approach is also taken, in particular in Publication 1. Also, this thesis concentrates on the linear mixing model in Formula (2.2). Nonlinear mixing is briefly discussed in Chapter 6.

ICA can also be seen as a method of dimensionality reduction as far as we interpret dimensionality reduction as finding a parsimonious representation of the data. Dimensionality reduction is not the primary aim of ICA and in fact most ICA algorithms favor moderate dimensionalities (say a few dozens compared to a few hundreds or more) of data — this will be discussed more in Section 2.3. In any case, assuming the ICA model $\mathbf{X} = \mathbf{AS}$ holds and the data matrix $\mathbf{X}$ is of size $m \times N$, the mixing matrix $\mathbf{A}$ is of size $m \times n$ with $m > n$, and the source matrix $\mathbf{S}$ is of size $n \times N$, we have $mN > mn + nN$ and thus we are able to

[1] In [39], Eriksson discusses real valued signals. The results generalize to complex valued signals as well, although not in a straightforward manner (Eriksson, personal communication).

present the observed data with fewer parameters using the ICA model.

2.2 Estimation of the ICA model

The task in ICA is to find both the latent variables or sources s_j and the mixing process; in the linear case, the latter task consists of finding the mixing matrix $\mathbf{A}$. A popular approach is to find a demixing or separating matrix $\mathbf{W}$ so that variables y_j in $\mathbf{y} = \mathbf{W}\mathbf{x}$ are estimates of s_j up to scaling and permutation. Hence $\mathbf{W}$ is an estimate of the (pseudo)inverse of $\mathbf{A}$ up to scaling and permutation of the rows of $\mathbf{W}$. Often the latent variables s_j are estimated one by one, by finding a column vector $\mathbf{w}_j$ (this will be stored as a row of $\mathbf{W}$) such that $y_j = \mathbf{w}_j^T\mathbf{x}$ is an estimate of s_j.

There are several approaches to estimating the independent components and the mixing matrix, resulting in different algorithms. Some of the approaches are briefly reviewed here. In all approaches, an objective or a *contrast function*[2] G is first chosen. G is a smooth scalar valued function of $\mathbf{w}$ that measures the goodness of the result of the estimation in one way or another, and different G are chosen in different approaches. Its derivative g, sometimes called an activation function, typically appears in the algorithm as a nonlinear function.

The first approach is maximization of non-Gaussianity of the components. According to the central limit theorem, sums of independent non-Gaussian random variables are closer to being Gaussian than the original random variables. Thus a linear combination $y = \sum_i b_i x_i$ of the observed variables x_i (which in turn are linear combinations of the independent components s_j) will be maximally non-Gaussian if it equals one of the independent components s_j. This is seen by a counterexample: if y does not equal one of the s_j but is a mixture of two or more s_j, then by spirit of the central limit theorem, y is more Gaussian than each of the s_j.[3] Thus the task is to find $\mathbf{w}_j$ such that the distribution of $y_j = \mathbf{w}_j^T\mathbf{x}$ is as far from Gaussian as possible. Non-Gaussianity is often measured by higher order cumulants such as kurtosis or skewness, although they are not robust against outliers. Robust measures have been presented in, e.g., [70]. Non-Gaussianity can be shown to have a rigorous connection to minimization of mutual information (discussed next), so we do not rely on the heuristic justification given by the central limit theorem only.

The second approach to solving the ICA problem is to use information-theoretic measures. Statistical independence between two random variables is obtained when their mutual information is zero. Mutual information is expressed in terms of marginal entropies of the variables. Among all random variables of unit variance, a Gaussian variable has the largest entropy. Negentropy is a convenient measure of entropy: it is always nonnegative, and zero for Gaussian variables. To maximize the independence between random variables, one can make the variables as non-Gaussian as possible. Thus this approach is in line with the first one. Information-theoretic measures are described in detail in [32] and their connection to ICA estimation is explained in, e.g., [72]. Negentropy is difficult to compute, and in practice it is approximated by cumulants. Again, the instability of the cumulants in the case of

[2]To be exact, the contrast function is $J_G(\mathbf{w}) = E\{G(\mathbf{w}^T\mathbf{x})\}$ in several references, but for brevity of notation, G is used when referring to the contrast function.

[3]To be precise, the central limit theorem only speaks about the asymptotic behavior of sums of random variables.

outliers suggests using some other contrast functions that have more desirable properties.

The third approach to estimating the ICA model is maximum likelihood (ML) estimation. In ML, one selects those parameter values that give the highest probability to the observations. If prior information on the parameters is taken into account, the method becomes the maximum a posteriori (MAP) method. ICA algorithms based on the ML method include the Bell-Sejnowski algorithm, also called the Infomax principle [13], and the natural gradient algorithm [6]. Mutual information is a unifying framework for the ML principle, too [72].

The fourth approach to ICA estimation are tensorial methods. The most well-known among these are the FOBI (first-order blind identification) [22] and JADE (joint approximate diagonalization of eigenmatrices) [25] algorithms. Tensors are generalizations of linear operators — in particular, cumulant tensors are generalizations of the covariance matrix. Minimizing the higher order cumulants approximately amounts to higher order decorrelation, and can thus be used to solve the ICA model. However, the statistical properties of the tensor methods may be inferior to the methods described above, and they are very burdensome in high dimensions [72].

An algorithm that can be used in all the previously listed ICA approaches is the FastICA algorithm[4] [76, 70, 72]. The algorithm is an iterative fixed-point algorithm with the following update for $\mathbf{w}$:

$$\mathbf{w} \leftarrow E\{\mathbf{x}g(\mathbf{w}^T\mathbf{x})\} - E\{g'(\mathbf{w}^T\mathbf{x})\}\mathbf{w} \tag{2.3}$$

where $\mathbf{w}$ is one of the rows of the unmixing matrix $\mathbf{W}$. In practice, the expectations are replaced by their empirical estimates. The nonlinear function g is chosen so that it is the derivative of the non-quadratic contrast function G that measures negentropy, non-Gaussianity, or whatever is our objective function. The algorithm was first suggested for the kurtosis cost function in [76]. Other choices of G are discussed in [70] and [72] — robust choices are non-polynomial functions such as $\log\cosh$ or $\exp(-y^2)$. Contrary to many other algorithms, in FastICA the choice of the contrast function does not severely restrict the type of the independent components that we are able to estimate. The choice of G is important only if one wishes to optimize the performance of the algorithm in some way.

Before running the algorithm (2.3), the data are transformed such that they have zero mean and preprocessed by whitening (described in Section 2.3.1). An initial unit norm vector $\mathbf{w}$ is chosen randomly. After each iteration step (2.3), $\mathbf{w}$ is again normalized to have unit norm. The iteration is continued until the direction of $\mathbf{w}$ does not change significantly.

In the so called deflationary approach, the independent components s_j are estimated one by one, and it must be ensured that the rows $\mathbf{w}_j$ of the unmixing matrix are orthogonal. This is done after every iteration step (2.3) by subtracting from the current $\mathbf{w}_j$ the projections of all previously estimated $\mathbf{w}_p$, $p = 1, \ldots, j-1$. The vector $\mathbf{w}_j$ is normalized and its convergence is tested only after this orthogonalization step. The cubic convergence of the deflationary algorithm was proved in [76].

In the symmetric approach, all independent components s_j are estimated simultaneously. The iteration step (2.3) is computed for all $\mathbf{w}_j$, and after that the matrix $\mathbf{W}$ containing $\mathbf{w}_j$ as its rows is orthogonalized. This is done at each round. The orthogonalization of $\mathbf{W}$ is

[4] http://www.cis.hut.fi/projects/ica/fastica/

accomplished either by

$$\mathbf{W} \leftarrow (\mathbf{W}\mathbf{W}^T)^{-1/2}\mathbf{W} \tag{2.4}$$

or iteratively by [70]

1. $\mathbf{W} \leftarrow \mathbf{W}/||\mathbf{W}||$ (2.5)
2. $\mathbf{W} \leftarrow \frac{3}{2}\mathbf{W} - \frac{1}{2}\mathbf{W}\mathbf{W}^T\mathbf{W}$ (2.6)
3. If $\mathbf{W}\mathbf{W}^T$ is not close enough to identity, go back to step 2. (2.7)

The good convergence properties of the symmetric FastICA algorithm are discussed in [111].

2.3 Data preprocessing for ICA

It is often beneficial to reduce the dimensionality of the data before performing ICA. It might well be that there are only a few latent components in the high-dimensional observed data, and the structure of the data can be presented in a compressed format. Estimating ICA in the original, high-dimensional space may lead to poor results. For example, several of the original dimensions may contain only noise. Also, overlearning is likely to take place in ICA if the number of the model parameters (i.e., the size of the mixing matrix) is large compared to the number of observed data points [74]. Care must be taken, though, so that only the redundant dimensions are removed and the structure of the data is not flattened as the data are projected to a lower dimensional space. In this section two methods of dimensionality reduction are discussed: principal component analysis and random projection.

In addition to dimensionality reduction, another often used preprocessing step in ICA is to make the observed signals zero mean and decorrelate them. The decorrelation removes the second-order dependencies between the observed signals. It is often accomplished by principal component analysis which will be briefly described next.

2.3.1 Principal component analysis

In *principal component analysis* (PCA) [122, 67], an observed vector $\mathbf{x}_{orig}$ is first centered by removing its mean (in practice, the mean is estimated as the average value of the vector in a sample). Then the vector is transformed by a linear transformation into a new vector, possibly of lower dimension, whose elements are uncorrelated with each other. The linear transformation is found by computing the eigenvalue decomposition of the covariance matrix, which for zero-mean vectors is the correlation matrix $E\{\mathbf{x}_{orig}\mathbf{x}_{orig}^T\}$ of the data. The eigenvectors of $E\{\mathbf{x}_{orig}\mathbf{x}_{orig}^T\}$ form a new coordinate system in which the data are presented.

The decorrelating process is called *whitening* or *sphering* if also the variances of each element of the new data vector are set to unity. This can be accomplished by scaling the vector elements by the inverses of the eigenvalues of the correlation matrix. In all, the whitened data have the form

$$\mathbf{x} = \mathbf{D}^{-1/2}\mathbf{E}^T\mathbf{x}_{orig} \tag{2.8}$$

where $\mathbf{x}$ is the whitened data vector, $\mathbf{D}$ is a diagonal matrix containing the eigenvalues of the correlation matrix and $\mathbf{E}$ contains the corresponding eigenvectors of the correlation matrix as its columns. In practice, the expectation in the correlation matrix is computed as the sample mean. Subsequent ICA estimation is done on $\mathbf{x}$ instead of $\mathbf{x}_{orig}$. For whitened data it is enough to find an orthogonal demixing matrix if the independent components are also assumed white.

Dimensionality reduction is performed by PCA simply by choosing the number of retained dimensions, m, and projecting the d-dimensional observed vector $\mathbf{x}_{orig}$ to a lower dimensional space spanned by the m ($m < d$) dominant eigenvectors (that is, eigenvectors corresponding to the largest eigenvalues) of the correlation matrix. Now the matrix $\mathbf{E}$ in Formula (2.8) has only m columns instead of d, and similarly $\mathbf{D}$ is of size $m \times m$ instead of $d \times d$, if whitening is desired.

There is no clear way to choose the number of retained dimensions in practice. In theory, the rank of $\mathbf{X}$ is equal to the rank of $\mathbf{S}$ in the noiseless case, so it is enough to compute the number of non-zero eigenvalues of $\mathbf{X}$. The problem is discussed in, e.g., [72, 149]. One often chooses the number of largest eigenvalues so that the chosen eigenvectors explain the data well enough, for example, 90 per cent of the total variance in the data. As PCA preprocessing for ICA always involves the risk that the true independent components are not in the space spanned by the dominant eigenvectors, it is often advisable to estimate fewer independent components than what is the dimensionality of the data after PCA. Trial and error are often needed in determining both the number of eigenvectors and the number of independent components estimated.

PCA is a convenient method for estimating the structure of the data, assuming that the distribution of the data is roughly symmetric and unimodal. PCA finds the orthogonal directions in which the data have maximal variance. PCA is an optimal method of dimensionality reduction in the mean-square sense: data points projected into the lower dimensional PCA subspace are as close as possible to the original high dimensional data points, meaning that

$$\sum_t ||\mathbf{x}_{orig}(t) - \mathbf{x}(t)||^2 \tag{2.9}$$

is minimized. Here we denote by $\mathbf{x}_{orig}(t)$ the t-th original observation vector and by $\mathbf{x}(t)$ its projection.

2.3.2 Random projection

Computing the PCA of a high-dimensional data set is computationally burdensome. In this thesis it is proposed that *random projection* (RP) is a suitable preprocessing method for ICA: using RP before PCA significantly reduces the computational load without introducing severe distortions in the data set.

Random projection is a method of dimensionality reduction. In Publication 2 of the thesis, several examples of its use are given, together with discussions on its suitability. In random projection, the original high-dimensional data matrix $\mathbf{X}^{orig}_{d\times N}$ is projected into a lower-dimensional space using a random matrix $\mathbf{R}_{k\times d}$ ($k \ll d$) whose columns have unit lengths,

resulting in a $k \times N$ dimensional matrix $\mathbf{X}^{RP}$:

$$\mathbf{X}^{RP} = \mathbf{R}\mathbf{X}^{orig} \tag{2.10}$$

The usefulness of random projections stems from the Johnson-Lindenstrauss lemma [79]: if points in a vector space are projected onto a randomly selected subspace of suitably high dimension, then the distances between the points are approximately preserved. Strictly speaking, (2.10) is not a projection because $\mathbf{R}$ is generally not orthogonal. A linear mapping such as (2.10) can cause significant distortions in the data set if $\mathbf{R}$ is not orthogonal. Orthogonalizing $\mathbf{R}$ is unfortunately computationally expensive. Instead, we can rely on a result presented by Hecht-Nielsen [58]: in a high-dimensional space, there exists a much larger number of almost orthogonal than orthogonal directions. Thus vectors having random directions are sufficiently close to orthogonal with a high probability, and equivalently $\mathbf{R}^T\mathbf{R}$ approximates an identity matrix.

Consider the linear ICA model for the original data vectors $\mathbf{x}_{orig}$, as in Formula (2.2). Reducing the dimensionality by random projection does not violate the identifiability of the model, as the independent components stay intact and only the mixing matrix is changed:

$$\mathbf{x}^{RP} = \mathbf{R}\mathbf{x}_{orig} = \mathbf{R}\mathbf{A}\mathbf{s} = \mathbf{A}^{RP}\mathbf{s} \tag{2.11}$$

where we define $\mathbf{A}^{RP} = \mathbf{R}\mathbf{A}$ to emphasize that the ICA model still holds: $\mathbf{x}^{RP} = \mathbf{A}^{RP}\mathbf{s}$. Here it is assumed that k, the dimensionality of $\mathbf{x}^{RP}$, is still larger or equal to n, the dimensionality of $\mathbf{s}$, making $\mathbf{A}^{RP}$ of full column rank.

Thus we propose that random projection could be used prior to PCA, to reduce the dimensionality from the original d to some k ($k \ll d$). The whitening of the data by PCA in the new, lower-dimensional space is significantly less demanding. (The computational complexities of random projection and PCA are discussed in Publication 2.) One may then either reduce the dimensionality further by PCA or directly estimate ICA in the k-dimensional space.

Achlioptas [1] suggests the use of sparse random matrices instead of a random matrix whose entries are Gaussian distributed (which is the usual choice in the random projection literature). An entry of $\mathbf{R}$ is then

$$\mathbf{R}(i,j) = \sqrt{a} \cdot \begin{cases} +1 & \text{with probability } \frac{1}{2a} \\ 0 & \text{with probability } 1 - \frac{1}{a} \\ -1 & \text{with probability } \frac{1}{2a} \end{cases} \tag{2.12}$$

where $a > 1$ is some constant. To the knowledge of the authors of Publication 2 and Achlioptas himself, Publication 2 is the first one in which experimental results on sparse random projection are presented.

