NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

Sparse Separation of Under-Determined Speech Mixtures

A dissertation submitted
for the degree of
Doctor of Philosophy

by

Paul D. O'Grady M.Eng., B.Eng.

Supervisor: Prof. Barak A. Pearlmutter

Department of Computer Science
National University of Ireland, Maynooth
Ollscoil na hÉireann, Má Nuad

· June 2007 ·

Contents

Declaration

I hereby declare that this dissertation, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy in Computer Science, is entirely my own work save and to the extent that such work has been cited and acknowledged within the dissertation.

Signed: ..... ID No: 63124637

Date: .....

Acknowledgements

The completion of a Doctor of Philosophy degree represents the culmination of many years of learning and hard work. Thankfully, I have not been alone in this endeavour. To their great credit, the following people (probably unknown to them) have contributed so much to the completion of this thesis.

First, I would like to thank my supervisor Prof. Barak Pearlmutter for his continued support and helpful contributions throughout my Ph.D. studies. Furthermore, I would like to thank the Higher Education Authority of Ireland for funding my research over this time.

Due mention is deserved to all the great people at the Hamilton Institute; my time there has been a truly enlightening and enjoyable experience. This has been due in large part to my lab mates (past and present): Ross Myles O'Neill, Steven Strachan, Stuart Tilbury, Vamsi Potluru, Santiago Jaramillo, Orla McGann, Simon McCann and Fiachra Matthews. Furthermore, I would like to thank Dr Oliver Mason for his helpful suggestions on matrix calculus and in-depth discussions on Gaelic football.

Thanks to all those who attend our DublinICA meetings, particularly Dr Scott Rickard who provided support in the final push to completion of this thesis.

I am greatly indebted to many teachers in the past. Particularly my secondary school teachers Mr Leslie McKay, who instilled in me a great interest in engineering, and Mr Charles Ferguson, who always provided support and encouragement.

I would like to give a special mention to Brian McCarthy and all those at

the Bujinkan Brian Dojo in Dublin, who have taught me that not all things worth learning need be academic.

Great thanks go to Aoife Reilly for proofreading this thesis and for providing very helpful suggestions.

Finally, I would like to give an extra special thanks to all my family for their never-ending support and encouragement. This thesis is dedicated to you!

Abstract

We are all familiar with the shape of sound from our secondary school science classes; the typical oscillatory form of a string under tension that decays over time is widely know. At first sight, this representation of sound imparts to the observer nothing more than its duration and amplitude. So how does the brain separate different sounds given such a representation? Over millions of years the Mammalian auditory cortex has evolved to effectively understand the sounds of its natural environment. By learning the distinguishing features of a sound, the brain can recognise and classify many different sounds. It has long been desired to replicate this ability using a machine, which has been the genesis for a topic of study known as *source separation.*

In this thesis we utilise two contrasting strategies for the Separation of under-determined speech mixtures, *i.e.*, the case where there are more sources than mixtures. Furthermore, we impose a sparseness requirement on the sources.

First, we introduce a blind source separation method called the LOST algorithm, which is based on a Expectation-Maximisation procedure. The LOST algorithm assumes an instantaneous mixing model, and estimates the columns of the mixing matrix by identifying corresponding linear subspaces in a scatter plot. This method combined with a transformation into a sparse domain and an L_1-norm minimisation, constitutes a blind source separation algorithm for the under-determined case, where there are at least two mixtures.

Second, we investigate Convolutive Non-negative Matrix Factorisation,

which is parts-based dimensionality reduction technique for the approximate factorisation of non-negative data. We extend the algorithm by introducing a sparseness constraint on the activations, and derive an algorithm that results in multiplicative updates. We demonstrate how the algorithm can be used to extract convolutive speaker phone sets, which exhibit sparse activations, and utilise such phone sets in supervised separation scheme that separates multiple speakers from a monophonic mixture.

Additionally, we present a perceptual evaluation of speech reconstructions created by a Non-negative Matrix Factorisation algorithm that utilises the beta divergence, and compare the results to a perceptually weighted NMF algorithm.

"Lord grant me the serenity to accept the things I cannot change, the courage to change the things I can, and the wisdom to know the difference."

Saint Francis of Assisi

List of Figures

List of Tables

List of Notation

$\mathbf{A}$	Matrix
$\mathbf{a}_i$	Constituent vector of $\mathbf{A}$ indexed for some purpose
a_{ij}	Element of the matrix $\mathbf{A}$
$[\mathbf{A}]_{ij}$	Element of the matrix $\mathbf{A}$
$\mathbf{a}$	Vector
$\hat{\mathbf{a}}$	Estimate of the vector $\mathbf{a}$
$\mathbf{A}_i$	Matrix indexed for some purpose
$\|\mathbf{A}\|$	L_2-norm of the matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	The inverse of the matrix $\mathbf{A}$
$\mathbf{A}^{+}$	The pseudo inverse of the matrix $\mathbf{A}$
$\mathbf{A}^{\mathsf{T}}$	The transpose of the matrix $\mathbf{A}$
$\mathbf{A}^{\mathsf{H}}$	Hermitian (complex-conjugate transpose) of $\mathbf{A}$
$\bar{\mathbf{A}}$	Matrix rescaled to unit L_2-norm
$\mathbf{\Sigma}_{\mathbf{X}}$	Covariance matrix for $\mathbf{X}$
$\mathbf{x}(t)$	Vector indexed at time t
$\mathbf{a} \cdot \mathbf{b}$	Dot product of two vectors
$P(x)$	PDF of the random variable x
$P(x\|s)$	Conditional PDF of x given s
$H(x)$	Entropy of the random variable x
$\overset{i\rightarrow}{(\cdot)}$	Column shift operator
$\otimes$	Element-wise matrix multiplication
$\|\cdot\|$	Absolute value
$\leftarrow$	Substitute

$\approx$	Approximately equal to
$\propto$	Proportional to
$\langle \cdot \rangle$	Expectation
$\in$	Belongs to
$\mathbb{R}$	The set of real numbers
e^x	Natural exponent of x

CHAPTER 1

Introduction

We live in a world of superposition: When two quantities exist at the same position, the laws of our universe reconcile this fact by calculating a new quantity that is the sum of the two constituent quantities. The consequences of this phenomenon are, that the existence of the constituent quantities is not immediately evident, and all observations of the universe are a mixture of many quantities. This is one of the most rudimentary principles of physics and is continually encountered in our perceptible world. Over our lifetime, the perceptual mixtures we experience are wide and varied. These range from the chaotic cross-modal perception of the world we experience when we are babies, in which the information from disparate senses are themselves mixed, to the more orderly multi-modal view of the world we experience from infancy onwards, in which the information we receive from our senses may be a mixture of many underlying components.

In this context, the most obvious example of superposition is in the perception of sound. The sound percept we experience is ultimately the result of sound pressure waves that emanate from a vibrating object and cause the ear drum to vibrate. Many objects may be the source for such pressure waves at the same time, resulting in superposition of many different vibrations at the ear drum. The ability of the brain to separate the different vibrations that lead to the perception of individual sounds is truly remarkable, and has

[1]Some material in this chapter appeared in O'Grady et al. (2005)

been the subject of study in many different areas of science. In the computer science community, the ability of the brain to separate individual sounds is described metaphorically as the *cocktail party problem* (Cherry, 1953); that is, the separation of individual voices from a myriad of voices in an uncontrolled acoustic environment such as a cocktail party.

All but the most uninteresting of sounds are composed of many individual components; these may include the individual speakers when engaged in conversation, or the individual instruments when listening to a piece of music. The brain is capable of separating the individual components of a sound into distinct auditory objects by exploiting cues such as timbre, location, reverberation and timing. It is obvious that without such cognitive processes our perception of the world would be very limited. Furthermore, these processes also lead to other experiences. Music, for example, is an organised mixture of sound. The brain recognises the instruments present as distinct auditory objects organised in time, from which emerges one of the most profound and enjoyable human experiences.

From a technological viewpoint, the separation of sound into its constituent components has many possible applications. Taking human-computer interaction as an example, if humans are to interact with machines as efficiently as they do with each other, then the most obvious mode of interaction is speech. A necessary pre-processing step for such a mode of interaction would be to separate speech from the cacophony of unwanted sound that exists in a natural environment. Furthermore, by developing technologies that effectively understand perceptual data, a more human-centred system of interaction arises. Other possible applications of source separation include diagnostics and event detection, where speech is replaced as the sound of interest by a sound that characterises some fault or event. Here too, the sound of interest is combined with unwanted sounds, which need to be separated, upon which the machine signals the fault or event.

Now that we have established the importance of unmixing speech, how is it possible to replicate such behaviour using a machine? There are two complementary approaches, these are *auditory scene analysis* (Bregman, 1990) and *Blind Source Separation* (BSS). Auditory scene analysis has its roots in psychoacoustics and endeavours to elucidate the cognitive processes within the brain that enable hearing, which is achieved by observing the response of a subject to specific sound stimuli. Such methods identify individual compo-

nents by replicating the observed perceptual grouping mechanisms in the frequency domain. Blind source separation defines a generative model for source separation, whereby a set of linearly mixed sources are recovered solely from their observed mixtures, without any knowledge about the mixing process. BSS can be achieved by a number of methods including *Independent Component Analysis* (ICA)(Comon, 1994) and *Non-Negative Matrix Factorisation* (NMF) (Lee and Seung, 2001). ICA is an information-theoretic approach that describes independent components in terms of random distributions, where the statistics of these random processes are used to characterise each individual component. In contrast, NMF is a parts-based approach that makes no statistical assumption about the data. Instead, it achieves separation by the factorisation of non-negative data into matrices of an appropriate dimension.

In this thesis, we utilise methods for the separation of mixtures where there are more sources than observations, while focusing our attention on speech data. Two contrasting separation strategies are studied. First, we present an ICA type method for the blind source separation of an arbitrary number of sources, where two or more observations are available. Second, we investigate NMF, which can separate multiple sources from a single (monophonic) mixture. In both methods we utilise a sparseness assumption, and for NMF we investigate the perceptual properties of its reconstruction objective.

In this chapter, we provide some background to the methods that will be used in the remainder of this thesis. In Section 1.1 we present the BSS generative model and statistical preliminaries that are relevant to this thesis. In Section 1.2 we introduce blind source separation and overview a selection of sparse and non-sparse source separation methods. In Section 1.3 non-negative matrix factorisation is discussed, with both conventional and convolutive algorithms being presented. Finally, we conclude with comments on the organisation of the thesis.

1.1 Statistics for Source Separation

In this section, we present the generative model for BSS, and discuss the statistical concepts that will be used by the methods presented in the following chapters.

1.1.1 BSS Generative Model

When presented with a set of observations from sensors such as microphones, the process of extracting the underlying sources is called source separation. Doing so without strong additional information about the individual sources, or constraints on the mixing process, is called blind source separation. The problem is stated as follows: Given M linear mixtures of N sources mixed via an unknown $M \times N$ mixing matrix $\mathbf{A}$, estimate the underlying sources, $\mathbf{S}$, from the mixtures, $\mathbf{X}$. The dimensionality of $\mathbf{A}$ influences the complexity of source separation. If $M = N$, then $\mathbf{A}$ is defined by an even-determined (*i.e.*, square) matrix, and provided that it is non-singular, $\mathbf{S}$ can be estimated by constructing an unmixing matrix, $\mathbf{W} = \mathbf{A}^{-1}$. If $M > N$, then $\mathbf{A}$ is defined by an over-determined (*i.e.*, tall) matrix, and provided that it is full rank, $\mathbf{S}$ can be estimated by least-squares optimisation or linear transformation involving matrix pseudo-inversion. If $M < N$, $\mathbf{A}$ is defined by an under-determined (*i.e.*, fat) matrix. Consequently, source estimation becomes more involved and is usually achieved by some non-linear technique.

Environmental assumptions about the surroundings in which the sensor observations are made also influence the complexity of the problem. Sensor observations in a natural environment are confounded by signal reverberations; consequently the estimated unmixing process needs to identify a source arriving from multiple directions at different times as one individual source. Generally, source separation techniques depart from this difficult real-world scenario, and make less realistic assumptions about the environment in order to make the problem more tractable. Three assumptions are typically made about the environment: The most rudimentary of these is the *instantaneous case*, where sources arrive instantly at the sensors but with differing signal intensity. An extension of this case is the *anechoic case*, where arrival delays between sensors are also considered. The anechoic case can be further extended by considering multiple paths between each source and each sensor, resulting in the *echoic case*.

Generative Model

The generative model for BSS is presented as follows: A set of T observations of M sensors

$$\mathbf{X} = \Big[\mathbf{x}(1)| \quad \cdots \quad |\mathbf{x}(T)\Big] = \begin{bmatrix} x_1(0) & x_1(p) & \cdots & x_1((T-1)p) \\ x_2(0) & x_2(p) & \cdots & x_2((T-1)p) \\ \vdots & \vdots & \ddots & \vdots \\ x_M(0) & x_M(p) & \cdots & x_M((T-1)p) \end{bmatrix}$$

consist of a linear mixture of N source signals

$$\mathbf{S} = \Big[\mathbf{s}(1)| \quad \cdots \quad |\mathbf{s}(T)\Big] = \begin{bmatrix} s_1(0) & s_1(p) & \cdots & s_1((T-1)p) \\ s_2(0) & s_2(p) & \cdots & s_2((T-1)p) \\ \vdots & \vdots & \ddots & \vdots \\ s_N(0) & s_N(p) & \cdots & s_N((T-1)p) \end{bmatrix}$$

by way of an unknown linear mixing process characterised by an $M \times N$ mixing matrix $\mathbf{A}$

$$\mathbf{A} = \Big[\mathbf{a}_1| \quad \cdots \quad |\mathbf{a}_N\Big] = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{MN} \end{bmatrix}$$

yielding the equation

$$\mathbf{x}(t) = \mathbf{A} \star \mathbf{s}(t) + \epsilon(t), \qquad t = 1, \ldots, T,$$

where $\epsilon(t)$ is noise (usually white and Gaussian), p is the sample period and $\star$ denotes the model dependent linear operator. The form of the elements of the mixing matrix, a_{ij}, and the linear operator in the above equation are mixing model dependent, and define whether the mixing process is instantaneous, anechoic or echoic. Table 1.1 presents the linear operators and mixing matrix elements specific to the three cases of blind source separation, where the operator $\delta(t - \delta_{ij})$ is used to denote a delay between source j to sensor i, c_{ij} is a scalar attenuation factor between source j to sensor i, δ_{ij}^k and c_{ij}^k are the delay and attenuation parameters for the k-th arrival path, L is the number

Table 1.1: Mixing Model Specific Linear Operators and Mixing Parameters

Mixing	**Linear Operator**	**Generative Model**	a_{ij}
Instantaneous	Matrix Multiply	$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$	c_{ij}
Anechoic	Delay	$\mathbf{x}(t) = \mathbf{A} * \mathbf{s}(t)$	$c_{ij}\,\delta(t - \delta_{ij})$
Echoic	Convolution	$\mathbf{x}(t) = \mathbf{A} * \mathbf{s}(t)$	$\sum_{k=1}^{L} c_{ij}^{k}\,\delta(t - \delta_{ij}^{k})$

of paths the sources can take to the sensors.

For the purposes of this thesis, we will restrict ourselves to the instantaneous mixing model and generally assume that there is no additive noise,

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t), \qquad t = 1, \ldots, T. \tag{1.1}$$

For $M = N$ the source estimates $\hat{\mathbf{s}}(t)$ are retrieved by,

$$\hat{\mathbf{s}}(t) = \mathbf{W}\mathbf{x}(t), \qquad t = 1, \ldots, T. \tag{1.2}$$

1.1.2 Principal Component Analysis

Principal Component Analysis (PCA) (Pearson, 1901)—also known as the Karhunen-Loève transform or the Hotelling transform—is a technique for the dimensionality reduction of data, which retains the features of the data that contribute most to its variance. PCA is a linear transformation that does not have a fixed set of basis vectors. Instead, PCA transforms the data to a coordinate system that corresponds to the directions of the variance of the data. The coordinate system is orthogonal, where the first coordinate corresponds to the direction of greatest variance, the second corresponds to the direction of second greatest variance and so on. The vectors that define the directions of variance are known as the *principal components* of the data. Reduction in the dimensionality of multi-dimensional data can be achieved by projecting the data onto a subset of its principal components. This subset may be arranged by selecting those principal components that have an associated variance above some threshold. Consequently, the projection conserves the *most interesting* features of the data and provided that the threshold is selected appropriately, the error introduced is low. If the data is projected on to all its principal components, then the input data is decorrelated, *i.e.*, the correlation matrix for the data is a diagonal matrix.

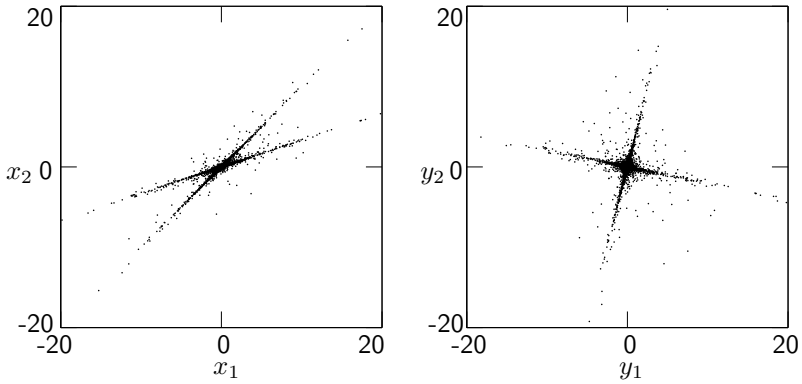

Figure 1.1: Scatter plots of two linear mixtures of two Laplacian distributed sources, before (left) and after (right) PCA. From the left scatter plot it is evident that the principal components (sources) of the data are non-orthogonal and therefore correlated. When PCA is performed the principal components become orthogonal, indicating that the components are uncorrelated.

PCA is presented as follows: Given a set of zero-mean observations, $\langle \mathbf{X} \rangle = 0$, the covariance matrix of the data is

$$\mathbf{\Sigma_X} = \langle \mathbf{X}\mathbf{X}^\mathsf{T} \rangle. \tag{1.3}$$

In order to decorrelate the data we need to perform a transformation on $\mathbf{X}$ that will diagonalise $\mathbf{\Sigma_X}$. This is usually achieved by eigenvector decomposition:

$$\mathbf{\Sigma_X} = \mathbf{U}\mathbf{D}\mathbf{U}^{-1}, \qquad \mathbf{U}^{-1}\mathbf{\Sigma_X}\mathbf{U} = \mathbf{D}, \tag{1.4}$$

where the matrix $\mathbf{U}$ contains the eigenvectors of $\mathbf{\Sigma_X}$ and the diagonal matrix $\mathbf{D}$ contains its associated eigenvalues $\lambda_i \ldots \lambda_N$. Since $\mathbf{\Sigma_X}$ is positive semidefinite, its eigenvectors are orthogonal and $\mathbf{U}^{-1} = \mathbf{U}^\mathsf{T}$. Combining Eq. 1.3 and Eq. 1.4 gives

$$\mathbf{D} = \mathbf{U}^\mathsf{T}\langle \mathbf{X}\mathbf{X}^\mathsf{T} \rangle \mathbf{U} = \langle (\mathbf{U}^\mathsf{T}\mathbf{X}\mathbf{U})(\mathbf{U}^\mathsf{T}\mathbf{X}\mathbf{U})^\mathsf{T} \rangle. \tag{1.5}$$

Therefore, by projecting $\mathbf{X}$ on $\mathbf{U}^\mathsf{T}$, we successfully decorrelate $\mathbf{X}$. Additionally, $\mathbf{X}$ can be normalised by $\mathbf{D}^{-1/2}$ resulting in $\mathbf{\Sigma_X} = \mathbf{I}$, which is referred to as *whitening* the data, *i.e.*, the spectral energy (variance) associated with each principal components is unity. By Assigning $\mathbf{W} = \mathbf{U}\mathbf{D}^{-1/2}\mathbf{U}^\mathsf{T}$ the data is whitened, $\mathbf{Y} = \mathbf{W}\mathbf{X}$.

The effect of PCA on two mixtures of two Laplacian-distributed sources is

presented in Figure 1.1. In the first illustration a scatter plot of the mixtures before PCA is presented. Here, the lines represent the sources contained in the mixtures, which correspond to the principal components of the data. After PCA is applied, the lines are orthogonal, indicating that the principal components (sources) are uncorrelated. It is evident from the scatter plot that PCA successfully decorrelated the sources in the mixtures but failed to separate the sources, as the sources would be separated if the line orientations corresponded to the axes of the scatter plot. This implies that *uncorrelatedness* is not a sufficient criterion for achieving source separation.

1.1.3 Independent Component Analysis

As has been demonstrated, PCA is insufficient for source separation. Although the variates of $\mathbf{Y}$ are decorrelated after PCA, they exhibit *mutual dependence* and the sources remain mixed. This implies that in order to separate $\mathbf{S}$, a stronger assumption of source *independence* needs to be considered; independent component analysis is a method that achieves separation by making such an assumption. ICA (Comon, 1994) originated in the context of blind source separation (Herault and Jutten, 1986) and assumes the same instantaneous generative model presented in Eq. 1.1. The conditions that must be satisfied to guarantee separation are given by Darmois' Theorem (Darmois, 1953), and are stated as follows:

1. The sources are assumed to be mutually independent, which can be described in probabilistic terms as

$$P(\mathbf{s}) = P(s_1, \cdots, s_N) = \prod_{i=1}^{N} P(s_i). \tag{1.6}$$

2. At most one of the independent components can have a Gaussian distribution. This is a requirement because $\mathbf{A}$ cannot be identified for more than one Gaussian source.

ICA provides a linear mapping that factors the joint probability distribution of the sources into independent components. This may be achieved by identifying the rotation needed to separate $\mathbf{S}$ after PCA. Furthermore, there are a number of ambiguities that characterise the ICA solution:

1. The order of the elements within the rows of the estimated $\mathbf{A}$ cannot be determined correctly. This is known as the *permutation ambiguity.*

2. The variances of the independent sources cannot be determined correctly, as both $\mathbf{A}$ and $\mathbf{s}(t)$ are unknown, and any scalar multiplication on $\mathbf{s}(t)$ will be lost in the mixing. This is known as the *scaling ambiguity.*

Therefore, $\mathbf{W} = \mathbf{A}^{-1}$ up to permutation and scaling of the rows.

ICA is generally solved as an optimisation problem, where $\mathbf{W}$ is discovered by maximising some measure of independence. Such measures include mutual information (Comon, 1994), entropy (Bell and Sejnowski, 1995), non-gaussianity (Hyvärinen and Oja, 1997), and sparseness (Zibulevsky and Pearlmutter, 2001). An overview of some ICA methods is presented in Section 1.2.

1.1.4 Sparseness Assumption

One increasingly popular and powerful assumption is that the sources have a parsimonious representation in a given basis. This assumption has come to be known as the *sparseness assumption*: A signal is said to be sparse when it is zero, or nearly zero, more than might be expected from its variance. Such a signal has a probability density function or distribution of values with a sharp peak at zero and fat tails. This shape can be contrasted with a Gaussian distribution, which has a smaller peak and tails that taper quite rapidly (Figure 1.2). A standard sparse distribution is the Laplacian distribution

$$P(c) = \frac{1}{\sqrt{2}}\, e^{-\sqrt{2}|c|}, \tag{1.7}$$

which has led to the sparseness assumption being sometimes referred to as a Laplacian prior. The sparseness of a distribution can be measured by a variety of methods, such as those based on tanh-functions (Karvanen and Cichoki, 2003) and the Gini index (Rickard and Fallon, 2004). However, the most commonly used measure for unimodal symmetric sparse distributions is kurtosis, which is the degree of peakedness of a distribution:

$$\mathrm{kurt}(c) = \frac{\langle (c-\mu)^4 \rangle}{\sigma^4} - 3, \tag{1.8}$$

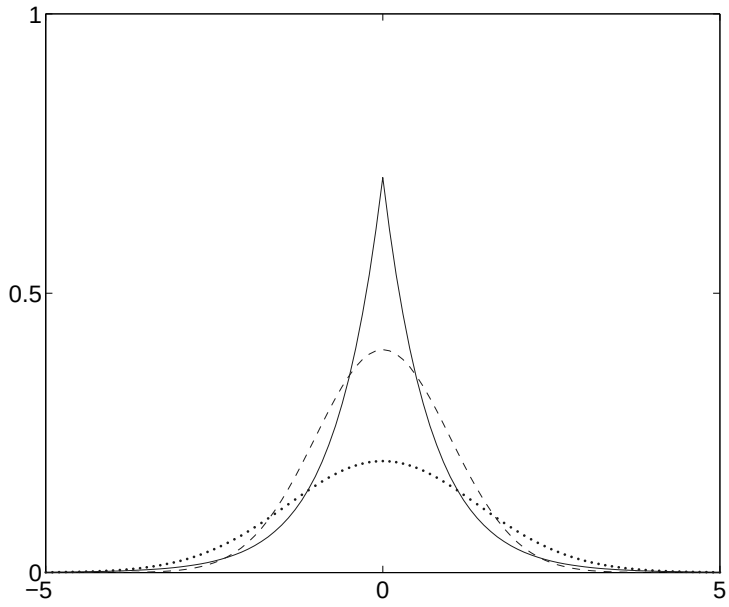

Figure 1.2: A Plot of the probability densities of a selection of random distributions. Solid line: Laplacian distribution; Dashed Line: Gaussian distribution; Dotted Line: Sub-Gaussian distribution.

where a random variable, c, drawn from a super-Gaussian distribution such as the Laplacian has a $\text{kurt}(c) > 0$.

A sparse representation of an acoustic signal can often be achieved by a transformation into a Fourier, Gabor or Wavelet basis. For a set of independent and identically distributed sparse variables, the probability of multiple variables being non-zero simultaneously is low. Thus, sparse representations lend themselves to good separability (Zibulevsky and Pearlmutter, 2001). Additionally, sparseness may be used in many instances to perform source separation in the case when there are more sources than mixtures (Lewicki and Sejnowski, 1998).

Sparse representations have an interpretation in information-theoretic terms, where the representation of a signal using a small number of coefficients corresponds to transmission of information using a code that utilises a small number of bits. Such representations occur in the natural world; in the brain neurons are thought to encode data in a sparse way if their firing pattern is characterised by long periods of inactivity (Földiák and Young, 1995; Körding et al., 2002): Recent work indicates that such firing patterns exist in the auditory cortex, suggesting that the brain encodes sound using a sparse code (DeWeese et al., 2003).

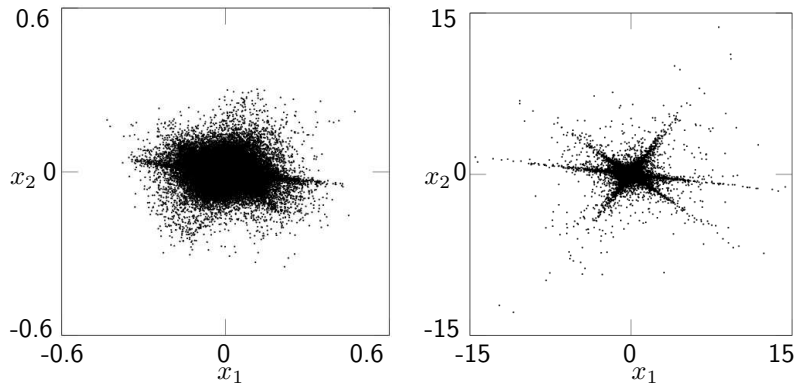

Figure 1.3: Scatter plot of two linear mixtures of three zero-mean sources, in both the time domain (left) and the transform domain (right). The *sparse* transform domain consists of the coefficients of 512-point windowed FFTs. The figures axes are measured in arbitrary units of mixture coefficients.

Unmixing Sparse Sources

For the instantaneous case, $\mathbf{A}$ simply consists of scalars. Taking a simple example where there are three sources and two mixtures, the generative model takes the form

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \end{bmatrix} \tag{1.9}$$

and can be described as a linear mixture of N linear subspaces in M-space. This linear mixing imposes a structure on the resultant mixtures, which becomes apparent when the mixtures have a sparse representation (see Figure 1.3). The existence of this structure can be explained as follows: From Eq. 1.9 it is evident that if only one source is active, say s_1, then the resultant mixtures would be

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} s_1(t),$$

therefore the points on the scatter plot of $x_1(t)$ versus $x_2(t)$ would lie on the line through the origin whose direction is given by the vector $[a_{11}\ a_{21}]^{\mathsf{T}}$. When the sources are sparse, making it unusual for more than one source to be active at the same time, the scatter plot of coefficients constitute a mixture of lines, with the lines broadened due to noise and occasional si-

multaneous activity. These line orientations correspond to the columns of **A**. Therefore, the essence of the sparse approach is the identification of line orientation vectors (also known as basis vectors) from the observed data. In contrast, traditional non-sparse approaches exploit the statistics of the sources as opposed to the structure of the mixtures.

1.2 Blind Source Separation

Blind source separation has received wide attention and has been a topic of investigation for over two decades. The earliest approach traces back to Herault and Jutten (1986) whose goal was to separate an instantaneous linear even-determined mixture of non-Gaussian independent sources. They proposed a solution that used a recurrent artificial neural network to separate the unknown sources, the crucial assumption being that the underlying signals were independent. This early work led to the pioneering adaptive algorithm of Jutten and Herault (1991). Linsker (1989) proposed unsupervised learning rules based on information theory that maximise the average mutual information between the inputs and outputs of an artificial neural network. Comon (1994) proposed that mutual information was the most natural measure of independence and demonstrated that maximising the non-Gaussianity of the source signals was equivalent to minimising the mutual information between them. He also gave the concept of determining underlying sources by maximising independence the name Independent Component Analysis. Bell and Sejnowski (1995) developed a BSS algorithm called *BS-Infomax*, which is similar in spirit to that of Linsker and uses an elegant stochastic gradient learning rule that was proposed by Amari et al. (1996). The idea of non-Gaussianity of sources was used by Hyvärinen and Oja (1997) to develop their *fICA* algorithm. As an alternative approach to separation using mutual information, Gaeta and Lacoume (1990) proposed maximum likelihood estimation, an approach elaborated by Pham et al. (1992). However, Pearlmutter and Parra (1996) and Cardoso (1997) later demonstrated that the BS-Infomax algorithm and maximum likelihood estimation are essentially equivalent. The early years of BSS research concentrated on solutions for even-determined and over-determined mixing processes. It was not until recent years that a solution for the under-determined case was proposed when Belouchrani and Cardoso (1994) presented a *Maximum A Posteriori*

(MAP) probability approach for discrete QAM sources. An approach for sparse sources was later proposed by Lewicki and Sejnowski (1998). The first practical algorithm for separation in an anechoic environment was the *DUET* algorithm, which was initially proposed by Jourjine et al. (2000) and further explored by Yilmaz and Rickard (2004). The first algorithms for the anechoic separation of moving speakers were presented by Rickard et al. (2001) and Anemüller and Kollmeier (2003). A selection of source separation algorithms and their characteristics are presented in Table 2.1.

Blind source separation techniques are not confined to acoustic signals, and have been applied to a wide variety of data types. BSS has been applied to the decomposition of functional brain imaging data such as electroencephalography (Jung et al., 1999, 2000), functional magnetic resonance imaging (McKeown et al., 1998) and magnetoencephalography (Vigário et al., 2000; Tang et al., 2000; Ziehe et al., 2000; Wübbeler et al., 2000; Pearlmutter and Jaramillo, 2003). BSS has also been applied to such diverse areas as real time robot audition (Nakadai et al., 2002), digital watermark attacks (Du et al., 2002) and financial time series analysis (Back and Weigend, 1997; Roth and Baram, 1996). It has even been conjectured that blind source separation will have a role in the analysis of the *cosmic microwave background* (Cardoso et al., 2003), potentially helping to elucidate the very origins of the universe.

1.2.1 Mixing Parameters Estimation

In a staged algorithm approach the first step is to estimate $\mathbf{A}$, the form of which is dependent on the environment considerations and the dimensionality of the problem. The following subsections explore the estimation of $\mathbf{A}$ using sparse and non-sparse methods in the context of instantaneous mixing. For an overview of sparse and non-sparse methods in anechoic and echoic mixing see O'Grady et al. (2005).

Non-Sparse Methods

The following methods do not make the sparseness assumption and instead estimate $\mathbf{A}$ by exploiting the statistics of the observations in the time domain. These methods are usually restricted to the over-determined and even-

Table 1.2: A Comparison of Various Source Separation Techniques (†denotes sparse methods)

Algorithm	**#Sources (N)**		**#Sensors (M)**			**Mixing Model**			**Representation**				**Unmixing**	
	M	$> M$	1	2	> 2	Inst.	Anechoic	Echoic	Time	Freq.	Signal	Dict.	Parameter. Est.	Separation
Wiener (1949) Filter		×	×			×	×	×		×			Power Spectrum Reshaping	Linear Filtering
Herault and Jutten (1986)	×			×	×	×			×				Independence Maximisation	Adaptive Feedback Network
JADE (Cardoso and Souloumiac, 1993)	×			×	×	×			×				Joint Cumulant Diagonalisation	Linear Transformation
ICA (Comon, 1994)	×			×	×	×			×				Mutual Information Minimisation	Linear Transformation
BS-InfoMax (Bell and Sejnowski, 1995)	×			×	×	×			×				Entropy Maximisation	Linear Transformation
Lambert (1995)	×			×	×			×	×				Multichannel Blind Deconvolution	Current Estimate Feedback
Lin et al. (1997) †	×	×		×		×			×				Hough Transform	Hard Assignment
SOBI (Belouchrani et al., 1997)	×			×	×	×			×				Joint Unitary Diagonalisation	Linear Transformation
fICA (Hyvärinen and Oja, 1997)	×			×	×	×			×				Non-Gaussianity Maximisation	Linear Transformation
Lee et al. (1999) †	×	×		×	×	×			×				Gradient Ascent Learning	Maximum a Posteriori
DUET (Jourjine et al., 2000) †	×	×		×		×	×			×			2D Histogram Clustering	Binary Time-Freq Masking
Bofill and Zibulevsky (2000) †	×	×		×	×	×				×			Potential Function Clustering	L_1-norm Minimisation
Zibulevsky and Pearlmutter (2001) †	×	×		×	×	×					×		MAP with Laplacian Prior	Joint Optimisation
Roweis (2001)		×	×			×				×			Hidden Markov Models	Spectral Masking and Filtering
Pearlmutter and Zador (2004) †		×	×					×	×				Known HRTF and Dictionary	L_1-norm Minimisation

determined case.

Entropy Maximisation

The BS-Infomax algorithm (Bell and Sejnowski, 1995) demonstrated that for signals with a positive kurtosis, such as speech, minimising the mutual information between the source estimates, and maximising the joint entropy of the source estimates are essentially the same: Given the *entropy*,

$$H(\mathbf{s}) = -\int P(\mathbf{s}) \log P(\mathbf{s}) d\mathbf{s}, \tag{1.10}$$

the *mutual information* between the sources is

$$I(s_1, s_2, \ldots, s_N) = \sum_{i=1}^{N} H(s_i) - H(\mathbf{s}), \tag{1.11}$$

which is zero if the variables $s_1, s_2, \ldots, s_N$ are independent. For the ICA generative model, the mutual information between the variates of the estimated sources, $\hat{s}$, is given by

$$I(\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_N) = -\sum_{i=1}^{N} H(\hat{s}_i) - H(\mathbf{x}) - \log|\det \mathbf{W}|. \tag{1.12}$$

For BS-Infomax, $\mathbf{Wx}$ is operated on by a transforming function, $\hat{\mathbf{s}} = \phi(\mathbf{Wx})$, which is monotonic, invertible and bounded—this function ensures that $H(\hat{\mathbf{s}})$ is bounded. Separation is achieved by optimising $\mathbf{W}$ such that it minimises the mutual information in Eq. 1.12, which effectively maximises the joint entropy, $H(\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_N)$. Minimisation of Eq. 1.12 can be implemented using stochastic gradient ascent, which results in the following update rule,

$$\mathbf{W} \leftarrow \mathbf{W} + \eta \Delta \mathbf{W}, \tag{1.13}$$

$$\Delta \mathbf{W} \propto (\mathbf{W}^{\mathsf{T}})^{-1} + \phi(\mathbf{Wx})\mathbf{x}^{\mathsf{T}}, \tag{1.14}$$

where η is the learning rate, $\Delta \mathbf{W}$ is the update of $\mathbf{W}$ and $\phi(\cdot)$ is defined as

$$\phi(\mathbf{Wx}) = \phi(\mathbf{u}) = [\phi_1(u_1), \phi_2(u_2), \ldots, \phi_2(u_N)]^{\mathsf{T}} \tag{1.15}$$

$$\phi_i(u_i) = -\frac{1}{P(u_i)}\frac{\partial P(u_i)}{\partial u_i}. \tag{1.16}$$

For super-Gaussian sources, *i.e.*, signals with a high kurtosis, $\phi_i(\mathbf{u}) = -2\tanh(\mathbf{u})$. BS-Infomax was subsequently improved by Amari et al. (1996), who realised that the parameter space is not Euclidean but has a Riemannian metric structure. They proposed a gradient update that better reflected this fact, which resulted in better convergence characteristics and improved adaptation speed. The new rule multiplied Eq. 1.14 by $\mathbf{W}^\mathsf{T}\mathbf{W}$,

$$\Delta\mathbf{W} \propto (\mathbf{I} + \phi(\mathbf{u})\mathbf{u}^\mathsf{T})\mathbf{W}. \tag{1.17}$$

The Infomax principal has also been extended to the anechoic case (Torkkola, 1996).