Let us present a simple example on using random projection as a preprocessing method in ICA. A total of 24 monochrome images of natural scenes were randomly mixed to 600 mixtures using a (600×24)-dimensional mixing matrix: $\mathbf{X}_{600\times32768} = \mathbf{A}_{600\times24}\mathbf{S}_{24\times32768}$; the number of pixels in each image was 32768. The demixing matrix was found by applying FastICA either on PCA preprocessed data or on data that were first randomly projected to a lower dimensional space and then PCA preprocessed. Table 2.1 lists the separation accuracies

and the number of floating point operations needed. The separation accuracy is measured as the sum of squared errors between the product of mixing and unmixing matrices (where the matrices used for preprocessing were taken into account) and a permutation matrix. The results in Table 2.1 are averages over 10 runs. In the first case, the dimensionality was directly reduced to 24 (which is the number of independent components) by PCA. In the second case, ordinary random projection with a Gaussian distributed random matrix was used to reduce the dimensionality from 600 to 30, and then PCA was used to further reduce the dimensionality to 24. PCA is computationally cheap in this low-dimensional random projected space. In the third case, a sparse random projection matrix was used instead of the Gaussian distributed random projection matrix, still somewhat lessening the computational burden. The sparse random matrix was generated using Formula (2.12) with $a = 3$. We see that random projection gives computational savings but almost no loss in separation accuracy.

Table 2.1: Estimation errors and computational loads with different preprocessing methods in ICA

Preprocessing method	SSE	Flops
PCA to $k = 24$ directly	1.63	$2.84 \cdot 10^{10}$
RP to $k = 30$ before PCA to $k = 24$	1.60	$1.98 \cdot 10^{9}$
Sparse RP to $k = 30$ before PCA to $k = 24$	1.65	$9.10 \cdot 10^{8}$

In Publication 2 the performance of random projection was compared to several other methods of dimensionality reduction: principal component analysis, singular value decomposition, discrete cosine transform and median filtering. The application areas were text documents and both noisy and noiseless images. The measure of performance was the distortion in the similarity of randomly chosen data vectors that took place when the dimensionality of the data was reduced. The similarity of two data vectors was computed by using either their Euclidean distance or inner product. Also, the computational complexities of the dimensionality reduction methods were compared by measuring the number of floating point operations. The results of Publication 2 indicate that random projection is a promising method for dimensionality reduction that does not introduce a great distortion in the data, while being computationally very simple.

As random projection preserves the interpoint distances well, it is most suitable for those application areas where every dimension of the data is more or less equally important and has a similar scale, and the interpoint distances are meaningful — for example text document data (assuming that the vocabulary is chosen appropriately) or data sets where the Euclidean distance is a meaningful distance measure. In some other applications, for example in process monitoring, some measured quantities (that is, dimensions) might be closely correlated with each other or are scaled very differently, and the interpoint distances do not necessarily bear a clear meaning.

2.3.3 Other random low rank matrix approximations

Achlioptas and McSherry [2] have presented simple techniques for accelerating the computation of a low rank approximation of a matrix $\mathbf{X}$ in the case $\mathbf{X}$ has strong spectral structure (that is, the largest singular values of $\mathbf{X}$ are significantly greater than those of a random matrix with size and entries similar to $\mathbf{X}$). They sample and/or quantize the entries of $\mathbf{X}$, thus reducing the number of non-zero entries and/or the length of their representation. The theoretical validity of such procedures relies on the fact that sampling and/or quantization can be seen as adding a random matrix $\mathbf{V}$ to $\mathbf{X}$ — with high probability, $\mathbf{V}$ has very weak spectral structure, and the effects of sampling and quantization nearly vanish when a low rank approximation to $\mathbf{X} + \mathbf{V}$ is computed.

Achlioptas and McSherry show that sampling and quantization greatly accelerate algorithms such as orthogonal iteration and Lanczos iteration [53] that are used to compute the singular value decomposition (SVD) of a data matrix. Note that the dimensionality of the data is not reduced in sampling and quantization — dimensionality reduction can be performed afterwards using the results of SVD, if desired.

A natural question now arises: can we use similar procedures for speeding up ICA, too? The FastICA algorithm is somewhat similar to orthogonal iteration and Lanczos iteration in that the data are iteratively projected in some direction and normalized, a new direction of projection is computed, and the data are again projected and normalized. The motivation of this would be to make the whitening phase of ICA computationally simpler. The SVD is largely unaffected by sampling and/or quantization. Unfortunately, this does not imply that ICA would be unaffected by such procedures — the results of [2] only show that the second-order characteristics of the data remain intact, while in ICA the higher order characteristics are taken into account, too. In fact, in the experiments conducted by the author of this thesis (details not shown), information-theoretic measures such as the negentropy were severely affected by random sampling and quantization of the data.

In the case of random sampling of the data, there is a fundamental reason why ICA cannot be estimated. Denote by $\mathbf{x}(t)$ the t-th observation vector and by $\hat{\mathbf{x}}(t)$ its sampled version. The procedure of sampling can be written as

$$\hat{\mathbf{x}}(t) = \rho(t)\mathbf{x}(t); \quad \rho(t) = diag\{\rho_1(t), \ldots, \rho_m(t)\} \tag{2.13}$$

where $\rho(t)$ is a diagonal matrix, and its element $\rho_i(t)$ is non-zero if the i-th element of $\mathbf{x}(t)$ is sampled and zero otherwise. Now the ICA mixing model could be written as

$$\hat{\mathbf{x}}(t) = \rho(t)\mathbf{A}\mathbf{s}(t) = \hat{\mathbf{A}}(t)\mathbf{s}(t). \tag{2.14}$$

But now the new mixing matrix $\hat{\mathbf{A}}(t)$ depends on t and thus is not constant with respect to different observations, which violates the basic assumptions of ICA. Also, if $\mathbf{A}$ is square, $\hat{\mathbf{A}}$ is not invertible as the determinant of $\rho(t)$ is zero; for non-square $\mathbf{A}$, $\hat{\mathbf{A}}$ might not have full column rank either.

Quantization of the observed data is described in [2] as finding the largest absolute value b in the data matrix, and then setting each entry of the data matrix either to $+b$ or $-b$ with a probability depending on the original value of the entry. Quantization may be realized in other ways, too, and the remarks made here apply to a more general setting. Quantization is

not a linear operation; instead, quantizing corresponds to a so called post-nonlinear mixture where the source signals have been mixed linearly, but a nonlinear transformation takes place before the measurement is done. Some ICA theory has been developed for such post-nonlinear mixtures [143] with invertible nonlinearities. Quantization is not invertible, so those ICA methods cannot be applied. Also, quantization destroys the structure of the data more severely than sampling and is very sensitive to outliers; thus it does not seem to be a promising method of preprocessing in ICA.

Generally, the random perturbation $\mathbf{V}$ can be seen equivalent to adding sensor noise in ICA. Sensor noise can be tolerated by the existing ICA methods if the noise is Gaussian or if its covariance structure is known and can be restricted to a special form [72]. Here neither of these requirements is satisfied.

2.4 Other latent variable decompositions

As mentioned in Section 1.1, one of the aims of this thesis is to discuss methods for latent variable decomposition in high dimensional data. In this section, some methods other than ICA are briefly discussed. All of them can be cast in the broader framework of (linear) generative models, overviews of which have been given in [60, 112, 131, 145].

Principal component analysis (PCA), described in Section 2.3.1, is a method for latent variable decomposition in its own right, in addition to being a method for data decorrelation or whitening. One way to write the data model in PCA is $\mathbf{x} = \mathbf{A}\mathbf{y}$ where $\mathbf{y} = (y_1, \ldots, y_n)$ is Gaussian, zero mean and white, and $\mathbf{A}$ has the eigenvectors of the data covariance matrix as its columns. Probabilistic versions of PCA have been given by [28, 130, 146]. The first of these generalizes the case to other than Gaussian latent variables.

Factor analysis, originally developed in social sciences and psychology [57], tries to find relevant and meaningful factors y that explain the observed data. The data model is $\mathbf{x} = \mathbf{A}\mathbf{y} + \mathbf{n}$; the interpretations of its components are the same as in PCA except for the vector $\mathbf{n}$ whose elements, the so called specific factors, are uncorrelated with the factors y, and have a diagonal covariance matrix. The unknown matrix $\mathbf{A}$ of factor loadings can be assumed to absorb the variances of the y. The matrix $\mathbf{A}$ is solved in such a way that the observed variables x in $\mathbf{x}$ have a high loading only on a small number of factors y — reminiscent of a sparse mixing matrix in ICA [73], although solved in a different way.

Projection pursuit [47, 81, 69] tries to find directions in which the data have an interesting structure — here "interesting" often refers to non-Gaussian or otherwise structured, and the aims are data visualization and exploratory data analysis. Again, the data are linearly projected.

In *nonnegative matrix factorization* (NMF) by Lee and Seung [97, 98], an observed data matrix $\mathbf{X}$ is decomposed into the product of two unknown matrices: $\mathbf{X} = \mathbf{A}\mathbf{Y}$. All three matrices $\mathbf{X}$, $\mathbf{A}$ and $\mathbf{Y}$ have nonnegative entries. Typically the dimensionality of the observed vectors (the columns of $\mathbf{X}$) is larger than that of the columns of $\mathbf{Y}$, so NMF is yet another method of dimensionality reduction. Lee and Seung give two algorithms for finding the unknown matrices but no probabilistic interpretation of the results. Computationally, the

methods seem very demanding and there are no clear results on the quality of the solutions [98]. The problem setting of NMF was already presented by Paatero and Tapper in [115, 114]. Recently, Hoyer [68] has combined the nonnegativity constraints with sparsity constraints. Note that the assumption of nonnegativity of $\mathbf{A}$ and $\mathbf{Y}$ already imposes some kind of sparsity on the estimated matrices, as that is the only way to restrict $\mathbf{AY}$ from growing too large compared to the observed data $\mathbf{X}$. Welling and Weber [153] present a fixed point algorithm for positive tensor factorization, for tensors of arbitrary orders.

Hofmann's *probabilistic latent semantic analysis* (PLSA) [64, 65] is a strong matrix decomposition method for matrices of probabilities: $\mathbf{P} = \mathbf{AY}$. The decomposition resembles that of NMF except that all matrix entries have values between 0 and 1, and they sum to 1 at each column. PLSA is typically used in document analysis, with the aim of modeling the observed term and document frequencies by latent topics of discussion. The probability $\mathbf{P}(i, j)$ of observing term i in document j is presented as a convex combination of n *aspects* $\mathbf{A}(i, l)$, $l = 1, \ldots, n$. The terms are conditionally independent given the topic. The model has the form $p(\mathcal{W} = w|\mathcal{D} = d) = \sum_z p(\mathcal{Z} = z|\mathcal{D} = d)p(\mathcal{W} = w|\mathcal{Z} = z)$, where $\mathcal{Z}$, $\mathcal{D}$ and $\mathcal{W}$ are random variables corresponding to the topics, documents and terms, respectively. In matrix form, $\mathbf{P}(i, j)$ gives $p(\mathcal{W} = w_i|\mathcal{D} = d_j)$, $\mathbf{A}(i, l)$ gives $p(\mathcal{W} = w_i|\mathcal{Z} = z_l)$[5] and $\mathbf{Y}(l, j)$ gives $p(\mathcal{Z} = z_l|\mathcal{D} = d_j)$. One main difference to NMF is that the probabilities $\mathbf{P}$ are not observed, only the multinomial document vectors as columns of $\mathbf{X}$ (an entry of a document vector gives the number of occurrences of a term in a document). The model is solved using the expectation-maximization (EM) algorithm [34].

Latent Dirichlet allocation (LDA) and *multinomial PCA* (MPCA) as presented by [18], [102] and [20] are methods somewhat similar to Hofmann's PLSA in that they are probabilistic in nature. In particular in MPCA, an observed document vector's distribution is multinomial with a parameter vector $\mathbf{p} = \mathbf{Ay}$. Here $\mathbf{y}$, the proportions of different latent variables in this document, is first sampled from a Dirichlet distribution. The matrix $\mathbf{A}$ again gives the probabilities of terms in different latent variables. For inference and learning in the LDA/MPCA model, a variational approximation of the data likelihood is done in [18], followed by an EM algorithm for maximum likelihood parameter estimation. Ways to enhance the estimation are presented by [102] and [20]. All of these approaches are computationally quite demanding. In a recent paper, Girolami and Kabán [52] have discussed the equivalence between PLSA and LDA.

A popular method for analyzing multidimensional data is *mixture modeling* where the observed data distribution is assumed to be a convex combination of some underlying latent distributions: $p(\mathbf{x}) = \sum_{j=1}^{n} \pi_j p_j(\mathbf{x}|\theta_j)$ where π_j is the probability that a data vector $\mathbf{x}$ is generated by the jth component density p_j with parameters θ_j; it also holds $\sum_{j=1}^{n} \pi_j = 1$. All components x_i of the observed vector $\mathbf{x}$ have the same probabilities π_j of being generated by the jth underlying distribution. This is an important difference to ICA-type methods where the components x_i of $\mathbf{x}$ may arise to different degrees $\mathbf{A}(i, j)$ from different latent components s_j. In contrast to ICA, in mixture models it is also often assumed that one data vector is generated by one latent distribution, although generation probabilities are given for all latent distributions. The observed vectors can then be clustered corresponding to

[5]To be exact, $p(\mathcal{W} = w_i|\mathcal{Z} = z_l)$ must be interpreted as the probability that topic z_l generates word w_i; this is not the same as the probability of observing w_i when z_l is active, as other topics than z_l may contribute to observing w_i, too.

these latent distributions. Originally, the latent distributions were often assumed univariate Gaussian. In more recent papers, the observed data distribution is seen as a mixture of PCA's [62], a mixture of probabilistic PCA's [145], or a mixture of factor analyzers [49], to name a few; all of these are locally linear decompositions.

Local PCA [48] can be used in dimensionality reduction as discussed in [87]: the data space is first partitioned into disjoint regions, and PCA is then performed separately in each region. This approach is closely related to mixtures of PCA's.

Chapter 3

ICA for complex valued signals

3.1 Introduction

The first theoretical development of ICA in this thesis is the separation of linearly mixed complex valued signals as presented in Publication 1. Here the problem is reviewed briefly and some new insights are given. The reader is referred to Publication 1 for more discussions and derivations.

A complex random variable z can be written as the sum of its real and imaginary parts, $z = u + iv$ where u and v are real random variables. We denote by $\mathrm{Re}(z)$ the real part u of z and by $\mathrm{Im}(z)$ the imaginary part v of z. Alternatively, a complex random variable can be presented in polar coordinates as $z = \rho e^{i\phi}$ where ρ is the modulus (also called radius) and ϕ is the phase of the variable.

Complex valued signals are often encountered in, e.g., the fields of telecommunications or audio separation where convolutive (that is, time-lagged) signals are mixed: the sources are located so far away from the measurement locations that the source signals arrive at different instances in time, with possible echoes from nearby walls and so on. Moving into the frequency domain changes the convolution into multiplication, and an ICA-type mixing is obtained, where the mixtures, the sources and the mixing matrix are complex valued. A common practice is to divide the frequency domain into bins; this helps for example in noise cancellation, if colored noise is observed. Then a complex source separation task is solved in each bin. In the following section, an algorithm for the separation of complex valued signals is given.

3.2 A fast fixed-point algorithm

We assume that the ICA model $\mathbf{x} = \mathbf{A}\mathbf{s}$ holds and both the independent component variables or source signals $\mathbf{s}$ and the observed variables $\mathbf{x}$ are complex valued. The mixing matrix $\mathbf{A}$

may be complex valued if desired. The source signals s_j are assumed to have zero mean and unit variance, with uncorrelated real and imaginary parts of equal variances. (The last assumption implies that s_j must be strictly complex; the imaginary part may not vanish everywhere.)

In the case of real valued signals, the independent components are typically found up to permutation and scaling. In the complex case, these indeterminacies hold as well, and in particular the scaling is complex valued. In other words, there is an unknown phase for each s_j. This indeterminacy is an inherent property of complex ICA and not a consequence of the assumptions made in our approach.