Cross-Statistics

In addition to the assumption of independence of sources, Parra and Sajda (2003) also consider statistical assumptions related to the structure of neighbouring source samples. The approach taken formalises the problem of finding the unmixing matrix, $\mathbf{W}$, as a generalised eigenvector decomposition of two matrices, which include the covariance matrix of $\mathbf{X}$, $\mathbf{\Sigma_X}$, and an additional symmetric matrix, $\mathbf{\Gamma_X}$. The matrix $\mathbf{\Sigma_X}$ can be described in terms of both $\mathbf{A}$ and the covariance matrix of $\mathbf{S}$, $\mathbf{\Sigma_S}$,

$$\mathbf{\Sigma_X} = \langle \mathbf{X}\mathbf{X}^\mathsf{T} \rangle, \qquad \mathbf{\Sigma_X} = \mathbf{A}\mathbf{\Sigma_S}\mathbf{A}^\mathsf{T}, \tag{1.18}$$

where $\mathbf{\Sigma_S}$ is diagonal if the variates of $\mathbf{S}$ are independent or decorrelated. For non-Gaussian, non-stationary, and non-white sources there exists, in addition to the covariance matrix, other cross-statistics, $\mathbf{\Gamma_X}$, which have the same diagonalisation property,

$$\mathbf{\Gamma_X} = \mathbf{A}\mathbf{\Gamma_S}\mathbf{A}^\mathsf{T}. \tag{1.19}$$

Both these conditions together are sufficient for source separation. In order to determine $\mathbf{W}$, some algebraic manipulation needs to be performed on Eq. 1.18 and Eq. 1.19; by multiplying both by $\mathbf{W}$, and Eq. 1.19 by $\mathbf{\Gamma_S}^{-1}$, the following generalised eigenvalue equation emerges:

$$\mathbf{\Sigma_X}\mathbf{W} = \mathbf{\Gamma_X}\mathbf{W}(\mathbf{\Sigma_S}\mathbf{\Gamma_S}^{-1}). \tag{1.20}$$

The solution provides $\mathbf{W}$ for different statistical assumptions on the sources. This formulation combines subspace analysis and mixing matrix construction into a single computation, making for simple implementation. The form of $\mathbf{\Gamma_X}$ is signal dependant, and is selected appropriately to give the diagonal cross-statistics required to solve Eq. 1.20. For example, if the sources are non-stationary and decorrelated, $\mathbf{\Gamma_X} = \mathbf{\Sigma_X}$ for different periods of stationarity, which results in $\mathbf{W}$ that performs simultaneous decorrelation. If the sources are non-Gaussian and independent, the cross-statistic used is the 4-th order moment, which corresponds to Cardoso's ICA method which also exploits 4-th order moments (Cardoso, 1990).

Second Order Statistics

The *SOBI* algorithm (Belouchrani et al., 1997) exploits the time coherence of the sources and achieves separation by using only second order statistics. The algorithm is based on the unitary diagonalisation of the whitened data covariance matrix, where the time coherence of the original sources is exploited by creating a set of such matrices from observations taken at different time delays, these matrices are then jointly diagonalised. Unitary diagonalisation can be explained as follows: Given a whitening matrix $\mathbf{B}$ and the observations $\mathbf{X}$, which may be complex, the covariance matrix of the whitened observations is

$$\langle \mathbf{B}\mathbf{X}\mathbf{X}^{\mathsf{H}}\mathbf{B}^{\mathsf{H}} \rangle = \mathbf{B}\mathbf{\Sigma_X}\mathbf{B}^{\mathsf{H}}, \tag{1.21}$$

by using Eq. 1.18 this becomes

$$\mathbf{V}\mathbf{A}\mathbf{\Sigma_S}\mathbf{A}^{\mathsf{H}}\mathbf{B}^{\mathsf{H}} = \mathbf{I}, \tag{1.22}$$

where superscript H denotes the complex conjugate transpose of a matrix. The source signals are assumed to have unit variance and are uncorrelated, so $\mathbf{\Sigma_S} = \mathbf{I}$. Eq. 1.22 now states that if $\mathbf{B}$ is a whitening matrix, $\mathbf{VA}$ is an $M \times M$ unitary matrix. It follows that for any whitening matrix $\mathbf{B}$, there exists a unitary matrix $\mathbf{U}$ such that $\mathbf{VA} = \mathbf{U}$. Consequently, $\mathbf{A}$ can be factored as

$$\mathbf{A} = \mathbf{B}^{-1}\mathbf{U}. \tag{1.23}$$

The problem of unmixing is to find a unitary matrix which jointly diagonalises all the covariance matrices generated from the different time delays. In order to achieve diagonalisation, we need a matrix operation that provides a measure of the distance from diagonality. Such a measure is the *off* matrix operator: Given a matrix $\mathbf{M}$ with entries m_{ij}, the off operator is defined as

$$\text{off}(\mathbf{M}) = \sum_{1 \leq i \neq j \leq M} |m_{ij}|^2. \tag{1.24}$$

Subsequently, the unitary diagonalisation of $\mathbf{M}$ is equivalent to reducing $\text{off}(\mathbf{QMQ})$ to zero by some unitary matrix $\mathbf{Q}$. In addition, if the matrix $\mathbf{M}$ is of the form $\mathbf{U}^{\mathsf{H}}\mathbf{D}\mathbf{U}$, where $\mathbf{U}$ is unitary and $\mathbf{D}$ is diagonal, then it may be unitary diagonalised only by unitary matrices that are essentially equal to $\mathbf{U}$, that is if $\text{off}(\mathbf{QMQ}) = 0$, then $\mathbf{Q} = \mathbf{U}$. For the case where $\mathbf{M} = \mathbf{\Sigma_X}$, $\mathbf{Q}$ will contain the principal components of the covariance matrix, which is effectively an eigenvector decomposition of $\mathbf{\Sigma_X}$. Consider a set of K covariance matrices, $\Sigma = \{\mathbf{\Sigma_X}(\tau_1), \ldots, \mathbf{\Sigma_X}(\tau_K)\}$, calculated at different time delays, τ_j for $\tau_1, \ldots, \tau_K$, the joint diagonality criterion for such a set is

$$D(\Sigma, \mathbf{Q}) = \sum_{k=1}^{K} \text{off}(\mathbf{Q}^{\mathsf{H}}\mathbf{\Sigma_X}(\tau_k)\mathbf{Q}). \tag{1.25}$$

A unitary matrix is said to be a joint diagonaliser of the set Σ if it minimises Eq. 1.25 over the set of all unitary matrices. This joint diagonalisation operation can be computed efficiently using a generalisation of the Jacobi technique for the exact diagonalisation of a single Hermitian matrix.

Non-Gaussianity Maximisation

The *Fast-ICA* algorithm of Hyvärinen and Oja (1997) maximises non-Gaussianity as a measure of statistical independence. The utility of the non-Gaussianity criterion as a measure of statistical independence can be explained by the *central limit theorem*, which states that the addition of two or more independent random variables produces a distribution that is more Gaussian than any of the independent variables alone. The algorithm uses *negentropy*, J, to measure the non-Gaussianity of the estimated sources

$$\text{kurt}(\hat{s}) = \langle \hat{s}^4 \rangle - 3\langle \hat{s}^2 \rangle^2, \tag{1.26}$$

$$J(\hat{s}) \approx \frac{1}{12}\langle \hat{s}^3 \rangle^2 + \frac{1}{48}\,\mathrm{kurt}(\hat{s})^2, \tag{1.27}$$

where kurt calculates normalised kurtosis of a random variable. Due to the non-robustness encountered with kurtosis, negentropy is usually calculated using the following approximation, which is based on the maximum entropy principle,

$$J(\hat{s}) \propto \langle G(\hat{s}) \rangle - \langle G(v) \rangle, \tag{1.28}$$

where v is a Gaussian random variable of zero-mean and unit variance, and G is a non-quadratic function. Fast-ICA uses a hierarchal decorrelation to discover the unmixing matrix: The algorithm discovers the first column, $\mathbf{w}_1$, of $\mathbf{W}$ by maximising the non-Gaussianity of the projection $\mathbf{w}_1^T\mathbf{x}$ for the data $\mathbf{x}$, and proceeds to calculate the remaining columns, ensuring that the newly estimated column is orthogonal to those previously estimated. Therefore, a necessary step in the fast-ICA procedure is the pre-whitening of the data. Fast-ICA is typically implemented using a fixed point algorithm. For whitened data the algorithm is derived as follows: The maxima of the approximation of the negentropy of $\mathbf{w}^T\mathbf{x}$ are obtained at certain optima of $\langle G(\mathbf{w}^T\mathbf{x}) \rangle$. According to Kuhn-Tucker conditions, the optima of $\langle G(\mathbf{w}^T\mathbf{x}) \rangle$ under the constraint $\langle G(\mathbf{w}^T\mathbf{x}) \rangle = \|\mathbf{w}\|^2 = 1$ are obtained at the points where

$$F(\mathbf{x}, \mathbf{w}) = \langle \mathbf{x} g(\mathbf{w}^T\mathbf{x}) \rangle - \beta\mathbf{w} = 0, \tag{1.29}$$

where β is a constant and $g(\hat{s}) = dG(\hat{s})/d\hat{s}$. A common choice for $g(\cdot)$ is

$$g(\hat{s}) = \tanh(c\hat{s}), \qquad \text{where } 1 \leq c \leq 2. \tag{1.30}$$

Applying Newton's method to solve Eq. 1.29 results in the Jacobian,

$$\frac{\partial F}{\partial \mathbf{w}} = \langle \mathbf{x}\mathbf{x}^\mathsf{T} g'(\mathbf{w}^T\mathbf{x}) \rangle - \beta\mathbf{I} \approx \mathbf{\Sigma}_\mathbf{x} \langle g'(\mathbf{w}^T\mathbf{x}) \rangle - \beta\mathbf{I} = (\langle g'(\mathbf{w}^T\mathbf{x}) \rangle - \beta)\mathbf{I}. \tag{1.31}$$

Since the data is pre-whitened, $\mathbf{\Sigma}_\mathbf{x} = \mathbf{I}$. The Newton's method update rule is given by

$$\mathbf{w}^+ \leftarrow \mathbf{w} - \left[\frac{\partial F}{\partial \mathbf{w}}\right]^{-1} F$$

$$\mathbf{w}^+ \leftarrow \mathbf{w} - [\langle \mathbf{x} g(\mathbf{w}^T\mathbf{x}) \rangle - \beta\mathbf{w}]/[\langle g'(\mathbf{w}^T\mathbf{x}) \rangle - \beta]. \tag{1.32}$$

Multiplying both sides of Eq. 1.32 by $\beta - \langle g'(\mathbf{w}^T\mathbf{x})\rangle$ gives the following fixed point algorithm

$$\mathbf{w}^+ \leftarrow \langle \mathbf{x} g(\mathbf{w}^\mathsf{T}\mathbf{x})\rangle - \langle g'(\mathbf{w}^\mathsf{T}\mathbf{x})\rangle \mathbf{w}, \tag{1.33}$$

$$\mathbf{w} = \mathbf{w}^+ / \|\mathbf{w}^+\|, \tag{1.34}$$

where normalisation has been introduced for stability. The above algorithm will discover only a single column of $\mathbf{W}$. In order to estimate all the components the algorithm is run N times, ensuring that the new components are orthogonal to those previously estimated.

Sparse Methods

The following methods provide a solution for under-determined blind source separation. For this case, $\mathbf{A}$ is defined by an over-complete set of basis functions, resulting in a matrix that has no inverse. Consequently, ICA methods that have a constraint of orthogonality on $\mathbf{W}$ after whitening, such as those previously discussed, cannot be used for the under-determined case and an alternative approach is needed.

Clustering Approach

Zibulevsky et al. (2002) use a *fuzzy-c-means* clustering method to identify the lines in a scatter plot. The approach exploits the sparseness of speech in the wavelet domain; the multi-scale nature of which provides a number of different classes of coefficients that represent the same signal. Furthermore, each class can be used to create a scatter plot. The selection of the most appropriate class is determined by the sparsity of the coefficients in that class. The clustering procedure is as follows:

1. Normalise the observation vectors to the unit sphere

$$\mathbf{x} = \mathbf{x} / \|\mathbf{x}\|, \tag{1.35}$$

   data points where $\|\mathbf{x}\| \approx 0$ may be removed.

2. Map all data points to the unit half sphere; this mapping is required because the line orientation for each source exists in both hemispheres, producing two clusters for each source. Each cluster pair is consolidated

by this mapping, and line orientations are represented as clusters of data points on the unit hemisphere.[2]

3. A *fuzzy-c-Means* algorithm is used to identify the cluster centres, which are adjoined to create $\hat{\mathbf{A}}$.

A similar approach is presented by Theis et al. (2004)

Bayesian Approach

Lewicki and Sejnowski (1998) formed a Bayesian approach to under-determined BSS. They consider the general case with additive noise,

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \epsilon.$$

Assuming that the noise, ϵ, is Gaussian, the data likelihood is

$$\log P(\mathbf{x}|\mathbf{A}, \mathbf{s}) \propto \frac{1}{2\sigma^2}(\mathbf{x} - \mathbf{A}\mathbf{s})^2, \tag{1.36}$$

where σ^2 is noise variance. Contrary to other ICA methods, Lewicki and Sejnowski's approach estimates $\mathbf{A}$, given an estimate of $\mathbf{s}$. The sources can be estimated by maximising the *a posteriori* value of $\mathbf{s}$:

$$\begin{aligned} \hat{\mathbf{s}} &= \max_{\mathbf{s}} P(\mathbf{s}|\mathbf{x}, \mathbf{A}) \\ &= \max_{\mathbf{s}} P(\mathbf{x}|\mathbf{A}, \mathbf{s})P(\mathbf{s}). \end{aligned} \tag{1.37}$$

Given $\mathbf{A}$ and $\mathbf{x}$, Eq. 1.37 can be optimised by gradient ascent on the log posterior distribution under a Laplacian prior, which is equivalent to L_1-norm Minimisation (will be explained later in Section 1.2.2). The source estimates can then be used to estimate $\mathbf{A}$ by maximising the probability of the data,

$$\max_{\mathbf{A}} P(\mathbf{x}(1), \ldots, \mathbf{x}(T)|\mathbf{A}) = \max_{\mathbf{A}} \prod_{t=1}^{T} P(\mathbf{x}(t)|\mathbf{A}), \tag{1.38}$$

[2]Except when line orientation lies on the equator, in which case the mapping can fail to consolidate its two halves.

which assumes temporal independence. Computation of this likelihood requires marginalising over all possible sources,

$$\max_{\mathbf{A}} P(\mathbf{x}(t)|\mathbf{A}) = \max_{\mathbf{A}} \int P(\hat{\mathbf{s}})P(\mathbf{x}(t)|\mathbf{A}, \hat{\mathbf{s}})d\hat{\mathbf{s}}. \tag{1.39}$$

For an under-determined mixing process, this integral is intractable. However, an approximation can be made by fitting a multivariate Gaussian around $\hat{\mathbf{s}}$. The update for $\mathbf{A}$ can then be derived by performing gradient ascent on the log of Eq. 1.38. The resulting update is

$$\Delta\mathbf{A} \propto \mathbf{A}\mathbf{A}^{\mathsf{T}} \frac{\partial}{\partial \mathbf{A}} \log P(\mathbf{x}|\mathbf{A}) \approx -\mathbf{A}(\phi(\hat{\mathbf{s}})\hat{\mathbf{s}}^{\mathsf{T}} + \mathbf{I}), \tag{1.40}$$

where $\phi(\cdot)$ represents the activation function, which is typically chosen to be $\tanh(\cdot)$ assuming a sparse prior. It is worth noting that the update resembles the BS-Infomax update of Eq. 1.14. The algorithm can be summarised as follows:

1. Randomly initialise $\mathbf{A}$.
2. Initialise source estimates $\hat{\mathbf{s}}$ with the pseudo-inverse, $\mathbf{A}^{+}$.
3. Use the estimated $\hat{\mathbf{s}}$ to calculate a new estimate for $\mathbf{A}$,

$$\mathbf{A} \leftarrow \mathbf{A} - \eta\mathbf{A}(\phi(\hat{\mathbf{s}})\hat{\mathbf{s}}^{\mathsf{T}} + \mathbf{I}), \tag{1.41}$$

   where η is the learning rate.
4. Given the estimate for $\mathbf{A}$, recalculate the source estimates using L_1-norm Minimisation.
5. Repeat steps 3 & 4 until convergence.

2D Techniques

Feature detection techniques from image processing have also been used to locate line orientations. Lin et al. (1997) present an algorithm that uses a Hough transform to identify lines. In contrast to the clustering approach used in under-determined BSS, *edge features* are used to identify line orientations:

1. The mixture domain data is partitioned into bins, creating an image representation of a scatter plot.

2. The image is convolved with an edge detection operator, and the resultant image is normalised and thresholded to form a binary image, which now contains only edge information.

3. This image is Hough transformed using line orientations as the feature of interest, and the line orientation vectors of the scatter plot are identified as peaks in the Hough transform space. The line orientation parameters associated with each peak in Hough space are combined to form $\hat{\mathbf{A}}$.

Monaural Separation

For the convolutive case, a biologically inspired technique that exploits spectral cues is presented in Pearlmutter and Zador (2004). When sound reaches an organism's inner ear, the sound's spectrum is *coloured* by the head and the shape of the ears. This spectral colouring or filtering is know as the *Head-Related Transfer Function* (HRTF), and is defined by the direction of the sound and the acoustic properties of the ear. When using the HRTF it is assumed that each source has a unique position in space. The auditory scene is segregated into a number of different locations, with each having a different HRTF filter. Sources coming from these locations will be coloured by its associated filter, which indicates the source's position. Thus, identification of the HRTF filter applied to each source will lead to separation of sources. This can be achieved by sparsely representing the sources in an over-complete (under-determined) dictionary which is performed as follows:

1. A monaural recording is made using a microphone setup that includes a moulding of a Pinna around the microphone—this is required to perform the necessary spectral filtering of the signal.

2. A known HRTF is used with a given N element signal dictionary, where each dictionary element is filtered by the HRTF filter. This procedure is repeated for F HRTFs, one for each location in the auditory scene, resulting in an over-complete signal dictionary containing $N \times F$ elements.

3. The monaural mixture signal is decomposed into the dictionary elements by L_1-norm Minimisation. The energy of the resultant coefficients indicate the location of the sources in the signal; the dictionary

elements of the locations identified are then scaled by their coefficients and linearly combined to create the estimated sources.

By estimating the coefficients using a post-HRTF (sensor space) dictionary and reconstructing using a pre-HRTF (source space) dictionary, separation and deconvolution can be simultaneously achieved.

1.2.2 Separation Techniques

Subsequent to the estimation of the $\mathbf{A}$, separation of the underlying sources can be performed. The complexity of the separation process is influenced by the mixing model used, and the relative number of sources and sensors. This section presents a number of techniques used in the separation stage of blind source separation algorithms.

Linear Unmixing

Separation in the even-determined case can be achieved by a linear transformation using $\mathbf{W}$,

$$\hat{\mathbf{s}}(t) = \mathbf{W}\mathbf{x}(t), \qquad t = 1, \ldots, T, \tag{1.42}$$

where $\hat{\mathbf{s}}(t)$ holds the estimated sources at time t, $\hat{\mathbf{A}}$ is the estimated mixing matrix and $\mathbf{W} = \hat{\mathbf{A}}^{-1}$ up to permutation and scaling of the rows. For over-determined mixing, where $M > N$, the pseudo-inverse can be used,

$$\hat{\mathbf{s}}(t) = \mathbf{A}^{+}\mathbf{x}(t), \tag{1.43}$$

where $\mathbf{A}^{+} = (\hat{\mathbf{A}}^{\mathsf{T}}\hat{\mathbf{A}})^{-1}\hat{\mathbf{A}}^{\mathsf{T}}$ is the Moore-Penrose pseudo-inverse.

L_1-norm Minimisation

For the under-determined case, which is usually restricted to sparse methods, a linear transformation is not possible since $\hat{\mathbf{A}}\hat{\mathbf{s}}(t) = \mathbf{x}(t)$ has more unknowns in $\mathbf{s}$ than knowns in $\mathbf{x}$, and is therefore non-invertible. Furthermore, the Moore-Penrose pseudo-inverse, which corresponds to the minimum L_2-norm solution, cannot be used as it is unable to remove the correlation between samples, and separate the mixtures. Consequently, some non-linear technique is needed to estimate the sources. These techniques usually involve

assigning observed data $\mathbf{x}(t)$ to the columns of $\hat{\mathbf{A}}$, which characterise each source. The most rudimentary technique is to hard assign each data point to only one source based on some measure of proximity to columns of $\hat{\mathbf{A}}$ (Vielva et al., 2000, 2002; Jourjine et al., 2000; Lin et al., 1997). A logical extension of this is the partial assignment of each data point to multiple sources. If the sources are sparse, it is desirable that the data point assignment results in one significantly non-zero coefficient, with the other coefficient values being close to zero. Such assignment schemes are usually formulated as optimisation problems, where the required behaviour is specified by enforcing a constraint on the norm, $\|\mathbf{c}\|_p = (\sum_{i=1}^{n} |c_i|^p)^{\frac{1}{p}}$, of the solution. The effect of using different norms is illustrated in Figure 1.4. Here, we can see three non-orthogonal vectors in 2D space, each black dot is generated by one significantly non-zero coefficient, which results in linear clouds of data points around each vector. Taking the red dot (farthest point along the centre vector) as the data point under consideration, a minimum solution using a L_2, L_1 and L_0 constraint is illustrated using red arrows. For the L_2 case it is evident that three significantly non-zero vectors are present, this solution does not reflect the structure of the data and is not sparse. Ideally, a sparse representation will have only one active coefficient; the L_0-norm can be used to measure the active coefficients in a vector. For the L_0 case the minimum solution is represented by a significantly non-zero and a close to zero vector, which produces a sparse representation. Unfortunately, minimisation of the L_0-norm is *NP-complete* and cannot be computed in polynomial time. A computationally tractable alternative is to use L_1-norm minimisation, which produces similarly sparse solutions.

L_1-norm minimisation—sometimes referred to as basis pursuit (Chen et al., 1998) or the shortest-path algorithm (Bofill and Zibulevsky, 2000)—is a piecewise linear operation that partially assigns the energy of $\mathbf{x}(t)$ to the M columns of $\hat{\mathbf{A}}$ that form a cone around $\mathbf{x}(t)$ in $\mathbb{R}^M$ space, with the remaining $N - M$ columns assigned zero coefficients. When the number of sources active at any one time is less than or equal to M, more accurate source estimates are produced. L_1-norm minimisation can be accomplished by formulating the problem as a linear program,

$$\arg \min_{\hat{\mathbf{s}}(t) \in \mathbb{R}^N} \|\hat{\mathbf{s}}(t)\|_1 \text{ subject to } \hat{\mathbf{A}}\hat{\mathbf{s}}(t) = \mathbf{x}(t), \qquad t = 1, \ldots, T, \tag{1.44}$$

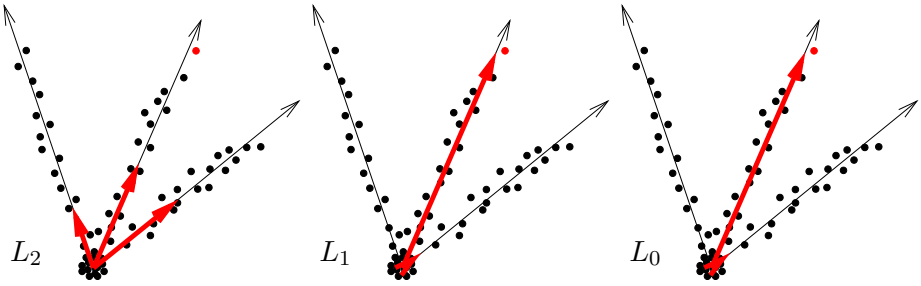

Figure 1.4: L_1-norm minimisation: Computationally tractable way to find a sparse representation. Example: 3 non-orthogonal basis vectors (thin black arrows), each black dot generated by one significantly non-zero coefficient c_i. 3 basis vectors in 2D therefore many possible solutions. Red vectors: solution found for red point. Constraints minimise norm of c: Right: L_0-norm (NP-complete); Left: L_2-norm (not sparse). Centre: L_1-norm (efficient and sparse).

where the observations $\mathbf{x}(t)$ are in a sparse domain and the coefficients $\hat{\mathbf{s}}(t)$, properly arranged, constitute the estimated sources, $\hat{\mathbf{S}} = [\hat{\mathbf{s}}(1) \,|\, \cdots \,|\, \hat{\mathbf{s}}(T)]$.

An illustration of the L_1-norm minimisation of speech mixtures in both the sparse STFT (real coefficients) and non-sparse time domain is presented in Figure 1.5. In this example there are 2 mixtures of 3 sources (see Appendix A). The sources are plotted in 3D space, and sparse and non-sparse scatter plots of the mixtures are presented. Each datapoint is partially assigned to the two column vectors that create a cone around that data point. In the non-sparse scatter plot it is evident that much of the data has no specific bias towards its encompassing column vectors, whereas in the sparse domain the data is clustered around those vectors. The effect of the sparseness of the mixtures can be observed from the plots of their minimum L_1-norm solutions. For the non-sparse mixtures, it is evident that L_1-norm minimisation results in a linear embedding of an M-dimensional space into the higher N-dimensional space. For the sparse mixtures, there is an embedding of a 1-dimensional space into the higher N-dimensional space, where the embedding is along the direction of the non-orthogonal column vectors of $\mathbf{A}$. In this case each linear subspace represents the coefficients for each individual source, which results in better estimates as can be seen from Figure 1.6. For complex data, such as FFT coefficients, the real and imaginary parts are treated separately, thus doubling the number of coefficients. Alternatively, L_1-norm minimisation of complex data can be solved using *second order conic programming*.

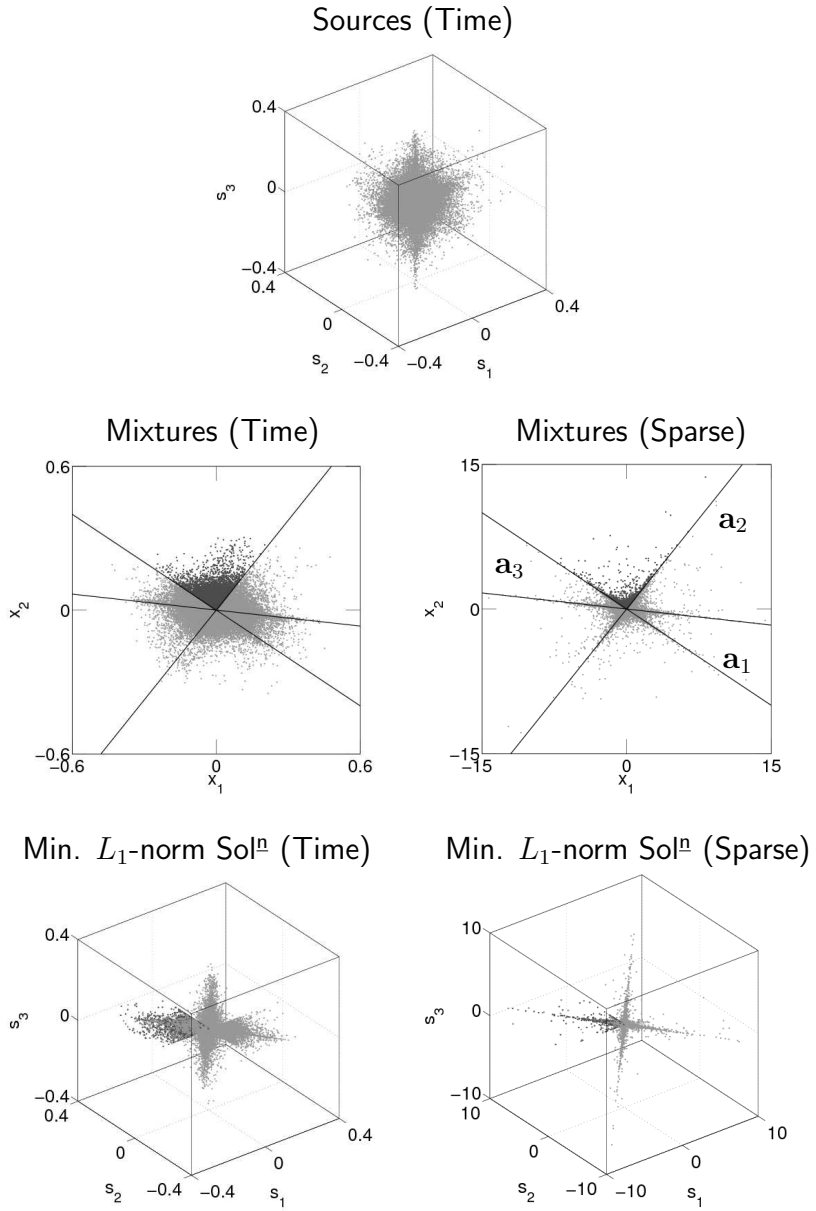

Figure 1.5: Illustration of L_1-norm minimisation for sparse and non-sparse mixtures. Top: 3D plot of 3 sources in the time domain; Centre: Scatter plots of the 2 mixtures in both a non-sparse and sparse domain, the column vectors of $\mathbf{A}$ intersect to form 6 distinct regions; Bottom: L_1-norm minimisation performs a linear embedding of each region into N-dimensional space. For the sparsely represented mixtures, the minimum L_1-norm solutions form a linear subspace along the direction of the non-orthogonal column vectors of $\mathbf{A}$, which results in better source estimates.

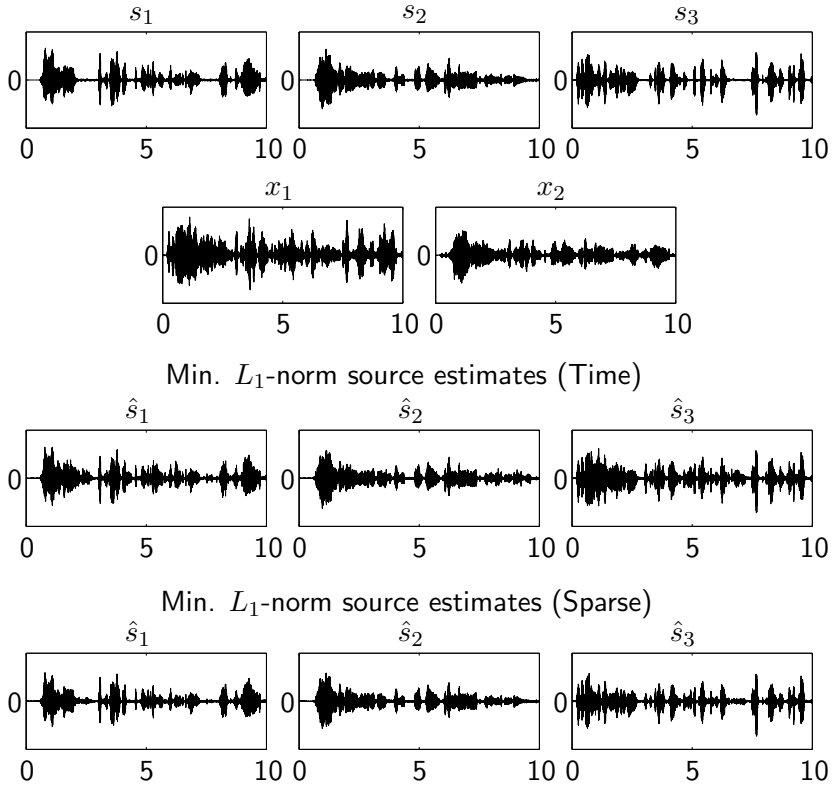

Figure 1.6: Source estimates using L_1-norm minimisation and the original mixing matrix for 2 mixtures of 3 speech sources. The estimated sources produced by L_1-norm minimisation on the mixtures in the time domain are presented in the third row, while the estimates produced in the sparse domain are presented in the fourth row. The quality of the estimates are measured using the signal-to-noise ratio and are as follows: Non-sparse; 4.86 dB, 10.18 dB, & 4.60 dB. Sparse: 9.90 dB, 15.10 dB & 9.49 dB. It is evident that the estimates calculated in the sparse domain provide better results

1.3 Non-negative Matrix Factorisation

Non-negative Matrix Factorisation (NMF) is a technique for the decomposition of multivariate data (Paatero and Tapper, 1994; Lee and Seung, 2001). The NMF algorithm has a generative model that is similar to instantaneous BSS, the difference being an additional non-negativity constraint on the data. NMF is a parts-based approach that makes no statistical assumption about the data. Instead, it assumes that for the domain at hand—for example grey-scale images—negative numbers are physically meaningless. The ICA decomposition of a grey-scale image may result in basis vectors that have both positive and negative components. The image is represented by a linear combination of these ICA basis vectors weighted by both positive and negative coefficients, with some basis vectors being cancelled out by others.

Negative basis components have no real-world representation in a grey-scale image context, which has led researchers to argue that the search for a basis should be confined to a non-negative basis. Formally, this idea can be interpreted as decomposing a non-negative matrix $\mathbf{V}$ into two non-negative factors $\mathbf{W}$ and $\mathbf{H}$. The lack of statistical assumptions makes it difficult to prove that NMF will give correct decompositions. However, it has been shown geometrically that NMF provides a correct decomposition for some classes of images (Donoho and Stodden, 2004).

Data that contains negative components, *e.g.* audio, must be transformed into a non-negative form before NMF can be applied. Here, we use the magnitude spectrogram. Spectrograms have been used in audio analysis for many years (Potter et al., 1947) and in combination with NMF have been applied to a variety of problems such as speech separation (Virtanen, 2003; FitzGerald et al., 2006; Smaragdis, 2004) and automatic transcription of music (Abdallah and Plumbley, 2004).

1.3.1 Conventional NMF

Given a non-negative matrix $\mathbf{V} \in \mathbb{R}^{\geq 0, M \times T}$, the goal is to approximate $\mathbf{V}$ as a product of two non-negative matrices $\mathbf{W} \in \mathbb{R}^{\geq 0, M \times R}$ and $\mathbf{H} \in \mathbb{R}^{\geq 0, R \times T}$,

$$\mathbf{V} \approx \mathbf{WH}, \qquad v_{ik} \approx \sum_{j=1}^{R} w_{ij} h_{jk}, \tag{1.45}$$

where $R \leq M$, such that the reconstruction error is minimised. Two NMF algorithms were introduced by Lee and Seung (2001), each optimising its own measure of reconstruction quality: These quality measures are the Euclidean distance,

$$D(\mathbf{V}, \mathbf{W}, \mathbf{H}) = \frac{1}{2} \|\mathbf{V} - \mathbf{WH}\|^2, \tag{1.46}$$

and a generalised version of the Kullback-Leibler divergence,

$$D(\mathbf{V} \| \mathbf{W}, \mathbf{H}) = \sum_{ik} \Big(v_{ik} \log \frac{v_{ik}}{[\mathbf{WH}]_{ik}} - v_{ik} + [\mathbf{WH}]_{ik} \Big). \tag{1.47}$$

NMF is formulated as an optimisation problem that minimises the proceeding objectives,

$$\min_{\mathbf{W}, \mathbf{H}} \; D(\mathbf{V} \| \mathbf{W}, \mathbf{H}) \qquad \mathbf{W}, \mathbf{H} \geq 0,$$

NMF with Euclidean distance

```
Obj=0.5*sum(sum((V-W*H).^2));
```

```
W=W.*(V*H')./(W*H*H'+1e-9);
H=H.*(W'*V)./(W'*W*H+1e-9);
```

NMF with Kullback-Leibler divergence

```
Obj=sum(sum((V.*log((V./(W*H+1e-9))+1e-9))-V+W*H));
```

```
W=W.*((V./(W*H+1e-9))*H')./(ones(M,1)*sum(H'));
H=H.*(W'*(V./(W*H+1e-9)))./(sum(W)'*ones(1,T));
```

Figure 1.7: Matlab notations for NMF algorithms.

both are convex in $\mathbf{W}$ and $\mathbf{H}$ individually but not together. Therefore algorithms usually alternate updates of $\mathbf{W}$ and $\mathbf{H}$. The objectives can be minimised using a diagonally rescaled gradient descent algorithm (Lee and Seung, 2001), which leads to the following multiplicative updates for the Euclidean distance objective,

$$w_{ij} \leftarrow w_{ij} \frac{[\mathbf{V}\mathbf{H}^{\mathsf{T}}]_{ij}}{[\mathbf{W}\mathbf{H}\mathbf{H}^{\mathsf{T}}]_{ij}}, \qquad h_{jk} \leftarrow h_{jk} \frac{[\mathbf{W}^{\mathsf{T}}\mathbf{V}]_{jk}}{[\mathbf{W}^{\mathsf{T}}\mathbf{W}\mathbf{H}]_{jk}}, \tag{1.48}$$

and Kullback-Leibler divergence,

$$w_{ij} \leftarrow w_{ij} \frac{\sum_{k=1}^{T}(v_{ik}/[\mathbf{W}\mathbf{H}]_{ik})h_{jk}}{\sum_{k=1}^{T} h_{jk}}, \quad h_{jk} \leftarrow h_{jk} \frac{\sum_{i=1}^{M} w_{ij}(v_{ik}/[\mathbf{W}\mathbf{H}]_{ik})}{\sum_{i=1}^{M} w_{ij}}, \tag{1.49}$$

where Matlab notations for the update rules are presented in Figure 1.7. Alternatively, optimisation methods such as multiplicative exponentiated gradient descent can also been used (Cichocki et al., 2006). As each algorithm iterates, their factors converge to a local optimum of Eq. 1.46 and Eq. 1.47.