It will be assumed that s_j has a spherically symmetric distribution — thus the distribution of s_j depends on the modulus of s_j only and the scaling by a constant complex value does not change the distribution of s_j. This assumption simplifies our approach and is quite realistic in many applications, and it is also in line with the indeterminacy mentioned above.

In Publication 1 a fast fixed point algorithm for the separation of complex valued signals is given. It is somewhat similar in nature to the FastICA algorithm [76, 70] briefly discussed in Section 2.2; hence the algorithm of Publication 1 is sometimes called the *complex FastICA* algorithm. The fixed-point algorithm for estimating one component $y = \mathbf{w}^H\mathbf{x}$ is

$$\mathbf{w}^+ = E\{\mathbf{x}(\mathbf{w}^H\mathbf{x})^* g(|\mathbf{w}^H\mathbf{x}|^2)\} - E\{g(|\mathbf{w}^H\mathbf{x}|^2) + |\mathbf{w}^H\mathbf{x}|^2 g'(|\mathbf{w}^H\mathbf{x}|^2)\}\mathbf{w} \tag{3.1}$$

$$\mathbf{w}_{new} = \frac{\mathbf{w}^+}{||\mathbf{w}^+||} \tag{3.2}$$

where the asterisk denotes complex conjugation, and $\mathbf{w}^H$ is the vector $\mathbf{w}$ transposed and complex conjugated. The choice of the nonlinear function g is discussed in Publication 1. To estimate several components, the outputs $\mathbf{w}_j^H\mathbf{x}$ are decorrelated before the normalization (3.2) similarly to what was discussed in the end of Section 2.2. For details, please refer to Publication 1.

In Publication 1 we also give the conditions under which the estimator given by the algorithm is consistent.[1] We start from an arbitrary nonlinear smooth contrast function and prove that its extrema coincide with the independent components. The nonlinear contrast function can be chosen quite freely to optimize, e.g., the statistical behavior of the estimator. This approach is computationally simple in contrast to another approach, where independence is measured by mutual information, approximated by cumulants. As discussed in Section 2.2, it is advisable to avoid cumulant nonlinearities as they are not robust against outliers in the data.

One practical implication of the consistency of the estimator is that the signs of the values of the contrast function for true independent components need not be known — in some ICA algorithms, the sign of the kurtosis (or some other function of the true sources) must be known.

Experimental results are given in Publication 1 to illustrate the performance of the fixed point algorithm and the theorem on the consistency of the estimator. Also, the connection

[1] In the theorem on page 4 of the paper we assume that $G : \mathbb{R}^+ \cup \{0\} \to \mathbb{R}$ is a sufficiently smooth even function. To be exact, there is a misprint here: the parity of G is undetermined as $G(y)$ only exists for $y \in \mathbb{R}^+ \cup \{0\}$, and thus G cannot be even.

to independent subspace methods [75] and multidimensional ICA [23] is discussed: complex ICA is a restricted form of independent subspace methods.

Apart from the experimental results given in Publication 1, Fiori and Burrascano [45] compare the algorithm of Publication 1 with JADE [24] in electromagnetic source localization. No significant differences between the algorithms were found with respect to separation accuracy or computational complexity. Ristaniemi and Joutsensalo [128] use the algorithm of Publication 1 in separating the codes of different users in a CDMA (code-division multiple access) wireless communication network.

Earlier it was mentioned that the frequency domain is divided into bins and a complex ICA problem is solved in each bin. Due to the indeterminacy of the ordering of the estimated ICA components (similarly to the case of real valued signals) a permutation problem now arises: the order of the estimated sources should be the same in each bin. This problem is tackled in some of the references listed in Section 3.4.

3.3 Random projection of complex signals

We next describe a small experiment on using random projection prior to ICA on high dimensional complex valued signals. The source signals are artificially generated complex random signals $s_j = \rho_j e^{i\phi_j}$ where for each signal j the modulus ρ_j is drawn from a different distribution (Exponential, Gamma, Poisson, Hypergeometric, Beta, Uniform, Weibull or Geometric) and the phase ϕ_j is uniformly distributed on $[-\pi, \pi]$. The uniform phase ensures that the distribution of s_j is spherically symmetric as discussed in Section 3.2. The sources have unit variance. Examples of such source distributions are seen in Figure 3.1. The number of sources is 8, each having 50 000 observations. The sources are randomly mixed using a (100×8)-dimensional complex valued mixing matrix.

The data described above are either random projected using a 10×100-dimensional complex random matrix and then PCA preprocessed to 8 dimensions, or directly PCA preprocessed to 8 dimensions. The algorithm described in the previous section is then used to separate the sources, with a nonlinearity $g(y) = 1/(2\sqrt{\varepsilon + y})$ where $\varepsilon = 0.1$.

Similarly to the experiment in Section 2.3.2, we study the sums of squared errors (SSE) between the product of the mixing and unmixing matrices and a permutation matrix. Here the product matrix is transformed into the real domain by taking element-wise absolute values (remember from Section 3.2 that the sources are only estimated up to scaling by a complex unit-norm constant, so in the case of perfect separation, we get a permutation matrix with one unit-norm element in each row and each column). Figure 3.2 shows the average convergence of ICA estimation in the cases of random projected and original data, over 20 trials. We can see that both cases converge quickly and the SSE's are almost equal.[2] Thus at least in this small experiment, random projecting the high dimensional data prior to PCA preprocessing does not distort the data. Computing the random projection, PCA and ICA takes 77.3 seconds of CPU time on the average; directly computing PCA and then ICA

[2]The 95 per cent confidence intervals over 20 trials are also plotted, although it is questionable whether the SSE's are sufficiently Gaussian to permit the computation of the confidence intervals.

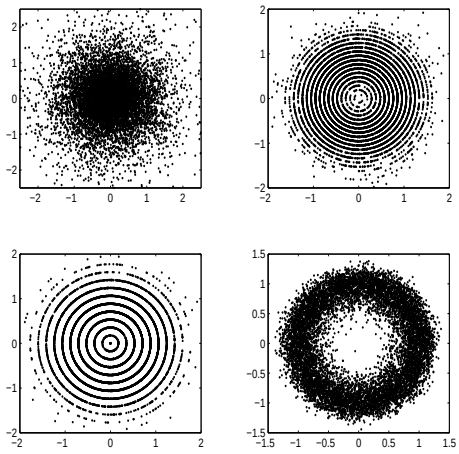

Figure 3.1: Examples of complex valued source signals with different modulus densities: Exponential (upper left), Poisson (upper right), Hypergeometric (lower left), Weibull (lower right). The source signals are spherically symmetric and have unit variance. The plane is the complex plane, the horizontal coordinate giving the real part and the vertical coordinate giving the imaginary part of a complex number.

takes 90.3 seconds of CPU time on the average[3]. This shows that the preprocessing of data by random projection again gives computational savings. Also, the theorem on the local consistency of the estimator, discussed in Publication 1, is still applicable to the random projected data.

3.4 Other approaches

The separation of complex signals is already discussed in Comon's seminal paper [29] from a cumulant point of view. The kurtosis of the estimated components is taken as a contrast function. In the complex case, kurtosis is not uniquely defined and its choice is discussed in [29]. The algorithm presented there is computationally quite demanding. A simpler algorithm is the cumulant-based JADE [25], also applicable to the complex case. Moreau and Macchi [105] give a cumulant-based algorithm which is also computationally heavy.

A somewhat different but still cumulant-based approach is Back and Tsoi's complex recurrent network [8] that is analogous to Jutten and Herault's algorithm [84]. Back and Tsoi's algorithm is computationally somewhat demanding. The algorithm works partly in the time domain and partly in the frequency domain and they claim that the permutation problem between different frequency bins is thus overcome.

Comon and Moreau [31] give a cumulant-based algorithm for finding a sequence of Givens

[3]Again, although computing the 95 per cent confidence intervals is a bit questionable, the interval is $[76.6, 77.9]$ for the case of random projection and $[89.7, 90.9]$ for the case without random projection.

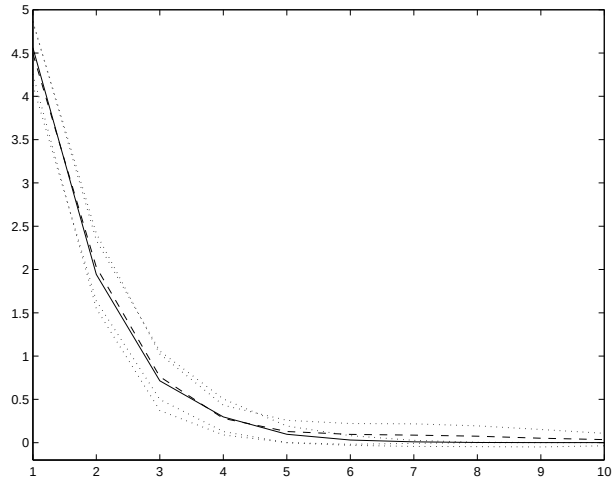

Figure 3.2: Convergence of ICA estimation of complex valued signals. Horizontal axis: number of fixed point iterations. Vertical axis: Sum of squared errors of the estimated mixing matrix. Observation data are random projected prior to PCA preprocessing (solid line) or directly PCA preprocessed (dashed line). Dotted lines give 95 per cent confidence intervals over 20 trials.

complex rotations for 2-dimensional observations and an arbitrary number (≥ 2) of sources. Givens rotations [53] are used in some ICA algorithms in the real-valued case, too: The observed data are first whitened and dimensionality reduced so that only an orthogonal square mixing matrix is left to find. Any orthogonal $m \times m$ matrix can be written as a product of $m(m-1)/2$ Givens rotation matrices and a diagonal matrix with diagonal elements ± 1. A Givens rotation is a plane rotation around the origin. The technique is useful in ICA in the two dimensional case but in higher dimensions several Givens rotations must be performed for each pair of components.

Smaragdis [140] presents a Bell-Sejnowski [13] type algorithm that is directly applicable to complex signals if transposes of vectors are simply changed to hermitians. An appropriate nonlinear function must be chosen: $g(x) = \tanh(x)$ used in the real case is unbounded in the complex domain, so he uses $g(z) = \tanh \mathrm{Re}(z) + i \tanh \mathrm{Im}(z)$. He proposes a heuristic coupling of adjacent frequency bins to make sure that the order of the estimated sources is the same in every frequency bin. The coupling approach is not always very effective, however.

In the fields of speech and radar signal processing, a popular and robust approach to solving the permutation problem is direction of arrival estimation (see, e.g., [149, 150]). It is assumed that the spatial locations of sources with respect to the locations of measurement do not change. Each frequency band must have the same direction of arrival for a chosen source signal; this gives the correct ordering of the sources within frequency bands. Names such as beampattern analysis or null beamforming also refer to this technique.

Since the appearance of Publication 1, new approaches to complex signal separation have been discussed in the literature. These are briefly reviewed in the following.

Clarke [27] avoids the use of cumulants by giving conditions of independence between two complex-valued signals. The problem in his approach is that only two signals are studied at a time. Fiori [44] presents a generalized Hebbian learning theory for a complex-weighted linear feedforward network and applies it to complex ICA. Zarzoso and Nandi [155, 156] define so called bicomplex numbers, by which they get an algorithm analogous to real ICA for finding 2-dimensional Givens rotations. This approach is applicable to the case of two source signals and two observed mixtures. In [157] Zarzoso and Nandi give closed-form estimators using bicomplex numbers.

Mitianoudis and Davies [104] tackle the permutation problem between different frequency bins by adding a time-dependent term that imposes frequency coupling between the bins. They study two fixed-point algorithms, a natural gradient type algorithm [76] and the complex FastICA algorithm of Publication 1. The time-dependent term $\beta(t)$ is integrated in the activation function $g(y)$ as $g_{new}(y) = 1/\beta(t) \cdot g(y)$ in both algorithms. They perform different experiments and conclude that the complex FastICA algorithm is slightly better: faster, more robust and more accurate.

Ristaniemi and Joutsensalo [128] prove that the convergence of the complex FastICA algorithm is cubic when kurtosis is chosen as a contrast function. The proof is analogous to the one for real valued signals, given by Hyvärinen and Oja in [76].

Recently, the separation of complex valued signals and the permutation correction are also discussed in, e.g., [7, 11, 21, 54, 80, 89, 91, 106, 108, 123, 127, 135].

Chapter 4

ICA in regression

4.1 The regression problem in the ICA framework

In this chapter we turn our attention to using ICA as a preprocessing method in nonlinear regression. The problem is treated in Publication 3 of this thesis. The contribution of the author of this thesis is smaller in Publication 3 than in the other Publications, and thus the problem is reviewed only briefly.

In a regression problem, one has a set of predictor variables and a set of predicted variables. The task is to generate a mapping between these sets so that given the values of the predictor variables, the values of the predicted variables can be estimated.

The regression problem can be cast into the ICA framework as follows. The set of mixtures $\mathbf{x}$ is divided into two, predictor ($\mathbf{x}_o$) and predicted ($\mathbf{x}_m$) variables[1]. Using a training set of both $\mathbf{x}_o$ and $\mathbf{x}_m$, we estimate the ICA model as

$$\begin{pmatrix} \mathbf{x}_o \\ \mathbf{x}_m \end{pmatrix} = \begin{pmatrix} \mathbf{A}_o \\ \mathbf{A}_m \end{pmatrix} \mathbf{s} \tag{4.1}$$

which gives us the joint probability density of $\mathbf{x}_o$ and $\mathbf{x}_m$ as densities of sums of independent random variables $\mathbf{s}$. With this joint density, we can estimate the expected value of the predicted variables $\mathbf{x}_m$ given the predictor variables $\mathbf{x}_o$. Using the general rules of densities of transforms (see, e.g., [72], pages 20 and 36), we can write

$$E\{\mathbf{x}_m|\mathbf{x}_o\} = \mathbf{A}_m \int_{\mathbf{A}_o\mathbf{s}=\mathbf{x}_o} \mathbf{s}p(\mathbf{s})d\mathbf{s} \tag{4.2}$$

where $p(\mathbf{s})$ is the joint density of the independent components. In Publication 3 an approximation for the integral formula (4.2) is given. First, the data are linearly preprocessed and the $\mathbf{x}_m$ are replaced by the residuals of linear regression. Then the approximation reads

$$E\{\mathbf{x}_m|\mathbf{x}_o\} \approx \mathbf{A}_m \mathbf{g}(\mathbf{A}_o^T \mathbf{x}_o) \tag{4.3}$$

[1] In Publication 3, the predictor variables were called "observed" and the predicted variables were called "missing"; hence the subscripts o and m.

where $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^n$ is a multidimensional function that consists of applying a (possibly) different function g_i on each component of its argument: $g_i(u) = p_i'(u)/p_i(u) + cu$. We denote by p_i the density of the i-th independent component, and by c a constant scaling term. The arguments of g are initial linear estimates of the independent components s as $\mathbf{A}$ is orthogonal due to prewhitening and $\mathbf{A}_o^T$ is equal to the pseudoinverse of $\mathbf{A}_o$. The approximation is derived for independent components whose distribution is not far from a Gaussian distribution.

The approximative formula (4.3) shows that ICA regression can be seen as regression by a multilayer perceptron (MLP) network with one hidden layer. The activation functions of the hidden layer are g_i, the number of hidden units equals the number of estimated components, the weights between the input and the hidden layer are given by $\mathbf{A}_o$ and the weights between the hidden and the output layer are given by $\mathbf{A}_m$.

ICA can now be seen as a preprocessing method in regression: instead of forming the regression directly on $\mathbf{x}_o$, we compute estimates of the sources $\mathbf{s}$ using $\mathbf{A}^T\mathbf{x}_o$ and form a nonlinear regression on them, as seen in Formula (4.3). The nonlinearly mapped sources are transformed back to $\mathbf{x}_m$ by multiplication with $\mathbf{A}_m$.

In Publication 3, experimental results on three different source densities — strongly super-Gaussian, Laplace distributed (somewhat super-Gaussian) and Cosh distributed (very weakly super-Gaussian) sources — are given for validating the approach. It is shown that the approximative formula (4.3) nicely matches the exact integral formula (4.2) in all cases; the better the less super-Gaussian the sources are. On the other hand, the very principle of ICA regression seems to be plausible only if the sources are very super-Gaussian; this is natural as ICA regression is nonlinear and assumes that all linear dependencies are first removed from the data. For Gaussian sources, linear regression captures all dependencies there are, and there is nothing left to be explained by the nonlinear structure of ICA regression.

In contrast to other ICA settings studied in this thesis, the use of random projection as a preprocessing method is not discussed here. In a regression problem, the number of predictor variables is often moderate, and dimensionality reduction is not needed.

4.2 Related methods

Publication 3 compares ICA regression with multilayer perceptrons and also discusses its relation to projection pursuit regression and wavelet shrinkage. Some other related approaches are described here.

Density shaping by Roth and Baram [129] was one of the first ICA-type approaches to regression, although the concepts of ICA or BSS were not mentioned in the paper. They give a neural network that performs a similar task as the Infomax [13] in maximizing the entropy of the network's output. A linear conditional expectation estimator $x_m = E\{x_m|\mathbf{x}_o\}$ is given for a one-dimensional predicted variable.

Cascade correlation by Fahlman and Lebiere [41] is a feedforward neural network architecture that can also be used for regression. The network is built incrementally, adding new hidden

units one by one until the performance of the network is satisfactory. In the beginning, the network has no hidden units and the weights of the connections between the input and the linear output layer are estimated — this corresponds to finding the linear dependencies in the data, as is first done in ICA regression. Then a nonlinear hidden unit is included in the model to explain what was left to be explained after the linear regression, and the incoming weights of the hidden unit are optimized. After this, all connections to the output layer are trained (including those directly from the input layer.) If there still remains some residual between the output of the network and the training data, another hidden unit is added, with incoming connections both from the input layer and the first hidden unit. The process of adding hidden units, training its incoming weights and training all weights connected to the output layer is repeated until the residual error in the training set vanishes.

The name "cascade" stems from the architecture of the network: new hidden units are connected both to the input layer and all preceding hidden units. The architecture is thus somewhat different from the MLP that corresponds to our ICA regression, where only one hidden layer with several units is used. Also, the training in cascade correlation is different in that after including new hidden units, the outgoing weights of the previous units are also updated. This is reminiscent of stepwise linear regression where the predictor variables are not totally independent and thus including new predictor variables (which are nonlinear functions of the previous ones) changes the explanatory power of the previously added predictor values, too. In our ICA regression the source estimates $\mathbf{A}_o^T \mathbf{x}_o$ in (4.3) are independent and decorrelated so that they explain orthogonal aspects of the observed data. There is no clearly defined way to choose the nonlinearities in the cascade correlation network, in contrast to ICA regression where the nonlinearities are directly obtained from the densities of the latent sources.

Back and Weigend [9] estimate the ICA components of stock returns and reconstruct the observed data either as linear combinations of the independent components or as sums of thresholded independent components. Reconstruction is shown to be better than with PCA, which is a well established tool in finance.

Eltoft and Kristiansen [38] use ICA and regression for filling in gaps in time series. Nonlinear predictions are computed in the independent component domain, and prediction errors in the observation domain.

Chapter 5

ICA in text mining

5.1 Introduction

In times of huge information flow, there is a strong need for automatic textual data analysis tools. Methods developed for this task fall in the broader category of *statistical natural language processing* (SNLP). This includes all quantitative approaches to automated language processing, such as probabilistic modeling, information theory and linear algebra [99]. In this thesis, the focus is on text document data presented in a matrix format (discussed in more detail in the sequel), and various linguistic aspects of SNLP remain untouched.

Another umbrella under which textual data analysis partially lies is *information retrieval* (IR). Quoting a textbook on IR [10], "Information retrieval deals with the representation, storage, organization of, and access to information items." Textual documents are but one source of information; others include images, music and so on. In the IR research, the emphasis is typically on finding the information relevant to a user's need. Examples of the information need might be "Find documents containing information on fitness boxing and places for that in Helsinki" or "What is the total length of illuminated ski tracks at the Saariselkä ski resort?". This information need is translated into a query that can be processed by an IR system such as a search engine [10]. In addition to [10], introductions into IR include [148, 133].

The third framework of textual data analysis is data mining; this point of view is discussed in, e.g., [3, 56]. The name "text mining" has been chosen for this chapter as aspects of data mining are touched elsewhere in this thesis, too: the general aim of finding the underlying structure in data is common to both data mining and statistical data analysis.

The approach to textual data analysis taken in this thesis fits into the modeling phase of SNLP. The results of the methods presented in this thesis help in seeing the underlying structure of a large text corpus. For example, analyzing the topics of the documents aids in topic-based document retrieval; this will be discussed in Sections 5.2 and 5.3.

In Publication 4 of this thesis we describe a way of using independent component analysis in

text mining. ICA was originally developed for signal processing purposes, in particular for continuously distributed signals. Text documents are a very different application area. However, in statistical natural language processing it has been observed that if text documents are presented in a numerical format, then many numerical and computational methods can be used to analyze the textual data.

The vector space model [133], also called bag-of-words model, is a popular format for presenting text documents. In this model, each document forms one d-dimensional vector where d is the number of distinct terms in the vocabulary. Forming the vocabulary is a preprocessing step which is discussed in, e.g., [10] and [99]; in our work, it is realized using the Bow toolkit [100]. The main task is to limit the size of the vocabulary by choosing only a subset of all terms appearing in the documents. This involves giving weights to terms to reflect their importance. The i-th element of the document vector indicates the frequency (or some function of the frequency) of the i-th vocabulary term in the document. The document vectors are collected as the columns of the data matrix, also called term by document matrix.

In the vector space model, the documents are treated as points in a high dimensional space. As a tradeoff for the computational simplicity of the representation, all information about the order of the words inside the document or the structure of the document is lost. On the other hand, using the vector space model makes it possible to see text document data in the same framework as other high dimensional data sets encountered in data mining applications, e.g., customer transaction data or web log data. The observations might be documents, customers or web users; the observed variables are then terms, products bought or web pages visited, respectively.

A common practice in text mining is to compute the singular value decomposition (SVD) of the data matrix and project the data into the subspace spanned by the left singular vectors corresponding to the largest singular values. Thus the observed documents are represented as linear combinations of some orthogonal features, called latent semantic factors. This method is known as latent semantic analysis (LSA) in text mining, first discussed in [33]. LSA is said to tackle the problem of synonymy and partially also polysemy[1], and take advantage of the implicit structure in how terms are associated with documents [33].

LSA uses only second-order moments of the data, so a natural step forward is to apply more powerful methods such as independent component analysis. First approaches to using ICA in the context of text data were presented by Isbell and Viola [77], Kolenda et al. [96] and Kabán and Girolami [85]. In these approaches, a text document is seen as an instantaneous mixture of independently occurring latent *topics*. In the text mining parlance, a topic is a probability distribution on the universe of terms. The estimated ICA mixing matrix of text data reveals to which degree the terms belong to different topics, and the estimated sources show which topics are active in each document, as shown in Figure 5.1. (Alternatively, one may also analyze the transpose of the data matrix, in which case the mixing matrix reveals which documents are good examples of each topic, and the sources show which terms represent each topic the best. In Publication 4 we have analyzed both the original term by document matrix and its transpose, the document by term matrix. The estimated sources then reveal the association of terms with topics and the association of

[1]Synonymous words have the same meaning, such as "car" and "automobile". A polysemous word has several different meanings, such as "branch".

documents with topics, respectively.) In practice, the high dimensional term by document matrix is often first preprocessed by SVD or PCA, and the dimensionalities of the matrices change; this is discussed in more detail in Section 2 of Publication 4.

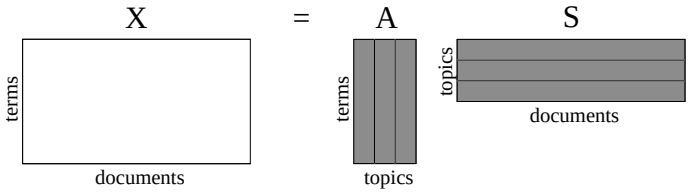

Figure 5.1: The observed matrix **X** has terms as its rows and documents as its columns. A column of the mixing matrix **A** tells the degrees of activity of each term in a topic, and a row of the source matrix **S** tells the degree of activity of a topic in each document.

In the rest of this chapter, we discuss the analysis of a dynamically evolving text stream in Section 5.2. In Section 5.3 we compare ICA and the self-organizing map (SOM) in document clustering and discuss the use of random projection as a preprocessing step.

5.2 Analysis of dynamically evolving text

In Publication 4 we have extended the use of ICA in text mining by analyzing dynamically evolving text. In basic ICA and in the above references, the ordering of the observed data vectors is not taken into account in estimating the model; in the approach of Publication 4, the observed data are instead seen as time series. Other latent variable methods for analyzing time-varying text data include Kolenda and Hansen's [94, 95] second-order approach, Kabán and Girolami's [86] hidden Markov model type algorithm and Slaney and Ponceleon's [138] LSA-based segmentation algorithm. Outside latent variable methods, there is a rich literature under the name "topic detection and tracking" for finding topically related material in streams of data. See, e.g., [151, 5, 4].

The dynamically evolving text data that are analyzed in Publication 4 are chat line data. The discussion found in chat lines on the Internet is an ongoing stream of text generated by the chat participants and the chat line moderator. Typically, several different discussions are going on simultaneously. Assuming that the discussions are more or less independent of each other, we can view the situation as an ICA mixing, and use an ICA-type algorithm to extract the different topics of the discussions.

The algorithm used in Publication 4 is based on Hyvärinen's complexity pursuit [71] algorithm. The idea in complexity pursuit is to find interesting structure in multidimensional time series data. Interestingness is measured as a low Kolmogorov coding complexity of a projection of the data. Intuitively stated, projections with a short coding length are typically structured in some way, that is, they are far from random noise and/or Gaussianity. Connections between Kolmogorov complexity and ICA-type methods are discussed in [117]. The complexity pursuit method is quite similar to ICA except that it exploits the temporal

dependencies in the data in addition to higher order statistics. No special emphasis is put on the statistical independence of the estimated latent sources, though.

The details of the algorithm are discussed in Publication 4. Let us briefly describe its main characteristics here. The data are first whitened by PCA. At each iteration step, a first order autoregressive (AR) model is estimated for each latent variable s_j. Then an approximation of the Kolmogorov complexity of the residual of the AR model is minimized by gradient descent. The estimated projection directions $\mathbf{w}_j$ are decorrelated after every iteration step, similarly to the way described in Section 2.2.

For the experiments of Publication 4, chat line data from the CNN Newsroom chat line[2] were collected. The text stream of almost 24 hours was split into short sections, each of which was considered as a document, and the term by document matrix of 5000 terms and 7430 documents was formed as discussed in Section 5.1. We estimated 10 latent topics of discussion, shown in Publication 4. The estimated topics are very easy to interpret as they concentrate on different terms — for a human observer the lists of the most important terms of each topic are very meaningful. Also, looking at the estimated topics in the time domain, we see that different topics behave differently over time; they all have their own periods of activity, which seems very natural considering the problem setting. Participants of the chat line discussion come and go, and so do the topics they discuss.

To compare the performance of the complexity pursuit algorithm to that of other ICA-type or time series methods, we also analyzed newsgroup data which are labeled in the sense that each article belongs to one newsgroup whose identity is known. The data were from the 20 Newsgroup corpus[3] and consecutive articles were split into overlapping sections to emphasize the time-dependent nature of the data. We measured how well the estimated topic time series can be clustered into clusters corresponding to different newsgroups. We compared the complexity pursuit algorithm to ordinary FastICA [70], JADE$_{TD}$[4] [107], Kolenda's delayed decorrelation [94] and Stone's temporal predictability maximization[5] [142] and showed that complexity pursuit yields the smallest classification error.

Our results suggest that the method could serve in queries on temporally changing text streams, perhaps complementing other topic segmentation and tracking methods [5]. An important example of temporally changing text streams is online news services.

5.3 Preprocessing by random projection

Regarding the very high dimensionality of the document data, let us again discuss the use of random projection as a preprocessing step. Computing the PCA or SVD of the term by document matrix is a common practice prior to ICA estimation, similarly to what is done in many other application areas. As random projection does not severely distort the distances

[2] http://www.cnn.com/chat/channel/cnn_newsroom

[3] http://www.cs.cmu.edu/~textlearning

[4] The code was kindly provided by Mr Jukka Matilainen, who compiled Cardoso's JADE (from http://sig.enst.fr/~cardoso/icacentral/Algos/cardoso/JnS.tar) and Ziehe's TDSEP (from http://www.first.gmd.de/~ziehe/download.html) codes.

[5] The code was kindly provided by Dr J.V. Stone.

between data points, we may use it to reduce the dimensionality of the data and then compute the PCA or SVD in the lower dimensional space. The original term by document matrix typically has thousands of terms and is very sparse; reducing its dimensionality to a few hundred gives computational savings even though the sparsity of the data is then lost.

Random projection has been used successfully in another text mining approach, namely the WEBSOM [93] method that is based on the self-organizing map (SOM) by Kohonen [92]. In this approach, documents are ordered on a two dimensional map display in an unsupervised manner so that similar documents lie close to each other. A sparse binary random matrix is used in the WEBSOM system to reduce the dimensionality of the document vectors prior to forming the map. Experimental results on random projection with the WEBSOM system are given in [88, 93]. Note that PCA or SVD is usually not computed when forming the map, so the context of random mapping is different from what we have discussed so far in this thesis — the SOM is not a method for finding projection directions of the data matrix like PCA, SVD, ICA, factor analysis etc. are.

In [16] we have shown experimental results of using random projection prior to ICA and SOM. The documents analyzed in the project were segments of spoken dialogues carried out over the telephone in a customer service, transcribed into text. The topics of the discussion segments were analyzed and the segments were clustered. The quality of the clustering was assessed by comparing it to a manual labeling of the documents (that is, segments). ICA is not primarily intended for clustering but instead for presenting each observation vector as a combination of latent variables. Here one document typically is about one topic only, and thus one latent variable dominates in it, and we can cluster the document into this latent variable. SOM, on the other hand, arranges the documents onto a two dimensional plane in which more or less clear clusters can be seen.

In our experiments [16], we computed the ICA on the original data and both the ICA and the SOM on the random projected data, and compared the clustering accuracies. The nonlinearity in the FastICA algorithm was the "skewness" nonlinearity $g(u) = u^2$ to reflect the skewed distribution of the estimated latent components: the activities of the topics in documents are nonnegative and mostly zero, with only one (or a couple of) active topic(s) in one document. Random projection gave computational savings in ICA with a slight decrease in the quality of the clustering. The overall performance of SOM was a bit better than that of ICA. The most visible difference between their performances was seen in documents belonging to a small cluster containing documents that were manually assigned as "out of domain" (that is, impossible to classify). ICA could not find the correct clustering for these, probably mainly because these documents did not form a statistically meaningful and coherent entity, and also because the number of these documents was small. Apart from these documents that topically do not truly belong together although they were manually labeled so, ICA performed as well as SOM or even better. So in addition to studying random projection in the context of ICA of text documents, in [16] we have also given evidence that the performances of ICA and SOM in document clustering are comparable.

Chapter 6

Finding structure in binary data

6.1 Introduction

Let us consider the problem of finding a simple representation for a large data set that takes values in $\{0, 1\}$. (When referring to data that takes values in $\{0, 1\}$, we will talk about "binary data" or "0-1" data interchangeably.) Our general hypothesis is the same as before, namely that the observations are generated by some unknown latent components and their interactions. Specifically, we assume that an observed data vector is a realization of a few co-occurring *topics*. Intuitively, the topics are collections of variables whose occurrences are somehow connected to each other. In Chapter 5 we noted that in the text mining parlance, a topic is a probability distribution on the universe of terms — in this chapter we maintain this definition with some possible restrictions on the form of the distribution. For the ease of notation, we may also characterize a topic by listing the variables on which its distribution concentrates.