The parameter R, which is the number of columns in $\mathbf{W}$ and rows in $\mathbf{H}$, specifies the rank of the approximation. If $R < M$ then $\mathbf{W}$ is over-determined and NMF reveals low-rank features of the data. The columns of $\mathbf{W}$ contain the basis for the data while the rows of $\mathbf{H}$ contain activation patterns for each basis. The selection of an appropriate value for R usually requires prior knowledge, and is important to obtaining a satisfactory decomposition.

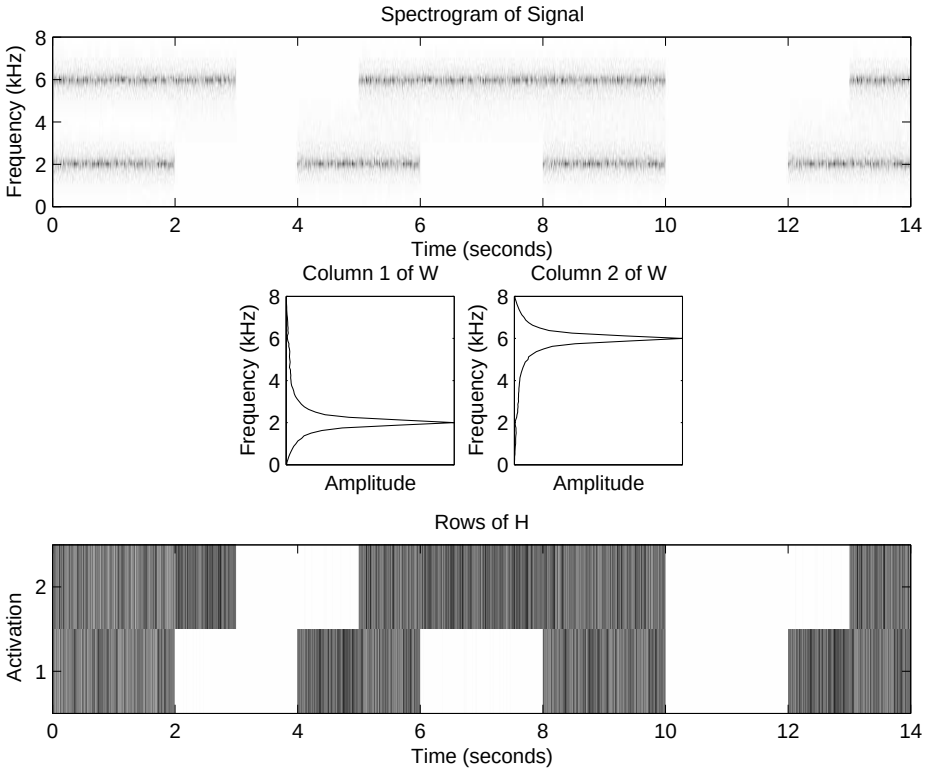

Figure 1.8: Spectrogram of a signal composed of band-limited noise bursts, and its factors obtained by NMF using the KLD objective.

NMF Applied to Audio Spectra

To illustrate the application of NMF to audio data, consider the example shown in Figure 1.8. The signal under consideration is composed of two band-limited noise bursts with magnitude spectra constant over time. The first burst is centred around 2 kHz and occurs four times, while the second burst is centred around 6 kHz and occurs three times. The signal's spectrogram is an $M \times T$ matrix $\mathbf{V}$ with magnitude information for M frequency bins at T time intervals. NMF is applied to $\mathbf{V}$ with $R = 2$ and the resultant factors shown. In this example, both the frequency spectra of the bursts (columns of $\mathbf{W}$) and their activations in time (rows of $\mathbf{H}$) have been identified. Therefore, this decomposition has successfully revealed the structure of $\mathbf{V}$ by correctly describing its constituent elements in both the frequency and time domains.

Now consider the example presented in Figure 1.9. Here, the signal under consideration is composed of two auditory objects that have different frequency sweeps over time. The first object is centred around 2 kHz and

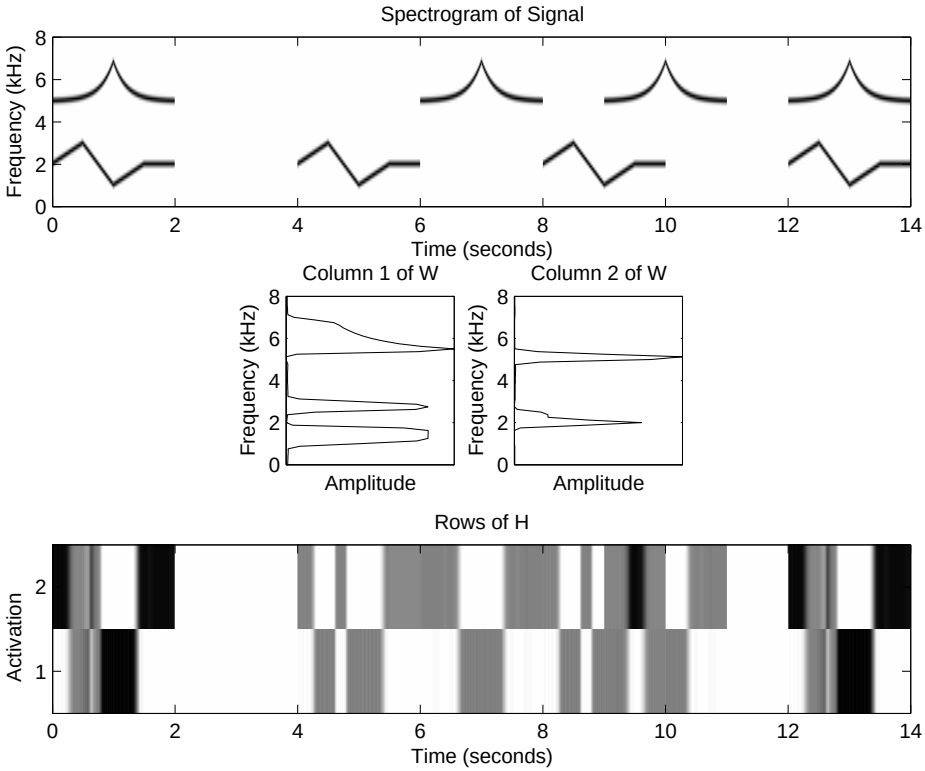

Figure 1.9: Spectrogram of a signal composed of auditory objects with time-varying spectra, and its factors obtained by NMF.

the second object is centred around 6 kHz, each occurring four times. NMF is applied to the data with the same parameters as above, and the resultant factors are shown. It is evident from the columns of $\mathbf{W}$ that the identified spectra contain frequency components that are centred around both 2 kHz and 6 kHz. Thus, NMF fails to identify the spectrum of each object, and instead discovers objects that are a combination of both. The reason for this is that NMF is not expressive enough to reveal the temporal structure of auditory objects that evolve over time. Therefore, in order to reveal a correct decomposition, the expressive properties of NMF need to be extended to consider the evolution of each object's spectrum.

1.3.2 Convolutive NMF

Typically, the temporal relationship between multiple observations over nearby intervals of time are discovered using a convolutive generative model. Such a model has previously been used to extend ICA (Lambert, 1996) and

NMF (Smaragdis, 2004), the latter constituting the algorithm we review in this section. For conventional NMF each object is described by its spectrum and corresponding activation in time, while for convolutive NMF (also known as *Non-Negative Matrix Deconvolution* (NMD)) each object has a sequence of successive spectra and corresponding activation pattern across time. The model of Eq. 1.45 is extended to the convolutive case

$$\mathbf{V} \approx \sum_{t=0}^{T_o-1} \mathbf{W}_t \overset{t\rightarrow}{\mathbf{H}}, \qquad v_{ik} \approx \sum_{t=0}^{T_o-1} \sum_{j=1}^{R} w_{ijt} (\overset{t\rightarrow}{h}_{jk}), \tag{1.50}$$

where $\mathbf{V} \in \mathbb{R}^{\geq 0, M \times T}$ is the input to be decomposed, $\mathbf{W}_t \in \mathbb{R}^{\geq 0, M \times R}$ and $\mathbf{H} \in \mathbb{R}^{\geq 0, R \times T}$ are its two factors, and T_o is the length of each spectrum sequence; the j-th column of $\mathbf{W}_t$ describes the spectrum of the j-th object t time steps after the object has begun. The function $\overset{i\rightarrow}{(\cdot)}$ denotes a column shift operator that moves its argument i places to the right; as each column is shifted off to the right the leftmost columns are zero filled. Conversely, the $\overset{\leftarrow i}{(\cdot)}$ operator shifts columns off to the left, with zero filling on the right:

$$\mathbf{D} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix} \quad \overset{0\rightarrow}{\mathbf{D}} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix} \quad \overset{1\rightarrow}{\mathbf{D}} = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 0 & 5 & 6 & 7 \end{bmatrix}$$
$$\overset{3\rightarrow}{\mathbf{D}} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 2 \end{bmatrix} \quad \overset{\leftarrow 2}{\mathbf{D}} = \begin{bmatrix} 3 & 4 & 0 & 0 \\ 7 & 8 & 0 & 0 \end{bmatrix} \quad \overset{\leftarrow 3}{\mathbf{D}} = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 \end{bmatrix}, etc...$$

Using the Kullback-Leibler divergence, the new objective function for the convolutive generative model is

$$D(\mathbf{V} \| \mathbf{\Lambda}) = \sum_{ik} \Big(v_{ik} \log \frac{v_{ik}}{[\mathbf{\Lambda}]_{ik}} - v_{ik} + [\mathbf{\Lambda}]_{ik} \Big) \tag{1.51}$$

where $\mathbf{\Lambda}$ is the approximation to $\mathbf{V}$ and is defined as

$$\mathbf{\Lambda} = \sum_{t=0}^{T_o-1} \mathbf{W}_t \overset{t\rightarrow}{\mathbf{H}}.$$

This new objective can be viewed as a set of T_o conventional NMF operations that are summed to produce the final result. Consequently, as opposed to updating two matrices ($\mathbf{W}$ and $\mathbf{H}$) as in conventional NMF, $T_o + 1$ matrices require an update ($\mathbf{W}_0, \ldots, \mathbf{W}_{T_o-1}$ and $\mathbf{H}$). The resultant convolutive NMF

Convolutive NMF

```
Obj=sum(sum((V.*log((V./(lambda+1e-9))+1e-9))-V+lambda));
```

```
for t=1:To
    Vt(:,:,t)=(W(:,:,t)*padshift(H,t-1));
end
lambda=sum(Vt,3);

for t=1:To
    Hs=padshift(H,t-1);
    W(:,:,t)=W(:,:,t).*((V./(lambda+1e-9))*Hs')./(ones(size(V))*Hs');
end

for t=1:To
    Qs=padshift(V./lambda,-(t-1));
    Ht(:,:,t)=H.*(W(:,:,t)'*Qs)./(W(:,:,t)'*ones(size(V)));
end
H=mean(Ht,3);
```

Figure 1.10: Matlab notations for convolutive NMF.

update equations are

$$w_{ijt} \leftarrow w_{ijt} \frac{\sum_{k=1}^{T} (v_{ik}/[\mathbf{\Lambda}]_{ik}) \overset{t\rightarrow}{h}_{jk}}{\sum_{k=1}^{T} \overset{t\rightarrow}{h}_{jk}}, \qquad h_{jk} \leftarrow h_{jk} \frac{\sum_{i=1}^{M} w_{ijt} (v_{ik}/\overset{\leftarrow t}{[\mathbf{\Lambda}]}_{ik})}{\sum_{i=1}^{M} w_{ijt}}, \quad (1.52)$$

with Matlab notations presented in Figure 1.10. At every iteration, both $\mathbf{H}$ and $\mathbf{W}_t$ are updated for each t. It is worth noting that $\mathbf{W}_t$ for $t = 0, \ldots, T_o$ is a tensor and contains a separate $\mathbf{W}$ for each t, while a shifted version of $\mathbf{H}$ is shared across all t. It is possible to update $\mathbf{W}_t$ and $\mathbf{H}$ at each t, however this is not advisable as it results in a biased estimate of $\mathbf{H}$, with the $t = T_o - 1$ update dominating over the others (Smaragdis, 2004). A more correct scheme is to update $\mathbf{H}$ to the average result of its updates for all t,

$$h_{jk} \leftarrow \left\langle h_{jk} \frac{\sum_{i=1}^{M} w_{ijt} (v_{ik}/\overset{\leftarrow t}{[\mathbf{\Lambda}]}_{ik})}{\sum_{i=1}^{M} w_{ijt}} \right\rangle, \forall t. \quad (1.53)$$

Convolutive NMF Applied on Audio Spectra

We have shown that conventional NMF reveals a correct decomposition for auditory objects with constant spectra, but fails for objects that exhibit

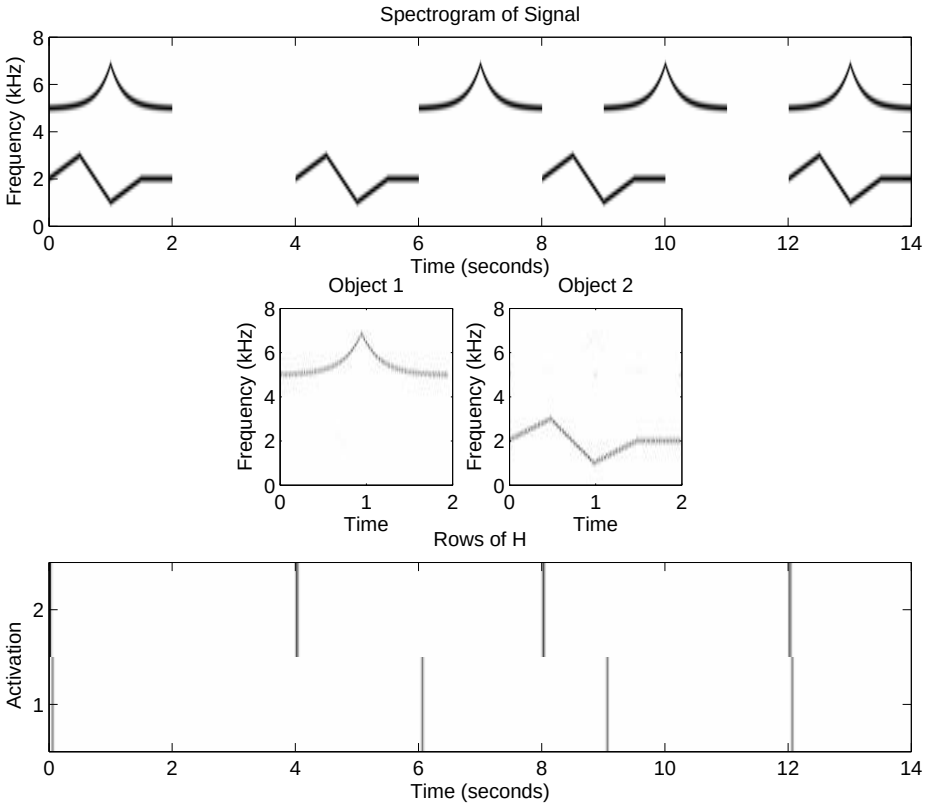

Figure 1.11: Spectrogram of a signal composed of auditory objects with time-varying spectra, and its factors obtained by convolutive NMF.

time-varying spectra. Let us now consider the application of convolutive NMF to this example. The performance of the algorithm now depends on two parameters R and T_o, where T_o must be larger than the temporal extent of each object. Convolutive NMF is applied to the data with $R = 2$ and $T_o = 2$ seconds, and the resultant factors are presented in Figure 1.11. It is evident from the spectral sequences obtained (j-th column of $\mathbf{W}_t$, for $t = 0, 1, \ldots, T_o - 1$) that the time-varying spectrum of each object is revealed, and that the rows of $\mathbf{H}$ identify the start of each object. Therefore, the decomposition has successfully revealed the structure of $\mathbf{V}$ by correctly describing the spectral evolution of each object and its position in time.

1.3.3 NMF Extensions

The previously discussed NMF algorithms can be extended to enforce additional constraints on either the discovered basis or activations patterns. This may be achieved by combining the NMF reconstruction objective with ad-

ditional cost functions that characterise the required constraints. The most widely used method for such multi-objective optimisation is the weighted sum method. This method creates an aggregate objective function by multiplying each constituent cost function by a weighting factor and summing the weighted costs,

$$J = w_1 J_1 + w_2 J_2 + \cdots + w_K J_K,$$

where w_i $(i = 1, \cdots, k)$ is a weighting factor for the i-th cost function J_i, and J is the sum of weighted costs. Combining the NMF reconstruction objective with a constraint on both $\mathbf{H}$ and $\mathbf{W}$ results in an objective of the following form,

$$J(\mathbf{V}\|\mathbf{W},\mathbf{H}) = D(\mathbf{V}\|\mathbf{W},\mathbf{H}) + w_{\mathbf{W}} f_{\mathbf{W}}(\mathbf{W}) + w_{\mathbf{H}} f_{\mathbf{H}}(\mathbf{H}), \tag{1.54}$$

where $f_{\mathbf{W}}(\cdot)$ and $f_{\mathbf{H}}(\cdot)$ are functions that enforce the required constraints, and $w_{\mathbf{W}}$ and $w_{\mathbf{H}}$ specify the weighting of each cost function. Additional constraints that have been introduced to NMF include sparseness (Hoyer, 2002) and temporal continuity (Virtanen, 2003).

1.4 Organisation and Overview

The focus of this thesis is the separation of under-determined speech mixtures by utilising sparseness. Additionally, we also investigate the perceptual properties of the NMF reconstruction objective. We employ two contrasting approaches to separation: First, we introduce a modified *Expectation Maximisation* (EM) procedure that separates under-determined speech mixtures, in the case when there are two or more mixtures. Second, we apply convolutive non-negative matrix factorisation with a sparseness constraint, to the problem of speaker separation from a monophonic mixture. An overview of the remaining chapters in this thesis are presented as follows:

Chapter 2: We present a blind source separation algorithm that estimates the mixing matrix for an under-determined mixing process. The algorithm exploits the sparseness of speech in the short time Fourier transform domain, in which a scatter plot of the mixtures reveals linear subspaces that cross the origin, such subspaces characterise the columns of the

mixing matrix. The mixture of lines is expressed as a Laplacian mixture model and the model parameters are estimated using an expectation-maximisation procedure. Furthermore, the sparseness of the sources are exploited when estimating the source estimates, by using L_1-norm minimisation.

Chapter 3: We investigate the perceptual quality of the NMF algorithm's reconstructions when applied to speech data. Here, we investigate the properties of the beta divergence reconstruction objective. The algorithm is tested for a range of β values, and the perceptual quality of the reconstructions are measured using the noise-to-mask ratio. In order to indicate the usefulness of our results, we also present an NMF algorithm that uses the noise-to-mask ratio as its objective.

Chapter 4: We introduce a convolutive NMF algorithm that includes a sparseness constraint on the activations. In contrast to previous methods, multiplicative updates are achieved for both $\mathbf{H}$ and $\mathbf{W}$. The algorithm is used to extract sparse phone sets from speech, which demonstrate superior performance over convolutive NMF when used in a monophonic speaker separation task. Furthermore, the extracted phones also exhibit superior coding efficiency.

Chapter 5: We conclude this thesis with an overview of the work presented and propose future directions.

CHAPTER 2

The LOST Algorithm

The ubiquity of stereophonic recordings, for example CD recordings, and the separation of such into constituent speakers or instruments, provides the most obvious example of how Blind Source Separation methods can be applied to the modern world. Such recordings are usually under-determined, where there are more than two sources in the recording. The separation of audio mixtures requires that some sort of implicit or explicit assumption be made about the sources and/or mixing process. For the under-determined case, the structure imposed by the mixing process is typically exploited in separation.

Here, we focus our attention on the blind source separation of under-determined instantaneous speech mixtures, where we encounter a mixture of oriented lines. BSS is described as follows: A set of M sensor observations, $\mathbf{X} = [\mathbf{x}(1)|\cdots|\mathbf{x}(T)]$, consist of a linear mixture of N source signals, $\mathbf{S} = [\mathbf{s}(1)|\cdots|\mathbf{s}(T)]$, by way of an unknown linear mixing process characterised by an $M \times N$ mixing matrix $\mathbf{A}$, *i.e.* $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$. When $M = N$ the underlying sources, $\mathbf{S}$, can be separated if one can find an unmixing matrix $\mathbf{W}$ such that $\hat{\mathbf{s}}(t) = \mathbf{W}\mathbf{x}(t)$, where $\hat{\mathbf{s}}(t)$ holds the estimated sources at time t and $\mathbf{W} = \mathbf{A}^{-1}$ up to permutation and scaling of the rows.

When the sources are sparse, a scatter plot of the mixtures reveals a structure composed of linear subspaces that cross the origin; these linear

[3]Some material in this chapter appeared in O'Grady and Pearlmutter (2004a)

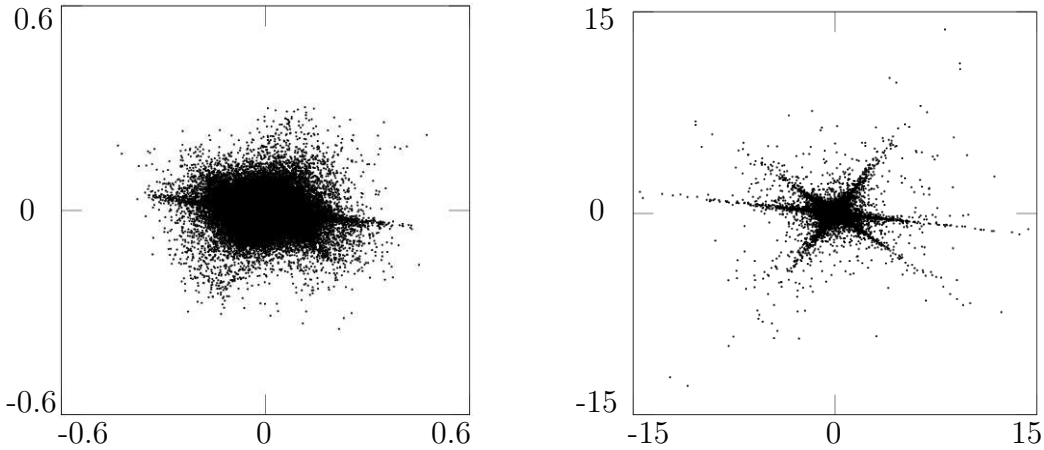

Figure 2.1: Scatter plot of two linear mixtures of three zero-mean speech sources, in both the time domain (left) and the transform domain (right). The *sparse* transform domain consists of the real coefficients of a 512-point windowed STFT. The figures axis are measured in arbitrary units of mixture coefficients.

subspaces correspond to the columns of $\mathbf{A}$. Therefore, if these lines can be estimated from the data, an estimate of the mixing matrix, $\hat{\mathbf{A}}$, can be trivially constructed. Furthermore, such a structure is evident for under-determined mixtures, which makes identification of $\mathbf{A}$ possible for this difficult case. For speech, a sparse representation can often be achieved by a transformation into a suitable domain such as the Short Time Fourier Transform (STFT) domain.

We introduce the LOST (*Line Orientation Separation Technique*) algorithm, which separates under-determined speech mixtures by identifying lines in a scatter plot using a *Laplacian Mixture Model* (LMM). The parameters of the LMM are estimated using an *Expectation-Maximisation* (EM) procedure, and the sources are estimated using L_1-norm minimisation. Furthermore, the LOST algorithm also separates even-determined and over-determined mixtures.

This chapter is organised as follows: In Section 2.1 we discuss the identification of overlapping linear subspaces in a scatter plot and present the LOST algorithm. In Section 2.2 we investigate the general separation performance of the algorithm, and provide an empirical assessment of the algorithms robustness to noise. Furthermore, we compare the performance of the LOST algorithm to that of the geoICA algorithm. We complete the chapter with a discussion and conclusion.

2.1 Oriented Lines Separation

It can be seen from the scatter plot of Figure 2.1 that the columns of $\mathbf{A}$, which represent the sources, manifest as linear subspaces that cross the origin in a sparse domain. Furthermore, it is evident that the points in each linear subspace are drawn from a distribution that is concentrated around the line. Such a distribution resembles a multivariate Laplacian density that is centred along the line. Since there are N sources, $s_i, \ldots, s_N$, each characterised by a different Laplacian density, the observations $\mathbf{x}(t)$ are generated by a linear combination of these Laplacian densities, which is known a Laplacian Mixture Model (LMM). By fitting an LMM to the observed density $P(\mathbf{x})$, the linear subspaces are identified by the Laplacian density centres.

2.1.1 Laplacian Mixture Model

The Laplacian density may be expressed by

$$\mathcal{L}(v|\gamma, x) = \gamma e^{-2\gamma|x-v|} \propto e^{-\gamma|x-v|}, \tag{2.1}$$

where v represents the centre of the Laplacian and γ controls the boundary of the density. The Laplacian density is expressed in terms of the absolute difference from the centre. For our purposes, the centre of the Laplacian, $\mathbf{v}$, and the observation $\mathbf{x}(t)$, are vectors that represent lines that cross the origin. Therefore, a metric that measures the distance between such lines is required; this is achieved by calculating the difference between $\mathbf{x}(t)$ and the projection of $\mathbf{x}(t)$ onto $\mathbf{v}$:

$$q_{it} = \|\mathbf{x}(t) - (\mathbf{v}_i \cdot \mathbf{x}(t))\, \mathbf{v}_i\|, \tag{2.2}$$

where $\cdot$ denotes the dot product. When the Laplacian centre and observation are coincident, q_{it} is at its minimum. We characterised each linear subspace by the following distribution,

$$\mathcal{L}(q_{it}, \gamma) = e^{-\gamma q_{it}}, \tag{2.3}$$

and define the LMM as

$$P(\mathbf{x}(t)) = \sum_{i}^{N} \mathcal{L}(q_{it}, \gamma) = \sum_{i}^{N} e^{-\gamma q_{it}}, \tag{2.4}$$

where γ is the same for each distribution.

2.1.2 LMM Parameter Estimation

Here, we describe the procedure used to estimate the parameters of the specified LMM: Since there are N lines, each with a different orientation vector $\mathbf{v}_i$, the observations are segregated into sets associated with each line. Segregation is achieved by estimating the probability of an observation belonging to a line,

$$\tilde{q}_{it} = P(\mathbf{x}(t)|\mathbf{v}_i) = \frac{e^{-\gamma q_{it}}}{\sum_{i'} e^{-\gamma q_{i't}}}, \tag{2.5}$$

where $\tilde{q}_{it}$ indicates the membership of the observation $\mathbf{x}(t)$ to the line $\mathbf{v}_i$. Calculating the probability of $\mathbf{x}(t)$ for all $\mathbf{v}_i$, represents a partial or *soft assignment* of the observation to each line, which can be contrasted with a hard (winner-takes-all) assignment where each observation is assigned to just one line. The data set associated with each line can be calculated using the observations, $\mathbf{X}$, and their soft assignments $\tilde{q}_{it}, \forall\, i, t$.

The orientation of a linear subspace can be thought of as the direction of its greatest variance. One method that can be used to determine the variance of a data set, and its direction, is Principal Component Analysis (Pearson, 1901). PCA is a dimensionality reduction technique that represents a data set by the variance of the data in orthogonal directions. The principal component with the largest variance, $\lambda_{\max}$, which corresponds to the principal eigenvector, $\mathbf{u}_{\max}$, of the covariance matrix for the weighted observations

$$\mathbf{\Sigma}_i = \mathbf{U}_i \mathbf{\Lambda}_i \mathbf{U}_i^{-1}, \tag{2.6}$$

identifies the centre of the line,

$$\mathbf{v}_i = \mathbf{u}_{\max}, \tag{2.7}$$

where the columns of the matrix $\mathbf{U}_i$ contain the eigenvectors of $\mathbf{\Sigma}_i$ and the

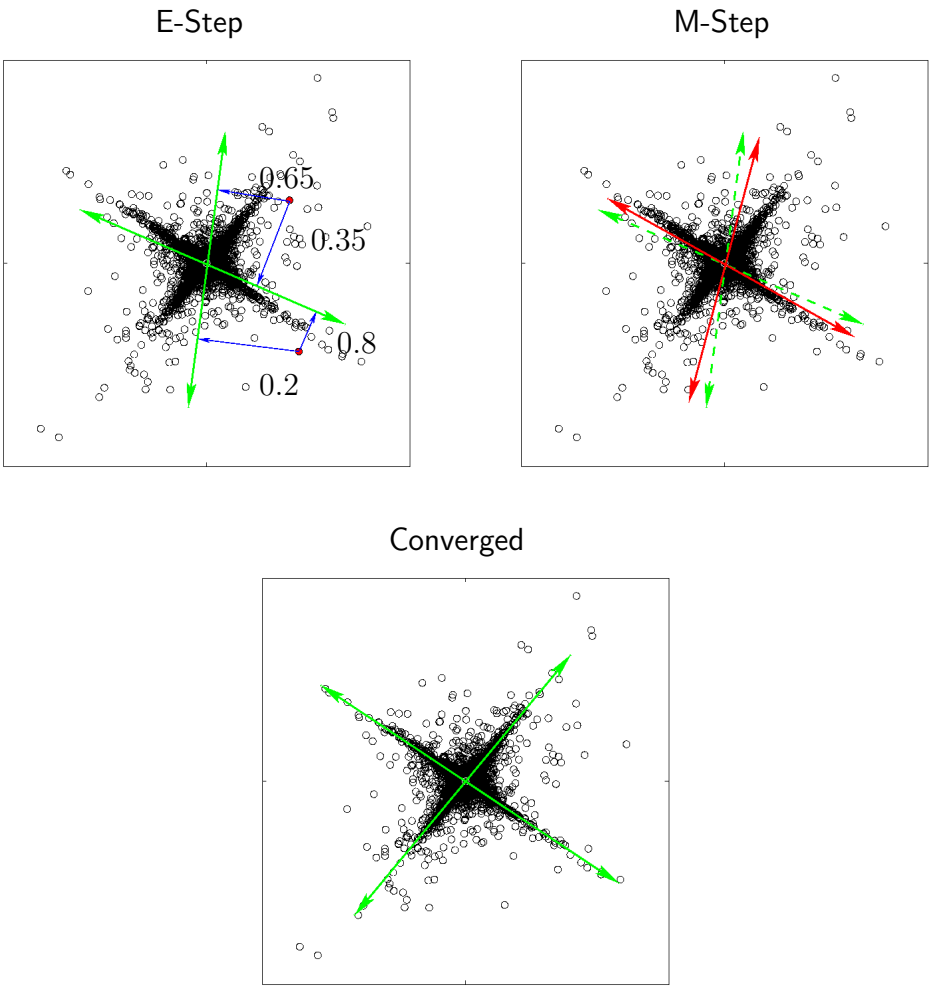

Figure 2.2: Illustration of the LOST algorithm's line estimation procedure: The E-step calculates posterior probabilities partially assigning data points to line orientation estimates, and the M-step repositions the line orientation estimates to the points assigned to them. After convergence the estimated line orientations coincide with the linear subspace directions in the scatter plot.

diagonal matrix $\mathbf{\Lambda}_i$ contains its associated eigenvalues $\lambda_i, \ldots, \lambda_M$. A similar approach to cluster centre re-estimation using Singular Value Decomposition is presented in Aharon et al. (2006), while an alternative approach that fits a straight line to the data points in a linear subspace is presented in Babaie-Zadeh et al. (2004).

The density boundary parameter γ represents the spread of the densities centred on each line. It is obvious from Figure 2.1 that such a spread may be represented by the variance of the linear subspace that is orthogonal to the line, *i.e.*, the second largest eigenvalue of $\mathbf{\Lambda}_i$. We estimate the value of

γ using a scheme that creates a set of second largest eigenvalues for all $\mathbf{\Sigma}_i$, and update γ to the reciprocal of the largest value in this set.

The procedure of soft assignment and line centre repositioning using PCA is repeated until $\mathbf{v}_i$ converge, at which point $\hat{\mathbf{A}}$ is constructed by adjoining the estimated line orientations to form the columns of the matrix

$$\hat{\mathbf{A}} = \left[\mathbf{v}_1 \middle| \cdots \middle| \mathbf{v}_N\right].$$

Such a procedure is an Expectation-Maximisation algorithm (Dempster et al., 1976), which finds maximum likelihood estimates of parameters in probabilistic models, where the model depends on unobserved latent variables. The EM algorithm alternates between an expectation (E) step, which calculates an expectation of the latent variables, and a maximisation (M) step, which calculates the maximum likelihood estimates of the parameters by maximising the expected likelihood found on the E-step. The parameters found on the M-step are then used to begin another E-step, and the process is repeated. In our case, the E-step calculates posterior probabilities assigning observations to lines and the M-step repositions the lines to match the points assigned to them. This EM procedure comprises the line estimation stage of the LOST algorithm (O'Grady and Pearlmutter, 2004b), and is illustrated in Figure 2.2.

Alternatively, the line estimation stage of the LOST algorithm can be thought of as a piecewise linear operation, where observations are soft assigned to lines, and PCA is performed for the data partially assigned to each line.

2.1.3 Sparse Transformation

In order for the linear subspaces in the scatter plot to be well defined, an appropriate sparse transformation is required. For the LOST algorithm, we exploit the sparseness of speech in the Short Time Fourier Transform (STFT) domain, which results in well defined lines (Figure 2.3). However, it is evident that some observations are perturbed by noise, broadening the lines. It is necessary that the lines are as well defined as possible, as the line estimation stage of the LOST algorithm is dependent on the quality of the sparse representation.