We assume that the topics cover more or less different aspects of the data set and that their occurrences or activities are independent of each other. Using the basic ICA model notation, the topics correspond to the columns of the mixing matrix $\mathbf{A}$, and the topic activities are given by the rows of the source matrix $\mathbf{S}$. The linear model $\mathbf{X} = \mathbf{AS}$ does not exactly hold but instead we discuss a few other ways of writing the data model.

As an example, consider market basket data where for each customer (an observed data vector) we list which products she/he buys among all products available in the store. The actual number of items bought is not modeled, but only the occurrence or non-occurrence of each product in the customer's basket. Market basket data are usually sparse, as one customer only buys a small subset of a large set of alternative products. We assume that the data consist of a small number of independent product groups: a customer typically buys products belonging to one or a few groups (e.g., baby-care products and dairy products). These unknown product groups are the latent topics of interest. Similarly, consider a large collection of text documents represented as a binary term by document matrix. The data often contain several distinct topics, one particular document dealing with only one or a

couple of them. Here, a topic is characterized by a subset of terms. A third example of binary data are web log data. A typical user of a large web site might visit pages concerning only one or a few specific topics within a broad choice of pages. Again, it is of interest to find these latent groups of web pages, given only the sets of pages that each user visited. Of course, such a representation omits important temporal relationships between the page accesses.

We assume that the observations are sparse, that is, there are a lot more 0s in the data than 1s. We also emphasize that the roles of 1 and 0 in the data are very different and not interchangeable: observing a 1 in a data vector means that there must have been an occurrence of at least one topic that generates the 1. Observing a 0 results either if no topic generates the 1, or simply if none of such topics occur in the observed data vector.

It is also natural to assume that the noise present in the observations is binary. Most ICA approaches assume Gaussian noise which has convenient properties. The case of binary noise is more problematic.

The problem of decomposing 0-1 data can be tackled by several different approaches depending on what assumptions we are willing to make about the latent structure of the data. Given an observation data matrix whose values are in $\{0, 1\}$, basic linear ICA would give matrices $\mathbf{A}$ and $\mathbf{S}$ whose entries are real valued. In this chapter we discuss cases in which we wish to restrict the values of $\mathbf{A}$ or $\mathbf{S}$ or both somehow. We first consider the case when the latent topics or their occurrences or both are binary. In Section 6.3 we assume that both are probabilities, that is, values in the range $[0, 1]$.

6.2 Binary sources and/or binary mixing

6.2.1 Problem setting and related methods

Let us first restrict both the latent topics (in matrix $\mathbf{A}$) and topic occurrences (in matrix $\mathbf{S}$) to binary values. Given a large data set of binary observations, our task is to find a reasonably small number of binary latent topics such that the observations can be reconstructed by simple "OR" operations or unions between a few topics. The data model in this "truly binary" approach is not the matrix product $\mathbf{X} = \mathbf{AS}$ but

$$\mathbf{X}(i,t) = \bigvee_j (\mathbf{A}(i,j) \wedge \mathbf{S}(j,t)) \vee \mathbf{B}(i,t) \tag{6.1}$$

where the noise in matrix $\mathbf{B}$ is binary, unless omitted.

To better understand this approach, consider again the examples of binary data listed at the beginning of this chapter. Restricting our attention to binary topics means in the case of document data that the topic of a document is characterized by a subset of all terms in the vocabulary, and the terms in this subset do not have a particular order of importance. Similarly, the supermarket products or web pages forming the latent topics in market basket data or web log data, respectively, are all equally important. Also, by saying that the occurrences of topics are binary we mean that for one observation vector, a subset of topics

is active and the rest are inactive, and the degree of activity is equal for all active topics.

This truly binary approach has several drawbacks. First is the above mentioned restriction on the nature of the variables or the topic activity. Second is the deterministic nature of the data model. If a topic is active, then all variables belonging to the topic will be observed. This problem is shared by the basic noiseless linear ICA, too, but here the problem is more pronounced due to the binary nature of $\mathbf{A}$ and $\mathbf{S}$. In Section 6.3 we will give a probabilistic interpretation for our binary data.

One can even argue that there is a fundamental reason why (linear) ICA cannot solve binary mixtures of binary sources: the intuition based on the central limit theorem discussed in Section 2.2 does not hold in this case. A disjunction of two binary vectors is still a binary vector, and Gaussianity does not increase as sources are mixed, as the mixing is not a linear combination. Thus ICA methods based on non-Gaussianity maximization cannot be used, and information-theoretic measures should not be approximated by measures of non-Gaussianity.

In the framework of basic linear ICA, the problem of binary topics and binary topic occurrences can be solved approximately if we assume that both $\mathbf{A}$ and $\mathbf{S}$ are binary and sparse. For such data, the "OR" operation is practically equivalent to a linear combination. A few resulting entries of $\mathbf{X} = \mathbf{AS}$ typically yield values larger than 1 but they can be thresholded later to 1. Also, one may regard the data being generated by the model $\mathbf{X} = f(\mathbf{AS})$ where f is a unit step function that operates on each element of the matrix $\mathbf{AS}$ individually: $f(u) = 1$ for $u \geq 0$, and $f = 0$ otherwise. This is a post-nonlinear mixture as discussed by, e.g., [143]. The nonlinear function f is not continuously differentiable, and ICA methods developed for post-nonlinear mixtures cannot be used. (In the post-nonlinear setting, one might as well assume $\mathbf{A}$ and $\mathbf{S}$ not binary but nonnegative in general, and the model $\mathbf{X} = f(\mathbf{AS})$ would give exactly the same observations.)

Himberg and Hyvärinen [59] have presented experimental results where the sources, mixing, observations and noise are binary and the data are generated by Formula (6.1). In estimating the mixing matrix $\mathbf{A}$, they threshold the estimate to binary values. In their paper the emphasis is on finding the mixing matrix instead of the binary sources. The FastICA algorithm is used, with the nonlinearity measuring either the skewness or the kurtosis of the estimated sources. The results on sparse simulated data are quite promising.

ICA was originally developed for continuously distributed latent source signals. Assuming binary valued sources poses the difficulty that the source density is not differentiable but instead consists of a peak at 0 and another one at 1. Maximum likelihood (ML) based methods such as Infomax [13] and natural gradient [6] use the source density and its derivative in the ICA estimation; now the derivative is not available. Instead, the source density must be approximated by a differentiable density. Palmieri et al. [119] assume that the data are continuous valued and they approximate the desired source density as a two-mode mixture of Gaussians.

Several other authors have discussed the ICA of binary (or in general, discrete) sources and non-binary observations with Gaussian noise. Belouchrani and Cardoso [14, 15] suggest a version of the ML for discrete sources in which the source distribution is known. The maximization of the likelihood is performed via the expectation-maximization (EM) [34]

algorithm. In their approach it is possible to separate more sources than observed signals. The EM algorithm becomes nevertheless quite demanding as the computational complexity grows exponentially with the dimension of the data. Discrete sources are often encountered in telecommunications. Comon and Grellier [30] assume more sources than observed signals and use the MAP method to estimate the sources in a telecommunications setting. Miskin [103] gives ensemble learning type algorithms for the task. In his approach, more sources than sensors can be estimated, as the prior distribution for the sources is sparse. Cheung and Xu's [26] method is based on clustering the observations and thereby determining the correct number of sources. Højen-Sørensen et al. [66] discuss a Bayesian approach to ICA in which several source distributions, including binary, can be studied. In all of these approaches, the noise is assumed to be Gaussian, which is not suitable for our problem of analyzing binary valued data.

To conclude, assuming that binary data are generated by the interaction of binary components seems a problem not suitable for ICA as such. Instead, we will turn our attention to non-binary latent spaces, discussed in the following section.

6.3 Topic models

6.3.1 Data model and problem setting

In Publications 5 and 6 we discuss the problem of finding latent structure in binary valued data. In contrast to what was discussed in Section 6.2, we do not assume that the latent variables are binary but instead both $\mathbf{A}$ and $\mathbf{S}$ contain probabilities or "activations" similarly to the case in Chapter 5. In Publications 5 and 6 we present a probabilistic model and give two algorithms for estimating the structure in the data. The publications are briefly reviewed and some new insights are given in the following, together with pointers to related work.

We assume that the data are generated by interactions between independent latent topics: Each topic has a probability s_j of being active in an observation vector. The topics j generate occurrences of variables[1] x_i according to some topic-variable probabilities $\mathbf{A}(i, j)$. For each observation $\mathbf{x}$, some topics are first selected according to their individual activity probabilities $\mathbf{s}$. The selected topics then generate observations according to the topic-variable probabilities $\mathbf{A}$.

We assume that $\mathbf{A}(i, j)$ are probabilities constant over the observed binary data vectors (as in ICA) but we do not restrict them otherwise; for example we do not require that they sum to 1 over i or j. (The term "probability" is not the most precise here, but as it is used in Publications 5 and 6, we continue to use it here.) Similarly, the s_j are probabilities drawn from some predefined distribution, not summing to 1 over j. The drawing can either be done once for the whole data set, making the s_j constant over observations, or repeatedly for each observation vector. In the latter case $\mathbf{S}(j, l)$ denotes the probability of topic j in observation l.

[1] The observed variables are called "attributes" in Publications 5 and 6.

As mentioned earlier, in information retrieval the term "topic" refers to a probability distribution on the universe of terms. Although the columns of $\mathbf{A}$ do not sum to 1 in our approach and thus do not form a probability distribution, we choose to call them topics in this overview and in the publications.

Figure 6.1 shows an example of our topic model. The topics are denoted by 1, 2 and 3 and the variables by A, B and so on. We assume that different topics give rise to mostly non-overlapping sets of variables. That is, for a topic j, $\mathbf{A}(i, j)$ is large for a subset of variables i, and for another topic j', $\mathbf{A}(i, j')$ is large for another subset of variables i, mostly disjoint from the former subset. This is a sensible assumption for several real world problems. For example in document data, mainly different terms are used to discuss different topics, except for homonyms and polysemes[2] that may belong to several topics and bear a different meaning in each topic.

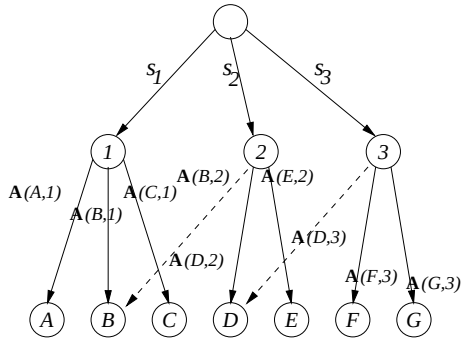

Figure 6.1: An example topic model. Topics 1, 2 and 3 are generated independently of each other with probabilities s_1, s_2 and s_3. The topics then generate observed variables with probabilities $\mathbf{A}(i, j)$. The dashed arrows indicate that a variable may be generated by several topics.

Specifically, to achieve what we earlier described vaguely as "mostly non-overlapping sets of variables", we can restrict the values of $\mathbf{A}$ in two different ways. In Publication 5 we use the concept of *ε-separability*, first presented by Papadimitriou et al. [120]. This states that each topic j has a disjoint set of *primary variables* U_j, and of the total probability mass $\sum_i \mathbf{A}(i, j)$ of topic j, a small fraction of size ε with $0 \leq \varepsilon \leq 1$ belongs to variables other than the primary variables U_j of the topic. That is, $\sum_{i \notin U_j} \mathbf{A}(i, j) \leq \varepsilon \sum_i \mathbf{A}(i, j)$. Let us now stop for a moment and analyze what this actually means. From the viewpoint of an individual variable i, this does not restrict at all the way in which the probabilities $\mathbf{A}(i, j)$ are distributed — even if i is the primary variable of some topic j', there might be another topic j'' for which $\mathbf{A}(i, j'') > \mathbf{A}(i, j')$. We might argue that the concept of "primary variables" is not always intuitive and the estimation of the main structure of the data may be cumbersome.

To overcome the problem discussed above, we restrict the distribution of $\mathbf{A}$ in another way in Publication 6. Instead of ε-separability which controls the "outgoing" probabilities of

[2]A homonymous word has several different, unrelated meanings, such as "bank". A polysemous word has several different but related meanings, such as "branch" [99].

the topics, in Publication 6 we use the concept of θ-*bounded conspiracy* which controls the "incoming" probabilities of the variables. The θ-bounded conspiracy condition states that every variable i has a *primary topic* j' for which $\sum_{j \neq j'} \mathbf{A}(i,j) \leq \theta \mathbf{A}(i,j')$. That is, the probability that any of the non-primary topics generates i is at most θ times the probability that the primary topic generates i. From the viewpoint of an individual variable i it is now clear into which topic it mainly belongs to. In Figure 6.1 the dashed arrows indicate that either $\varepsilon > 0$ or $\theta > 0$, depending on how we wish to restrict the structure of the data.

Unfortunately, we cannot write a topic model simply as $\mathbf{P} = \mathbf{AS}$ where $\mathbf{P}(i,l)$ would give the probability $P(\mathbf{X}(i,l) = 1)$, that is, the probability of seeing variable x_i in observation l. As several topics may generate a variable, the formula becomes more cumbersome. For example, in Figure 6.1 we have $P(B = 1) = s_1\mathbf{A}(B,1) + s_2\mathbf{A}(B,2) - s_1 s_2 \mathbf{A}(B,1)\mathbf{A}(B,2)$. Generally, allowing any topic to generate any variable[3], we have

$$P(\mathbf{X}(i,l) = 1) = \sum_j \mathbf{S}(j,l)\mathbf{A}(i,j) - \sum_j \sum_{j'<j} \mathbf{S}(j,l)\mathbf{S}(j',l)\mathbf{A}(i,j)\mathbf{A}(i,j') + \mathcal{O}(\mathbf{S}(j,l)^3 \mathbf{A}(i,j)^3). \quad (6.2)$$

Thus $\mathbf{P}$ equals $\mathbf{AS}$ plus some extra terms. However, if the topics are almost disjoint (that is, $\varepsilon \approx 0$ or $\theta \approx 0$), we can omit the extra terms. Also, for any ε or any θ, if the probabilities are quite small, the first term dominates. The first term is a sum of n terms of order $\mathcal{O}(\mathbf{A}(i,j)\mathbf{S}(j,l))$, the second is a sum of $n(n-1)/2$ terms of order $\mathcal{O}(\mathbf{A}(i,j)^2\mathbf{S}(j,l)^2)$ and so on. For example, if the probabilities are 0.2 on the average, the first term dominates as long as the number of topics is $n < 51$, a limit seldom exceeded in practical data sets.

The approximation $\mathbf{P} = \mathbf{AS}$ bears close resemblance to both ICA, PLSA and NMF. The latter two were discussed in Section 2.4; the ICA variant we will study here is the nonnegative ICA by Plumbley [126], described in more detail in Section 6.3.4. As we do not observe the probabilities in $\mathbf{P}$ but instead the binary outcomes in the matrix $\mathbf{X}$, we must decompose the data as $\mathbf{X} = \mathbf{AS}$ instead when using ICA, PLSA or NMF. This is of course a crude approximation but hopefully hints which elements of $\mathbf{A}$ and $\mathbf{S}$ are non-zero and thus sheds light on the structure of the data. Comparative results on different methods are given in Section 6.3.3.

6.3.2 Algorithms

In Publications 5 and 6 we give two methods for finding the latent structure in the data. The first method is called the *Probe* algorithm, discussed in Publication 5. Intuitively, if two variables belong to the same topic, then they behave similarly with respect to any third variable C. For every pair of variables A, B we compute the probe distance

$$d(A,B) = \sum_{C \neq A,B} |P(C|A) - P(C|B)| \quad (6.3)$$

[3]In Publication 5, Section 3, in the sentence "In the ε-separable case, any variable may in principle be generated by any topic ..." the higher order interactions in $p(A)$ are missing. Although not mentioned in the paper, it is assumed that ε is very small and all probabilities are quite small so that the higher order terms can be omitted.

where the probabilities P are estimated as frequencies in the data. Optionally, the probe distances can be scaled so that each variable's average distance to all other variables is the same. This is done by scaling the sum of all probe distances to a variable to 1. The scaling is beneficial if some variables are very rare.