The broadening of the lines may be reduced by controlling the effects

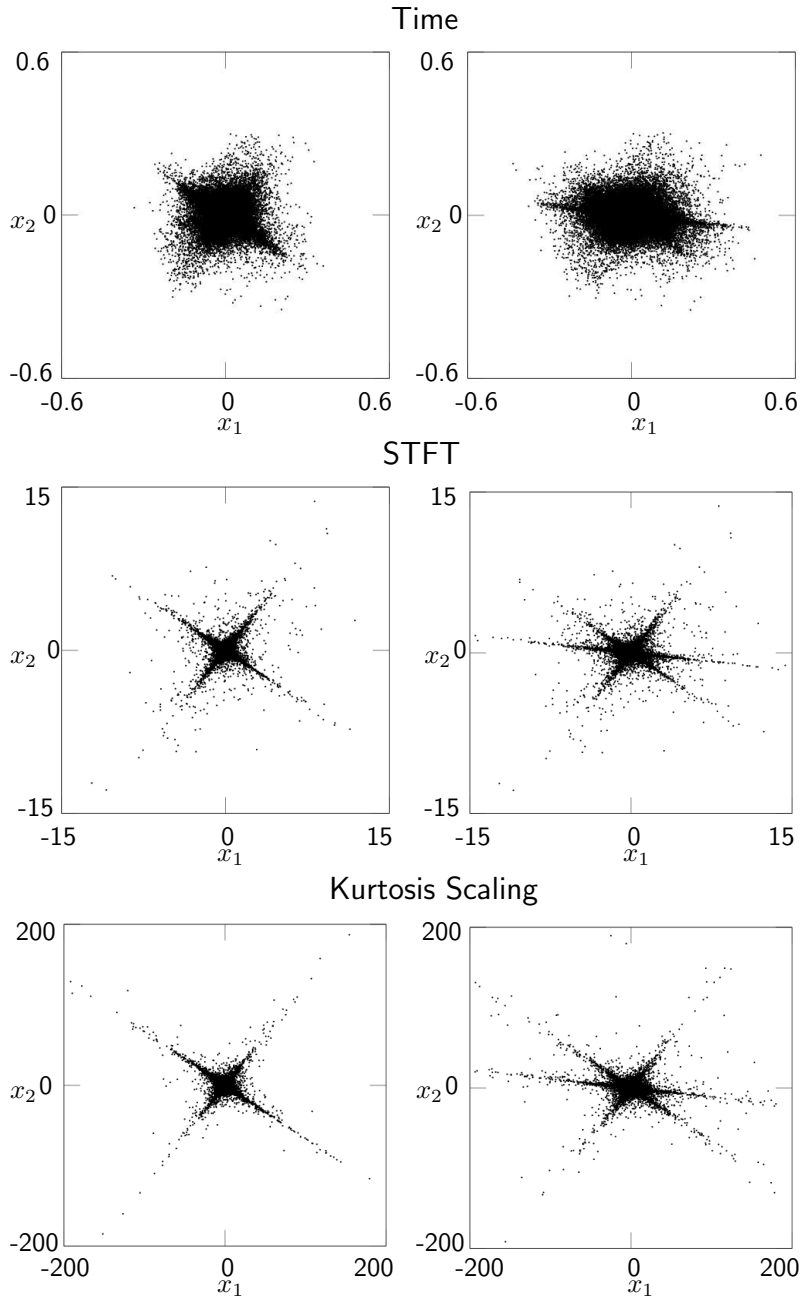

Figure 2.3: Scatter plots for two mixtures of two sources and two mixtures of three sources in the time domain (top), real coefficients of the 512-point STFT domain (middle) and kurtosis weighted STFT domain (bottom). It can be seen that the kurtosis scaled STFT domain produces the best defined lines, which is especially evident for the two mixtures of two sources scatter plot. The figures axes are measured in arbitrary units of mixture coefficients.

of the perturbing noise, which may be achieved by segregating the STFT coefficients into different classes based on some notion of noise level. Here, we examine the levels of noise present in each frequency bin over all STFT frames. Since speech is sparse in the STFT domain, we can assume that fre-

quency bins that have a distribution of coefficients that reflect a Gaussian are mostly noise, while frequency bins that exhibit a Laplacian distribution contribute most to the definition of the lines; the distinguishing feature between the two distributions being their *peakedness*. We measure the peakedness of the distribution of coefficients for each bin using kurtosis,

$$\text{kurt}(c_k) = \frac{\langle (c_k - \mu)^4 \rangle}{\sigma^4} - 3, \tag{2.8}$$

where c_k is the distribution of coefficients for the k-th frequency bin. Each bin is subsequently scaled by its kurtosis, $\text{kurt}(c_k)$. Weighting the frequency bins that have a Laplacian distribution of values greater than those that have a Gaussian pushes those observations away from the origin while pulling the noisy observations toward the origin, resulting in better defined lines and improved line estimates.

The effects of kurtosis scaling are illustrated in Figure 2.3. It can be seen that the kurtosis weighted STFT domain produces the best defined lines, which is especially evident for the two mixtures of two sources scatter plot. The effectiveness of kurtosis scaling is discussed in Section 2.2.3.

2.1.4 Source Unmixing

As discussed in Section 1.2.2, the dimensionality of $\hat{\mathbf{A}}$ determines the procedure used to estimate $\hat{\mathbf{s}}$. Therefore, so as to be applicable to separation problems that exhibit an arbitrary number of sources and mixtures, the LOST algorithm supports three different source-unmixing methods. For the even-determined case, where $M = N$, $\hat{\mathbf{A}}$ is square and the data points can be assigned to line orientations using $\hat{\mathbf{s}}(t) = \mathbf{W}\mathbf{x}(t)$. When there are more observations than sources, *i.e.*, the over-determined case ($M > N$), data points can be assigned to sources by finding the least squares solution. When $M < N$, the under-determined case, $\mathbf{A}$ is not invertible therefore $\mathbf{S}$ needs to be estimated by some other means. One technique is so-called *hard assignment* of coefficients (Rickard and Dietrich, 2000; Roweis, 2001; Vielva et al., 2000, 2002; Lin et al., 1997). Another is partial assignment, where each coefficient can be decomposed into more than one source. This is generally done by minimisation of the L_1-norm, which can be seen as a maximum likelihood reconstruction under the assumption that the coefficients are drawn from a

Laplacian distribution—this being the method used by the LOST algorithm.

2.1.5 The LOST Algorithm Summary

The following is a summary of the LOST algorithm, describing both line orientation estimation and source unmixing.

Line Estimation

1. Create a scatter plot of $\mathbf{X}$ in a sparse domain: Transform the observations, $x_1, \ldots, x_M$, using an STFT and perform kurtosis scaling of the coefficients; the transformed observations are subsequently plotted against each other.

2. Randomly initialise the N line orientation vectors $\mathbf{v}_i$, and initialise γ to a sufficiently large value.

3. Partially assign each observation, $\mathbf{x}(t)$, to each line orientation vector, $\mathbf{v}_i$, using a soft data assignment:

$$\begin{aligned} q_{it} &= \|\mathbf{x}(t) - (\mathbf{v}_i \cdot \mathbf{x}(t))\,\mathbf{v}_i\|^2, \\ \tilde{q}_{it} &= \frac{e^{-\gamma q_{it}}}{\sum_{i'} e^{-\gamma q_{i't}}}, \end{aligned} \tag{2.9}$$

   where γ controls the boundary between the regions attributed to each line, and $\tilde{q}_{it}$ are the computed weightings of observation at time t for each line i.

4. Calculate the covariance matrix for the weighted observations assigned to each line. The covariance matrix expression and assignment weightings are combined as follows:

$$\mathbf{\Sigma}_i = \frac{\sum_t \tilde{q}_{it}(\mathbf{x}(t) - \mu)(\mathbf{x}(t) - \mu)^T}{\sum_t \tilde{q}_{it}}, \tag{2.10}$$

   where μ is a vector of the mean values of the rows of $\mathbf{X}$, which is typically zero for speech, and $\mathbf{\Sigma}_i$ is the covariance of weighted observations

associated with line i.

5. Update the line orientation estimates to the principal eigenvector of each covariance matrix: The eigenvector decomposition of $\mathbf{\Sigma}_i$ is

$$\mathbf{\Sigma}_i = \mathbf{U}_i \mathbf{\Lambda}_i \mathbf{U}_i^{-1}, \tag{2.11}$$

where the columns of the matrix $\mathbf{U}_i$ contain the eigenvectors of $\mathbf{\Sigma}_i$ and the diagonal matrix $\mathbf{\Lambda}_i$ contains its associated eigenvalues $\lambda_i, \ldots, \lambda_M$. The new line orientation vector estimate is the principal eigenvector of $\mathbf{\Sigma}_i$,

$$\mathbf{v}_i \leftarrow \mathbf{u}_{\max} \tag{2.12}$$

where $\mathbf{u}_{\max}$ is the principal eigenvector, *i.e.*, the eigenvector with the largest eigenvalue, $\lambda_{\max}$.

6. Update γ using the variances that are orthogonal to the direction of the lines: Select the second largest eigenvalue from each diagonal matrix $\mathbf{\Lambda}_i$, and update to the reciprocal of the largest eigenvalue from this set,

$$\gamma \leftarrow \frac{1}{\max(\lambda_i, \ldots, \lambda_M)}, \tag{2.13}$$

where λ_i is the second largest eigenvalue of $\mathbf{\Sigma}_i$. Return to step 3 and repeat until $\mathbf{v}_i$ converge.

7. After convergence, adjoin the line orientations estimates to form $\hat{\mathbf{A}}$,

$$\hat{\mathbf{A}} = \left[\mathbf{v}_1 \middle| \cdots \middle| \mathbf{v}_N\right]. \tag{2.14}$$

As discussed in Section 1.2, previous methods for mixing matrix estimation include: *fuzzy C-means* clustering (Zibulevsky et al., 2002), kernel methods (Bofill and Zibulevsky, 2001), clustering using topographic maps (van Hulle, 1999), feature extraction using the Hough transformation (Lin et al., 1997), joint unitary diagonalisation (Belouchrani et al., 1997), entropy maximisation (Bell and Sejnowski, 1995) and independence maximisation (Herault and Jutten, 1986).

Source Unmixing

1. Perform line estimation to calculate $\hat{\mathbf{A}}$.

2. (a) Even-determined case: Source estimates are calculated using linear transformation,

$$\hat{\mathbf{s}}(t) = \mathbf{W}\mathbf{x}(t), \qquad t = 1, \ldots, T,$$

where $\mathbf{W} = \hat{\mathbf{A}}^{-1}$.

(b) Over-determined case: Source estimates are calculated by finding the least squares solution,

$$\text{minimise } \|\hat{\mathbf{A}}\hat{\mathbf{s}}(t) - \mathbf{x}(t)\|_2, \qquad t = 1, \ldots, T.$$

(c) Under-determined case: Source estimates are calculated using L_1-norm minimisation[4] for each observation in the sparse STFT domain such that

$$\arg \min_{\hat{\mathbf{s}}(\omega) \in \mathbb{R}^N} \|\hat{\mathbf{s}}(\omega)\|_1 \text{ subject to } \hat{\mathbf{A}}\hat{\mathbf{s}}(\omega) = \mathbf{x}(\omega).$$

Subsequent to which, an inverse transformation is performed, $\hat{\mathbf{s}}(\omega) \mapsto \hat{\mathbf{s}}(t)$.

3. The final result is an $N \times T$ matrix $\hat{\mathbf{S}}$ that contains the source estimates, $\hat{s}_1, \ldots, \hat{s}_N$, in each row.

2.2 Experiments

To demonstrate the effectiveness of the LOST algorithm, we investigate its separation performance when applied to speech mixtures: We use speech

[4]The solution can be found efficiently using linear programming (Chen et al., 1998). We introduce vectors $\mathbf{s}^+$ and $\mathbf{s}^-$, each with the same dimensionality as $\hat{\mathbf{s}}(t)$, and use the linear constraints $\mathbf{s}^+, \mathbf{s}^- \geq 0$ and $\hat{\mathbf{A}}\mathbf{s}^+ - \hat{\mathbf{A}}\mathbf{s}^- = \mathbf{x}(t)$. The minimisation of $\|\hat{\mathbf{s}}\|_1 = \sum_i |\hat{s}_i|$ becomes the linear objective of minimising $\sum_i (s_i^+ + s_i^-)$. After solving this system, the desired coefficients are $\hat{\mathbf{s}}(t) = \mathbf{s}^+ - \mathbf{s}^-$. When using complex data, as in the case of a STFT representation, we treat the real and imaginary parts separately, thus doubling the number of coefficients.

sources that are extracted from a commercial audio CD of poems read by their authors (Paschen et al., 2001); each source is a ten second segment of a poem, which has been down-sampled to 8 kHz; details of the extraction procedure and the poems used are presented in Appendix A.

Throughout this section we use the notation `MmNs` to denote the mixtures, where `M` and `N` indicate the number of mixtures and sources used, *e.g.*, `4m6s` indicates an instantaneous mixture that has 4 observations of 6 sources. For all experiments, we evaluate the separation performance of the LOST algorithm when applied to the following mixtures: `2m2s`, `2m3s`, `3m2s`, `3m3s`, `3m4s`, `4m3s`, `4m4s`, `4m5s` and `4m6s`; which includes even-determined, over-determined and under-determined mixtures.

2.2.1 Performance Measurement

For the purposes of ease of comparison with existing separation methods, we evaluate the separation performance of the LOST algorithm using the measures provided by the `BSS_EVAL` toolbox (Févotte et al., 2005). The performance measures are based on the principal that a given source estimate, $\hat{s}$, is composed as a sum that includes the original source and different classes of noise:

$$\hat{s}(t) = s(t) + \epsilon_i(t) + \epsilon_n(t) + \epsilon_a(t), \tag{2.15}$$

where $\epsilon_i(t)$ is noise due to interference from other sources, $\epsilon_n(t)$ is perturbating noise (such as Gaussian noise) and $\epsilon_a(t)$ is the noise due to artifacts (such as musical noise). The noise introduced by each class is estimated by the toolbox and used in the following global performance measures:

- *Source-to-Artifact Ratio* (SAR): Measures the level of artifacts in the source estimate,
$$\text{SAR} = \frac{\|s + \epsilon_i + \epsilon_n\|^2}{\|\epsilon_a\|^2}. \tag{2.16}$$

- *Source-to-Interferences Ratio* (SIR): Measures the level of interference from the other sources in the source estimate,
$$\text{SIR} = \frac{\|s\|^2}{\|\epsilon_i\|^2}. \tag{2.17}$$

- *Source-to-Distortion Ratio* (SDR): Provides an overall separation per-

formance criterion,

$$\text{SDR} = \frac{\|s\|^2}{\|\epsilon_i + \epsilon_n + \epsilon_a\|^2}. \tag{2.18}$$

All performance measures are expressed in dB, with higher performance values indicating better quality estimates.

2.2.2 Transform Sparseness

We achieve a sparse representation of the mixtures by exploiting the sparseness of speech in the Short Time Fourier Transform (STFT) domain. In order to find the optimal transform parameters for the data, we perform separation over a wide parameter space and evaluate the estimates. Specifically, we perform an STFT on each mixture where each frame is windowed using a Hamming function over a range of FFT sizes, $\{128, 256, 512, 1024, 2048, 4096\}$, and FFT frame advances, $\{\frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1\}$ (expressed in fractions of FFT size). We perform this procedure for each of the previously specified mixtures and repeat for 20 Monte Carlo runs, resulting in a total of 5400 ($6 \times 5 \times 9 \times 20$) LOST algorithm experiments. Furthermore, the sources used in each mixture are randomly selected from the set of source signals (Appendix A), and are mixed using a random mixing matrix. The procedure for each experiment is as follows:

1. N source signals are randomly selected from the set of sources presented in Appendix A, and are mixed using a randomly generated $\mathbf{A}$ resulting in a matrix of observations, $\mathbf{X} = \mathbf{AS}$.

2. The LOST algorithm (Section 2.1.5) is applied to $\mathbf{X}$, and the source estimates, $\hat{\mathbf{S}}$, are constructed.

3. The estimates and the original sources are used to evaluate the separation performance of the LOST algorithm, using the following performance measures SIR, SAR and SDR.

Results

The results from all experiments are collated and separation performance is calculated as follows: The performance values of the source estimates for each

experiment are averaged, which are themselves averaged over 20 Monte Carlo runs. The worst, median and best performances results, and the transform parameters that achieved these results are tabulated in Table 2.1; average values for γ and iterations are also tabulated. As indicated in Figure 2.1 the sparseness of the coefficients in the transform domain will have an important effect on how well defined the line orientations will be, which ultimately controls the separation performance of the LOST algorithm. The results show that a frame size of 2048 or 4096 produce the worst separation performance for all three measures, which indicates that speech that is sampled at 8 kHz is not sufficiently sparse in this domain. Median performance is achieved for a frame size of 128 or 256, while the best performance is achieved for 256, 512, 1024. It is evident that the average γ values obtained for the best performance values are smaller than all others, indicating that the line orientations are well defined when using the associated STFT parameters. Furthermore, the best performance experiments typically converge the fastest. Therefore, the sparseness of the transform domain effects not only the separation performance but convergence speed also.

To analyse the performance of the LOST algorithm for STFT parameters that achieve good separation, we select a subset of the experiments that have a frame size of 512 or 1024 (which results in a total of 200 experiments for each mixture) and represent the results using box plots: Each box presents information about the median and the statistical dispersion of the results. The top and bottom of each box represents the upper and lower quartiles, while the length between them is the interquartile range; the whiskers represent the extent of the rest of the data, and outliers are represented by +. Box plots for SDR, SIR and SAR are presented in Figure 2.4, Figure 2.5 and Figure 2.6 respectively.

The performance values for SDR indicate that over-determined mixtures produce the best results, while under-determined mixtures produce the worst, which is to be expected for over-determined mixtures, as there are more knowns in $\mathbf{s}$ than unknowns in $\mathbf{x}$. The general trend in SDR performance is that as M increases, separation performance decreases, which decreases further as N increases relative to M.

For SAR performance, the large distances between the median values for the even-determined and under-determined performance results illustrate the high level of artifacts present in the under-determined mixture source

Table 2.1: The Relationship Between Transform Parameters and the Separation Performance of the LOST Algorithm; Average Separation Performance over 20 Monte Carlo Runs for Each Experiment.

Measure	Rand. Mix.	Worst Performance					Median Performance					Best Performance				
		FFT Param.		γ	Iter.	Avg. Res. (dB)	FFT Param.		γ	Iter.	Avg. Res. (dB)	FFT Param.		γ	Iter.	Avg. Res. (dB)
		Frame	Adv.				Frame	Adv.				Frame	Adv.			
SDR	2m2s	128	16	47.4	11.05	**33.28**	256	16	44.67	8.85	**39.9**	1024	256	18.66	7.4	**45.76**
	2m3s	4096	4096	59.52	22.5	**1.97**	128	8	108.37	29.95	**7.29**	512	64	12.76	19.6	**10.26**
	3m2s	4096	256	10.02	12.2	**36.33**	256	16	26	6.2	**41.83**	512	32	15	8.4	**49.05**
	3m3s	128	32	88.04	20.65	**21.11**	256	16	24.8	16.85	**27.03**	1024	256	8.22	17.05	**34.71**
	3m4s	4096	4096	39.18	30.75	**4.09**	128	16	84.4	27.75	**9.93**	512	32	7.16	24.3	**14.25**
	4m3s	4096	256	5.95	31.45	**18.32**	128	32	90.88	14.75	**31.21**	256	32	21.58	12.3	**37.37**
	4m4s	2048	2048	10.56	34.4	**15.55**	128	16	66.68	24.95	**23.4**	256	64	19.55	18.35	**27.88**
	4m5s	2048	2048	9.33	41.45	**6.46**	128	8	66.81	35.45	**11.54**	512	32	9.05	33.55	**14.35**
	4m6s	4096	4096	32.91	39.65	**1.31**	128	8	63.86	48.2	**6.64**	256	16	14.4	46.7	**9.72**
SIR	2m2s	128	16	47.4	11.05	**33.32**	256	16	44.67	8.85	**39.99**	1024	256	18.66	7.4	**46.97**
	2m3s	4096	4096	59.52	22.5	**9.79**	128	8	108.37	29.95	**14.72**	512	256	25.03	15.1	**17.98**
	3m2s	4096	256	10.02	12.2	**36.33**	256	16	26	6.2	**41.89**	512	32	15	8.4	**49.58**
	3m3s	128	32	88.04	20.65	**21.11**	256	16	24.8	16.85	**27.03**	1024	256	8.22	17.05	**34.71**
	3m4s	4096	512	5.04	47.3	**10.27**	128	16	84.4	27.75	**16.04**	256	64	21.86	24.65	**20.69**
	4m3s	4096	256	5.95	31.45	**18.32**	128	32	90.88	14.75	**31.21**	256	32	21.58	12.3	**37.38**
	4m4s	2048	2048	10.56	34.4	**15.55**	128	16	66.68	24.95	**23.4**	256	64	19.55	18.35	**27.88**
	4m5s	4096	2048	7.2	44.3	**11.48**	256	16	15.66	36.2	**17.01**	512	512	9.97	25.9	**22.04**
	4m6s	4096	2048	5.99	59.05	**7.72**	128	8	63.86	48.2	**13.17**	256	64	14.04	31.15	**18.05**
SAR	2m2s	4096	4096	59.86	11.1	**66.27**	256	128	58.18	9.35	**70.45**	1024	512	15.18	7.7	**72.56**
	2m3s	4096	4096	59.52	22.5	**4.18**	256	16	40.88	34.1	**11.51**	256	16	40.88	34.1	**15.58**
	3m2s	128	8	68	12	**68.99**	256	16	26	6.2	**71.78**	256	16	26	6.2	**73.01**
	3m3s	2048	128	7.02	26.85	**63.09**	128	64	77.12	18	**65.87**	1024	128	4.6	14.7	**68.37**
	3m4s	4096	4096	39.18	30.75	**7.53**	128	8	88.21	36.55	**15.06**	512	64	7.15	38.15	**18.68**
	4m3s	4096	2048	12.93	22.3	**65.7**	128	16	88.94	15.05	**68.66**	128	32	90.88	14.75	**70.5**
	4m4s	128	128	82.53	22.2	**61**	128	16	66.68	24.95	**64.02**	1024	1024	10.46	22.5	**66.89**
	4m5s	4096	4096	33.08	30.4	**9.03**	128	8	66.81	35.45	**17.84**	256	32	14.51	43.75	**22.2**
	4m6s	4096	4096	32.91	39.65	**3.98**	128	8	63.86	48.2	**11.16**	512	32	4.53	46.8	**15.31**

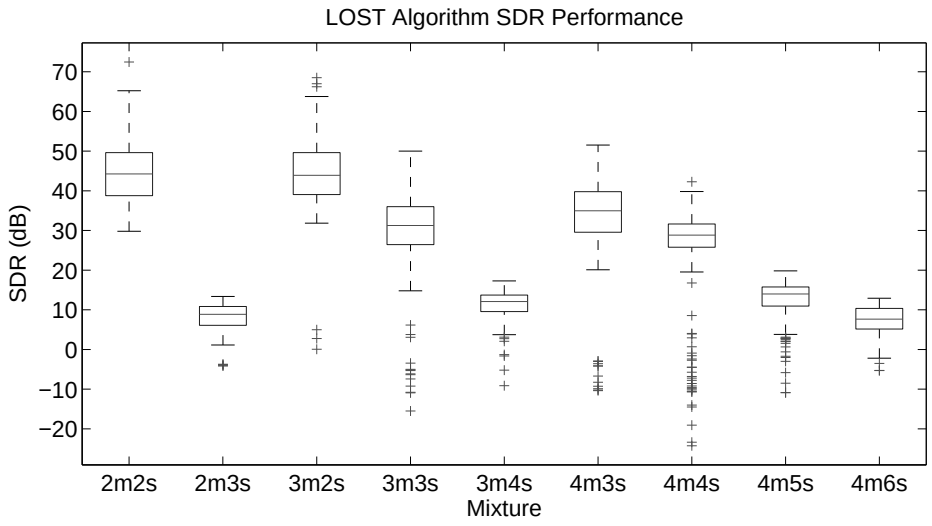

Figure 2.4: SDR results for the LOST algorithm: Box plots are used to illustrate the performance results for each mixture, with each box representing the median and the interquartile range of the results. For SDR, which represents overall separation performance, separation performance decreases as M increases, which decreases further as N increases relative to M.

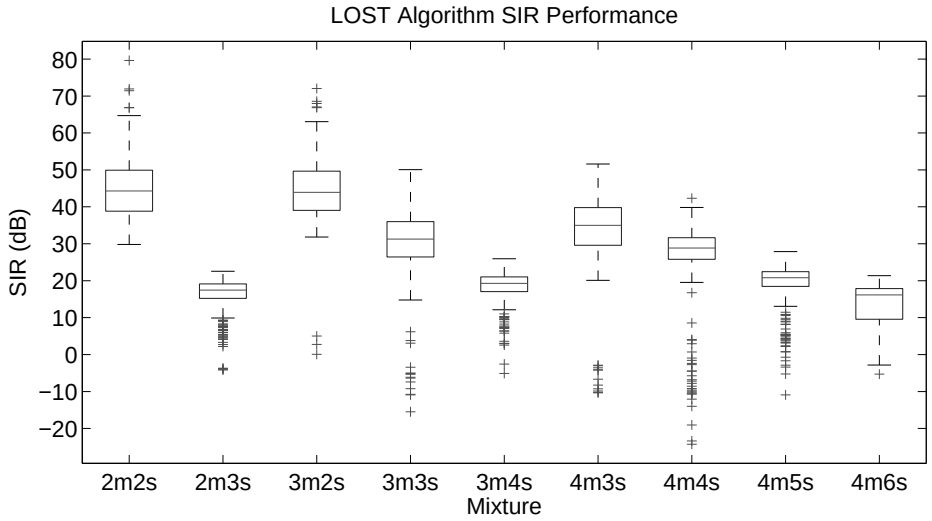

Figure 2.5: SIR results for the LOST algorithm: Box plots are used to illustrate the performance results for each mixture, with each box representing the median and the interquartile range of the results. The results indicate that the source estimates become more resilient to interference from other sources as M increases relative to N.

estimates. Listening to these estimates reveals the presence of clipping sounds and portions of the other sources in the estimates. Such artifacts are not audible for the even-determined or over-determined source estimates, and are

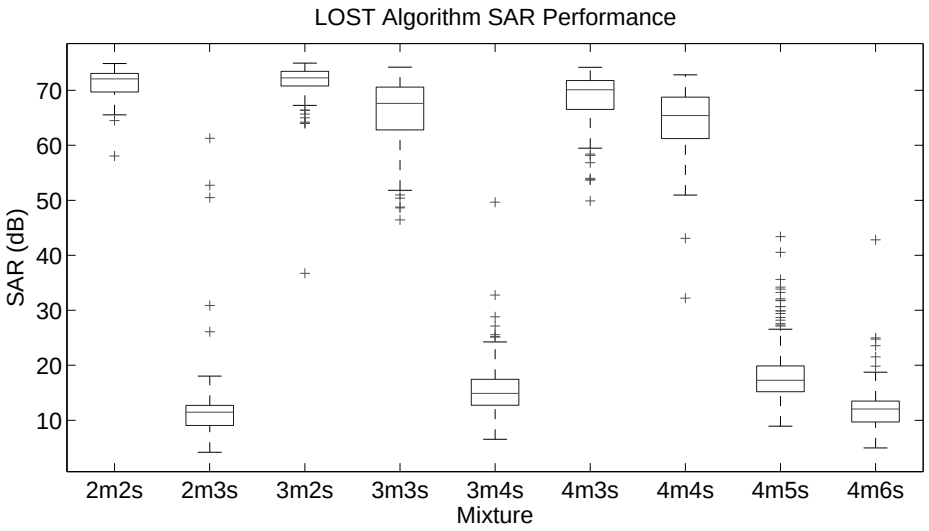

Figure 2.6: SAR results for the LOST algorithm: Box plots are used to illustrate the performance results for each mixture, with each box representing the median and the interquartile range of the results. For SAR, it is evident that there are dramatic differences between the performances achieved for even-determined and under-determined mixtures, which is a consequence of the artifacts produced by L_1-norm minimisation.

produced by L_1-norm minimisation when more than M sources are active at the same time. This contrasts with SIR, where the difference between even-determined and under-determined performance is not so dramatic.

It is worth noting that over all performance measures, increasing the number of observations for an even-determined mixture, does not dramatically improve separation performance. For example, we can see from inspection of the results for the mixtures `3m3s` & `4m3s` that the additional observation provides a small increase in performance, the same is also true for `2m2s` & `3m2s`. Such an incremental improvement may defy preconceptions, but is typical of BSS algorithms.

The outliers that are evident in the box plots may be due to the random mixing matrices used to generate our mixtures. Such randomly generated mixtures may produce scatter plots that contain lines that are too close for the LOST algorithm to separate effectively, *i.e.*, $\mathbf{A}$ is an ill-conditioned matrix. A plot of the estimates for `4m6s` produced by the LOST algorithm is presented in Figure 2.7.

Overall, the LOST algorithm provides very good results for the blind source separation of even-determined and over-determined mixtures, and suc-

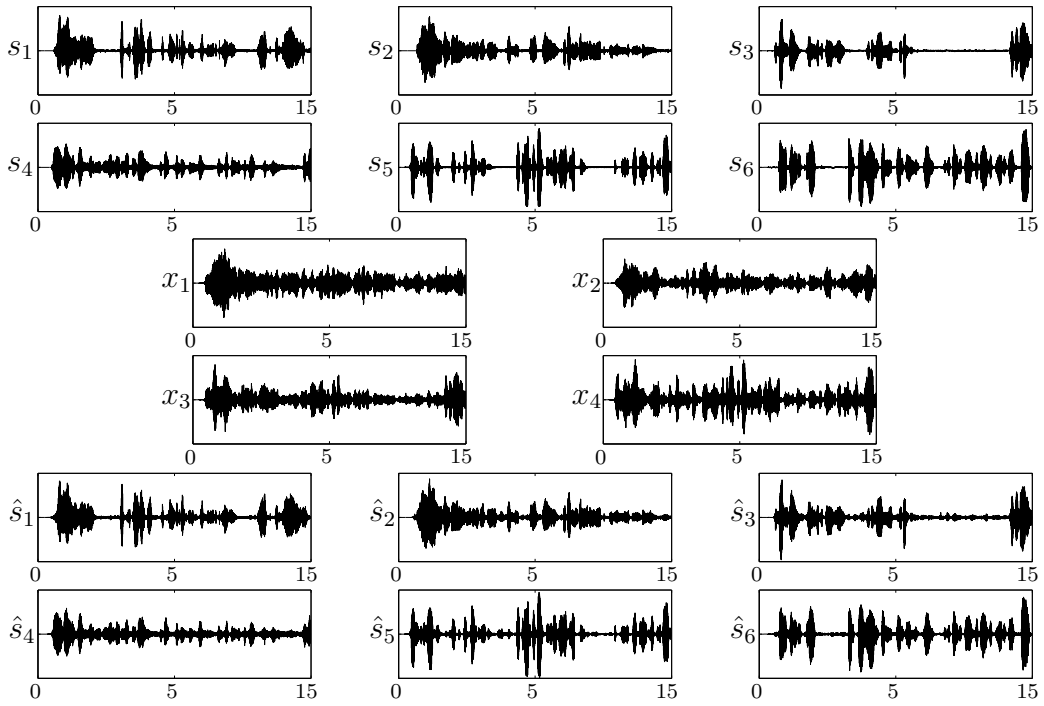

Figure 2.7: Source estimate plots for the LOST algorithm. The plots above show ten second clips of six acoustic sources, $s_1, \ldots, s_6$; 4 mixtures, $x_1, \ldots, x_4$; and 6 source estimates, $\hat{s}_1, \ldots, \hat{s}_6$. Sound wave pressure is plotted against time in units of seconds.

cessfully achieves separation of under-determined mixtures with good separation performance.

2.2.3 Robustness to Noise

We perform an empirical investigation on the separation performance of the LOST algorithm when Gaussian noise is added to $\mathbf{S}$. The noise added to each source is measured using the signal-to-noise ratio and is expresses in dB. We perform experiments where Gaussian noise of the following intensities is added to the each source: 20 dB, 15 dB, 10 dB, 5 dB and 2 dB. As a means of comparison, we also perform an experiment where no noise (∞ dB) is added to the sources. We run the LOST algorithm using an FFT frame size of 512 and frame advance of 128. In contrast to the experimental procedure presented in Section 2.2.2, each mixture is generated using a fixed mixing matrix and fixed set of sources, which is necessary as we are only interested in robustness to noise and not general separation performance.

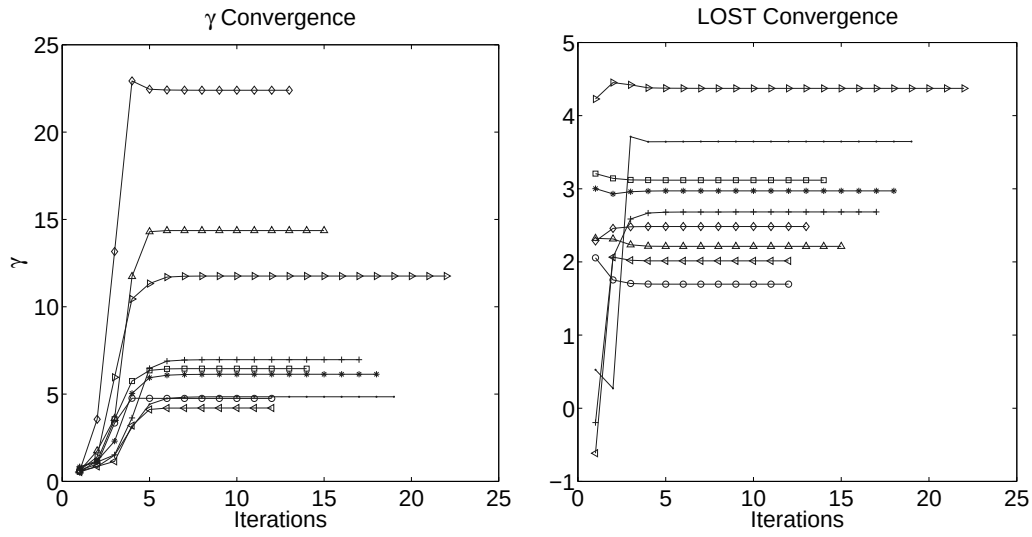

Figure 2.8: LOST algorithm convergence plots for the following experiments: 2m2s ○, 2m3s □, 3m2s ◇, 3m3s △, 3m4s ▷, 4m3s ◁,4m4s +,4m5s ⋆,4m6s •; the convergence of the mixing matrix, $\hat{\mathbf{A}}$ is presented on the right, while convergence of the boundary value, γ, is presented on the left. It is evident that both γ and $\hat{\mathbf{A}}$ quickly converge to stable values.

Additionally, we evaluate the performance of the LOST algorithm with and without kurtosis scaling of the STFT coefficients.

Results

The results from all experiments are collated and averaged as before, and separation performance for each experiment is presented in Table 2.2. It is evident that the SIR performance results degrade for all mixtures as the level of noise increases, this reflects the perturbation of the line orientations by the random noise, which influences the level of interference from other sources that will be present in the source estimates.

The SAR performance remains relatively constant for the even-determined and over-determined mixtures over all noise levels, while the results for the under-determined results gradually degrade as noise increases. This degradation in performance demonstrates that L_1-norm minimisation is generally unstable for perturbation of $\mathbf{A}$. Furthermore, the results show that SAR is largely unaffected by the kurtosis scaling of the transform coefficients, which demonstrates that kurtosis scaling has no effect on the presence of artifacts.

Overall performance, as indicated by SDR, demonstrates that the LOST algorithm achieves good separation results over all noise levels. Further-

Table 2.2: Average Separation Performance for the LOST Algorithm on Noisy Mixtures, With and Without Kurtosis Scaling.

Measure	Fixed Mix.	FFT Param.		With Kurtosis Scaling						Without Kurtosis Scaling					
		Frame	Adv.	Avg. Res. (dB) for Added Noise (SNR)						Avg. Res. (dB) for Added Noise (SNR)					
				None	20 dB	15 dB	10 dB	5 dB	2 dB	None	20 dB	15 dB	10 dB	5 dB	2 dB
SDR	2m2s	512	128	55.96	59.2	49.11	50.29	49.25	33.74	41.88	41.81	42.59	42.57	46.65	31.34
	2m3s			9.19	8.85	8.31	7.16	5.15	2.84	9.07	8.72	8.19	7.06	5.1	2.91
	3m2s			35.04	33.14	31.05	28.39	25.81	23.08	27.03	26.52	25.82	24.1	20.94	16.33
	3m3s			30.24	29.12	27.75	24.73	20.58	14.36	30.25	29.67	28.46	25.06	19.29	11.22
	3m4s			12.01	11.88	11.25	9.88	7.68	4.42	11.95	11.79	11.14	9.7	7.22	3.04
	4m3s			36.85	35.35	33.76	30.48	27.99	22	34.67	33.52	32.05	27.93	21.62	13.08
	4m4s			29.87	28.66	27.33	24.84	20.73	15.27	27.35	26.32	25.02	22.14	16.66	9.63
	4m5s			16.52	15.2	13.85	11.99	8.99	6.17	15.47	14.68	12.91	11.19	8.61	5.77
	4m6s			9.87	10.12	8.97	7.55	5.13	2.53	10.7	10.32	9.46	7.56	5.06	-0.68
SIR	2m2s	512	128	55.99	59.32	49.12	50.3	49.25	33.74	41.88	41.81	42.59	42.57	46.69	31.34
	2m3s			17.27	16.66	15.86	14.13	11.17	7.69	17.1	16.53	15.76	14.13	11.14	7.57
	3m2s			35.04	33.14	31.05	28.39	25.81	23.08	27.03	26.52	25.82	24.1	20.94	16.33
	3m3s			30.24	29.12	27.75	24.73	20.58	14.36	30.25	29.67	28.46	25.06	19.29	11.22
	3m4s			19.81	19.47	18.51	16.56	13.02	7.69	19.63	19.18	18.19	16.14	12.09	6.49
	4m3s			36.85	35.35	33.76	30.49	27.99	22	34.67	33.52	32.05	27.93	21.62	13.08
	4m4s			29.87	28.66	27.33	24.84	20.73	15.27	27.35	26.32	25.02	22.15	16.66	9.63
	4m5s			27.1	25.19	23.17	20.35	15.62	11.13	25.47	24.12	20.99	18.35	14.2	9.36
	4m6s			18.52	18.96	17.1	15.26	11.76	7.89	20.1	19.46	18.18	15.34	11.51	1.49
SAR	2m2s	512	128	77.47	77.29	77.29	77.32	78.19	79.28	77.47	77.26	77.28	77.31	78.16	79.3
	2m3s			10.04	9.76	9.29	8.34	6.83	5.51	10.01	9.72	9.25	8.29	6.8	5.61
	3m2s			74.24	74.07	74.05	74.67	74.52	76.22	74.07	73.97	73.97	74.54	74.36	75.94
	3m3s			74.52	74.51	74.58	74.72	75.05	75.4	74.5	74.48	74.57	74.71	74.96	74.87
	3m4s			13.22	13.14	12.6	11.53	10.23	10.93	13.07	13.01	12.51	11.47	10.43	11.31
	4m3s			76.3	76.33	76.33	76.61	77.23	78.15	76.29	76.34	76.34	76.6	77.06	77.81
	4m4s			66.87	66.88	66.88	66.83	66.94	68.57	66.77	66.77	66.76	66.63	66.42	67.46
	4m5s			18.33	17.6	16.01	12.8	10.24	8.34	16.14	15.42	13.9	12.39	10.45	9.58
	4m6s			10.73	10.87	9.93	8.64	6.65	4.9	11.35	11.02	10.22	8.62	6.68	11.44

Table 2.3: Typical Run Times for the LOST Algorithm on 10 Second Mixtures, Using a Frame Size of 512 and Frame Advance of 128.