Using the above distance measure, we then cluster the variables into possibly overlapping sets which correspond to the topics. The term "clustering" usually refers to non-overlapping clusters; terms such as relaxed clustering and multi-faceted clustering are used in the literature when an object is allowed to belong to multiple clusters simultaneously. We accomplish relaxed clustering by hierarchical linkage clustering that is interrupted before all variables are clustered[4]. The remaining variables typically have a small distance to variables in several different clusters, and thus they are inserted into more than one cluster.

The second method for estimating the latent topics is called the *Ratio* or *Lift* algorithm, discussed in Publications 5 and 6[5]. It computes for every pair of variables A, B a statistic called *lift*:

$$\mathrm{lift}(A, B) = \frac{P(A \mid B)}{P(A)} = \frac{P(A, B)}{P(A)P(B)} \tag{6.4}$$

where again the probabilities are estimated as frequencies in the data. The lift statistic equals 1 if variables A and B are independent (that is, they belong to different topics) and the larger the lift statistic is, the more dependent the occurrences of A and B are. Furthermore, if both A and B belong to one topic only, then $\mathrm{lift}(A, B) = 1$ if they belong to different topics and $\mathrm{lift}(A, B) = s_j^{-1}$ if they belong to the same topic j. Thus such variables are easy to cluster into topics.

The case of variables belonging to several topics (let us call them *multi-topic* variables) is more cumbersome and is analyzed in Publication 6. Assume that there are some variables belonging to one topic only (let us call them *single-topic* variables). Then the lift between a multi-topic variable and any single-topic variable can be approximately written as a linear combination of lifts between single-topic variables. The coefficients of this linear combination indicate which single-topic variables share a topic with the multi-topic variable, that is, into which topics the multi-topic variable belongs.

We have some reasons to believe that the above linear approximation also holds for the probe distances in some form or another. This is a topic of a further study.

Both methods estimate the latent structure in the data mainly by telling which variables belong to the same topic, and thus present the topics as lists of variables. In the ICA terminology, this information is given by the $\mathbf{A}$ matrix, as discussed previously in Section 6.3.1. We regard this information more important than that given by the ICA source matrix $\mathbf{S}$ that tells the probabilities s of the topics in the observation vectors. Using the lift statistic, we can approximate these topic probabilities by averaging $\mathrm{lift}(A, B) = s_j^{-1}$ over all single-topic variables of topic j, if needed. Methods such as PLSA, LDA and MPCA (discussed in Section 2.4) estimate the topic probabilities, too.

[4]Mr. Johan Himberg's help in programming the interrupted hierarchical linkage clustering is appreciated.

[5]We used the name "Ratio" in Publication 5 and the name "Lift" in Publication 6. Before writing Publication 6 we became aware that the term "Lift" had been used in the literature.

6.3.3 Experimental results

Preliminary experimental results on the Probe and Lift algorithms are given in Publication 5. In Publication 6, the mean squared errors in estimating the topic-variable probability matrix $\mathbf{A}$ are presented for the Lift algorithm, NMF [97], PLSA [65], and K-means. The corresponding results for the Probe algorithm and nonnegative ICA [126] are given in this section.

We generated artificial data according to our θ-bounded topic model presented in Section 6.3.1. The model has 10 topics and 100 variables; other details of the model are given in Publication 6. The Lift algorithm gives the topic-variable probabilities as explained in Publication 6. The Probe algorithm assigns the variables into overlapping topics but does not give estimates of the topic-variable probabilities. Instead, we can estimate the probabilities of single-topic variables similarly to the Lift algorithm, and for the multi-topic variables we approximate each probability as a mean over all probabilities in the corresponding topic. For NMF, PLSA[6] and nonnegative ICA[7], we decompose the data as $\mathbf{X} = \mathbf{AS}$ as discussed at the end of Section 6.3.1; this gives us directly an estimate of $\mathbf{A}$. A naive alternative to these latent variable methods is the simple K-means algorithm which clusters the variables into non-overlapping sets.

Figure 6.2 shows the mean squared errors (MSE's) of the estimated topic-variable probabilities, compared to the true probabilities used to generate the data. The conspiracy parameter θ runs from 0 to 1 with intervals of 0.02. For each θ, the topic probabilities s are sampled anew, so there is great variability in the generating models. In Figure 6.2 we see that for smaller θ, the Lift and Probe algorithms estimate the topic-variable probabilities and thus the structure of the data very nicely. When θ grows very large, the model is more difficult to estimate with these two algorithms. The behaviors of nonnegative ICA, NMF and PLSA do not depend on θ, which is natural: the methods are not primarily aimed for such θ-bounded data, but instead are able to estimate the structure also when the topics are totally overlapping. Nonnegative ICA does not force the matrix $\mathbf{A}$ to nonnegative values, and therefore we need to threshold the negative values to zero, which perhaps gives unfair advantage; the mean squared errors are very small. In favor of NMF and PLSA it must be noted that although their MSE is large at small θ, they might still be able to reconstruct the data. The Probe algorithm does not always work as well as the Lift algorithm, perhaps due to the lack of the linear approximation that we presented for the Lift algorithm in Publication 6 (briefly discussed in Section 6.3.2). The K-means algorithm estimates the structure of the data poorly for all θ.

To conclude, we suggest that the Probe and Lift algorithms approximate the structure of the binary data quite well if the latent topics only overlap to a small or moderate degree. One of their advantages is the simplicity of computations.

As an example of real word data we use the same data set as in Publications 5 and 6: a collection of bibliographical data on computer science[8]. The number of documents (that is, bibliographical entries) is 67066. We remove a small set of stop words and, for the results

[6]The PLSA [65] method was kindly programmed by Mr. Teemu Hirsimäki. No simulated annealing was used in the EM algorithm of the PLSA in our experiments.

[7]The code was kindly provided by Dr. Mark Plumbley.

[8]Available at http://liinwww.ira.uka.de/bibliography/Theory/Seiferas/

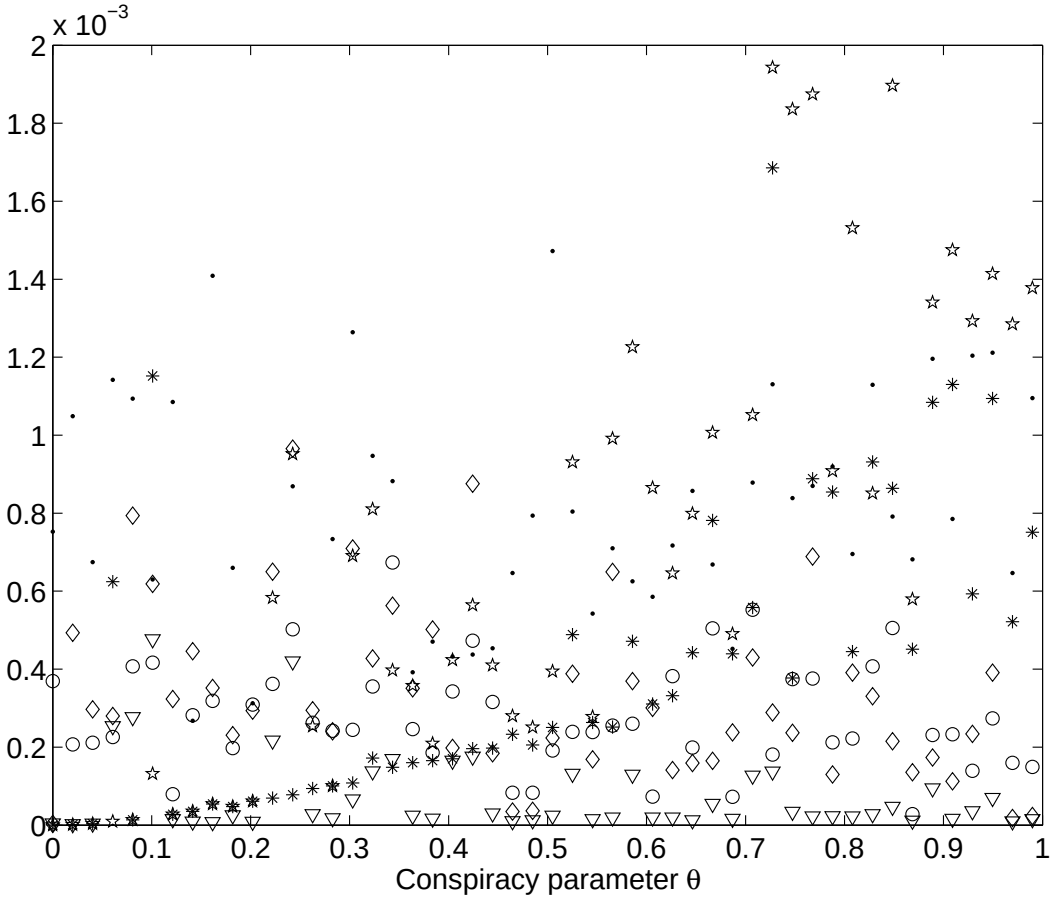

Figure 6.2: Mean squared errors of topic-variable probabilities at different conspiracy parameters θ. NMF $\Diamond$, PLSA $\circ$, Nonnegative ICA $\triangledown$, K-means $\cdot$, Probe $\star$, Lift $*$.

shown in this chapter and in Publication 6, we select the 100 most common terms. An entry of the data matrix is 1 if the term appears in the document and 0 otherwise. The data are quite sparse: about 2 per cent of the entries of the data matrix are non-zero.

We use a hierarchical average linkage clustering algorithm (available in Matlab) on the pairwise distances between the terms. The distances are given either as the scaled probe distances (6.3) or as the inverses of the lift statistics (6.4) (the lift is a similarity measure, so a convenient distance measure is obtained by taking its inverse). We cluster the terms into topics; the results are shown in Publication 6 for the lift statistic and in Table 6.1 for the case of probe distances. The number of topics was chosen as 21 in Publication 6, and the same number is chosen here for the ease of comparison of the results.

In Table 6.1 we see that the clusters are of different sizes and encompass very different terms. The structure our method yields is immediately familiar for a theoretical computer scientist. The estimated topics are surprisingly coherent in their term lists. Some topics concentrate on a very specific area within computer science: the terms of topic 1, for instance, are clearly words that occur frequently in the titles of papers on graph algorithms; topic 5 is about programming and topic 11 deals with formal languages. Topics 3, 6, 7, 9, 12, 13 and 21 are also about quite well-defined areas in computer science. Some topics only encompass a few terms that behave similarly because they are almost synonyms (topic 14) or appear frequently together (topics 10, 15, 16, 19 and 20). The rest of the topics — 4, 8 and 17 — contain terms whose meaning suits many different contexts. Some topics (2 and 18) correspond to publication forums[9]; typically only one of these terms appears in a document. This is in contrast to the topics listed above, the topics with "scientific content", several of whose terms appear in one document.

In addition to the terms listed in Table 6.1 there are five "outliers" whose probe distance is large to all other terms and thus they do not get clustered into any topic: 'approximation', 'codes', 'communication', 'dynamic' and 'scheduling'.

Note that it can well happen that several topics apply to one document: the "content-bearing" terms in the title and the publication forum are represented in different topics. A subjective comparison between Table 6.1 here and Table 1 in Publication 6 suggests that the topics found by the Probe algorithm are even more coherent than the ones found by the Lift algorithm. We conclude that the probe distances and lift statistics are fruitful ways of finding related term sets in document data.

[9]Explanations of the abbreviations in Table 6.1: In topic 2, 'actainf' is Acta Informatica, 'beatcs' is Bulletin of the European Association for Theoretical Computer Science, 'damath' is Discrete Applied Mathematics, 'dmath' is Discrete Mathematics, 'focs' is IEEE Symposium on Foundations of Computer Science, 'icalp' is International Colloquium on Automata, Languages and Programming, 'infctrl' is Information and Computation (formerly Information and Control), 'ipl' is Information Processing Letters, 'jacm' is Journal of the ACM, 'jcss' is Journal of Computer and System Sciences, 'libtr' is one kind of a technical report, 'mfcs' is International Symposium on Mathematical Foundations of Computer Science, 'sicomp' is SIAM Journal on Computing, 'stacs' is International Symposium on Theoretical Aspects of Computer Science, 'stoc' is ACM Symposium on Theory of Computing, 'tcs' is Theoretical Computer Science, and 'tr' is technical report. In topic 18, 'cacm' is Communications of the ACM, 'crypto' is International Cryptology Conference, 'ieeetc' is IEEE Transactions on Computers, and 'lncs' is Lecture Notes in Computer Science.

topic	terms
1	algorithm algorithms efficient fast graph graphs matching optimal parallel problem set simple
2	actainf beatcs damath dmath focs geometry icalp infctrl ipl jacm jcss libtr mfcs sicomp stacs stoc tcs tr
3	complexity functions machines probabilistic
4	applications problems some
5	approach de logic model programming programs system systems van
6	network networks routing sorting
7	computational information theory
8	linear new two
9	binary search tree trees
10	polynomial time
11	algebraic automata finite languages note properties sets theorem
12	data structures
13	analysis design distributed using
14	computation computing
15	bounds lower
16	computer science
17	from learning
18	cacm crypto ieeetc lncs
19	number random
20	abstract extended
21	finding minimum planar

Table 6.1: Topics in bibliographical data, computed using the probe distances. The terms are in alphabetical order; the order of the topics is not relevant.

6.3.4 Related methods

Our model can be interpreted as a *Bayesian network* (a graphical model): the topics and the observed variables are the nodes of the graph. The nodes are discrete valued, the relations between them are directed and acyclic, and child nodes (here the observed variables) are independent of each other given the values of their parent nodes (here the topics). For an introduction to graphical models, see e.g. [82, 83].

Our model is also an example of a *multiple cause model*. A seminal paper presenting such models for binary valued data is by Saund [134], stating that "a multiple cause model accounts for observed data by combining assertions from many hidden causes, each of which can pertain to varying degree to any subset of the observable dimensions". The data likelihood of our model, presented in Publication 6, is similar but not identical to that in [134] where a so called "soft disjunction" or "noisy-OR" mixing is applied. Sahami et al. [132] apply the model to text categorization. The model is estimated by iterative gradient descent in [132, 134]. A similar model is presented by Jaakkola [78] and solved by variational methods.

Our data model is somewhat similar to that of Papadimitriou et al. [120] (discussed in Publication 5) and the one in latent Dirichlet allocation and multinomial PCA (LDA and MPCA, see Section 2.4). The task of decomposing the data is similar to Hofmann's PLSA [64, 65], Lee and Seung's NMF [97, 98] and Hoyer's nonnegative sparse coding [68]; all of these were discussed in Section 2.4 and PLSA and NMF were used in the experiments in the previous section. However, these references do not give special emphasis on binary data.

Hinton et al. [61, 63] have presented a method called *contrastive divergence*. It is a graphical model that uses a restricted Bolzmann machine [141] whose units are binary valued and the hidden units are not connected to each other. All hidden units are connected to all visible units with bidirectional connections. This corresponds to our topic model where the topics are in the hidden layer and the observed variables in the visible layer, except that in our model the dependence relations are not bidirectional but only the observed variables depend on the topics. When the values of the variables are observed, the hidden units of the Bolzmann machine are conditionally independent. In contrast, in our model and in PLSA, LDA and MPCA, the observed variables are conditionally independent given the topics.

Bernoulli mixtures (see, e.g., [40, 55]) are a common choice for analyzing binary data. However, we do not wish to use them since mixture models assume that all entries of an observation vector have the same probabilities of being generated by the jth latent component distribution, as mentioned in Section 2.4.

There are a few ICA approaches to estimating nonnegative $\mathbf{A}$ and $\mathbf{S}$. Generally, the matrix entries are not assumed to be probabilities but any nonnegative real numbers. Nuzillard [110] gives a method that constrains both the mixing matrix and latent variables to nonnegative values. She first performs original ICA and then uses an iterative method called Alternated Least Squares to restrict the mixing matrix and the latent variables to nonnegative values. Parra et al. [121] present a constrained ICA method in which the problem of estimating nonnegative mixing and latent variables is solved by a maximum a posteriori (MAP) approach. The priors for the mixing matrix and for the latent variables impose the nonnegativities.