Mixture	Time (sec.)	Mixture	Time (sec.)
2m2s	7	4m3s	20
2m3s	30	4m4s	18
3m2s	8	4m5s	45
3m3s	14	4m6s	60
3m4s	40		

more, kurtosis scaling improves separation performance for all mixtures at all noise levels, however it is particularly effective for even-determined and over-determined mixtures. The tabulated results demonstrate that the LOST algorithm is an effective algorithm for blind source separation of over-determined, even-determined and under-determined mixtures, even in the presence of noise.

To illustrate the convergence of the LOST algorithm, convergence curves for both γ and the norm of $\hat{\mathbf{A}}$ are presented for each mixture in Figure 2.8; the curves correspond to the experiments presented in Table 2.2 where kurtosis scaling is performed and no noise is added. It is evident that both γ and $\hat{\mathbf{A}}$ converge to stable results after a small number of iterations, demonstrating the fast convergence properties of the LOST algorithm.

We implemented the LOST algorithm in C, and all the experiments presented were run on a 3.06 GHz Intel Pentium-4 based computer with 768 MB of RAM running the Debian GNU/Linux operating system. Typical run times for a frame size of 512 and frame advance of 128 are presented in Table 2.3. Our C code implementation of the LOST algorithm is freely available and can be downloaded from `http://www.hamilton.ie/paul/`

2.2.4 LOST Vs geoICA

One of the main advantages of the LOST algorithm is that it provides a solution for the under-determined case. In order to demonstrate the usefulness of the LOST algorithm when applied to under-determined mixtures, we compare its performance to the geoICA[5] algorithm (Theis et al., 2004), which also

[5]Matlab implementations for geoICA and GCE are available at `http://www.biologie.uni-regensburg.de/Biophysik/Theis/research/geoICA.zip`

Table 2.4: Average GCE with Standard Deviations for LOST and geoICA over 20 Monte Carlo Runs for Each Experiment, where smaller values indicate better performance.

Mixture	Algorithm		
	LOST	geoICA	geoICA+STFT
2m2s	0.42±0.88	0.48±0.55	$\mathbf{0.05 \pm 0.09}$
2m3s	$\mathbf{0.12 \pm 0.17}$	0.95±0.55	0.91±0.59
3m2s	$\mathbf{0.02 \pm 0.02}$	0.37±0.61	0.02±0.02
3m3s	$\mathbf{0.12 \pm 0.22}$	1.29±0.69	1.05±0.58
3m4s	$\mathbf{0.3 \pm 0.49}$	1.82±0.42	1.25±0.42
4m3s	$\mathbf{0.19 \pm 0.57}$	1.16±0.77	0.76±0.82
4m4s	$\mathbf{0.55 \pm 1.02}$	2.06±0.48	1.61±0.8
4m5s	$\mathbf{0.62 \pm 0.92}$	3±0.85	2.13±0.76
4m6s	$\mathbf{0.76 \pm 0.79}$	3.7±0.78	2.68±0.77

provides a solution for the under-determined case. We test both algorithms using the previously specified mixtures; where $\mathbf{A}$ is randomly generated and the sources are randomly selected as in Section 2.2.2. Furthermore, each experiment is repeated for 20 Monte Carlo runs. For the LOST algorithm a FFT size of 512 and frame advance of 128 is used, geoICA does not specify a STFT. However, in order to place both algorithms in an even setting, we perform geoICA using speech that is STFT transformed using the same parameters as those specified for the LOST algorithm. Furthermore, we use geoICA with its default number of iterations, which is $10 \times$ #samples.

The geoICA algorithm specifies no method to separate the sources once $\hat{\mathbf{A}}$ is found (such as L_1-norm minimisation), therefore we measure the performance of the algorithms using the *Generalised Crosstalk Error* (GCE) (Theis et al., 2004) between $\mathbf{A}$ and $\hat{\mathbf{A}}$:

$$\text{GCE} = \min_{\mathbf{L} \in \Pi} \|\mathbf{A} - \hat{\mathbf{A}}\mathbf{L}\|, \tag{2.19}$$

where the minimum is taken over the group Π of all invertible matrices having only one non-zero entry per column. When $\mathbf{A}$ and $\hat{\mathbf{A}}$ are equivalent, GCE vanishes, which indicates that GCE decreases as performance increases.

The results for each experiment are collated, and the average GCE performances, along with their standard deviations, are presented in Table 2.4. It is evident from the results that the LOST algorithm achieves superior per-

formance over geoICA when applied to the separation of speech mixtures. While geoICA performs well for `2m2s`, `2m3s` and `3m2s`; it performs badly for all other mixtures, even when the observations are transformed to the STFT domain. The general trend of the results show that geoICA does not perform well when $M > 2$, and while the LOST algorithm does exhibit decreased performance, the scale of degradation is not as dramatic as that exhibited by geoICA. The reason for this may be that geoICA maps the observations to the unit half-sphere, which may cause difficulties when the sources lie near the equator. Another reason may be the fact that geoICA is a simple clustering approach that does not specify any particular prior, unlike the LOST algorithm, which assumes a Laplacian prior.

2.3 Discussion

One of the main benefits of our approach is that a solution for the underdetermined case can be found. In contrast to other similar approaches (Mitianoudis and Stathaki, 2005), the LOST algorithm is not constrained to just two mixtures. Furthermore, by comparison with the geoICA algorithm, we have demonstrated that the LOST algorithm produces good results when $M > 2$.

The performance of the LOST algorithm is heavily influenced by how well defined the linear subspaces are in the transform domain. Therefore, the sparse domain transformation is an integral component of the algorithm, and appropriate selection of such is required to provide useful results. We use the STFT transform, which achieves good separation performance for speech mixtures when an FFT frame size of 512 or 1024 is used. Alternative transformations such as Gabor or wavelet could also be used. Although, this may not be necessary as the performance comparison between LOST and geoICA indicates that the STFT transformation is sufficient to achieve good separation performance.

The algorithm we present is a batch operation algorithm, which operates on the entire set of observations. Conversely, an online approach that operates on an observation-by-observation basis is also possible. We have previously presented such an algorithm (O'Grady and Pearlmutter, 2004a), where the PCA computations of the batch algorithm are replaced by the stochastic gradient algorithm, which converges to the direction of largest

variance of its input data.

The scheme we use for line estimation involves updating the current line estimates to the principal eigenvector of the covariance matrix associated with each line. While this is a perfectly acceptable assumption for small values of M. For very large hyper spaces, where M is large, such a scheme may not produce an optimal estimate of the direction of linear subspace. The same is also true for the γ update. Therefore, to more accurately estimate the direction and width of a linear subspace in a high dimensional space, a more sophisticated scheme using the provided eigenvalues may be required.

Throughout our experiments, we have observed on occasion that the random initialisation of $\mathbf{A}$ affects the performance of the line estimation procedure. Sensitivity to initial conditions is common among clustering algorithms, and in the case of the LOST algorithm, such a scenario is indicated when $\gamma < 1$. In this event, we suggest that $\mathbf{A}$ is reinitialised and that the experiment is repeated.

Finally, occasionally we observe that the scheme we use for the adaption of γ causes the parameter to grow without bounds. This typically happens when the transform parameters selected produce scatter plots that are not well defined. When this behaviour is observed, we recommend that γ is fixed to some suitably large value. Alternatively, we have observed that increasing the dynamic range of the mixtures works on occasion.

2.4 Conclusion

In this chapter, we presented an EM algorithm that identifies linear subspaces that cross the origin, we have illustrated how such a problem arises in the context of blind source separation of instantaneous mixtures, where mixture matrix columns correspond to linear subspaces in a scatter plot. This method, combined with a transformation into a sparse domain and an L_1-norm optimisation, constitutes the LOST algorithm, which provides a solution for the blind source separation of instantaneous mixtures with an arbitrary number of mixtures and sources. We performed an extensive investigation on the general separation performance of the LOST algorithm, which yielded good results, and demonstrated the algorithm's robustness in the presence of noise. Furthermore, we demonstrated that the LOST algorithm performs well when compared to the geoICA algorithm.

CHAPTER 3

Perceptual Evaluation of the NMF Objective

Since the introduction of the NMF algorithm by Lee and Seung (2001), the two originally proposed reconstruction objectives—Squared Euclidean Distance and Kullback-Leibler Divergence—have remained the most popular choice for implementation of the algorithm. These objectives can be used effectively to measure the reconstruction error of the factorisation. However, they may not necessarily reflect the subjective realities of how a person would perceive the signal. A more appropriate way to measure the reconstruction error for perceptible data is to develop a generalised receiver model of the target sense. In the case of audio, the receiver is ultimately the ear and the perception of sounds is determined by its psychoacoustic properties. Such evaluation methods apply greater weight to perceptually important features of the data.

For many years the speech compression community have employed psychoacoustic results in speech quality measures (Quackenbush et al., 1988). Audio encoders achieve compression by exploiting the fact that some of the audio information presented at the ear is not detectable by the listener (Painter and Spanias, 2000). In the case of NMF of speech data, it would be desirable that the reconstruction objective focus on the more perceptually relevant features of the data. With this in mind, we investigate the perceptual properties of a parameterisable divergence known as the beta divergence (Kompass, 2005) when used in an NMF algorithm that is applied

to speech data. The beta divergence is tested for a range of β values and the resultant reconstructions are perceptually evaluated using the noise-to-mask ratio (Brandenburg, 1987). By way of comparison, we also present a perceptually weighted version of NMF that utilises the noise-to-mask ratio as its reconstruction objective.

In contrast to the LOST algorithm, where the algorithm learns the parameters of an LMM that factorise probability densities, NMF factorises a non-negative matrix, $\mathbf{V}$, into matrices containing bases, $\mathbf{W}$, and activations, $\mathbf{H}$. Furthermore, the LOST algorithm estimates the sources by applying L_1-norm minimisation to the original observations in a sparse domain. Conversely, NMF does not separate sources using the original data and instead synthesises the sources using the discovered bases and activations. Since the LOST algorithm does not synthesise the sources from learned features, a perceptual evaluation of the algorithm is not required.

This chapter is organised as follows: We present an overview of psychoacoustic phenomena and discuss a perceptual evaluation method that is based on such phenomena in Section 3.1. We present the objective functions that are under investigation, discuss their symmetry properties, and present NMF algorithms that use these objectives in Section 3.2. The reconstructions produced by the presented algorithms are perceptually evaluated and the results are discussed in Section 3.3. The chapter closes with a discussion and conclusion.

3.1 Psychoacoustics

Psychoacoustics is the branch of psychology that studies human acoustical perception. Psychoacoustic phenomena are elucidated by observing the response of a human subject to different sound stimuli. These stimuli may include a single tone or a narrow-band noise signal. Psychoacoustic effects have been shown to occur between stimuli that occur at the same time, while other effects occur over intervals of time (Zwicker and Fastl, 1999). The discovered phenomena are also known to be very dependent on physiology of the ear. In this section, we introduce the psychoacoustic phenomena that affect how we perceive sound by discussing the experimental procedures involved in investigating such phenomena.

3.1.1 Psychoacoustic Experimental Methods

The basis of the following psychoacoustic experiments is the notion of *tonal detection thresholds*. Tonal detection thresholds are used to express the perceived intensity of the stimulus, and are measured in terms of *Sound Pressure Level* (SPL). The SPL expresses the intensity of stimulus sound pressure in decibels (dB) relative to an internationally defined reference level, *i.e.*, $L_{\text{SPL}} = 20\log_{10}(p/p_0)$ dB, where p is the sound level of the stimulus in Pascals, and p_0 is the standard reference level of 20 micropascals. The dynamic range of the intensity for the human auditory system is about 150 dB_{SPL}, which includes sounds that range from the limits of detection for low intensity (quiet) stimuli, up to the threshold of pain for high intensity (loud) stimuli. Typical SPL levels for some commonly occurring sounds include 30 dB_{SPL} for a whisper, 70 dB_{SPL} for conversational speech and 140 dB_{SPL} for a jet engine. The psychoacoustic phenomena addressed in this section are characterised in terms of SPL.

3.1.2 Hearing Threshold

A first step towards a perceptual model is to determine the tone detection threshold required for detection of each frequency in the hearing spectrum. This is known as the absolute threshold of hearing in quiet, and is measured experimentally by progressing through the frequencies of the spectrum allowing the subject to modulate the intensity of a tone until it becomes audible. The absolute threshold is measured in dB_{SPL} and is approximated by the following non-linear function, which is representative of a young listener with acute hearing (Terhardt, 1979),

$$T_q(f) = 3.64\left(\frac{f}{1000}\right)^{-0.8} - 6.5e^{-0.6\left(\frac{f}{1000-3.3}\right)^2} + 10^{-3}\left(\frac{f}{1000}\right)^4, \qquad (3.1)$$

where f is frequency in Hertz. The behaviour of the function can be described by its constituent terms: The first term describes the low-frequency cutoff of hearing, the second describes the increased sensitivity of the ear around 3 kHz, and the third describes the high-frequency cutoff. A plot of the threshold function is presented in Figure 3.1. It can be seen that for low frequencies, the threshold requires relatively high SPL reaching about 40

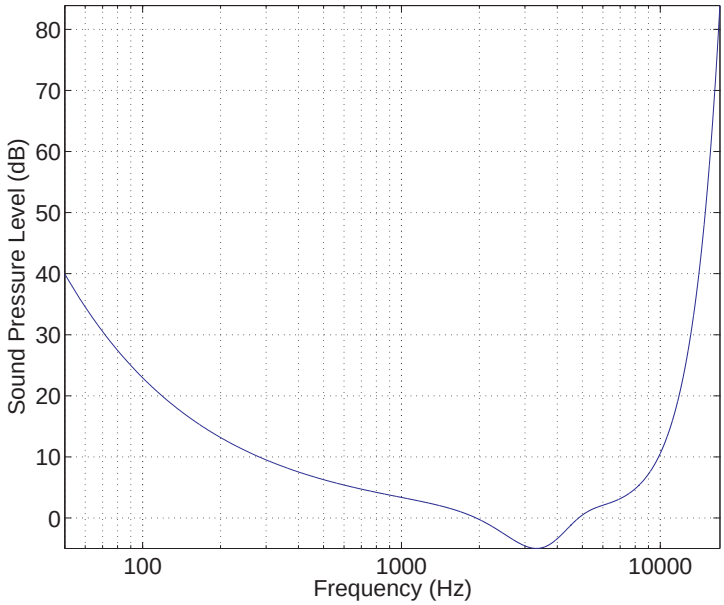

Figure 3.1: The absolute threshold of hearing in quiet. Across the spectrum of human hearing, the threshold quantifies the SPL necessary at each frequency such that an average listener detects a pure tone stimulus in a noiseless environment.

$\mathrm{dB_{SPL}}$ at 50 Hz. The level at 300 Hz has dropped by about 30 $\mathrm{dB_{SPL}}$. For frequencies between 0.5 and 2 kHz, the threshold remains almost independent of frequency, followed by a very sensitive bump at around 3 kHz. For frequencies above 10 kHz the threshold increases sharply with some frequencies remaining inaudible for older subjects irrespective of SPL.

3.1.3 Masking Effects

The threshold in quiet only considers a subject's response to single tones. When the subject's response to multiple simultaneous tones is considered, a phenomenon known as *simultaneous masking* is observed. Simultaneous masking describes the inaudibility of weaker tones (the maskee) when in close proximity to louder tones (the masker). Masking of one tone by another occurs more acutely when both tones reside in a predefined bandwidth known as the *critical bandwidth*. This behaviour results from the operation of the cochlea, which can be viewed from a signal processing perspective as a bank of overlapping bandpass filters, where the passbands are of nonuniform bandwidth and the bandwidths increase with frequency. These passbands define the critical bandwidths, and are determined experimentally using the

Table 3.1: Idealised Critical Band Parameters (After Scharf (1970)).

Critical Band No.	Centre Freq. (Hz)	Bandwidth (Hz)	Critical Band No.	Centre Freq. (Hz)	Bandwidth (Hz)
1	50	-100	14	2150	2000-2320
2	150	100-200	15	2500	2320-2700
3	250	200-300	16	2900	2700-3150
4	350	300-400	17	3400	3150-3700
5	450	400-510	18	4000	3700-4400
6	570	510-630	19	4800	4400-5300
7	700	630-770	20	5800	5300-6400
8	840	770-920	21	7000	6400-7700
9	1000	920-1080	22	8500	7700-9500
10	1175	1080-1270	23	10500	9500-12000
11	1370	1270-1480	24	13500	12000-15500
12	1600	1480-1720	25	19500	15500-
13	1850	1720-2000			

following procedure: A single tone and a narrow-band noise signal with time varying bandwidth are centred at the same frequency. As the noise bandwidth increases, the noise energy increases resulting in an elevated tone detection threshold. However, there will come a point where an increase in bandwidth does not result in an increase in tone detection threshold, indicating that only a limited band of noise acts to mask the tone; this is the critical bandwidth.

Idealised critical band parameters (Scharf, 1970) are presented in Table 3.1, and are approximated by the following expression (Zwicker and Fastl, 1999)

$$CB(f) = 25 + 75\left[1 + 1.4\left(\frac{f}{1000}\right)^2\right]^{0.69} \text{ (Hz)}, \tag{3.2}$$

which is plotted in Figure 3.2. For an average listener, the critical bandwidth remains constant at about 100 Hz up to 500 Hz, and increases to approximately 20% of centre frequency above 500 Hz. The critical bandwidth is usually represented using the *bark scale*, where the centre frequencies of each band are equally spaced, and a distance of one critical band is referred to as one Bark. The following function (Zwicker and Fastl, 1999) is used to convert from frequency in Hertz to the Bark scale,

$$B(f) = 13\arctan(0.00076f) + 3.5\arctan\left[\left(\frac{f}{7500}\right)\right]^2 \text{ (Bark)}. \tag{3.3}$$

A plot of Eq. 3.2 in the Bark domain is presented in Figure 3.3. It is evident from the graph that the spacing of critical bands is non-uniform on the Hertz scale, but uniform on a Bark scale.

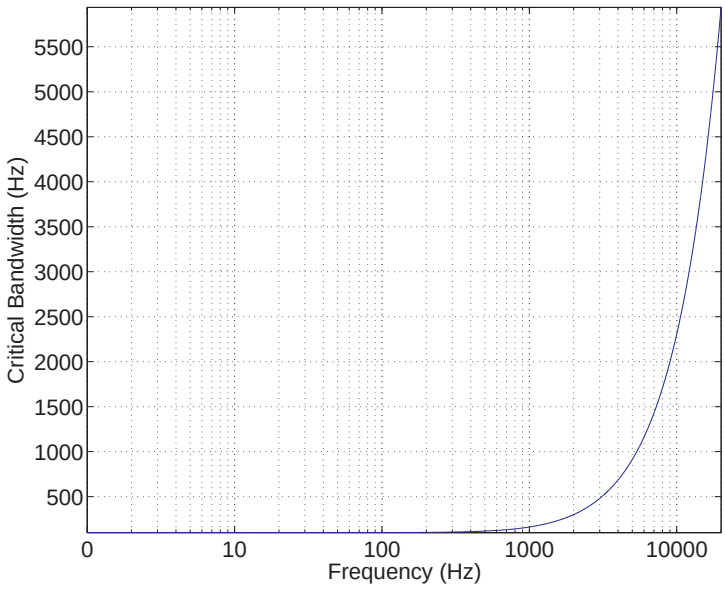

Figure 3.2: Critical bandwidth as a function of centre frequency.

Simultaneous masking effects can be measured experimentally using a similar procedure. The two most commonly used approaches are *Noise-Masking-Tone* (NMT) and *Tone-Masking-Noise* (TMN). In the NMT approach a fixed bandwidth narrow-band noise signal masks a tone within the same critical band: The intensity of the tone is fixed at a constant value, while the intensity of the narrow-band noise is increased until the tone becomes inaudible. The TMN approach employs the reverse of this procedure, where the intensity of the narrow-band noise is fixed and the tone intensity is variable. The difference between the intensity of the masker and maskee is known as the *Signal-to-Mask Ratio* (SMR). Numerous studies investigating NMT and TMN for random noise and pure tone stimuli have appeared over the years. For example, in a reported NMT study (Egan and Hake, 1950) a critical band noise masker centred at 410 Hz with an intensity of 80 dB_{SPL} masks a 410 Hz tone with an intensity of 76 dB_{SPL}, resulting in a SMR of 4 dB at the threshold of detection. The threshold SMR increases as the tone is shifted above or below 410 Hz. In a similar TMN study (Schroder et al., 1979), a critical band noise masker centred at 1 kHz with an intensity of 56 dB_{SPL} is masked by a tone centred at the same frequency at an intensity of 80 dB_{SPL}. Here, the SMR is 24 dB. An interesting observation from this comparison is that even though the intensity of the masker is identical in each case, the SMR is markedly different; indicating that simultaneous masking

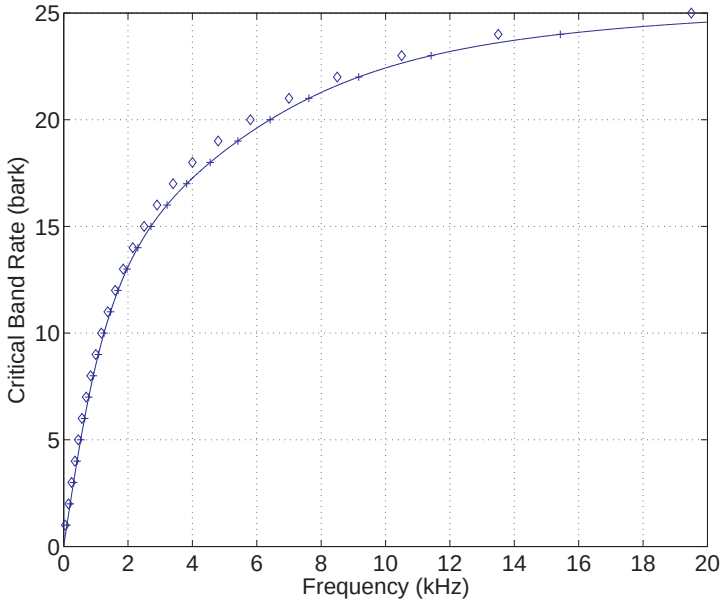

Figure 3.3: Plot of frequency to Bark domain mapping for Eq. 3.2, where + indicates the centre frequencies for each band and ◇ indicates the idealised centre frequencies of Table 3.1.

is an asymmetric process, and that significantly greater masking power is associated with noise maskers.

Simultaneous masking is not confined to the bandwidth of a single critical band; stimuli can also have a predictable effect on the detection thresholds in other critical bands. This effect is known as the *spread of masking*. The effects of the spread of masking can be determined experimentally using a *tone-masking-tone* procedure. The procedure is similar to determining the absolute threshold of hearing in quiet. The difference being that the tone detection threshold for each frequency is measured in the presence of a single tone fixed at a specified frequency and intensity. The effect of the masking is asymmetric, resulting in a steep decrease in tone detection threshold from the masker to lower frequencies, whereas a more gentle decrease is experienced in the direction of higher frequencies (Zwicker and Fastl, 1999). This behaviour is modelled on the Bark scale using a triangular spreading function, with the peak of the triangle corresponding to the critical band under consideration, z_c. The slope for Bark values less than z_c is fixed, while the slope for values larger than z_c is dependent on the intensity of the stimuli in the corresponding critical bands (Terhardt, 1979).

Masking effects also occur between stimuli over small intervals of time,

where a sound stimulus renders an immediately preceding (*backward masking*) or following (*forward masking*) stimulus inaudible. Temporal masking is characterised by exponential attenuation from the onset and offset of the masker, where the onset attenuation lasts for approximately 10 ms, and the offset attenuation approximately 50 ms.

3.1.4 Psychoacoustic Model

The results from studies of these psychoacoustic phenomena have created a wide body of knowledge about how the ear works. Over the years, this information has been used to form analytical approximations, which replicate the operation of the different processes within the ear. Such models result in sound representations that correspond to the physical activity of the hair cells along the basilar membrane. These patterns are known as *excitation patterns* and can be used to determine the *masking pattern* for the frequencies in a sound. This work has led to the development of models of auditory perception that utilise masking patterns to evaluate sound quality in a much more subjective way. Such perceptual evaluation methods have long been adopted by the speech compression community (Brandenburg, 1987), and are an important discovery necessary for the invention of lossy compression standards such as MP3.

In Appendix B we detail the psychoacoustic model used to create our masking patterns. The model is based on the PEAQ algorithm (Perceptual Evaluation of Audio Quality (Thiede, 1999)), which is an internationally recognised standard for the measurement of perceived audio quality.

3.1.5 Noise-to-Mask Ratio

The importance of the masking pattern is that it indicates the tone detection threshold for each frequency at each point in time. Signals that are beneath this threshold are inaudible to the listener. In the context of speech processing, a masking pattern can be used to formulate a perceptual evaluation measure called the *Noise-to-Mask Ratio* (NMR); the noise-to-mask ratio is calculated in the Bark domain, and measures the level of noise between a reference signal and its estimate, in relation to the masking pattern for the

reference signal:

$$\text{NMR} = \sum_{kn} \frac{[(\mathbf{A} - \mathbf{B})^2]_{kn}}{[\mathbf{M_A}]_{kn}}, \tag{3.4}$$

where $\mathbf{A}$ is the magnitude spectra of the reference signal, $\mathbf{B}$ is the magnitude spectra of the estimate and $\mathbf{M_A}$ is the masking pattern for $\mathbf{A}$; perceptual performance increases with decreasing NMR. Furthermore, NMR can be used as an objective to be minimised, with the result that the reconstruction noise, which is ordinarily spread throughout the spectrum, is concentrated in the areas of the spectrum that are masked.

3.2 The NMF Objective

An important component in the formulation of the NMF algorithm is the comparison of the matrix to be factorised ($\mathbf{V}$) and its reconstructed estimate ($\mathbf{WH}$). This comparison is performed using an objective function that utilises some function of dissimilarity between the two, and ensures non-negativity. The NMF algorithm minimises the specified objective while enforcing a non-negativity constraint on the resulting factors:

$$\min_{\mathbf{W},\mathbf{H}} D(\mathbf{V}\|\mathbf{W},\mathbf{H}) \qquad \mathbf{W},\mathbf{H} \geq 0,$$

where the resultant reconstructions are characterised by the penalty scheme enforced by the objective function, which is dependent on the form of the objective.

3.2.1 Symmetry Properties of Objective Functions

The asymmetric effects displayed by simultaneous and temporal masking motivates our interest in the symmetry properties of the objective function. An objective function that displays similar asymmetry may be used to shape the reconstruction error such that it is inaudible by the human ear. We discuss two notions of symmetry, which we refer to as *metrical symmetry* and *error symmetry*.

An objective function is metrically symmetric if $D(a, b) = D(b, a)$, which is a property of metric spaces and simply states that the distance from a to b is the same as from b to a. The importance of this property is that the

objective does not specify an *a priori* reference variable, *i.e.*, a and b are treated equally. Metrically symmetric objectives are defined by a distance metric, whereas metrically asymmetric objectives are typically defined by a divergence measure. An objective function can penalise additive error more than subtractive error, or vice versa, where the error ϵ is additive when $b = a + \epsilon$ and subtractive if $b = a - \epsilon$. Additive error is penalised more than subtractive error if

$$D(a, a + \epsilon) > D(a, a - \epsilon), \qquad a > \epsilon$$

and vice versa if the inequality is reversed; both types of error are penalised symmetrically if

$$D(a, a + \epsilon) = D(a, a - \epsilon), \qquad a > \epsilon.$$

3.2.2 Objectives Under Investigation

Below we present the objectives that are investigated in our experiments, discussing both symmetry properties and statistical considerations. A selection of these objective functions are plotted in Figure 3.4.

Squared Euclidean Distance

The Squared Euclidean Distance (SED) is a measure of the ordinary distance between two points. It is metrically symmetric and penalises both additive and subtractive error equally:

$$D_{\text{SED}}(\mathbf{A}, \mathbf{B}) = \frac{1}{2} \sum_{ij} (a_{ij} - b_{ij})^2 , \tag{3.5}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the matrices to be compared, and a_{ij} and b_{ij} are the matrix elements.

Kullback-Leibler Divergence

The Kullback-Leibler Divergence (KLD) (Kullback, 1959) is a directed divergence that has its roots in information theory, and is based on the discriminant information between two distributions. It measures the log likelihood

of an observation being in one distribution over another distribution. For speech processing applications, such a likelihood measure indicates the similarity between two speech spectrograms. KLD is metrically asymmetric and penalises additive error less than subtractive error. A generalised version of KLD is commonly used for NMF,

$$D_{\mathrm{KLD}}(\mathbf{A}\|\mathbf{B}) = \sum_{ij}\Big(a_{ij}\log\frac{a_{ij}}{b_{ij}} - a_{ij} + b_{ij}\Big). \tag{3.6}$$

Itakura-Saito Divergence

The Itakura-Saito Divergence (ISD) (Itakura and Saito, 1968) is a directed divergence, which was designed as a similarity measure for speech signals. ISD was formulated from the linear predictive coding analysis equations, and corresponds to a maximum likelihood estimate for the parameters of an i-th order Gaussian autoregressive process, on asymptotically long observed frames of speech. The heritage of ISD makes it an important candidate for the NMF of speech data; ISD is metrically asymmetric and penalises additive error less than subtractive error,

$$D_{\mathrm{ISD}}(\mathbf{A}\|\mathbf{B}) = \sum_{ij}\Big(\frac{a_{ij}}{b_{ij}} - \log\frac{a_{ij}}{b_{ij}} - 1\Big). \tag{3.7}$$

Beta Divergence

The Beta Divergence (BD) (proposed as an objective for NMF by Kompass (2005); also referred to as the modified alpha divergence (Cichocki et al., 2006)) is a parameterised divergence measure that encompasses SED, KLD and ISD. For $\beta = 2$, SED is obtained; for $\beta \to 1$, the divergence tends to KLD; and for $\beta \to 0$, it tends to ISD. The choice of the β parameter depends on the statistical distribution of the data, and requires prior knowledge. The utility of this divergence is that it enables the investigation of the above measures, and the fractional divergences between them.

$$D_{\mathrm{BD}}(\mathbf{A}\|\mathbf{B},\beta) = \sum_{ij}\Big(a_{ij}\frac{a_{ij}^{\beta-1} - b_{ij}^{\beta-1}}{\beta(\beta-1)} + b_{ij}^{\beta-1}\frac{b_{ij} - a_{ij}}{\beta}\Big) \tag{3.8}$$

Noise-to-Mask Ratio

The noise-to-mask ratio is metrically asymmetric and penalises both additive and subtractive error equally. This objective utilises masking thresholds that are constructed from excitation patterns, as described in Section 3.1. The objective minimises the reconstruction error variance under the additional constraint that the energy of the estimation error is beneath the masking threshold $\mathbf{M_A}$,

$$D_{\mathrm{NMR}}(\mathbf{A}\|\mathbf{B},\mathbf{C},\mathbf{M_A}) = \sum_{bj} \frac{\left(\sum_i c_{bi}\ a_{ij} - \sum_i c_{bi}\ b_{ij}\right)^2}{m_{bj}}. \tag{3.9}$$

Here, $\mathbf{A}$ and $\mathbf{B}$ contain magnitude spectra information, and $\mathbf{C}$ performs a frequency (Hz) to Bark domain transformation, which is necessary as $\mathbf{M_A}$ is in the Bark domain.

It is important to note that the NMR objective achieves masking using the additional constraint of the masking threshold, while the proceeding objectives exhibit auditory masking properties that are purely based on signal detection.

3.2.3 NMF Algorithms

For the purposes of our investigation, we select the beta divergence as the reconstruction objective for NMF. We perform experiments using this divergence for a range of different β values, and perceptually evaluate the reconstructions. The NMF objective utilising beta divergence is

$$D_{\mathrm{BD}}(\mathbf{V}\|\mathbf{W},\mathbf{H},\beta) = \sum_{ik}\Big(v_{ik}\frac{v_{ik}^{\beta-1} - [\mathbf{WH}]_{ik}^{\beta-1}}{\beta(\beta-1)} + [\mathbf{WH}]_{ik}^{\beta-1}\frac{[\mathbf{WH}]_{ik} - v_{ik}}{\beta}\Big), \tag{3.10}$$

which results in the following update rules

$$w_{ij} \leftarrow w_{ij}\frac{\sum_{k=1}^{N}(v_{ik}/[\mathbf{WH}]_{ik}^{2-\beta})h_{jk}}{\sum_{k=1}^{N}[\mathbf{WH}]_{ik}^{\beta-1}h_{jk}}, \qquad h_{jk} \leftarrow h_{jk}\frac{\sum_{i=1}^{M} w_{ij}(v_{ik}/[\mathbf{WH}]_{ik}^{2-\beta})}{\sum_{i=1}^{M} w_{ij}[\mathbf{WH}]_{ik}^{\beta-1}}. \tag{3.11}$$

We also investigate convolutive NMF: Replacing $\mathbf{WH}$ in Eq. 3.10 with the convolutive generative model, $\mathbf{\Lambda} = \sum_{t=0}^{T_o-1}\mathbf{W}_t \overset{t\rightarrow}{\mathbf{H}}$, results in the following

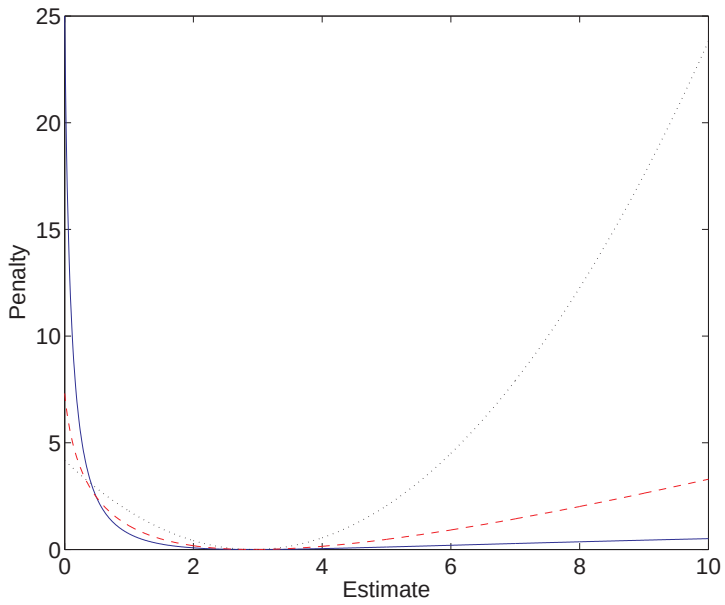

Figure 3.4: Plot of NMF objective functions: Solid line: Itakura-Saito divergence; Dashed line: Kullback-Leibler divergence; Dotted line: Squared Euclidean distance. The curves indicate the penalty scheme imposed by each objective, where the reconstruction error is represented on the x-axis, while the associated penalty is represented on the y-axis. Here, the reference variable is 3, where an estimate of 3 has no penalty.

update rules

$$w_{ijt} \leftarrow w_{ijt} \frac{\sum_{k=1}^{T} (v_{ik} / [\mathbf{\Lambda}]_{ik}^{2-\beta}) \overset{t\rightarrow}{h}_{jk}}{\sum_{k=1}^{T} [\mathbf{\Lambda}]_{ik}^{\beta-1} \overset{t\rightarrow}{h}_{jk}}, \qquad h_{jk} \leftarrow h_{jk} \frac{\sum_{i=1}^{M} w_{ijt} (v_{ik} / [\mathbf{\Lambda}]_{ik}^{2-\beta})}{\sum_{i=1}^{M} w_{ijt} [\overset{\leftarrow t}{\mathbf{\Lambda}}]_{ik}^{\beta-1}}. \tag{3.12}$$

As a means of providing a comparative benchmark for an *ideal* perceptual performance, we investigate an NMF algorithm that utilises NMR as its reconstruction objective;

$$D_{\text{NMR}}(\mathbf{V} \| \mathbf{W}, \mathbf{H}, \mathbf{C}, \mathbf{M}_{\mathbf{V}}) = \sum_{bk} \frac{\left(\sum_i c_{bi}\ v_{ik} - \sum_i c_{bi}\ [\mathbf{WH}]_{ik} \right)^2}{m_{bk}}, \tag{3.13}$$

which results in the updates

$$w_{ij} \leftarrow w_{ij} \frac{[\mathbf{C}^{\mathsf{T}} (\mathbf{M}_{\mathbf{V}}^{-1} \otimes (\mathbf{CV})) \mathbf{H}^{\mathsf{T}}]_{ij}}{[\mathbf{C}^{\mathsf{T}} (\mathbf{M}_{\mathbf{V}}^{-1} \otimes (\mathbf{CWH})) \mathbf{H}^{\mathsf{T}}]_{ij}}, \tag{3.14a}$$

NMF with Beta Divergence

```
Obj=sum(sum((V.*((((V+1E-9).^(b-1))-((W*H+1E-9).^(b-1))).*(b*(b-1)+1E-9))
(((W*H+1E-9).^(b-1)).*(((W*H)-V).*(b+1E-9))))));
```

```
W=W.*((V./(W*H+1e-9).^(2-b))*H').*((W*H+1e-9).^(b-1)*H');
H=H.*(W'*(V./(W*H+1e-9).^(2-b))).*(W'*(W*H+1e-9).^(b-1));
```

NMF with NMR Reconstruction Objective

```
Obj=0.5*sum(sum((C*V-C*W*H).^2./Mv));
```

```
Minv=Mv.^(-1);
W=W.*(C'*(Minv.*(C*V))*H').*(C'*(Minv.*(C*W*H))*H'+1e-9);
H=H.*((C*W)'*(Minv.*(C*V))).*((C*W)'*(Minv.*(C*W*H))+1e-9);
```

Convolutive NMF with Beta Divergence

```
Obj=sum(sum((V.*((((V+1E-9).^(b-1))-((lambda+1E-9).^(b-1))).*(b*(b-1)+1E-9))
(((lambda+1E-9).^(b-1)).*(((lambda)-V).*(b+1E-9))))));
```

```
for t=1:To
    Vt(:,:,t)=(W(:,:,t)*padshift(H,t-1));
end
lambda=sum(Vt,3);

for t=1:To
    Hs=padshift(H,t-1);
    W(:,:,t)=W(:,:,t).*((V./(lambda+1e-9).^(2-b))*Hs').*((lambda).^(b-1)*Hs'+1e-9);
end

for t=1:To
    Qs=padshift((V./(lambda).^(2-b)),-(t-1));
    Ps=padshift((lambda.^(b-1)),-(t-1));
    Ht(:,:,t)=H.*(W(:,:,t)'*Qs).*((W(:,:,t)'*Ps)+1e-9);
end
H=mean(Ht,3);
```

Figure 3.5: Matlab notations for NMF algorithms using beta divergence and NMR.

$$h_{jk} \leftarrow h_{jk} \frac{[(\mathbf{CW})^{\mathsf{T}}(\mathbf{M}_{\mathbf{V}}^{-1} \otimes (\mathbf{CV}))]_{jk}}{[(\mathbf{CW})^{\mathsf{T}}(\mathbf{M}_{\mathbf{V}}^{-1} \otimes (\mathbf{CWH}))]_{jk}}, \tag{3.14b}$$

where $\otimes$ denotes an element-wise multiplication, and $\mathbf{C}$ is a $N_c \times M$ matrix ($N_c < M$) that performs Bark domain transformation (Section B.4). By including the frequency grouping transformation in the objective, we avoid an inverse transformation from the Bark to magnitude spectral domain. Matlab notations for these algorithms are presented in Figure 3.5.

3.3 Experiments

In this section, we detail the procedure used in our investigation and discuss our results. We perform a number of experiments on a speech signal comprised of both male and female speakers, which is constructed as follows: 10 sentences are randomly selected from the TIMIT (Garofolo et al., 1993) speech database (5 male 5 female) producing 30 seconds of monophonic speech, which is subsequently down-sampled from 16 kHz to 8 kHz. The speech signal is calibrated to an intensity of 70 $\mathrm{dB_{SPL}}$, which corresponds to conversation-level speech, and a magnitude spectrogram is created: The signal is framed into overlapping blocks of a specified size and windowed by a hamming function, a short-term FFT is performed on each block and a magnitude spectrogram, $\mathbf{V}$, is constructed.

3.3.1 Conventional NMF

For our experiments, a number of parameters remain constant throughout. These include an FFT frame advance of half the FFT frame size, and an algorithm run limit of 150 iterations. For our variable parameters we use FFT frame sizes of 128, 256, 512, 1024 & 2048, where $M = \frac{\text{FFT size}}{2} + 1$, with the number of objects, R, specified as a fraction of M, $R = \{\frac{M}{16}, \frac{M}{8}, \frac{M}{4}, \frac{M}{2}, M\}$. The NMF algorithm is tested for a range of β values between 0 and 2, where β increases in steps of 0.1 resulting in 21 different values. As the experiment is run, the factors $\mathbf{W}$ & $\mathbf{H}$ are obtained from the algorithm every 5 iterations, and are used to create an estimate of $\mathbf{V}$. The reconstruction quality for each estimate is measured using the total noise-to-mask ratio,

$$\mathrm{NMR_{tot}} = 10\log_{10}\left[\frac{1}{T}\sum_{k=1}^{T}\left(\frac{1}{M}\sum_{i=1}^{M}\frac{([\mathbf{V}]_{ik} - [\mathbf{WH}]_{ik})^2}{[\mathbf{T}_V]_{ik}}\right)\right] \text{(dB)}, \qquad (3.15)$$

and the signal-to-noise ratio,

$$\mathrm{SNR} = 20\log_{10}\left[\frac{\|\mathbf{V}\|_{\mathrm{fro}}}{\|\mathbf{V} - (\mathbf{WH})\|_{\mathrm{fro}}}\right] \text{(dB)}. \qquad (3.16)$$

In this way, we observe the performance for each β as the algorithm converges. This process is repeated for 5 Monte Carlo runs resulting in a total of 2625 $(5 \times 5 \times 21 \times 5)$ experiments.

Table 3.2: Conventional NMF: β Values for Optimal NMR_{tot} and SNR.

	NMR_{tot} $R = M \times$					SNR $R = M \times$				
	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{2}$	1	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{2}$	1
128	0.7	0.8	0.8	0.7	0.6	2.0	1.8	1.6	1.3	0.9
256	1.0	1.0	0.9	0.8	0.7	1.9	1.8	1.6	1.3	1.1
FFT 512	1.2	1.0	1.0	0.9	0.9	2.0	1.9	1.6	1.6	1.3
Size 1024	1.1	1.2	1.2	1.0	0.9	1.8	1.8	1.6	1.6	1.3
2048	1.1	1.2	1.1	0.8	0.7	1.7	1.6	1.4	1.0	0.9
4096	1.1	0.8	0.7	0.7	0.5	1.2	1.0	1.0	0.7	0.7

Results

The results from the investigation are collated and presented in two figures: Figure 3.6 contains six rows and five columns of NMR_{tot} performance surfaces, where each row represents a different FFT size and each column a different R. For each surface the x-axis represents β, the y-axis represents iterations and the z-axis is the NMR_{tot} expressed in dB. Figure 3.7 provides corresponding SNR surfaces—note that in order to properly view the SNR surfaces the scales on the x-axis and y-axis are the reverse of Figure 3.6.

From inspection of the NMR_{tot} results we see that for each FFT size as R increases, the performance surface becomes more concave and the surface minima becomes more defined; it is also evident that the surface minima travels towards $\beta = 0$. As FFT size increases from 128 to 1024 for all R there is a slight degradation of performance, but essentially the results vary very little, for FFT size = 2048 and 4096 there is a jump in minimum NMR. This jump in performance may be due to the statistics of our speech data: Using the reduced 39 phoneme symbol set (Lee and Hon, 1989) our speech data contains 369 phonemes (not including silences) with a maximum length of 182 ms, an average length of 80 ms and minimum length of 25 ms. At these FFT sizes the length of each object is 256ms and 512ms respectively, which is larger than our maximum phoneme length. The extensive length of the FFT size, along with a comparatively large R, allow for a more accurate estimation as our auditory objects are represented by fewer basis vectors. Similarly, a corresponding performance jump is evident from the SNR surfaces.

In order to find the optimal β for each surface, we calculate the average NMR_{tot} value for each β over all iterations, and select the β value that corresponds to the minimum NMR_{tot}. We use the same procedure for SNR, this time selecting β that corresponds to the maximum SNR. The resultant optimal parameters for NMR_{tot} and SNR are presented in Table 3.2.

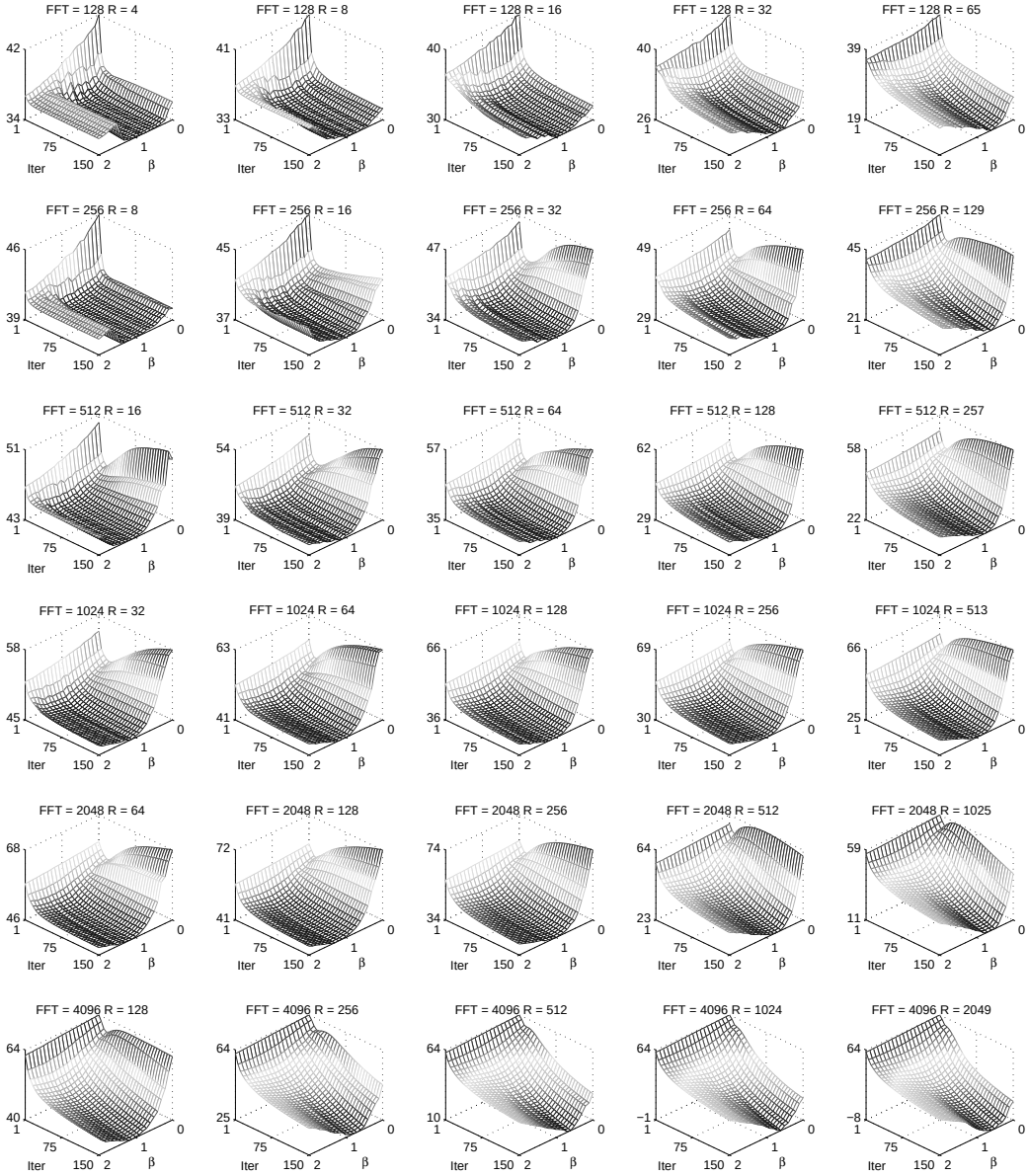

Figure 3.6: NMR_{tot} performance surfaces for the NMF algorithm, where smaller values indicate better performance. Each row represents a different FFT size, and each column represents a different R. For each plot the x-axis represents β, the y-axis represents iterations and the z-axis is the NMR_{tot}, which is expressed in dB. Note that the scales for the z-axis change for each plot. It is evident that for each FFT size as R increases, perceptual performance increases and the minima travels towards $\beta = 0$.

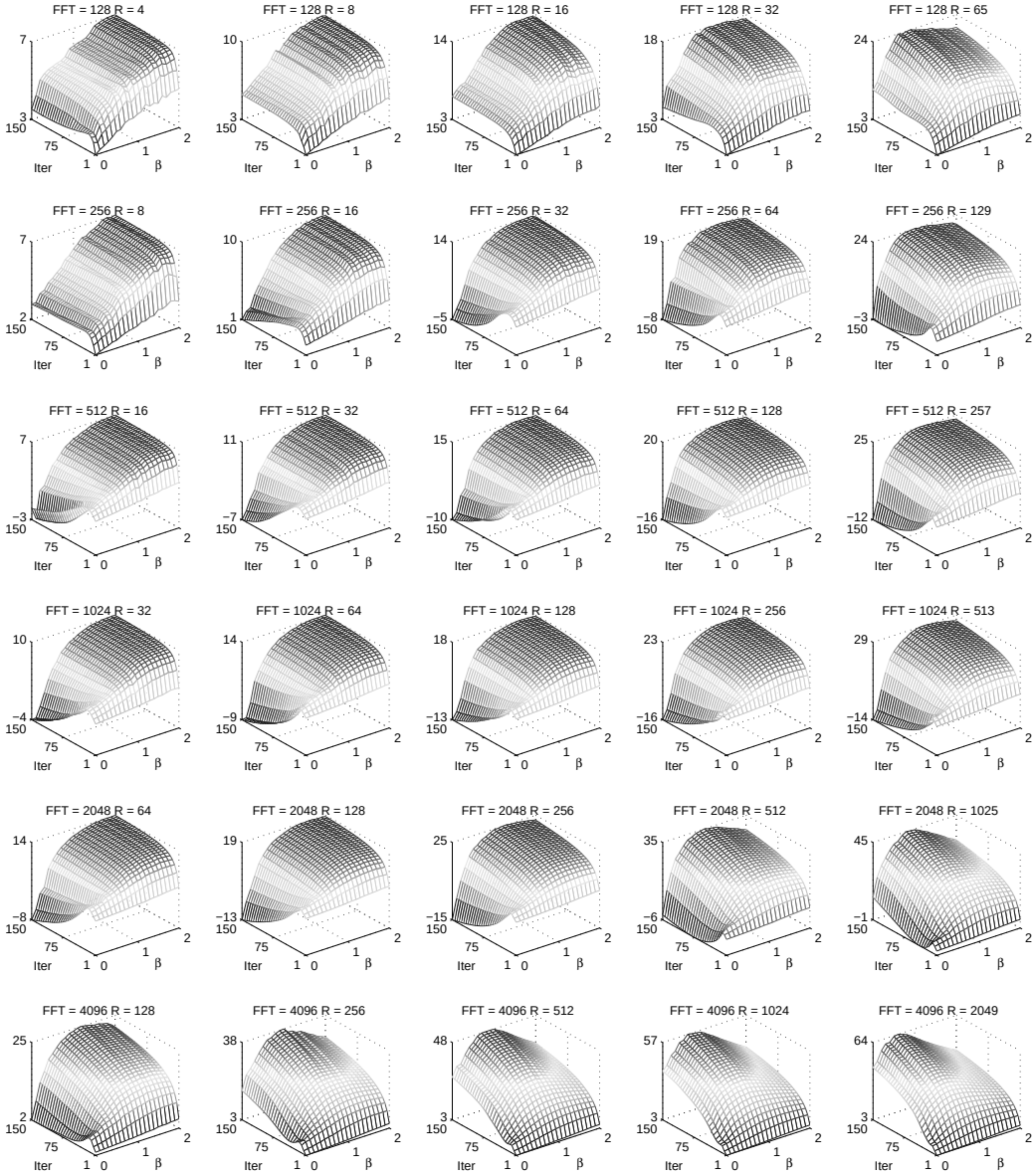

Figure 3.7: SNR performance surfaces for the NMF algorithm. Each row represents a different FFT size, and each column represents a different R. For each plot the x-axis represents β, the y-axis represents iterations and the z-axis is the SNR expressed in dB. Note that the scales for the z-axis change for each plot. In contrast to the NMR_{tot} performance surfaces, the SNR surface maximum, which indicates optimal reconstruction, is positioned at a different β . Although, the optimal β does exhibit the same drift towards 0 as R increases.

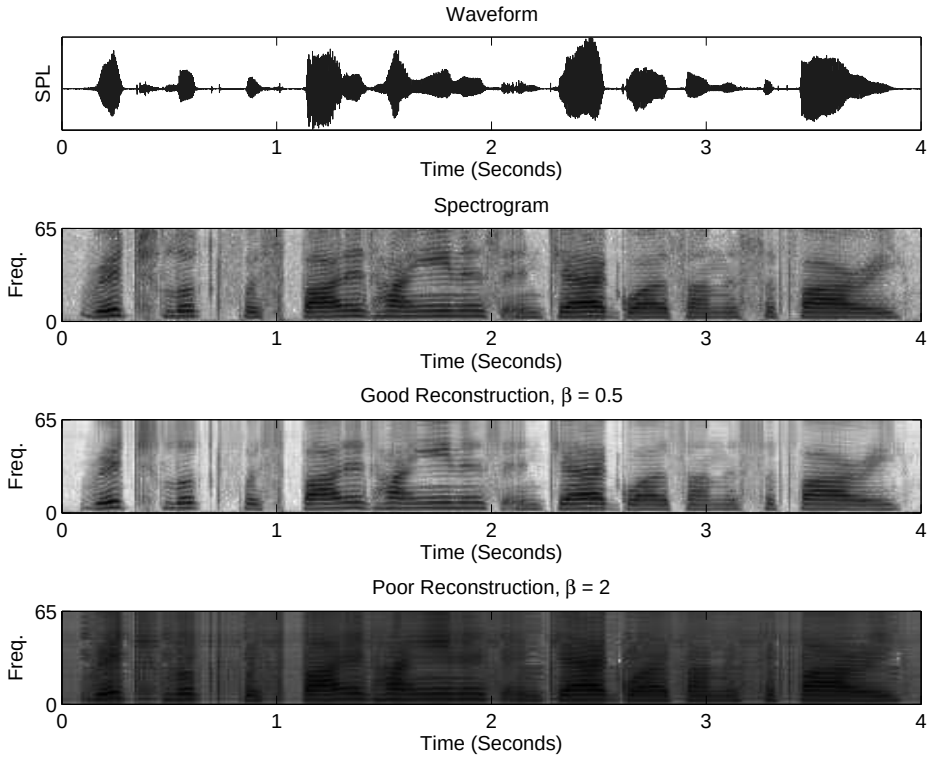

Figure 3.8: NMF reconstructions for a sentence (SX181) from a female speaker (SMA0). Row 1 & 2 contain the speech waveform and its log-power spectrogram, row 3 & 4 contain the NMF estimates using both *good* and *poor* selections for β. It is evident that when $\beta = 0.5$, the features of the original spectrogram are better preserved in the reconstruction.

NMF Reconstructions

To illustrate the performance of the algorithm with an optimal β parameter, we perform an experiment on a randomly chosen female (speaker: SMA0, sentid: SX181) and male (speaker: DMT0, sentid: SX302) sentence from the TIMIT database. The chosen speech segments have different speakers and sentences to those used in our beta divergence investigation. We perform the same preprocessing and use the same constant parameters specified previously. Variable parameters are set to the following values: FFT size = 128 and $R = 32$. For the selected algorithm parameters, Figure 3.6 indicates that $\beta = 0.5$ provides superior perceptual quality over $\beta = 2$, we use these values in our experiments. The reconstructions for the female and male experiments are presented in Figure 3.8 and Figure 3.9 respectively. Both figures contain the waveform and the log-power spectrogram of the sentence,

Table 3.3: Reconstruction Experiment Results.

	β	SNR (dB)	NMR_{tot} (dB)
Male	0.5	18	25
	2	20	29
Female	0.5	17	25
	2	22	29

along with the NMF reconstructions for the specified β values. From the results presented in Table 3.3 we can see that in both cases $\beta = 0.5$ provided the best NMR_{tot}. It is also evident that if we were to measure the quality of reconstruction by using the SNR measure, the experiment would indicate that the reconstruction with poorer perceptual performance provides better quality.

In order to subjectively validate these results, an audible reconstruction of the estimate is created. This can be achieved by combining the magnitude spectrum of the NMF estimate with the phase of the original input (which represents a Polar form of the complex FFT coefficients) and returning to Cartesian form where an inverse FFT transformation can be performed. The resultant waveform exhibits perfect phase, and its quality is uniquely dependant on the magnitude spectrum provided by NMF. Therefore, subjective listening tests should indicate the better β selection. In both of our experiments, the β values that indicated better perceptual performance exhibited superior subjective performance, thus providing support for our proposed β values.

The constituent components of speech are evident from the spectrograms in Figure 3.8 and Figure 3.9. These components are known as *phones* and are composed of harmonic series with various pitch inflections or wideband spectra. Comparison of the waveform spectrograms with their reconstruction estimates indicate the properties necessary for an objective function to exhibit good perceptual performance. It is evident that for the $\beta = 0.5$ the formants (harmonic peaks) in the speech are more closely preserved. This is especially evident for the female sentence at lower frequencies where there is more energy. The importance of this sensitivity to formant energy is that if these frequencies are properly represented in the reconstruction, the masking properties of such frequencies are elevated, resulting in heightened masking of the approximation error when perceived by a listener. In this way, the ob-

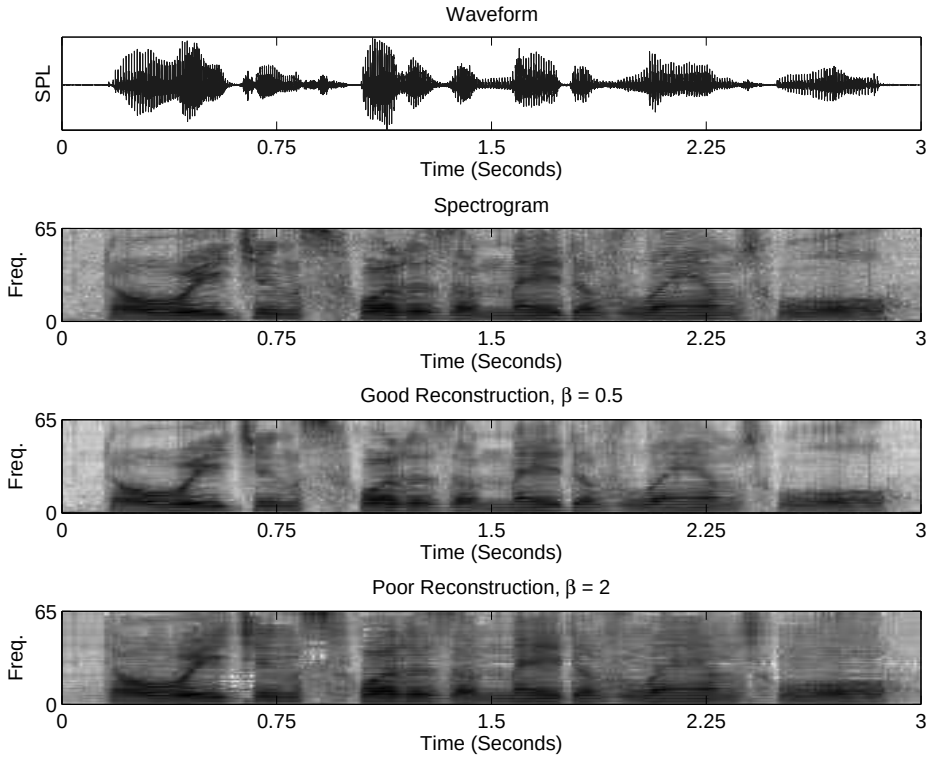

Figure 3.9: NMF reconstructions for a sentence (SX302) from a male speaker (DMT0). Row 1 & 2 contain the speech waveform and its log-power spectrogram, row 3 & 4 contain the NMF estimates using both *good* and *poor* selections for β. It is evident that when $\beta = 0.5$, the features of the original spectrogram are better preserved in the reconstruction.

jective exhibits auditory masking properties that are based solely on signal detection.

3.3.2 NMF Using NMR as an Objective

By way of comparison, we repeat the experiments of the proceeding section for an NMF algorithm that uses NMR as its reconstruction objective (Eq. 3.14). The experiment parameters remain the same except for the β values, which are not used in this context. Convergence curves that indicate the NMR_{tot} performance of the algorithm at each iteration are presented in Figure 3.11, and can be contrasted with the performance surfaces in Figure 3.6. It is evident that the rate of convergence for our perceptually weighted algorithm is much slower than for our previous experiments, and therefore requires more iterations to achieve a NMR_{tot} similar to those observed in

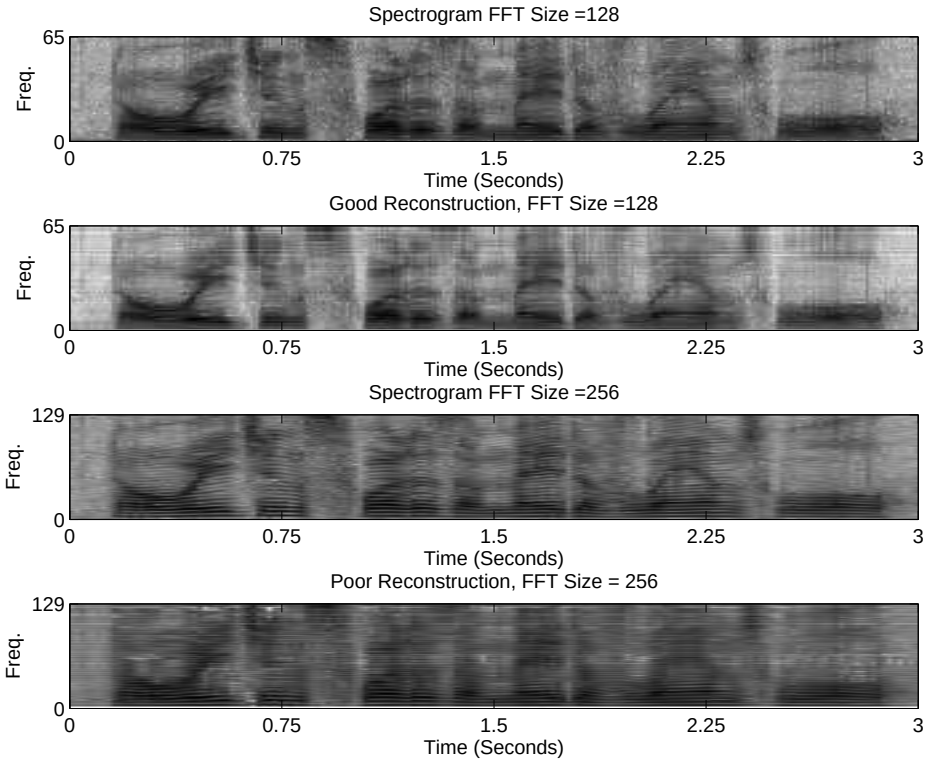

Figure 3.10: Reconstructions for NMF using NMR as an objective, where $N_c = 67$. A log-power spectrogram of a sentence from a male speaker using an FFT frame size of 128 is presented in row 1, while its reconstruction is presented in row 2. A spectrogram and reconstruction for an FFT size of 256 is presented in row 3 & 4, respectively. It is evident that when FFT frame size is much larger than N_c, as is the case for row 4, the reconstruction produced spreads the energy in the Bark domain over multiple FFT bins (reverse of frequency grouping), resulting in artifacts in its audible reconstruction.

Figure 3.6. Overall, the results indicate that NMF utilising beta divergence, and an appropriately selected β, results in faster convergence and comparable NMR_{tot} performance to the perceptually weighted algorithm. However, the perceptually weighted algorithm does not require the selection of additional parameters, and will produce better results if run for long enough.

There is an important caveat in relation to audible reconstructions: Since the noise-to-mask ratio objective is calculated in the Bark domain, and $\mathbf{V}$ is in the magnitude spectral domain, an implicit inverse frequency grouping procedure is performed. For our data, where the sample frequency is 8 kHz, the energy in each FFT frame is grouped into 67 critical bands, $N_c = 67$ (Appendix B). Since, the Bark domain transformation is specified by an under-determined matrix, the inverse frequency grouping procedure is ill-

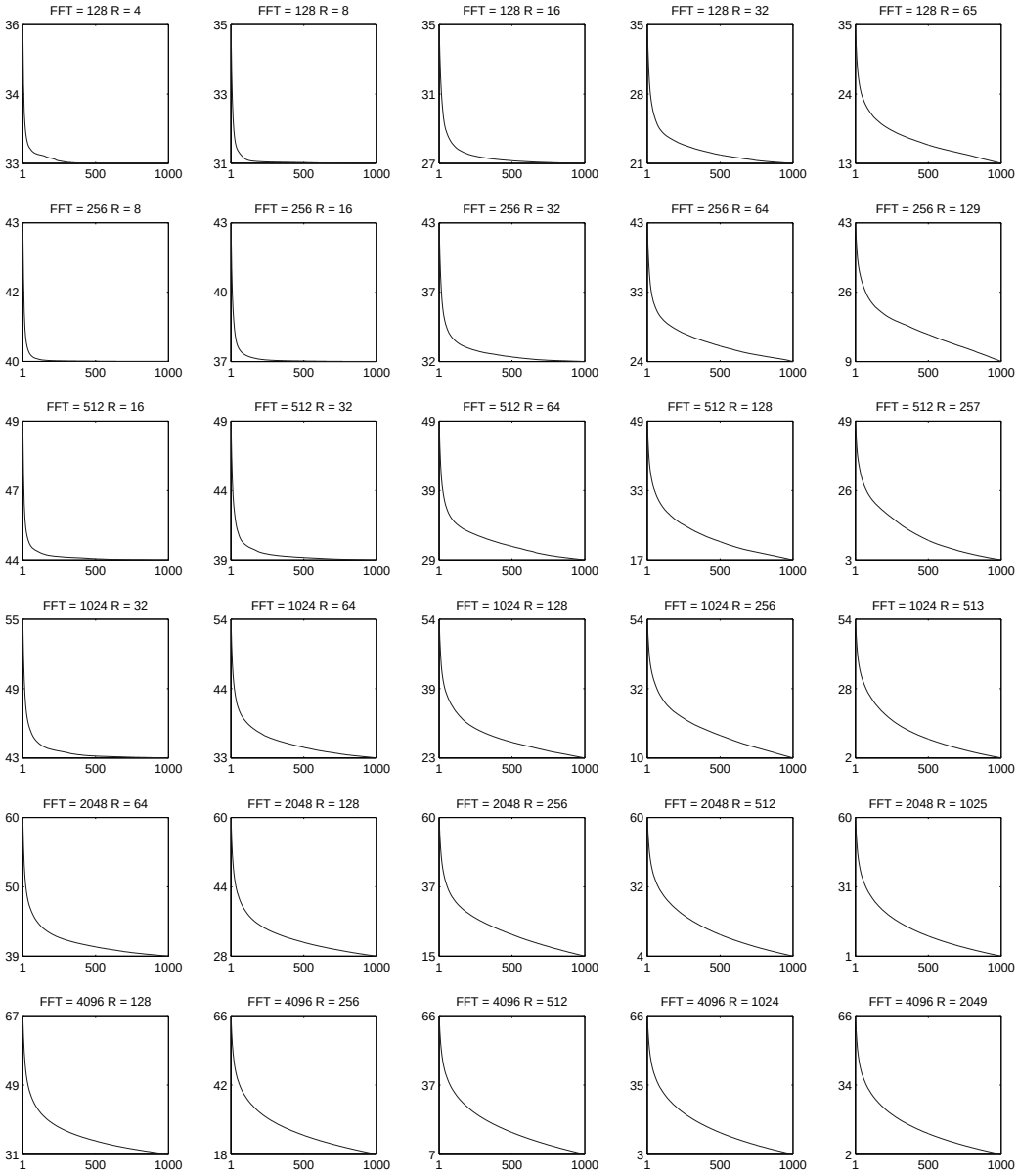

Figure 3.11: NMR_{tot} performance curves for NMF using NMR as an objective. Each row represents a different FFT size, and each column represents a different R. For each plot the x-axis represents iterations and the y-axis represents NMR_{tot}, which is expressed in dB. It is evident that the additional constraints introduced by the masking threshold scaling has a degenerative effect on the convergence rate of the algorithm, with the effect worsening as FFT Size and R increases. However, when the algorithm is run for enough iterations, it produces very good NMR_{tot} results.

Table 3.4: Convolutive NMF: β Values for Optimal NMR_{tot}.

		$R =$				
		20	40	80	120	240
FFT Size	128	0.8	0.8	0.7	0.7	0.7
	256	1.0	1.0	0.9	0.9	0.8
	512	1.0	1.2	1.1	1.0	0.9
	1024	1.3	1.3	1.2	1.1	1.1
	2048	1.4	1.3	1.3	1.3	1.1

defined and results in energy spreading when returning to the magnitude spectral domain, which produces artifacts in the audible reconstructions. This effect is more evident the larger FFT size is in relation to N_c, and may be ameliorated by decreasing FFT size (as illustrated in Figure 3.10), or increasing the critical band resolution in the Bark domain.

3.3.3 Convolutive NMF

The additional computation required for the convolutive NMF algorithm (effectively T_o instances of conventional NMF per reconstruction), together with the computationally intensive model used to create the masking patterns, make an extensive perceptual evaluation of the convolutive beta divergence NMF algorithm prohibitively time consuming. In contrast to our investigation for conventional NMF, in which we explored a wide parameter space, we fix T_o to a minimum of 0.176 seconds and investigate the perceptual performance of the algorithm using the following variable parameters: FFT size $= \{128, 256, 512, 1024, 2048\}$, $R = \{20, 40, 80, 120, 240\}$ and an FFT frame advance of half the FFT size. The FFT size and frame advance used affect our ability to achieve the desired T_o; consequently, we use $T_o = \{21, 10, 5, 2, 1\}$ (in order of corresponding FFT size), which results in the following temporal extents: $0.176, 0.176, 0.192, 0.192, 0.256\}$ (expressed in seconds). We use the same data and β values as those used in the conventional NMF investigation, and repeat each experiment for 5 Monte Carlo runs.