In his *nonnegative ICA*, Plumbley [124] suggests that decorrelating the observed data is

enough if nonnegativity constraints are imposed on the latent variables and the mixing matrix. Usually in ICA, decorrelation of observations is just a preprocessing step and not a sufficient condition for independence — according to Plumbley, the nonnegativity constraints are sufficient to fix the remaining underdetermined parameters of the solution, provided that the sources have a non-zero probability density function in the positive neighborhood of zero. In Plumbley [125, 126] and Oja and Plumbley [113] the case of nonnegative sources is studied, but the values of the mixing matrix $\mathbf{A}$ are not assumed nonnegative. Several algorithms are given, all starting with data whitening. After the whitening step, it remains to find a rotation which forces all of the data into positive values. This is done either by axis pair rotations, by nonnegative PCA, or by a geodesic search in the space of orthogonal matrices; the latter was used in our experiments in Section 6.3.3.

Generative topographic mapping (GTM, [144, 17]) gives a probabilistic visualization of high-dimensional data; in this case the latent variables are Gaussian and not binary valued. Pajunen [118] and Girolami [50, 51] have presented binary versions of GTM: the latent space is not continuous but an n-dimensional grid of discrete points to which the observations are mapped (typically $n = 2$). This gives a clustering of data points. The clusters are non-overlapping in the sense that one observation belongs to one cluster only.

A somewhat different method for analyzing binary (or in general, term-document data) is given by Dhillon's *co-clustering* [35], sometimes also called *double clustering*. In contrast to clustering only terms or documents, he clusters both dimensions of the data matrix simultaneously by a spectral graph partitioning algorithm. Dhillon does not present his approach as a latent variable model. However, an analogy can be drawn to the approach described in this chapter: a term can be clustered into those topics in which its probability of appearance is "large" or significantly non-zero, and similarly a document can be clustered into topics which have a "large" probability of occurrence in this document. (This analogy lends itself easily to NMF and PLSA, too.) The clustering in Dhillon's co-clustering is non-overlapping in the sense that a document (or similarly, a term) can only belong to one cluster; in our topic model the clusters may overlap. A method related to Dhillon's co-clustering and to principal component analysis is *correspondence analysis* [42] that displays a low dimensional projection of the data for two variables simultaneously (in [42], for genes and hybridizations).

Overlapping clusters are encountered in the field of *fuzzy systems* [154], too: assuming fuzzy topics, each observation would have a membership value ranging from 0 to 1 in each topic. The case of both observations and variables having a membership value in a topic is not straightforward in this setting, neither is the use of the concept of latent variables.

Another method related to clustering is the famous *information bottleneck* method of Tishby et al. [147]. More precisely, it is an information-theoretical approach to dimensionality reduction. Variables are clustered in a way that maximizes the mutual information within the cluster. An application of the method to term and document clustering is given in [139]: first, the terms are clustered so that the obtained clusters maximally preserve the information about the documents. Then the documents are clustered so that the information about the term clusters is preserved. The clusters are not overlapping in this double clustering approach. In an earlier paper, Becker and Hinton [12] also found coherent regions in the observed data by maximizing mutual information; the method is for the continuous space. A corresponding task, again in the continuous space, is accomplished by the *discriminative clustering* method by Sinkkonen et al. [137, 136].

Chapter 7

Conclusion

7.1 Summary

This thesis considered the problem of representing a large data set in a compressed format. It was assumed that the data are not completely random but there are some regularities that form a kind of internal structure in the data. A natural task in such a setting is to find out this internal or latent structure and thus obtain a simple representation of the data.

The hypothesis to start with was that an effective method for estimating latent structure in data, independent component analysis (ICA), could be extended into new problem settings. ICA, which originated in the field of signal processing, is a powerful and promising method for solving a problem that at first sight seems unsolvable: Assume that the observed data are generated by some unknown interactions between unknown but statistically independent latent variables. Using the observed data only, find these unknown latent variables and the way they interact. It has been shown [29] that the problem is indeed solvable if some requirements are met, as discussed in Section 2.1. ICA has been applied more or less successfully to various different problems in a multitude of application areas; some applications are listed in [72] and some others are discussed in this thesis.

Tempted by its promise, the work was started to extend ICA in various different ways. The first extension was from real valued to complex valued signals. A simple but computationally efficient algorithm for separating complex valued, linearly mixed signals was given in this thesis; this was not the first ICA algorithm for such a problem, but the elegance of the FastICA algorithm [76, 70] had not been previously applied to the problem. Conditions on the local consistency of the estimator given by the algorithm were also given in this thesis.

The second extension was to use ICA as a preprocessing method in nonlinear regression: First, using the labeled training examples of predictor and predicted variables, the independent latent variables and their linear mixing are estimated by ordinary ICA. Then instead of forming a nonlinear regression on the original predictor variables, the nonlinearities are applied to the independent latent variables and the result is transformed back to the original

observation space by projecting with the estimated mixing matrix. This procedure resembles closely regression by a multilayer perceptron (MLP) but is defined in a more disciplined way: choosing the nonlinearities and the weight matrices of the MLP network is now straightforward.

The third extension was the use of ICA in information retrieval. The original idea of finding the latent topics of written text had been presented elsewhere. In this thesis, the focus was on analyzing the topics of discussion in a dynamically evolving text stream; as an example of such data, chat line discussion appearing on the Internet was used. It was shown that by using an ICA-type method developed for time-dependent signals, called complexity pursuit [71], the topics in such data can be found and visualized in a convenient way. This has applications in the retrieval of text whose characteristics change over time, such as online news services.

The fourth extension in this thesis was the analysis of binary valued data that contain some hidden topics; again text document data may serve as an example. A restricting assumption on the latent structure of the data was that the observed variables (that is, terms) have unknown probabilities of belonging to each latent topic, and the latent topics themselves have unknown probabilities of appearance in each observed document. In short, a nonlinear mixture of the latent topics is observed, and as the parameters of the model are probabilities, ordinary (linear) ICA cannot be used as that would result in negative values for the parameters. Instead, two methods were presented for analyzing such data. The methods are based on the independence of the latent topics and can thus be seen as extensions of the original idea of ICA, although the name ICA might be misleading in this setting.

Apart from the objective of extending ICA into new problem domains, this thesis also had other objectives. As mentioned at the beginning of this chapter, a general aim was to study methods for representing a large data set in a compressed form by finding some latent structure in it. A somewhat overlapping aim was to discuss methods for dimensionality reduction (the focus was mainly on projection-based methods). The methods for finding the latent structure in the data can be viewed as performing dimensionality reduction, so these aims are intertwined. Specifically, the use of random projection as a method of dimensionality reduction was studied. It was also proposed that random projection is a suitable method of data preprocessing prior to ICA, and supporting empirical evidence was given in several different contexts.

This thesis also gave literature surveys on each addressed problem: latent variable decompositions, separation of complex valued signals, ICA-type methods in regression and in information retrieval, and the estimation of structure in binary valued data.

7.2 Further work

From the point of view of the ICA community, the linear ICA model has been studied extensively during the past years and it is natural to set the focus on new problem settings. The extensions presented in this thesis are by far not the only possible ones, nor are they yet conclusively studied here.

The problems encountered in information retrieval often involve non-continuous data: binary, count or (continuous) nonnegative data. The case of binary data was studied in this thesis; other types of data deserve attention in the future, too. Also, many kinds of restrictions on the type of the latent structure being sought for are possible. We concentrated on "additive" interactions of the latent variables. A structure worth studying is one in which some latent variables have inhibitory effects: for example, if some topic is active, then some other topic must stay inactive (of course, such topics are not independent of each other any more and the independence cannot be utilized in the algorithms). Another form of inhibition is one in which a topic generates exclusive-or appearances of terms: among two specific terms, only one may appear at a time. Or, a topic favors the appearance of a term and inhibits the appearance of another.

The analysis of the link structure in large graphs such as the World Wide Web is an interesting new field of application of many latent variable models. Questions such as "Are there some smaller subgraphs that are more or less independent of each other?" can be posed. Proposed techniques for the link analysis include spectral partitioning [37, 43, 101, 109, 152] and the HITS [90] and PageRank [116, 19] algorithms that can also be conveniently presented in matrix form [36]. The field of bioinformatics has several problems where latent variable methods could prove useful, too. For example, given a DNA sequence, the task would be to decompose it into short subsequences generated by some latent variables. We hope to be able to address these problem domains in the future, taking into account the specific restrictions posed by the application areas.

Bibliography

[1] Dimitris Achlioptas. Database-friendly random projections. In *Proceedings of the ACM Symposium on the Principles of Database Systems*, pages 274–281, 2001.

[2] Dimitris Achlioptas and Frack McSherry. Fast computation of low rank matrix approximations. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 611–618, 2001.

[3] Helena Ahonen, Oskari Heinonen, Mika Klemettinen, and A. Inkeri Verkamo. Applying data mining techniques in text analysis. Technical Report C-1997-23, University of Helsinki, Department of Computer Science, 1997.

[4] James Allan. Introduction to topic detection and tracking. In James Allan, editor, *Topic Detection and Tracking: Event-based Information Organization*, pages 1–16. Kluwer, 2002.

[5] James Allan, Jaime Carbonell, George Doddington, Jonathan Yamron, and Yiming Yang. Topic detection and tracking pilot study: Final report. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pages 194–218, 1998.

[6] Sun-Ichi Amari. Natural gradient works effectively in learning. *Neural Computation*, 10(2):251–276, 1998.

[7] Jörn Anemüller, Terrence J. Sejnowski, and Scott Makeig. Complex spectral-domain independent component analysis of electroencephalographic data. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 47–52, 2003.

[8] Andrew D. Back and Ah Chung Tsoi. Blind deconvolution of signals using a complex recurrent network. In *Neural Networks for Signal Processing 4, Proceedings of the 1994 IEEE Workshop*, pages 565–574. IEEE Press, 1994.

[9] Andrew D. Back and Andreas S. Weigend. A first application of independent component analysis to extracting structure from stock returns. *International Journal of Neural Systems*, 8(4):473–484, 1997.

[10] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, New York, 1999.

[11] Wolf Baumann, Dorothea Kolossa, and Reinhold Orglmeister. Maximum likelihood permutation correction for convolutive source separation. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 373–378, 2003.

[12] Suzanna Becker and Geoffrey E. Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355:161–163, 1992.

[13] Anthony J. Bell and Terrence J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[14] Adel Belouchrani and Jean-François Cardoso. Maximum likelihood source separation for discrete sources. In *Proceedings of the VII European Signal Processing Conference (EUSIPCO'94)*, pages 768–771, 1994.

[15] Adel Belouchrani and Jean-François Cardoso. Maximum likelihood source separation by the Expectation-Maximization technique: Deterministic and stochastic implementation. In *Proceedings of the International Symposium on Nonlinear Theory and its Applications (NOLTA'95)*, pages 49–53, Las Vegas, Nevada, USA, 1995.

[16] Ella Bingham, Jukka Kuusisto, and Krista Lagus. ICA and SOM in text document analysis. In *Proceedings of the 25th ACM SIGIR 2002 International Conference on Research and Development in Information Retrieval*, pages 361–362, 2002.

[17] Christopher M. Bishop, Markus Svensén, and Chris K.I. Williams. GTM: The generative topographic mapping. *Neural Computation*, 10(1):215–234, 1998.

[18] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. In *Advances in Neural Information Processing Systems 14*, pages 601–608, 2001.

[19] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[20] Wray Buntine. Variational extensions to EM and multinomial PCA. In *Machine Learning: ECML 2002*, number 2430 in Lecture Notes in Artificial Intelligence (LNAI), pages 23–34. Springer-Verlag, 2002.

[21] Vince Calhoun, Tulay Adali, Lars Kai Hansen, Jan Larsen, and Jim Pekar. ICA of functional MRI data: An overview. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 281–288, 2003.

[22] Jean-François Cardoso. Eigen-structure of the fourth-order cumulant tensor with application to the blind source separation problem. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'90)*, pages 2655–2658, 1990.

[23] Jean-François Cardoso. Multidimensional Independent Component Analysis. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'98)*, pages 1941–1944, 1998.

[24] Jean-François Cardoso. High-order contrasts for independent component analysis. *Neural Computation*, 11(1):157–192, 1999.

[25] Jean-François Cardoso and Antoine Soloumiac. Blind beamforming for non Gaussian signals. *IEE Proceedings-F*, 140(46):362–370, 1993.

[26] Yiu-Ming Cheung and Lei Xu. Rival penalized competitive learning based approach for discrete-valued source separation. *International Journal of Neural Systems*, 10(6):483–490, 2000.

[27] Ira J. Clarke. Blind separation of complex-valued signals by real-valued in-phase and quadrature rotations. In *Proceedings of the X European Signal Processing Conference (EUSIPCO 2000)*, volume II, pages 689–692, 2000.

[28] Michael Collins, Sanjoy Dasgupta, and Robert E. Schapire. A generalization of principal component analysis to the exponential family. In *Advances in Neural Information Processing Systems 14*, pages 617–624, 2001.

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

[30] Pierre Comon and Olivier Grellier. Non-linear inversion of underdetermined mixtures. In *Proceedings of the International Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 461–465, 1999.

[31] Pierre Comon and Eric Moreau. Improved contrast dedicated to blind separation in communications. In *Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97)*, pages 3453–3456, 1997.

[32] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[33] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[34] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[35] Inderjit S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. Technical Report TR 2001-05, Department of Computer Sciences, University of Texas, Austin, 2001.

[36] Chris Ding, Xiaofeng He, Parry Husbands, Hongyuan Zha, and Horst Simon. PageRank, HITS and a unified framework for link analysis. Technical Report 49372, Lawrence Berkeley National Laboratory, 2002.

[37] William E. Donath and Alan J. Hoffman. Lower bounds for the partitioning of graphs. *IBM Journal of Research and Development*, 17(5):420–452, 1973.

[38] Torbjørn Eltoft and Ørjan Kristiansen. ICA and nonlinear time series prediction for recovering missing data segments in multivariate signals. In *Proceedings of the Third International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001)*, pages 716–721, 2001.

[39] Jan Eriksson and Visa Koivunen. Identifiability and separability of linear ICA models revisited. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 23–27, 2003.

[40] Brian S. Everitt and David J. Hand. *Finite mixture distributions.* Chapman & Hall, London, 1981.

[41] Scott E. Fahlman and Christian Lebiere. The cascade-correlation learning architecture. Technical Report CMU-CS-90-100, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA, 1990.

[42] Kurt Fellenberg, Nicole C. Hauser, Benedikt Brors, Albert Neutzner, Jörg D. Hoheisel, and Martin Vingron. Correspondence analysis applied to microarray data. *Proceedings of the National Academy of Sciences of the United States of America*, 98(19):10781–10786, 2001.

[43] Miroslav Fielder. Algebraic connectivity of graphs. *Czechoslovak Mathematics Journal*, 23:298–305, 1973.

[44] Simone Fiori. Blind separation of circularly distributed sources by neural extended APEX algorithm. *Neurocomputing*, 34:239–252, 2000.

[45] Simone Fiori and Pietro Burrascano. Electromagnetic environmental pollution monitoring: Source localization by the independent component analysis. In *Proceedings of the Third International Conference on Independent Component Analysis and Signal Separation (ICA2001)*, pages 575–580, 2001.

[46] Jerome H. Friedman. Data mining and statistics: What's the connection? In *Proceedings of the 29th Symposium on the Interface: Computing Science and Statistics*, pages 3–9, 1997.

[47] Jerome H. Friedman and John W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions of Computers*, c-23(9):881–890, 1974.

[48] Keinosuke Fukunaga and Dan R. Olsen. An algorithm for finding intrinsic dimensionality of data. *IEEE Transactions on Computers*, 20(2):176–183, 1971.

[49] Zoubin Ghahramani and Geoffrey E. Hinton. The EM algorithm for mixtures of factor analyzers. Technical Report CRG-TR-96-1, University of Toronto, Department of Computer Science, 1996.