Results

NMR_{tot} results are presented in Figure 3.12, which contains a performance plot for each experiment, where each row represents a different FFT size and

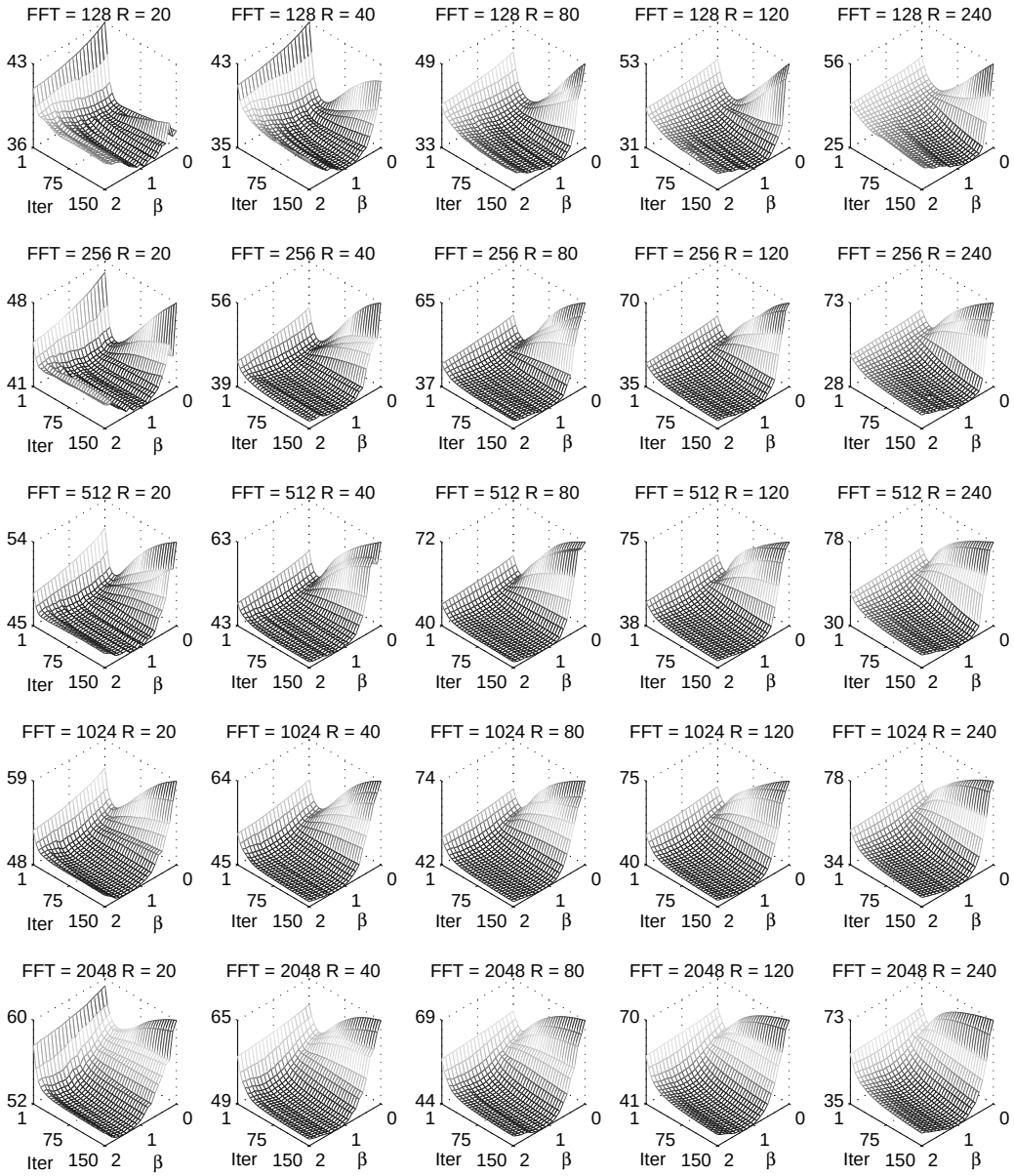

Figure 3.12: NMR_{tot} performance surfaces for convolutive NMF algorithm. Each row is for a different FFT size, and each column in the row is for a different R. For each plot the x-axis represents β, the y-axis represents iterations and the z-axis is the NMR_{tot}, which is expressed in dB. Note that the scales for the z-axis change for each plot. It is evident that for each FFT size as R increases the performance surface becomes more defined and travels towards $\beta = 0$.

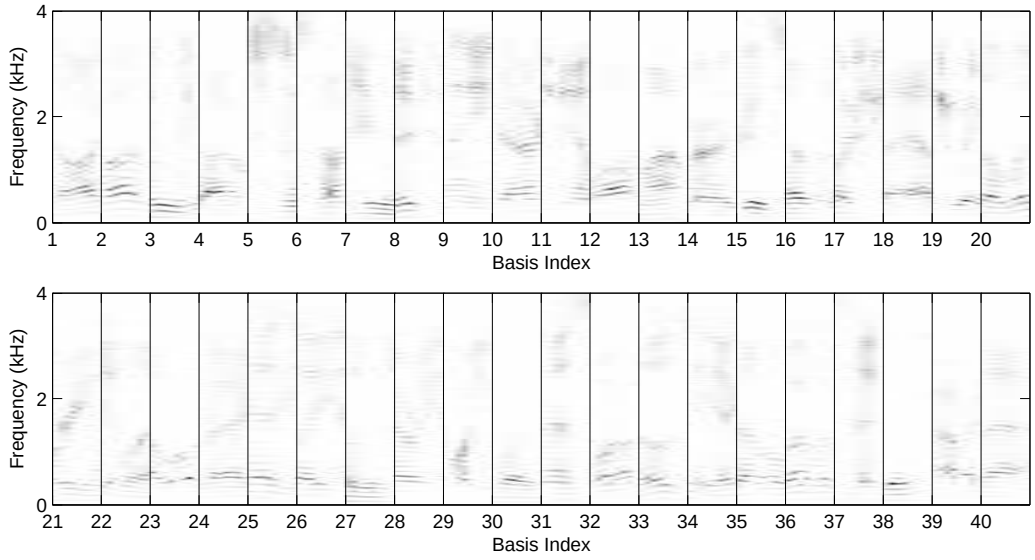

Figure 3.13: Auditory objects discovered from the magnitude spectrum of mixed speech using a *good* ($\beta = 1$) selection for β.

each column a different R. The optimal β values for each plot are presented in Table 3.4. We can see from the results, that convolutive NMF behaves in a similar way to conventional NMF, where β drifts towards 0 as the number of objects, R, increases. In General, for both investigations, the smaller the FFT size and the larger R is, the closer optimal β is to 0, with $\beta = 0.5$ being the smallest value encountered overall.

To illustrate the performance of convolutive NMF using the beta divergence objective, auditory objects that exhibit good and bad perceptual performance are extracted by the algorithm, where $\beta = 1$ (Figure 3.13) and $\beta = 0$ (Figure 3.14) respectively. We can see by inspection that for $\beta = 0$ the auditory objects are characterised by dark dots as opposed to phonetic pitch inflections ($\beta = 1$). This effect indicates that for our algorithm parameters, the penalty scheme of the Itakura-Saito divergence, *i.e.*, weak penalty for overestimation, results in some features dominating others. Furthermore, audible reconstructions reveal the presence of intermittent musical noise, where tonal elements in the phones are over overestimated.

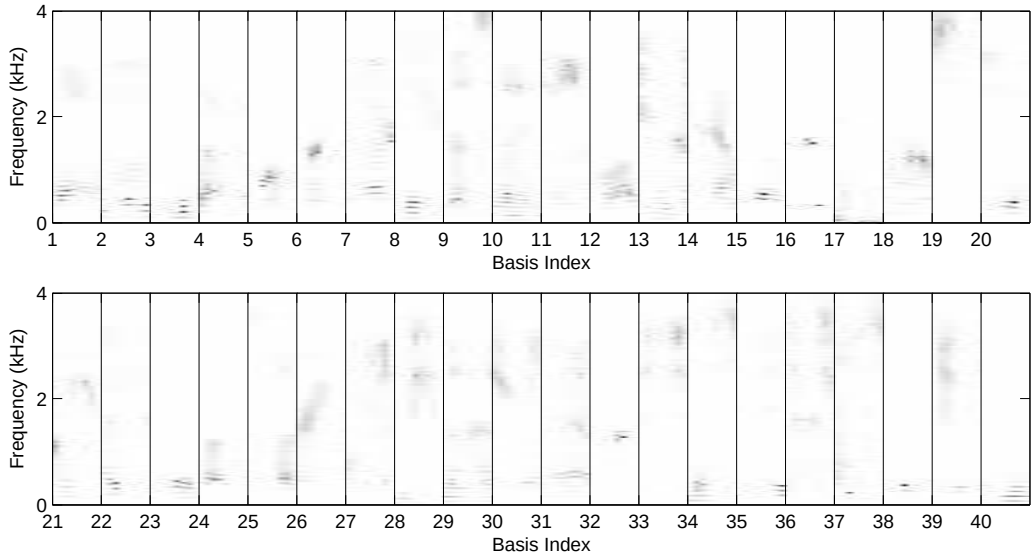

Figure 3.14: Auditory objects discovered from the magnitude spectrum of mixed speech using a *poor* ($\beta = 0$) selection for β.

3.4 Discussion

The quality of the reconstructions produced by the NMF algorithm ultimately depends on two parameters: The number of auditory objects to discover, as specified by R, and the number of iterations of the algorithm. Assuming that these parameters are fixed at appropriate values for the data under consideration, the selection of an appropriate reconstruction objective becomes important. For perceptible data such a speech, the subjective quality of NMF reconstructions can be improved by using an objective function that preserves perceptually important features. In the case of the NMF algorithm, the presence of such features in the reconstruction elevates the masking threshold associated with the features, which results in increased masking of the approximative error. This is auditory masking based purely on accurate signal detection.

Through our investigation of the beta divergence, we have endeavoured to illustrate the perceptual properties of the Squared Euclidean distance, the Kullback-Leibler divergence, the Itakura-Saito divergence, and the fractional divergences between them. As indicated by Figure 3.6, the optimal β parameter is dependant on the FFT size and R. The performance plots also reveal that the rate of convergence is essentially the same for our selected range of β values. It is evident from the $\mathrm{NMR}_{\mathrm{tot}}$ performance surfaces that, for selected

FFT sizes and R, the slope towards $\beta = 0$ may become very steep resulting in poor NMR_{tot} performance. Audible reconstructions reveal the presence of musical noise and loud tonal sounds, which is a result of the relatively low penalty for additive error enforced by the Itakura-Saito divergence. It is worth noting that optimal β, presented in Table 3.2, never exceeds 1.2; this may be due to the fact that the human ear perceives loudness in a semilogarithmic way, which is modelled by the log likelihood measures utilised in the beta divergence. As β tends towards 2 the log terms in the divergence vanish and the divergence loses its ability to model this psychoacoustic phenomena.

It is evident from the performance surfaces that the optimal β value can be determined after only a few iterations. A practical approach that can be used to select an appropriate β parameter, which is specific to the data under consideration, is to run the algorithm for a small number of iterations over a range of β parameters. This preliminary step indicates an appropriate β that can be used in subsequent experiments. It may be tempting to suggest that an exhaustive investigation such as ours can be avoided by learning an optimal β directly from the data using a Bayesian approach. However, such methods cannot be directly applied since β is not a parameter in a statistical model but rather a parameter that defines a divergence.

The versatility of the beta divergence is that by modulating the β parameter we modify the symmetry of the divergence, which enforces a different penalty scheme for subtractive and additive error. From our results, we can see that penalising subtractive and additive error equally results in reconstructions that are not of optimal perceptual quality. It is also evident that for the accurate comprehension of speech, it is acceptable to use a penalty scheme that enforces a weak penalty for overestimation, which is justified by the fact that it is more important to have some portion of a phonetic feature present as opposed to it being absent.

By way of comparison, we presented an NMF algorithm that utilises the masking patterns of the input data as an additional constraint. Such an approach requires additional processing to create the masking patterns, which are created using sophisticated psychoacoustic models, and detract from NMF's ease of implementation.

3.5 Conclusion

We have demonstrated the utility of the proposed beta divergence for the purposes of the NMF of speech spectrograms. By creating auditory masking patterns for our input data, and measuring the perceptual quality of NMF reconstructions using the noise-to-mask ratio, we have illustrated the reconstruction performance of the NMF algorithm for a range of β values. Furthermore, we have compared our results to a perceptually weighted NMF algorithm that utilises the noise-to-mask as its reconstruction objective. Finally, due to the fact that so many variables influence the selection of an optimal β, and the fact that terms involving fractional powers require additional computation, for most applications it may be better to resort to the Kullback-Leibler Divergence, as it has performed better than both the Squared Euclidean Distance and Itakura-Saito Divergence throughout our investigation.

CHAPTER 4

Convolutive NMF with a Sparseness Constraint

Discovering a representation that allows auditory data to be parsimoniously represented is useful for many machine learning and signal processing tasks. Such a representation can be constructed by Non-negative Matrix Factorisation (NMF) (Section 1.3). For some tasks it may be advantageous to perform NMF with additional constraints placed on either **W** or **H**. One increasingly popular and powerful constraint is that the rows of **H** have a parsimonious activation pattern for the basis contained in the columns of **W**. This is the so called *sparseness constraint* (Field, 1994; Olshausen and Field, 2004). Moreover, the addition of a sparseness constraint enables the discovery of an over-complete basis.

Although convolutive NMF produces activation patterns that tend to be sparse, the addition of the sparseness constraint on **H** provides a means of trading off the sparseness of the representation against accurate reconstruction. Previous algorithms for sparse NMF (Hoyer, 2002; Virtanen, 2003; O'Grady and Pearlmutter, 2006) have suffered from the scaling problem associated with the addition of a sparse constraint on **H**. In order for the algorithm to behave as required, an additional normalisation step on **W** is needed. As will be discussed in Section 4.1.2, this may result in **W** having an additive update rule. We overcome this restriction by using a normalised version of **W** explicitly in the reconstruction objective, and present an algo-

[6]Some material in this chapter appeared in O'Grady and Pearlmutter (2006)

rithm that has multiplicative updates for both $\mathbf{H}$ and $\mathbf{W}$. Furthermore, the proposed algorithm uses the beta divergence as its reconstruction objective, which we have shown to be versatile when used in NMF algorithms that are applied to speech (Chapter 3).

We extract phones from speech using both convolutive NMF and sparse convolutive NMF and apply them to a supervised separation scheme for monophonic mixtures. In contrast, the LOST algorithm is not an appropriate method for the separation of speakers from a monophonic recording as it requires a scatter plot of the observations to be constructed, which is not possible here as there is only one observation.

This chapter is organised as follows: In Section 4.1 we discuss convolutive NMF with an additional sparseness constraint on $\mathbf{H}$, and present an algorithm that has multiplicative updates. In Section 4.2 we apply sparse convolutive NMF to speech data, and demonstrate its utility in the extraction of speech phones. We apply such phone sets to a monophonic mixture separation task in Section 4.3, and discuss their use in speech coding in Section 4.4. We complete the chapter with a discussion and conclusion.

4.1 Sparse Convolutive NMF

The specifics of the convolutive NMF algorithm are presented in Section 1.3.2. For our sparse convolutive NMF algorithm, we use the beta divergence as the reconstruction objective,

$$D_{\mathrm{BD}}(\mathbf{V}\|\mathbf{\Lambda},\beta)=\sum_{ik}\Big(v_{ik}\frac{v_{ik}^{\beta-1}-[\mathbf{\Lambda}]_{ik}{}^{\beta-1}}{\beta(\beta-1)}+[\mathbf{\Lambda}]_{ik}{}^{\beta-1}\frac{[\mathbf{\Lambda}]_{ik}-v_{ik}}{\beta}\Big), \qquad (4.1)$$

where β controls the reconstruction penalty and $\mathbf{\Lambda}$ is the current estimate of $\mathbf{V}$,

$$\mathbf{\Lambda}=\sum_{t=0}^{T_o-1}\mathbf{W}_t\ \overset{t\rightarrow}{\mathbf{H}};$$

the j-th column of $\mathbf{W}_t$ describes the spectrum of the j-th object t time steps after the object has begun, $\overset{i\rightarrow}{(\cdot)}$ denotes a column shift operator and T_o is the length of each object sequence. Combining our reconstruction objective (Eq. 4.1) with a sparseness constraint on $\mathbf{H}$ results in the following objective

function:

$$G(\mathbf{V}\|\mathbf{\Lambda},\mathbf{H},\beta) = D_{\mathrm{BD}}(\mathbf{V}\|\mathbf{\Lambda},\beta) + \lambda \sum_{jk} h_{jk}, \tag{4.2}$$

where the left term of the objective function corresponds to convolutive NMF, and the right term is an additional constraint on $\mathbf{H}$ that enforces sparsity by minimising the L_1-norm of its elements. The parameter λ controls the trade off between sparseness and accurate reconstruction.

4.1.1 Basis Normalisation

The objective of Eq. 4.2 creates a new problem: The right term is a strictly increasing function of the absolute value of its argument, so it is possible that the objective can be decreased by scaling $\mathbf{W}_t$ up and $\mathbf{H}$ down ($\mathbf{W}_t \mapsto \alpha\mathbf{W}_t$ and $\mathbf{H} \mapsto (1/\alpha)\mathbf{H}$, with $\alpha > 1$). This situation does not alter the left term in the objective function, but will cause the right term to decrease, resulting in the elements of $\mathbf{W}_t$ growing without bound and $\mathbf{H}$ tending toward zero. Consequently, the solution arrived at by the optimisation algorithm is not influenced by the sparseness constraint.

To avoid the scaling misbehaviour of Eq. 4.2 another constraint is needed: The scale of the elements in $\mathbf{W}_t$ and $\mathbf{H}$ can be controlled by normalising the convolutive bases. Normalisation is performed for each object matrix, $\mathbf{W}_j$, by rescaling it to the unit L_2-norm,

$$\bar{\mathbf{W}}_j = \frac{\mathbf{W}_j}{\|\mathbf{W}_j\|}, \qquad j = 1,\ldots,R, \tag{4.3}$$

where the matrix $\mathbf{W}_j$ is constructed from the j-th column of $\mathbf{W}_t$ at each time step, $t = 0, 1, \ldots, T_o - 1$. Normalisation of $\mathbf{W}_j$ has no adverse effects on the NMF algorithm, as the objective function (Eq. 4.2) does not depend on the norm of the object matrices.

4.1.2 Additive $\mathbf{W}$ Update

An NMF algorithm that uses Eq. 4.2 as its objective and performs the necessary basis normalisation results in the following update for $\mathbf{H}$.

$$h_{jk} \leftarrow h_{jk} \frac{\sum_{i=1}^{M} w_{ijt} (v_{ik} / \overleftarrow{[\mathbf{\Lambda}]}_{ik}^{2-\beta})}{\sum_{i=1}^{M} w_{ijt} \overleftarrow{[\mathbf{\Lambda}]}_{ik}^{\beta-1} + \lambda}. \tag{4.4}$$

The additional unit norm constraint on each object, $\mathbf{W}_j$, complicates the update rule and impedes the discovery of a suitable diagonally rescaled learning rate, $\eta_{w_{ijt}}$, which would result in a multiplicative update. Consequently, the following additive update is used

$$w_{ijt} = w_{ijt} + \eta_{w_{ijt}} \left[\sum_{k=1}^{T} (v_{ik} / [\mathbf{\Lambda}]_{ik}^{2-\beta}) \overrightarrow{h}_{jk} - \sum_{k=1}^{T} [\mathbf{\Lambda}]_{ik}^{\beta-1} \overrightarrow{h}_{jk} \right]. \tag{4.5}$$

After this update, any negative values in the set of matrices $\mathbf{W}_t$ are set to zero (non-negativity constraint), and each $\mathbf{W}_j$ is normalised.

4.1.3 Multiplicative $\mathbf{W}$ Update

A multiplicative update can be obtained by including the normalisation requirement in the objective. Previously, this has been achieved for conventional NMF using the Squared Euclidean Distance reconstruction objective (Eggert and Körner, 2004). Here, we derive the multiplicative updates for a convolutive NMF algorithm utilising beta divergence. The classic NMF update rules (Lee and Seung, 2001) implement gradient descent, our new updates will also follow this approach. First, we will introduce our new reconstruction objective, which is a modification of Eq. 4.1 where each of the objects contained in $\mathbf{W}$ are normalised,

$$D_{\mathrm{BD}}(\mathbf{V} \| \mathbf{\Delta}, \beta) = \sum_{ik} \Big(v_{ik} \frac{v_{ik}^{\beta-1} - [\mathbf{\Delta}]_{ik}{}^{\beta-1}}{\beta(\beta-1)} + [\mathbf{\Delta}]_{ik}{}^{\beta-1} \frac{[\mathbf{\Delta}]_{ik} - v_{ik}}{\beta} \Big). \tag{4.6}$$

Here, $\mathbf{\Delta}$ is the current estimate of $\mathbf{V}$ following the normalisation of $\mathbf{W}_j$ (Eq. 4.3); this normalisation requires that each object is treated separately,

resulting in a column-by-column generative model,

$$\mathbf{\Delta} = \sum_{t=0}^{T_o-1} \sum_{j=1}^{R} \bar{\mathbf{w}}_{jt} (\overrightarrow{\mathbf{h}_j^t}), \tag{4.7}$$

where $\mathbf{w}_{jt}$ is a column vector and $\mathbf{h}_j$ is a row vector. By substituting Eq. 4.7 into Eq. 4.2 we obtain

$$G(\mathbf{V} \| \mathbf{\Delta}, \mathbf{H}, \beta, \lambda) = D_{\mathrm{BD}}(\mathbf{V} \| \mathbf{\Delta}, \beta) + \lambda \sum_{jk} h_{jk}. \tag{4.8}$$

We can now derive the gradient descent update for $\mathbf{H}$,

$$h_{jk} \leftarrow h_{jk} - \eta_{h_{jk}} \frac{\partial G}{\partial h_{jk}}. \tag{4.9}$$

Taking the gradient of Eq. 4.8 with respect to $\mathbf{H}$ gives

$$\frac{\partial G}{\partial h_{jk}} = \sum_{i=1}^{M} \bar{w}_{ijt} (v_{ik} / [\overleftarrow{\mathbf{\Delta}^t}]_{ik}^{2-\beta}) - \sum_{i=1}^{M} \bar{w}_{ijt} [\overleftarrow{\mathbf{\Delta}^t}]_{ik}^{\beta-1} + \lambda. \tag{4.10}$$

Diagonally rescaling the variables (Lee and Seung, 2001) and setting the learning rate to

$$\eta_{h_{jk}} = \frac{h_{jk}}{\sum_{i=1}^{M} \bar{w}_{ijt} [\overleftarrow{\mathbf{\Delta}^t}]_{ik}^{\beta-1} + \lambda} \tag{4.11}$$

gives the following multiplicative update rule for $\mathbf{H}$

$$h_{jk} \leftarrow h_{jk} \frac{\sum_{i=1}^{M} \bar{w}_{ijt} (v_{ik} / [\overleftarrow{\mathbf{\Delta}^t}]_{ik}^{2-\beta})}{\sum_{i=1}^{M} \bar{w}_{ijt} [\overleftarrow{\mathbf{\Delta}^t}]_{ik}^{\beta-1} + \lambda}, \tag{4.12}$$

which is the same as the update of Eq. 4.4.

Similarly, we derive a new update for $\mathbf{w}_{jt}$,

$$w_{ijt} \leftarrow w_{ijt} - \eta_{w_{ijt}} \frac{\partial G}{\partial w_{ijt}}. \tag{4.13}$$

To calculate the gradient of Eq. 4.8 with respect to $\mathbf{w}_{jt}$, we first need to

calculate the gradient of $\mathbf{\Delta}$ using the quotient rule,

$$\begin{aligned}\frac{\partial[\mathbf{\Delta}]_{ak}}{\partial w_{ijt}} &= \frac{\partial\left(\sum_{p=0}^{T_o-1}\sum_{q=1}^{R}\frac{w_{aqp}}{\|\mathbf{W}_q\|}\overset{p\rightarrow}{h}_{qk}\right)}{\partial w_{ijt}} \\ &= \frac{\|\mathbf{W}_j\|\overset{t\rightarrow}{h}_{jk} - (w_{ijt}\overset{t\rightarrow}{h}_{jk})\frac{\partial\|\mathbf{W}_j\|}{\partial w_{ijt}}}{\|\mathbf{W}_j\|^2},\end{aligned} \tag{4.14}$$

where $a = i; p = t; q = j$; and $\frac{\partial\|\mathbf{W}_j\|}{\partial w_{ijt}} = \bar{\mathbf{W}}_j$ for the L_2-norm. The gradient of Eq. 4.8 can now be expressed as

$$\frac{\partial G}{\partial w_{ijt}} = \sum_{k=1}^{T}\Big[\frac{v_{ik}}{[\mathbf{\Delta}]_{ik}^{2-\beta}} - [\mathbf{\Delta}]_{ik}^{\beta-1}\Big]\frac{\partial[\mathbf{\Delta}]_{ik}}{\partial w_{ijt}}. \tag{4.15}$$

Setting the learning rate to

$$\eta_{w_{ijt}} = \frac{w_{ijt}\|\mathbf{W}_j\|^2}{\sum_{k=1}^{T}\overset{t\rightarrow}{h}_{jk}\big[\|\mathbf{W}_j\|[\mathbf{\Delta}]_{ik}^{\beta-1} + \bar{w}_{ijt}(w_{ijt}(v_{ik}/[\mathbf{\Delta}]_{ik}^{2-\beta}))\big]}, \tag{4.16}$$

then rearranging Eq. 4.13 and scaling by $\|\mathbf{W}_j\|/\|\mathbf{W}_j\|$ results in the following element-wise update,

$$w_{ijt} \leftarrow w_{ijt}\frac{\sum_{k=1}^{T}\overset{t\rightarrow}{h}_{jk}\big[(v_{ik}/[\mathbf{\Delta}]_{ik}^{2-\beta}) + \bar{w}_{ijt}(\bar{w}_{ijt}[\mathbf{\Delta}]_{ik}^{\beta-1})\big]}{\sum_{k=1}^{T}\overset{t\rightarrow}{h}_{jk}\big[[\mathbf{\Delta}]_{ik}^{\beta-1} + \bar{w}_{ijt}(\bar{w}_{ijt}(v_{ik}/[\mathbf{\Delta}]_{ik}^{2-\beta}))\big]}, \tag{4.17}$$

and column-wise update,

$$\mathbf{w}_{jt} \leftarrow \mathbf{w}_{jt} \otimes \frac{\big[(\mathbf{V}/\mathbf{\Delta}^{2-\beta}) + (\bar{\mathbf{w}}_{jt}\bar{\mathbf{w}}_{jt}^{\mathsf{T}}\mathbf{\Delta}^{\beta-1})\big]\overset{t\rightarrow}{\mathbf{h}}_j}{\big[\mathbf{\Delta}^{\beta-1} + (\bar{\mathbf{w}}_{jt}\bar{\mathbf{w}}_{jt}^{\mathsf{T}}(\mathbf{V}/\mathbf{\Delta}^{2-\beta}))\big]\overset{t\rightarrow}{\mathbf{h}}_j}, \tag{4.18}$$

where $\otimes$ denotes an element-wise (also known as Hadamard or Schur product) multiplication, and division is also element-wise. The update for $\mathbf{w}_{jt}$ is now in terms of its normalised version, which is calculated (Eq. 4.3) after the update. As long as $\eta_{w_{ijt}}$ and $\eta_{h_{jk}}$ are sufficiently small, these updates should reduce Eq. 4.8. Matlab notations for sparse convolutive NMF are presented in Figure 4.1.

Sparse Convolutive NMF with Beta Divergence

```
Obj=sum(sum((V.*((((V+1E-9).^(b-1))-((delta+1E-9).^(b-1))).*(b*(b-1)+1E-9))
(((delta+1E-9).^(b-1)).*(((delta)-V).*(b+1E-9))))))+lambda*sum(sum(H));
```

```
for t=1:To
    Vt(:,:,t)=W(:,:,t)*padshift(H,t-1);
end
delta=sum(Vt,3);

for t=1:To
    Hs=padshift(H,t-1);
    for j=1:R
        NumMatW=((V./delta.^(2-b))+(W(:,j,t)*W(:,j,t)'*(delta.^(b-1))))*Hs(j,:)';
        DenMatW=(delta.^(b-1)+(W(:,j,t)*W(:,j,t)'*(V./(delta.^(2-b)))))*Hs(j,:)';
        W(:,j,t)=W(:,j,t).*(NumMatW)./(DenMatW+1e-9);
    end
end
for j=1:R
    scaling=sqrt(sum(sum(W(:,j,:).^2)));
    W(:,j,:)=squeeze(W(:,j,:))./((ones(M,To)*scaling)+1e-9);
end

for t=1:To
    Qs=padshift((V./(delta).^(2-b)),-(t-1));
    Ps=padshift(delta.^(b-1),-(t-1));
    Ht(:,:,t)=H.*(W(:,:,t)'*Qs)./((W(:,:,t)'*Ps)+lambda+1e-9);
end
H=mean(Ht,3);
```

Figure 4.1: Matlab notations for sparse convolutive NMF.

4.1.4 Sparse Convolutive NMF Applied to Audio Spectra

An interesting property of the sparseness constraint is that it enables the discovery of an over-complete basis, *i.e.*, a basis that contains more basis functions than are necessary to span the projection space.

To illustrate the performance of convolutive NMF on data generated from an over-complete basis, consider the example presented in Figure 4.2. The signal is composed of three auditory objects, each occurring at least twice: The first object is an exponentially decreasing then increasing frequency sweep centred around 4 kHz, the second object has a frequency sweep that is the reverse of the first and is also centred at 4 kHz, while the third object is a combination of the first two. Convolutive NMF is applied to the data with $R = 3$ and $T_o = 2$ seconds, and the resultant factors are presented. It is evident from the results that only the first two auditory objects are identified. The reason being that the third object can be expressed in terms

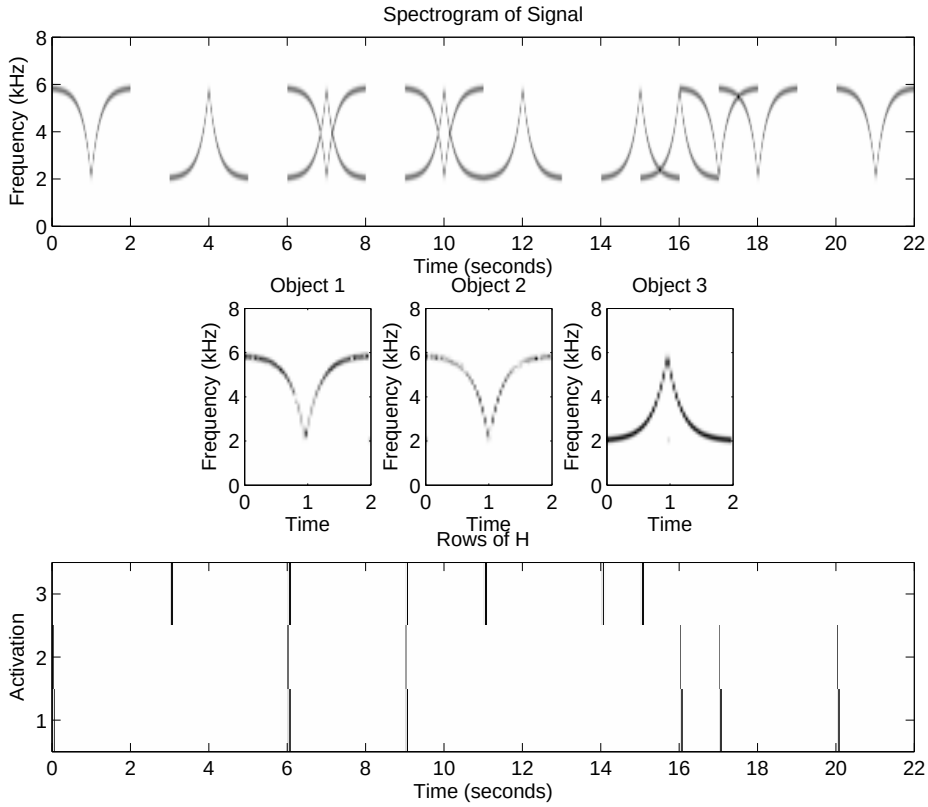

Figure 4.2: Spectrogram of a signal composed of an over-complete basis, and its factors obtained by convolutive NMF. It is evident that convolutive NMF fails to reveal the over-complete basis used to create the signal.

of the first two, and the signal can be adequetly described by using the first two objects. Therefore, convolutive NMF achieves its optimum with just the first two linearly independent objects, without the need for an over-complete representation.

When a sparseness constraint is introduced, the existence of an over-complete representation helps minimise the objective, allowing for a sparser description of the signal. Sparse convolutive NMF applied to the same signal (Figure 4.3) identifies all three objects and their associated activation patterns, successfully revealing the over-complete basis used to generate the signal.

4.1.5 Sparse Convolutive NMF Applied to Music

We now compare these algorithms on a simple music example. For illustrative clarity the music is composed of rudimentary synthesised guitar notes,

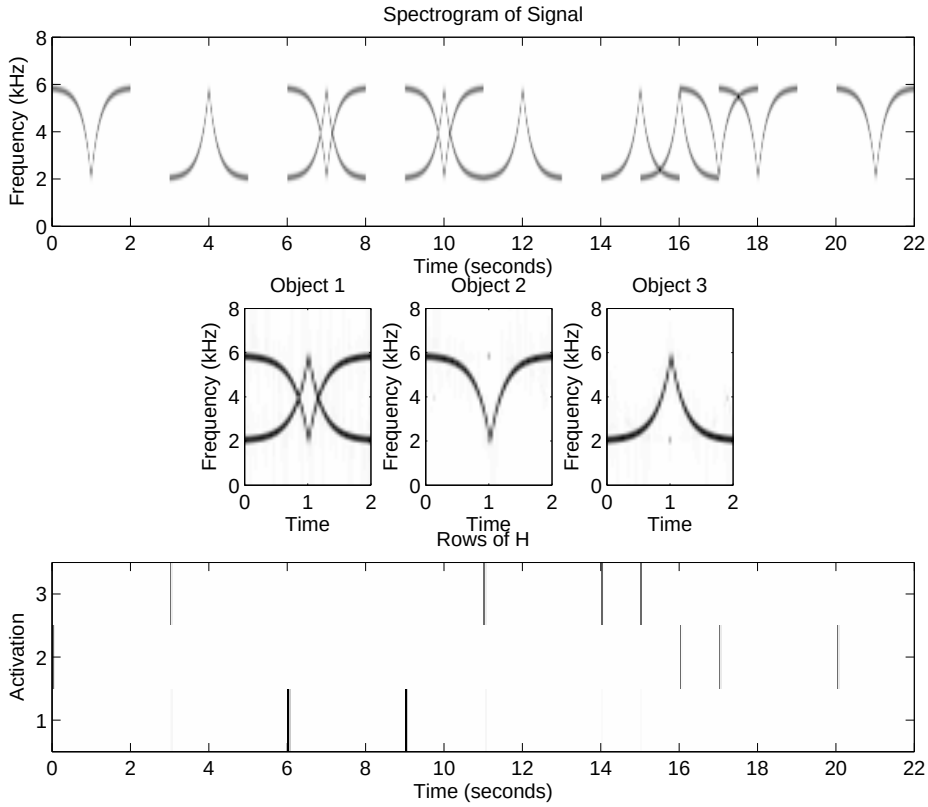

Figure 4.3: Spectrogram of a signal composed of an over-complete basis, and its factors obtained by sparse convolutive NMF. It is evident that sparse convolutive NMF successfully reveals the over-complete basis used to create the signal.

where each note produces only its fundamental frequency. The arrangement is simple, composed of three sections: The six notes of a G chord are played individually in descending order; all six notes of the chord are played simultaneously; and each note is played in reverse order. Each note is played for approximately one second, and the frequencies of the notes are 98.00 Hz (G), 123.47 Hz (B), 146.83 Hz (D), 196.00 Hz (G), 246.94 Hz (B) and 392.00 Hz (G).

Both sparse convolutive NMF and convolutive NMF are applied to the music and the resultant factors are presented in Figure 4.4. It is evident from the spectrogram that the music can be represented by an over-complete representation consisting of each individual note and the chord. Convolutive NMF is applied with $R = 7$ and $T_o = 1$ second, the resultant factors are presented in rows 5 & 6. As can be seen from the activation pattern, the algorithm has failed to represent the chord as an individual auditory object and instead represents it as a combination of notes. Sparse convolutive NMF

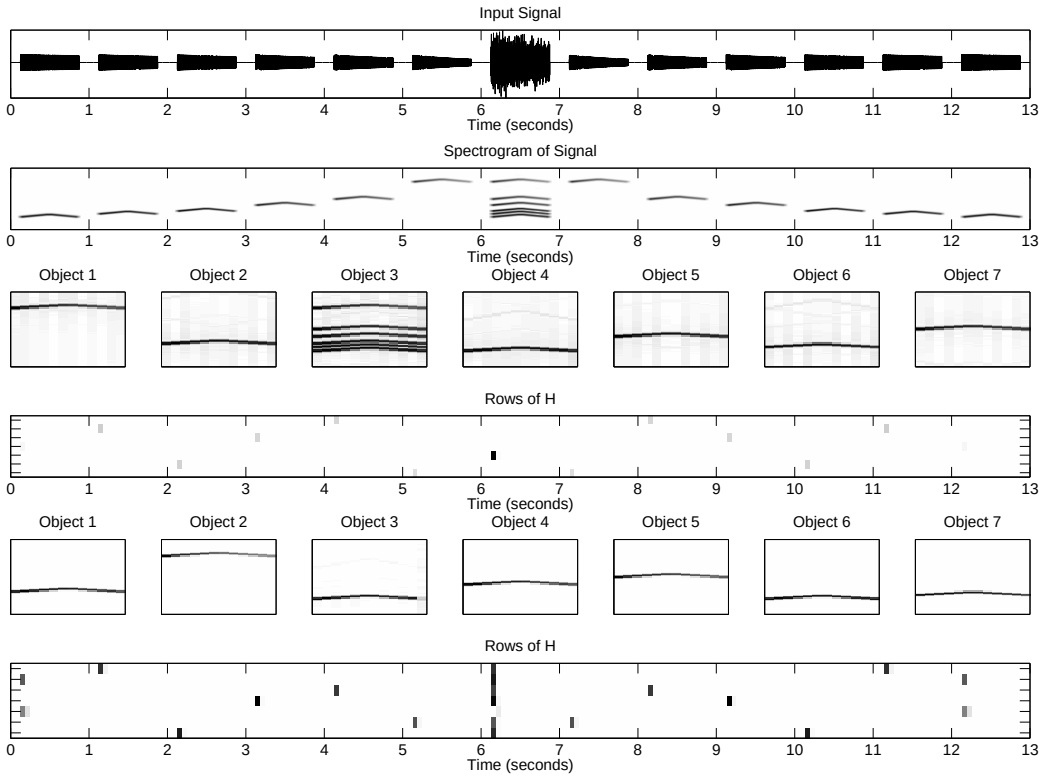

Figure 4.4: Music waveform and its associated spectrogram along with its factors obtained by sparse convolutive NMF (rows 3 & 4) and conventional convolutive NMF (rows 5 & 6).

is applied with the same parameters, where the additional parameter λ is selected on an *ad hoc* basis; the resultant factors are presented in rows 3 & 4. Here, it is evident that an over-complete representation is discovered, whereby the chord is represented as an individual auditory object.