[50] Mark Girolami. A generative model for sparse discrete binary data with non-uniform categorical priors. In *Proceedings of the European Symposium on Artificial Neural Networks*, pages 1–6, 2000.

[51] Mark Girolami. The topographic organization and visualization of binary data using multivariate-Bernoulli latent variable models. *IEEE Transactions on Neural Networks*, 12(6):1367–1374, 2001.

[52] Mark Girolami and Ata Kabán. On an equivalence between PLSI and LDA. In *Proceedings of the 26th ACM SIGIR 2003 International Conference on Research and Development in Information Retrieval*, pages 433–434, 2003.

[53] Gene H. Golub and Charles F. van Loan. *Matrix Computations*. North Oxford Academic, Oxford, UK, 1983.

[54] Hiromu Gotanda, Kazuyuki Nobu, Takeshi Koya, Kei-ichi Kaneda, Taka-aki Ishibashi, and Naomi Haratani. Permutation correction and speech extraction based on split spectrum through FastICA. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 379–384, 2003.

[55] Mats Gyllenberg, Timo Koski, Edwin Reilink, and Martin Verlaan. Non-uniqueness in probabilistic numerical identification of bacteria. *Journal of Applied Probability*, 31:542–548, 1994.

[56] David Hand, Heikki Mannila, and Padhraic Smyth. *Principles of Data Mining*. The MIT Press, 2001.

[57] Harry H. Harman. *Modern Factor Analysis*. University of Chicago Press, 2nd edition, 1967.

[58] Robert Hecht-Nielsen. Context vectors: general purpose approximate meaning representations self-organized from raw data. In Jacek M. Zurada, Robert J. Marks II, and Charles J. Robinson, editors, *Computational Intelligence: Imitating Life*, pages 43–56. IEEE Press, 1994.

[59] Johan Himberg and Aapo Hyvärinen. Independent component analysis for binary data: an experimental study. In *Proceedings of the Third International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001)*, pages 552–556, 2001.

[60] Geoffrey Hinton and Terrence J. Sejnowski, editors. *Unusupervised Learning. Foundations of Neural Computation*. MIT Press, 1999.

[61] Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. Technical Report GCNU TR 2000-004, Gatsby Computational Neuroscience Unit, 2000.

[62] Geoffrey E. Hinton, Peter Dayan, and Mike Revow. Modeling the manifolds of images of handwritten digits. *IEEE Transactions on Neural Networks*, 8(1):65–74, 1997.

[63] Geoffrey E. Hinton, Max Welling, Yee Whye Teh, and Simon Osindero. A new view of ICA. In *Proceedings of the Third International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001)*, pages 746–751, 2001.

[64] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57, 1999.

[65] Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42:177–196, 2001.

[66] Pedro A.d.F.R. Højen-Sørensen, Ole Winther, and Lars Kai Hansen. Mean field approaches to independent component analysis. *Neural Computation*, 14:889–918, 2002.

[67] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441, 498–520, 1933.

[68] Patrik O. Hoyer. Non-negative sparse coding. In *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, pages 557–565, 2002.

[69] Peter J. Huber. Projection pursuit. *The Annals of Statistics*, 13(2):435–475, 1985.

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

[71] Aapo Hyvärinen. Complexity pursuit: separating interesting components from time-series. *Neural Computation*, 13(4):883–898, 2001.

[72] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent Component Analysis*. Wiley Interscience, 2001.

[73] Aapo Hyvärinen and Raju Karthikesh. Imposing sparsity on the mixing matrix in independent component analysis. *Neurocomputing*, 49:151–162, 2002.

[74] Aapo Hyvärinen, Jaakko Särelä, and Ricardo Vigário. Spikes and bumps: Artefacts generated by independent component analysis with insufficient sample size. In *Proceedings of the International Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 425–429, 1999.

[75] Aapo Hyvärinen and Patrik Hoyer. Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation*, 12(7):1705–1720, 2000.

[76] Aapo Hyvärinen and Erkki Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9:1483–1492, 1997.

[77] Charles Lee Isbell and Paul Viola. Restucturing sparse high dimensional data for effective retrieval. In *Advances in Neural Information Processing Systems 11*, pages 480–486, 1998.

[78] Tommi S. Jaakkola. *Variational methods for inference and estimation in graphical models*. PhD thesis, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA, 1997.

[79] William B. Johnson and Joram Lindenstrauss. Extensions of Lipschitz mapping into Hilbert space. In *Conference in Modern Analysis and Probability*, volume 26 of *Contemporary Mathematics*, pages 189–206. American Mathematical Society, 1984.

[80] Marcel Joho and Philip Schniter. Frequency domain realization of a multichannel blind deconvolution algorithm based on the natural gradient. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 543–548, 2003.

[81] M. Jones and Robin Sibson. What is projection pursuit? *Journal of the Royal Statistical Society, ser. A*, 150:1–36, 1987.

[82] Michael I. Jordan, editor. *Learning in Graphical Models*. MIT Press, 1998.

[83] Michael I. Jordan and Terrence J. Sejnowski, editors. *Graphical Models: Foundations of Neural Computation*. MIT Press, 2001.

[84] Christian Jutten and Jeanny Herault. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10, 1991.

[85] Ata Kabán and Mark Girolami. Unsupervised topic separation and keyword identification in document collections: a projection approach. Technical Report 10, Department of Computing and Information Systems, University of Paisley, August 2000.

[86] Ata Kabán and Mark Girolami. A dynamic probabilistic model to visualize topic evolution in text streams. *Journal of Intelligent Information Systems, Special Issue on Automated Text Categorization*, 18(2), March 2002.

[87] Nandakishore Kambhatla and Todd K. Leen. Dimension reduction by local principal component analysis. *Neural Computation*, 9(7):1493–1516, 1997.

[88] Samuel Kaski. Dimensionality reduction by random mapping. In *Proceedings of the International Joint Conference on Neural Networks*, volume 1, pages 413–418, 1998.

[89] Wlodzimierz Kasprzak and Adam Okazaki. Blind deconvolution of timely-correlated sources by homomorphic filtering in Fourier space. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 1029–1034, 2003.

[90] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[91] Mirko Knaak, Shoko Araki, and Shoji Makino. Geometrically constrained ICA for robust separation of sound mixtures. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 951–956, 2003.

[92] Teuvo Kohonen. *Self-Organizing Maps*. Springer, Berlin, 1995 (Second, extended edition 1997).

[93] Teuvo Kohonen, Samuel Kaski, Krista Lagus, Jarkko Salojärvi, Vesa Paatero, and Antti Saarela. Organization of a massive document collection. *IEEE Transactions on Neural Networks, Special Issue on Neural Networks for Data Mining and Knowledge Discovery*, 11(3):574–585, May 2000.

[94] Thomas Kolenda and Lars Kai Hansen. Dynamical components of chat. Technical report, Technical University of Denmark, 2000.

[95] Thomas Kolenda, Lars Kai Hansen, and Jan Larsen. Signal detection using ICA: application to chat room topic spotting. In *Proceedings of the Third International Conference on Independent Component Analysis and Signal Separation (ICA2001)*, pages 540–545, 2001.

[96] Thomas Kolenda, Lars Kai Hansen, and Sigurdur Sigurdsson. Independent components in text. In Mark Girolami, editor, *Advances in Independent Component Analysis*, chapter 13, pages 235–256. Springer-Verlag, 2000.

[97] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, October 1999.

[98] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, pages 556–562, 2000.

[99] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999.

[100] Andrew Kachites McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/~mccallum/bow, 1996.

[101] Frank McSherry. Spectral partitioning of random graphs. In *IEEE Symposium on Foundations of Computer Science*, pages 529–537, 2001.

[102] Thomas Minka and John Lafferty. Expectation-propagation for the generative aspect model. In *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, pages 352–359, 2002.

[103] James W. Miskin. *Ensemble Learning for Independent Component Analysis*. PhD thesis, Selwyn College, University of Cambridge, 2000.

[104] Nikoloas Mitianoudis and Mike Davies. New fixed-point ICA algorithms for convolved mixtures. In *Proceedings of the Third International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001)*, pages 633–638, 2001.

[105] Eric Moreau and Odile Macchi. Complex self-adaptive algorithms for source separation based on higher order contrasts. In *Proceedings of the VII European Signal Processing Conference (EUSIPCO'94)*, volume II, pages 1157–1160, 1994.

[106] Ryo Mukai, Hiroshi Sawada, Shoko Araki, and Shoji Makino. Real-time blind source separation for moving speakers using blockwise ICA and residual crosstalk subtraction. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 975–980, 2003.

[107] Klaus-Robert Müller, Petra Philips, and Andreas Ziehe. JADE$_{TD}$: Combining higher-order statistics and temporal information for blind source separation (with noise). In *Proceedings of the International Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 87–92, 1999.

[108] Noboru Nakasako and Hisanao Ogura. Complex ICA for direction finding and separation of broadband sound sources — high-quality signal separation using an inverse filter. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 633–638, 2003.

[109] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856, 2001.

[110] Danielle Nuzillard. Separation of non orthogonal spectral data. In *Proceedings of the 2nd International Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000)*, pages 321–326, 2000.

[111] Erkki Oja. Convergence of the symmetrical FastICA algorithm. In *Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'02)*, 2002.

[112] Erkki Oja. Unsupervised learning in neural computation. *Theoretical Computer Science*, 287:187–207, 2002.

[113] Erkki Oja and Mark Plumbley. Blind separation of positive sources using non-negative PCA. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 11–16, 2003.

[114] Pentti Paatero. Least squares formulation of robust non-negative factor analysis. *Chemometrics and Intelligent Laboratory Systems*, (37):23–35, 1997.

[115] Pentti Paatero and Unto Tapper. Positive matrix factorization: A non- negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, (5):127–144, 1994.

[116] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the Web. Technical report, Stanford Digital Library Technologies Project, 1998.

[117] Petteri Pajunen. Blind source separation using algorithmic information theory. *Neurocomputing*, 22:35–48, 1998.

[118] Petteri Pajunen and Juha Karhunen. A maximum likelihood approach to nonlinear blind source separation. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 541–546, 1997.

[119] Francesco Palmieri, Alessandra Budillon, Michele Calabrese, and Dadive Mattera. Searching for a binary factorial code using the ICA framework. *Neurocomputing*, (22):131–144, 1998.

[120] Christos H. Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh Vempala. Latent semantic indexing: A probabilistic analysis. In *Proceedings of the 17th ACM Symposium on the Principles of Database Systems*, pages 159–168, 1998.

[121] Lucas Parra, Clay Spence, Paul Sajda, Andreas Ziehe, and Klaus-Robert Müller. Unmixing hyperspectral data. In *Advances in Neural Information Processing Systems 12*, pages 942–948, 2000.

[122] Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, 6(2):559–572, 1901.

[123] Din-Tuan Pham, Christine Servière, and Hakim Boumaraf. Blind separation of convolutive audio mixtures using nonstationarity. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 981–986, 2003.

[124] Mark Plumbley. Adaptive lateral inhibition for non-negative ICA. In *Proceedings of the Third International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001)*, pages 516–521, 2001.

[125] Mark Plumbley. Conditions for non-negative independent component analysis. *IEEE Signal Processing Letters*, 9(6):177–180, June 2002.

[126] Mark Plumbley. Algorithms for nonnegative independent component analysis. *IEEE Transactions on Neural Networks*, 14(3):534–543, 2003.

[127] Rajkishore Prasad, Hiroshi Saruwatari, Akinobu Lee, and Kyohiro Shikano. A fixed-point ICA algorithm for convoluted speech signal separation. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 579–584, 2003.

[128] Tapani Ristaniemi and Jyrki Joutsensalo. Advanced ICA-based receivers for block fading DS-CDMA channels. *Signal Processing*, 82:417–431, 2002.

[129] Ze'ev Roth and Yoram Baram. Multidimensional density shaping by sigmoids. *IEEE Transactions on Neural Networks*, 7(5):1291–1298, September 1996.

[130] Sam Roweis. EM algorithms for PCA and SPCA. In *Advances in Neural Information Processing Systems*, volume 10, pages 626–632, 1998.

[131] Sam Roweis. A unifying review of linear Gaussian models. *Neural Computation*, 11(2):305–345, 1999.

[132] Mehran Sahami, Marti Hearst, and Eric Saund. Applying the multiple cause mixture model to text categorization. In *Proceedings of ICML-96, 13th International Conference on Machine Learning*, pages 435–443, 1996.

[133] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[134] Eric Saund. A multiple cause mixture model for unsupervised learning. *Neural Computation*, 7(1):51–71, 1995.

[135] Hiroshi Sawada, Ryo Mukai, Shoko Araki, and Shoji Makino. A robust and precise method for solving the permutation problem of frequency-domain blind source separation. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 505–510, 2003.

[136] Janne Sinkkonen and Samuel Kaski. Clustering based on conditional distributions in an auxiliary space. *Neural Computation*, 14:217–239, 2002.

[137] Janne Sinkkonen, Samuel Kaski, and Janne Nikkilä. Discriminative clustering: optimal contingency tables by learning metrics. In *Machine Learning: ECML 2002*, number 2430 in Lecture Notes in Artificial Intelligence (LNAI), pages 418–430. Springer-Verlag, 2002.

[138] Malcom Slaney and Dulce Ponceleon. Hierarchical segmentation: finding changes in a text signal. In *Proceedings of the SIAM Text Mining 2001 Workshop*, pages 6–13, 2001.

[139] Noam Slonim and Naftali Tishby. Document clustering using word clusters via the information bottleneck method. In *Proceedings of the 23rd ACM SIGIR 2000 International Conference on Research and Development in Information Retrieval*, pages 208–215, 2000.

[140] Paris Smaragdis. Blind separation of convolved mixtures in the frequency domain. In *Proceedings of the International Workshop on Independence & Artificial Neural Networks*, 1998.

[141] Paul Smolensky. Information processing in dynamical systems: Foundations of harmony theory. In David E. Rumelhart and James L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1: Foundations, pages 194–281. MIT Press, 1986.

[142] James V. Stone. Blind source separation using temporal predictability. *Neural Computation*, 13(4):1559–1574, 2001.

[143] Anisse Taleb and Christian Jutten. Source separation in post-nonlinear mixtures. *IEEE Transactions on Signal Processing*, 47(10):2807–2820, 1999.

[144] Michael E. Tipping. Probabilistic visualization of high-dimensional binary data. In *Advances in Neural Information Processing Systems 11*, pages 592–598, 1998.

[145] Michael E. Tipping and Christopher M. Bishop. Mixtures of probabilistic principal component analysers. *Neural Computation*, 11(2):443–482, 1999.

[146] Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, 61(3):611–622, 1999.

[147] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, 1999.

[148] C.J. "Keith" van Rijsbergen. *Information Retrieval*. Butterworths, 1979.

[149] Mati Wax and Thomas Kailath. Detection of signals bby information-theoretic criteria. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 33(2):387–392, 1985.

[150] Mati Wax, Tie-Jun Shan, and Thomas Kailath. Spatio-temporal spectral analysis by eigenstructure methods. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(4):817–827, 1984.

[151] Charles L. Wayne. Topic detection & tracking (TDT): Overview & perspective. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[152] Yair Weiss. Segmentation using eigenvectors: A unifying view. In *International Conference on Computer Vision*, volume 2, pages 975–982, 1999.

[153] Max Welling and Markus Weber. Positive tensor factorization. *Pattern Recognition Letters*, 22(12):1255–1261, 2001.

[154] Lotfi A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[155] Vicente Zarzoso and Asoke K. Nandi. Unified formulation of closed-form estimators for blind source separation in complex instantaneous linear mixtures. In *Proceedings of the X European Signal Processing Conference (EUSIPCO 2000)*, volume I, pages 597–600, 2000.

[156] Vicente Zarzoso and Asoke K. Nandi. A general theory of closed-form estimators for blind source separation. In *Proceedings of the Third International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001)*, pages 25–30, 2001.

[157] Vicente Zarzoso and Asoke K. Nandi. Closed-form estimators for blind separation of sources – part II: Complex mixtures. *Wireless Personal Communications*, 21(29–48), 2002.