4.2 Sparse Convolutive NMF Applied to Speech

We have demonstrated the properties of sparse convolutive NMF when applied to synthetic audio data, we will now turn our attention to real-world data. We apply sparse convolutive NMF to speech, and present a learned basis for the sparse representation of speech using the TIMIT (Garofolo et al., 1993) database. Recently, such work has been presented for convolutive NMF (Smaragdis, 2007).

First, it is necessary to appropriately define the constituent elements of speech. At a conceptual level, the theoretical representation of a sound is

called a *phoneme*, which is a sound in the most neutral form. Different phonemes distinguish different words. A segment of speech that possess's distinct physical or perceptual properties is called a *phone*. Phones occur frequently within speech and are the constituent components that create a speech spectrogram. In this context, the features that are extracted by convolutive NMF are called *phones*.

4.2.1 Discovering a Phone-like Basis

To illustrate the differences between the phones extracted by convolutive NMF and sparse convolutive NMF we perform the following three experiments for each algorithm: We take around 30 seconds of speech from a single male speaker (`DMT0`), a single female speaker (`SMA0`), and around 15 seconds from both to create a contiguous mixture. The data is normalised to unit variance, down-sampled from 16 kHz to 8 kHz, and a magnitude spectrogram of the data is constructed. We use a FFT frame size of 512, a frame overlap of 384 and a hamming window to reduce the presence of sidelobes. We extract 40 bases, $R = 40$, with a temporal extent of 0.176 seconds, $T_o = 8$, and run convolutive NMF (with $\beta = 1$) for 200 iterations. The extracted bases for male, female and mixed speech are presented in Figure 4.5, Figure 4.6 and Figure 4.7, respectively. The experiments are repeated for sparse convolutive NMF with $\lambda = 15$, and the corresponding bases are presented in Figure 4.8, Figure 4.9 and Figure 4.10.

Convolutive NMF Basis

For convolutive NMF, it is evident that the extracted bases correspond to speech phones. The verification of which, can be achieved by listening to an audible reconstruction as described in Section 3.3.1. Most of the phones represent harmonic series with differing pitch inflections, while a smaller subset of phones contain wideband components that correspond to consonant sounds. The form of the extracted basis functions are very dependent on the data, and reflect the timbral characteristics of each speaker's voice. Comparison of the male and female phone sets reveal that the most important difference between the two is the spacing between the harmonics of the phones. For the male speaker, the harmonics are spaced much closer to-

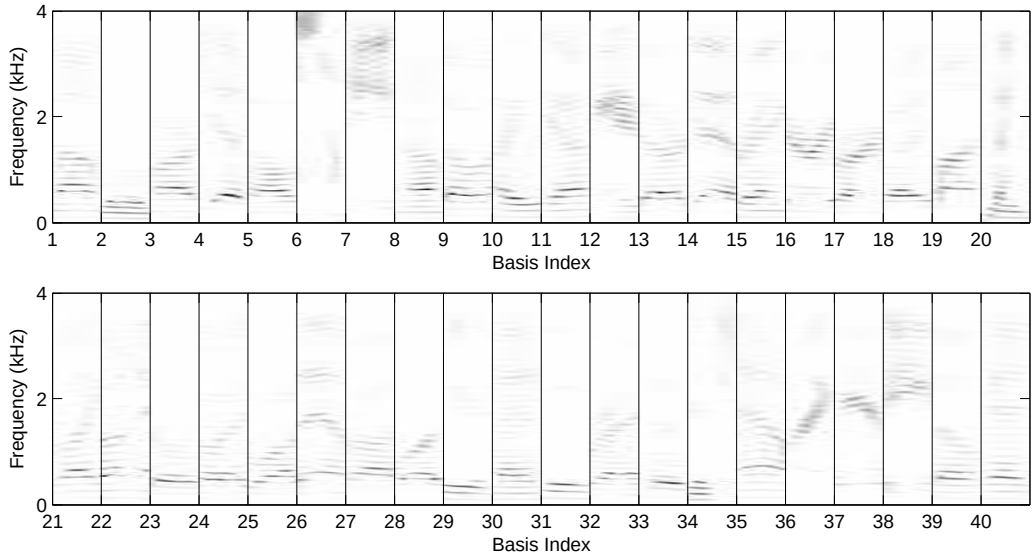

Figure 4.5: A collection of 40 phone-like basis functions extracted by convolutive NMF for a single male speaker (DMT0) taken from the TIMIT speech database, where the temporal extent of each basis is 176 ms.

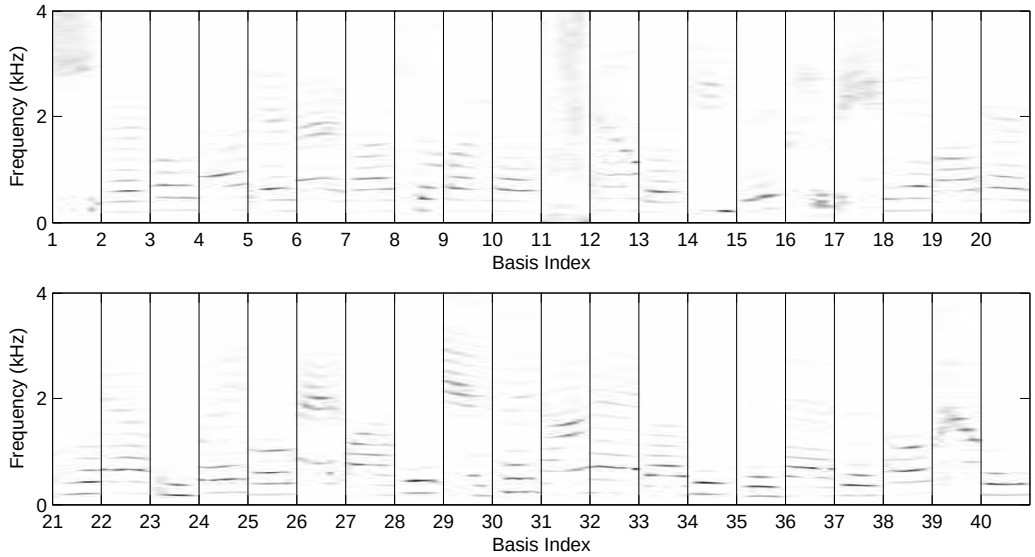

Figure 4.6: A collection of 40 phone-like basis functions extracted by convolutive NMF for a single female speaker (SMA0) taken from the TIMIT speech database, where the temporal extent of each basis is 176 ms.

gether, which is indicative of a lower pitched voice, while the female speaker phone set contains harmonics which are farther apart, indicating a higher pitched voice. Otherwise, both phone sets are quite similar. For the mixture phone set it is evident that the extracted phones correspond to either the

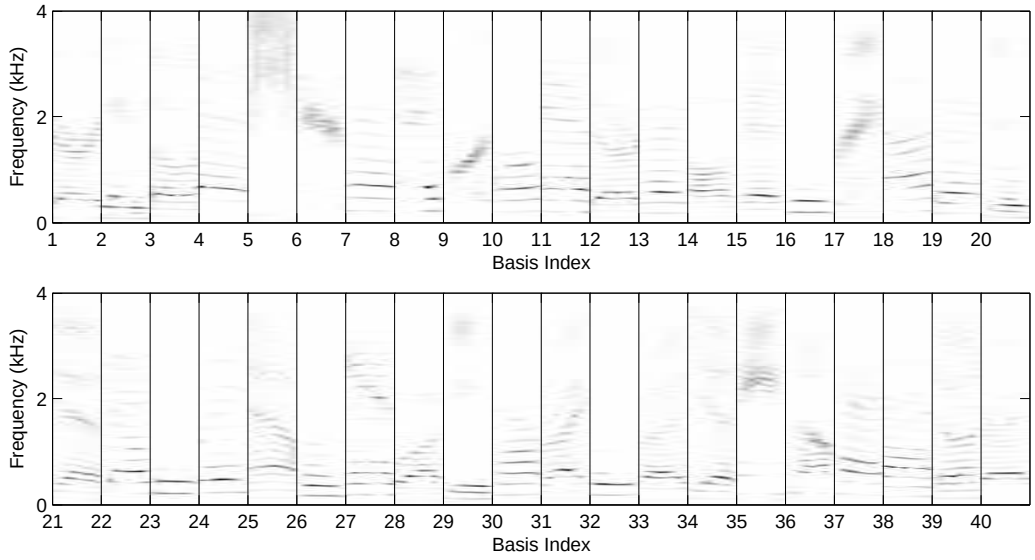

Figure 4.7: A collection of 40 phone-like basis functions extracted by convolutive NMF for a mixture of a male (DMT0) and female speaker (SMA0) taken from the TIMIT speech database, where the temporal extent of each basis is 176 ms.

male or female phone set. This indicates that the timbral characteristics of the male and female speaker are sufficiently different, such that phones that are representative of both cannot be extracted. Although, this may not be true for consonant phones.

Due to the approximative nature of NMF, the number of bases R and the temporal extent of each basis T_o affects the ability of the algorithm to represent phonetic content in a speech spectrogram. This is reflected by the SNR of the original spectrogram and its NMF reconstruction. For a large value of R, convolutive NMF can more accurately represent individual phones as individual basis functions, resulting in better reconstruction quality. For small values of R the resultant bases are forced to simultaneously represent multiple phones in each individual basis function, resulting in a blurry distinction between the bases, and poor reconstruction quality. For the purposes of our illustrative examples, the chosen algorithm parameters suffice.

Sparse Convolutive NMF Basis

By placing a sparseness constraint on the activations of the basis functions, we specify that the expressive power of each basis be extended such that it is capable of representing phones parsimoniously, much like an over-complete

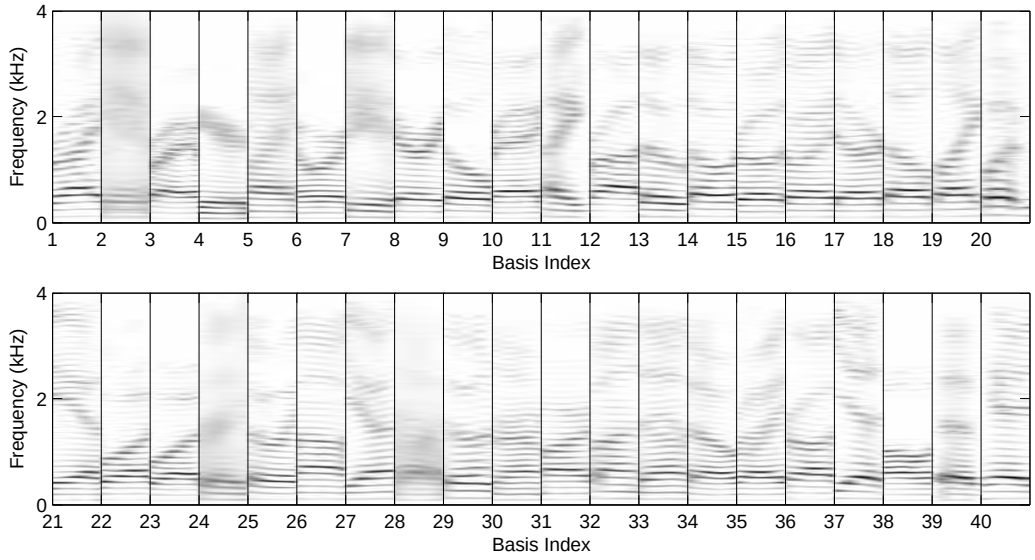

Figure 4.8: A collection of 40 phone-like basis functions for a single male speaker (DMT0) taken from the TIMIT speech database. The bases are extracted using Sparse Convolutive NMF with $\lambda = 15$, and a temporal extent of 176 ms.

dictionary. The result is that the extracted phones exhibit a structure that is rich in phonetic content, where harmonics at higher frequencies have a much greater intensity than seen in the phones extracted by convolutive NMF. This reflects the requirement that the basis functions in our new sparse phone set must contain enough features to produce a parsimonious activation pattern.

Analysis of the male and female sparse phone set reveals another important difference between the two speakers. In addition to difference in harmonic spacing, it is evident that the structure of the male phones are of a more complex nature, where changes over time are much more varied than for the female phone set. Furthermore, for the male sparse phone set, basis functions that contain both harmonic series and wideband components are extracted. For the mixture phone set, the effects are the same as those previously observed, where extracted phones correspond to either the male or female sparse phone set.

It is worth noting the effects of the selection of the weighting parameter λ. Since λ controls the tradeoff between accurate reconstruction and sparseness of the activations, larger values for λ will result in degradation of the quality of the approximation. Consequently, for the same algorithm parameters, convolutive NMF typically produces better reconstructions. This effect can be ameliorated by increasing R or reducing λ.

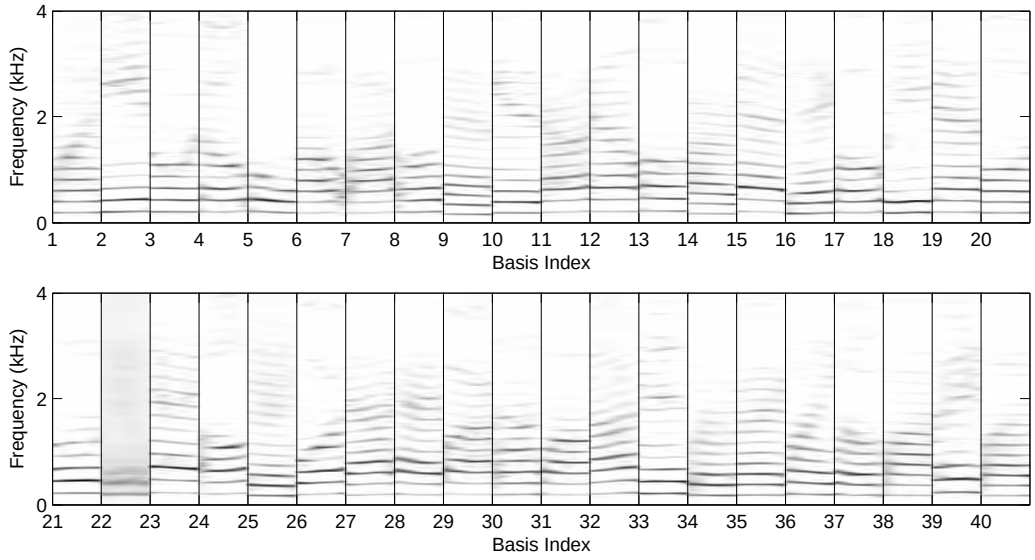

Figure 4.9: A collection of 40 phone-like basis functions for a single female speaker (SMA0) taken from the TIMIT speech database. The bases are extracted using Sparse Convolutive NMF with $\lambda = 15$, and a temporal extent of 176 ms.

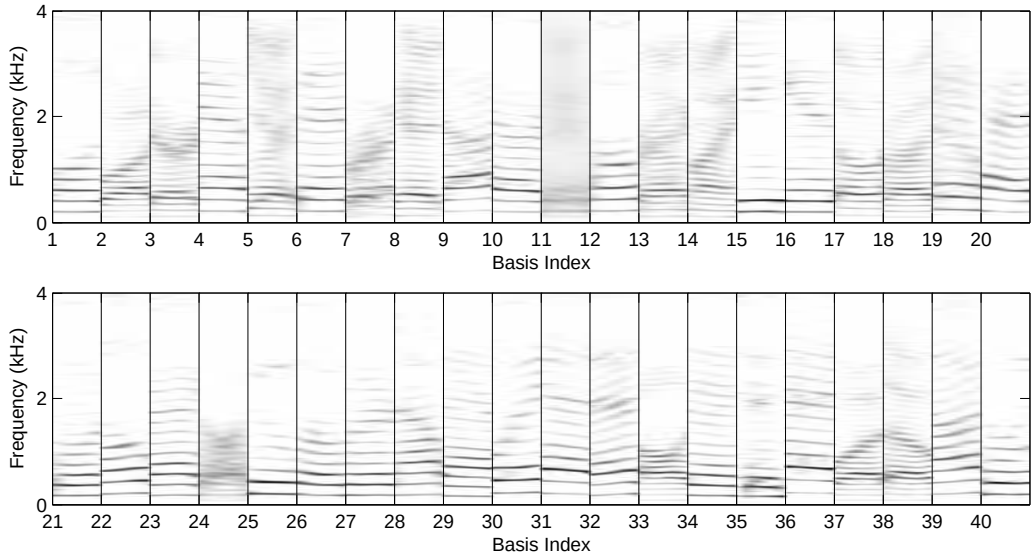

Figure 4.10: A collection of 40 phone-like basis functions for a a mixture of a male (DMT0) and female speaker (SMA0) taken from the TIMIT speech database. The bases are extracted using Sparse Convolutive NMF with $\lambda = 15$, and a temporal extent of 176 ms.

Sparsity of Activations

The sparsity of the activations produced by convolutive NMF, $\mathbf{H}^{\mathrm{c}}$, and sparse convolutive NMF, $\mathbf{H}^{\mathrm{sc}}$, can be compared using the *Kurtosis Ratio* (KR):

$$\mathrm{KR}(\mathbf{H}^{\mathrm{sc}}, \mathbf{H}^{\mathrm{c}}) = \frac{\frac{1}{R}\sum_{j=1}^{R} \mathrm{kurt}(\mathbf{h}_j^{\mathrm{sc}})}{\frac{1}{R}\sum_{j=1}^{R} \mathrm{kurt}(\mathbf{h}_j^{\mathrm{c}})}, \tag{4.19}$$

where kurt is given in Eq. 1.8. KR > 1 indicates that the $\mathbf{H}^{\mathrm{sc}}$ is sparser than $\mathbf{H}^{\mathrm{c}}$, and vice versa. The KR values for our male, female and mixed representations are 2.03, 1.74 and 1.98 respectively. Indicating that sparse convolutive NMF has indeed discovered a sparse representation for each.

4.3 Supervised Method for the Separation of Speakers

To demonstrate the utility of the extracted phone sets, we apply them to the separation of speakers from a monophonic mixture. From inspection of the NMF generative model, we can see that the estimate for $\mathbf{V}$ is constructed by taking the outer product of each column of $\mathbf{W}$ and row of $\mathbf{H}$, then summing the resultant matrices,

$$\mathbf{V} \approx \sum_{j=1}^{R} \mathbf{w}_j \mathbf{h}_j.$$

This reconstruction scheme together with a magnitude spectrogram representation, where overlapping spectra sum approximately, constitute a scheme whereby different sounds, represented by different basis functions, can be separated from the mixture. This scheme can be extended to convolutive NMF.

4.3.1 Monophonic Separation of Known Speakers

As illustrated in our previous experiments, the structure of the bases that are extracted from the speech data are uniquely dependent on the speaker (given the same algorithm parameters). In the context of speech separation, it is not unreasonable to expect that the bases extracted for a specific speaker adequately characterise the speaker, such that they can be used to discriminate them from other speakers. For a monophonic mixture where a number of speakers are added together, it is possible to separate the speakers in the mixture by constructing an individual magnitude spectrogram for each speaker, using the phones specific to that speaker.

It is evident that this scheme requires that the bases be categorised into

individual phone sets. If the speakers are known in advance, a phone set can be extracted for each speaker and used in this scheme in a supervised manner. For example, consider a mixture of a known male and female speaker. The set of male bases, $\mathbf{W}_t^m$, and female bases, $\mathbf{W}_t^f$, are learned from training data, and it is assumed that they will roughly correspond to bases extracted from any unknown sentences voiced by that speaker. By arranging the respective bases contiguously to form a combined basis, $\mathbf{W}_t^{mf} = [\mathbf{W}_t^m|\mathbf{W}_t^f]$, we can fit the mixture to the combined basis by fixing $\mathbf{W}_t = \mathbf{W}_t^{mf}$ and updating $\mathbf{H}$. Separation can be achieved by constructing an individual magnitude spectrogram using each speaker's bases and associated activations. The separation performance of such an approach is highly dependant on the *similarity* of each speaker's phone set. For a typical male and female mixture, the respective phone sets will be sufficiently different to achieve good results.

We use the following procedure for the separation of a known male and female speaker from a monophonic mixture:

1. Obtain training data for the male, $s_m(t)$, and female, $s_f(t)$, speaker, create a magnitude spectrogram for both, and extract corresponding phone sets, $\mathbf{W}_t^m$ and $\mathbf{W}_t^f$, using sparse convolutive NMF.

2. Construct a combined basis set $\mathbf{W}_t^{mf}$. This results in a basis that is twice as big as R.

3. Take a mixture that is composed of two unknown sentences voiced by our selected speakers, and create a magnitude spectrogram of the mixture. Fit the mixture to $\mathbf{W}_t^{mf}$ by performing sparse convolutive NMF with $\mathbf{W}_t$ fixed to $\mathbf{W}_t^{mf}$, and learn only the associated activations $\mathbf{H}$.

4. Partition $\mathbf{H}$ such that the activations are split into male, $\mathbf{H}^m$, and female, $\mathbf{H}^f$, parts that correspond to their associated bases, $\mathbf{H} = [\mathbf{H}^m|\mathbf{H}^f]^\mathsf{T}$.

5. Construct a magnitude spectrogram for both speakers, using their respective bases and activations: $\mathbf{S}^m = \sum_{t=0}^{T_o-1} \mathbf{W}_t^m \mathbf{H}^m$ and $\mathbf{S}^f = \sum_{t=0}^{T_o-1} \mathbf{W}_t^f \mathbf{H}^f$.

6. Use the phase information from the original mixture (as described in Section 3.3.1) to create an audible reconstruction for both speakers.

Table 4.1: Information on the Training Data for Each Speaker, Including Duration of Training Data and Phoneme Information (39 Phoneme Set, Lee and Hon (1989)).

Speakers		Training Len. (sec.)	Phoneme Information			
			Total No.	Time Len. (ms)		
				Min.	Avg.	Max.
Male	ABC0	23	322	17	70	206
	BJV0	24	331	23	73	175
	DWM0	27	328	15	82	186
Female	EXM0	32	350	17	96	213
	KLH0	26	327	24	87	179
	REH0	25	361	16	61	161

This procedure may also be used for convolutive NMF, and can be generalised for more than two speakers, and speakers of the same gender.

4.3.2 Separation Experiments

In this section, we compare the separation performance of convolutive NMF and sparse convolutive NMF. Our interest lies in how the algorithms perform for the same algorithm parameters, which may not necessarily be the optimal choice for each algorithm. For an extensive study of the relationship between parameter selection and separation performance for convolutive NMF, see Smaragdis (2007).

We randomly select three male and three female speakers from the TIMIT database, and create a training set for each that includes all but one sentence voiced by that speaker. We artificially generate a monophonic mixture by summing the remaining sentences for a selected male female pair, generating a total of nine mixtures in this way. More formally, each sentence pair is normalised to unit variance, down-sampled from 16 kHz to 8 kHz, and summed together. A magnitude spectrogram of each mixture is constructed using a FFT frame size of 512, a frame overlap of 256 and a hamming window. Information pertaining to the speakers and their training data is presented in Table 4.1, while information about the mixtures is presented in Table 4.2.

The separation performance for both algorithms is evaluated for each mixture over a selection of values for R ($R = \{40, 80, 140, 220\}$). For both al-

Table 4.2: The Speakers and Sentences Used for Each Male and Female Mixture, Including Information About Sentence Duration and Phoneme Content (39 Phoneme Set, Lee and Hon (1989)).

Mix.	Speaker		Sentence		Len. (sec.)		Phonemes	
	Male	Female	Male	Female	Male	Female	Male	Female
1	ABC0	EXM0	SX331	SX291	2.45	3.48	32	36
2	ABC0	KLH0	SX331	SX357	2.45	3.69	32	43
3	ABC0	REH0	SX331	SX325	2.45	1.93	32	25
4	BJV0	EXM0	SX347	SX291	3.62	3.48	59	36
5	BJV0	KLH0	SX347	SX357	3.62	3.69	59	43
6	BJV0	REH0	SX347	SX325	3.62	1.93	59	25
7	DWM0	EXM0	SX286	SX291	3.66	3.48	52	36
8	DWM0	KLH0	SX286	SX357	3.66	3.69	52	43
9	DWM0	REH0	SX286	SX325	3.66	1.93	52	25

gorithms the temporal extent of each phone is set to 0.224 seconds ($T_o = 6$), the number of iterations is 150, β is set to 1 and each experiment is repeated for 10 Monte Carlo runs. For convolutive NMF, a total of 24 speaker phone sets are extracted and used in 360 ($9 \times 4 \times 10$) separation experiments. For sparse convolutive NMF separation performance is tested for $\lambda = \{0.01, 0.1, 0.3, 1.0, 2.0\}$; resulting in 120 ($6 \times 4 \times 5$) speaker phone sets and 1800 ($9 \times 4 \times 5 \times 10$) separation experiments.

For the purposes of ease of comparison with existing separation methods, we evaluate the separation performance of both algorithms using the measures provided by the `BSS_EVAL` toolbox (Févotte et al., 2005), which are described in Section 2.2.1 and are briefly restated below:

- *Source-to-Artifact Ratio* (SAR): Measures the level of artifacts in the source estimates.
- *Source-to-Interferences Ratio* (SIR): Measures the level of interference from the other sources in each source estimate.
- *Source-to-Distortion Ratio* (SDR): Provides an overall separation performance criterion.

All performance measures are expressed in dB, with higher performance values indicating better quality estimates.

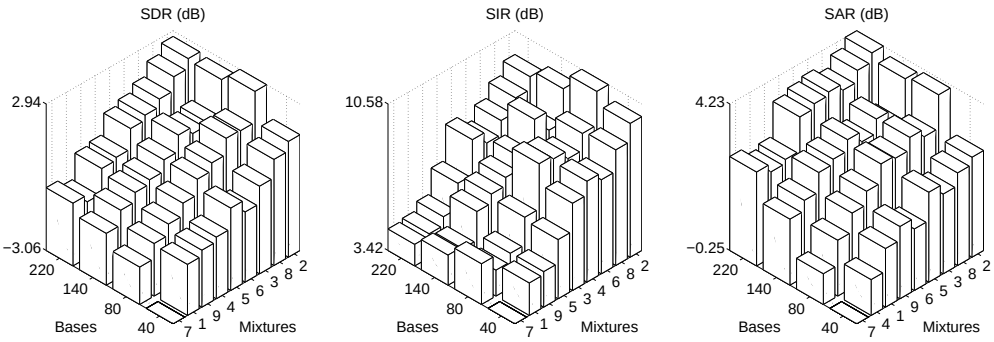

Figure 4.11: Separation performance for convolutive NMF: A bar chart for each performance measure (SDR, SIR and SAR) is presented, where the performance for each mixture, in ascending order, is plotted against the number of bases. Note that the scales for the z-axis are expressed in dB and change for each plot.

Convolutive NMF Separation Performance

Here, we examine the separation performance of convolutive NMF applied to our generated mixtures. The results for each experiment are averaged over all runs and are presented in Figure 4.11. Each separation measure is illustrated as a bar chart, where mixtures are plotted against the number of bases used, and bar height indicates performance. For visual clarity, the 9 mixtures are arranged in ascending order.

The resultant performance values are very dependant on the mixture under consideration, this may reflect similarity in the timbral characteristics of the speakers in each mixture. On average, mixture 7 performed worst for all performance measures, while mixture 2 performed best. The SDR results, which indicate overall performance, improve for most mixtures as the number of bases used increases. The average SDR over all mixtures range from -0.18 dB for 40 bases to 0.96 dB for 220 bases. The same is also true for SAR, where performance rises from 1.75 dB at 40 bases to 3.37 at 220 bases. For the SIR results, best performance is achieved when 80 bases are used.

Sparse Convolutive NMF Separation Performance

The results in Figure 4.11 can be compared with the corresponding results for sparse convolutive NMF in Figure 4.12, in which 4 sets of results pertaining to different values of λ are presented.

For added clarity, we statistically analyse the performance of convolutive

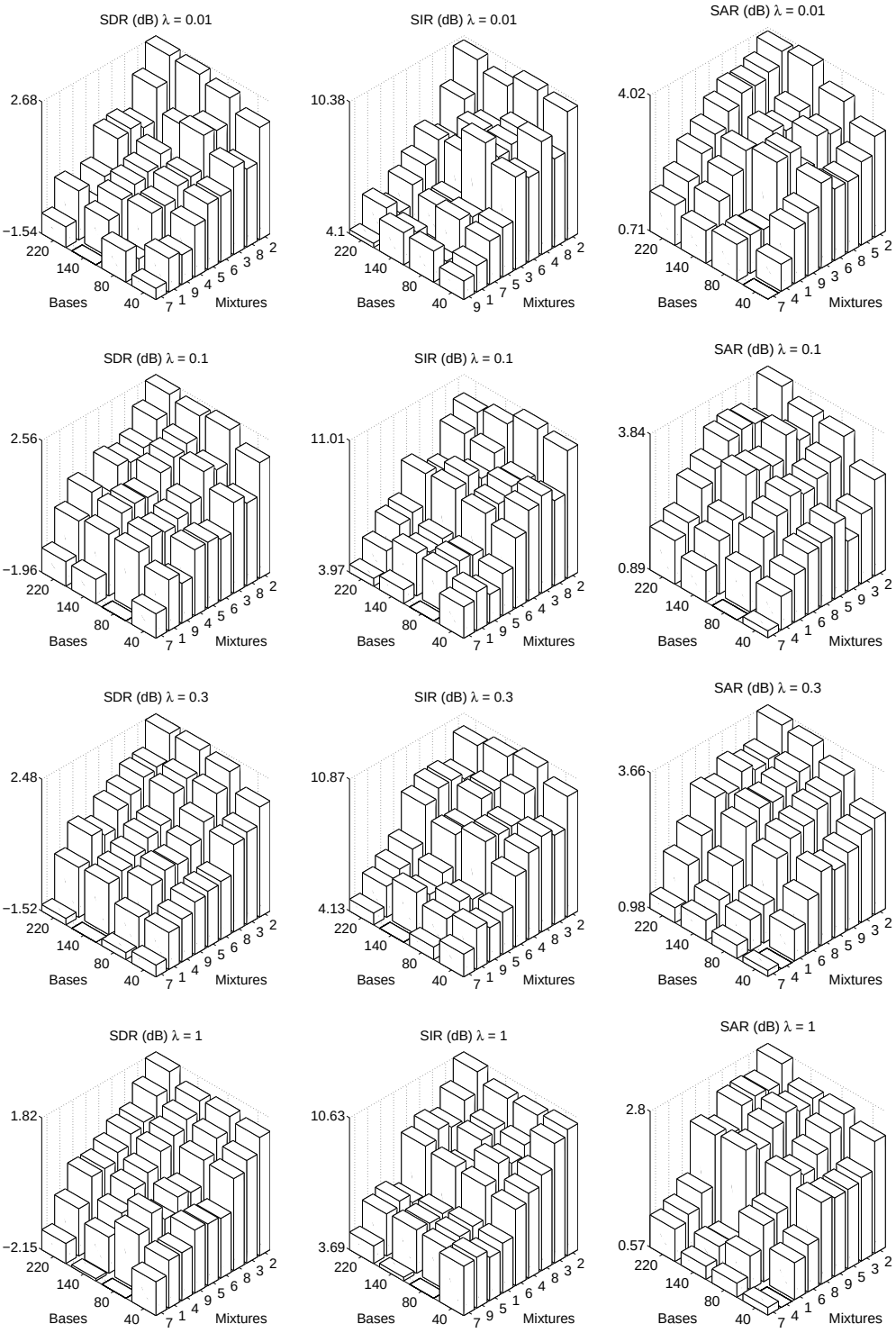

Figure 4.12: Separation performance for sparse convolutive NMF: A bar chart for each performance measure (SDR, SIR and SAR) is presented for a selection of λ values, where the performance for each mixture, in ascending order, is plotted against the number of bases. Note that the scales for the z-axis change for each plot and are expressed in dB.

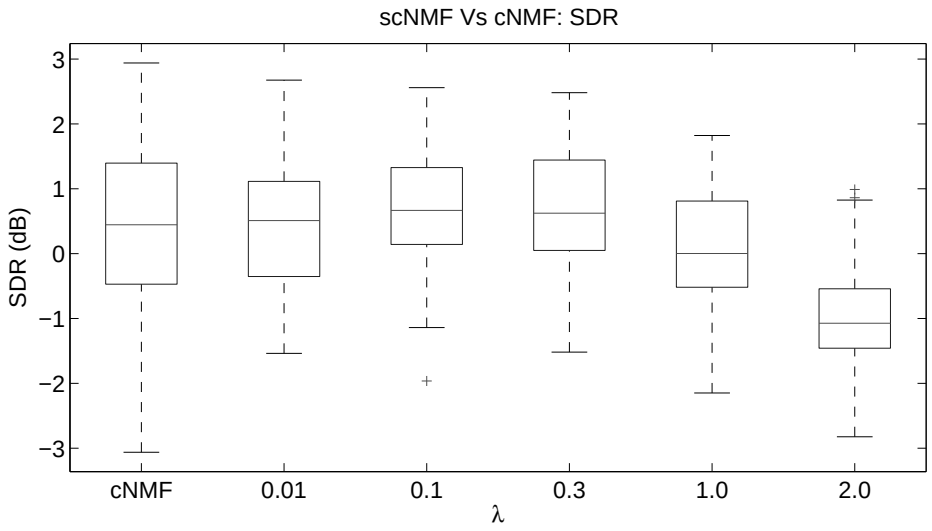

Figure 4.13: A comparison of the SDR results obtained by convolutive and sparse convolutive NMF: Box plots are used to illustrate the performance results, with each box representing the median and the interquartile range of the results. It is evident that for $\lambda = 0.1$, a better spread of results is obtained, indicating that sparse convolutive NMF achieves superior overall performance.

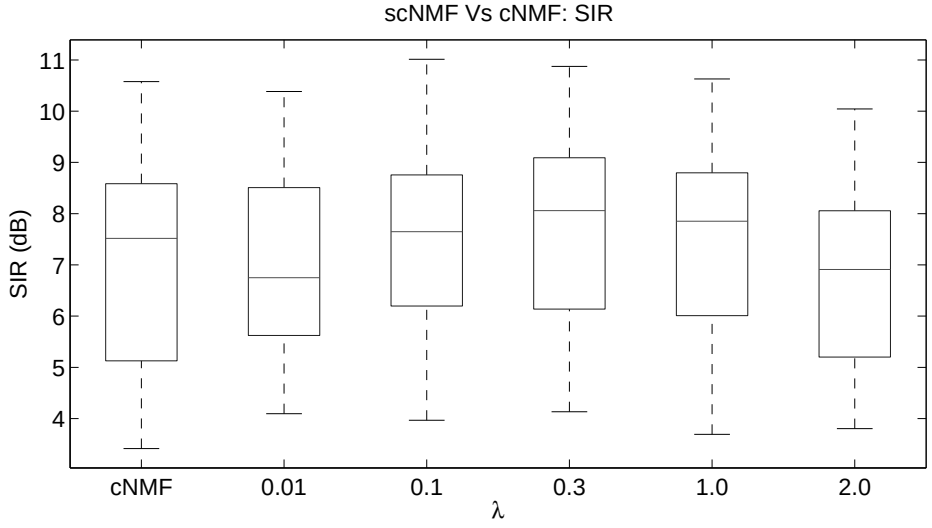

Figure 4.14: A comparison of the SIR results obtained by convolutive and sparse convolutive NMF: Box plots are used to illustrate the performance results. For $\lambda = 0.1$, a better spread of results is obtained, indicating that sparse convolutive NMF produces estimates that are more resilient to interference from other sources.

NMF and sparse convolutive NMF by collating the results from all experiments (Figure 4.11 & Figure 4.12), and represent the results using box plots: Each box presents information about the median and the statistical disper-

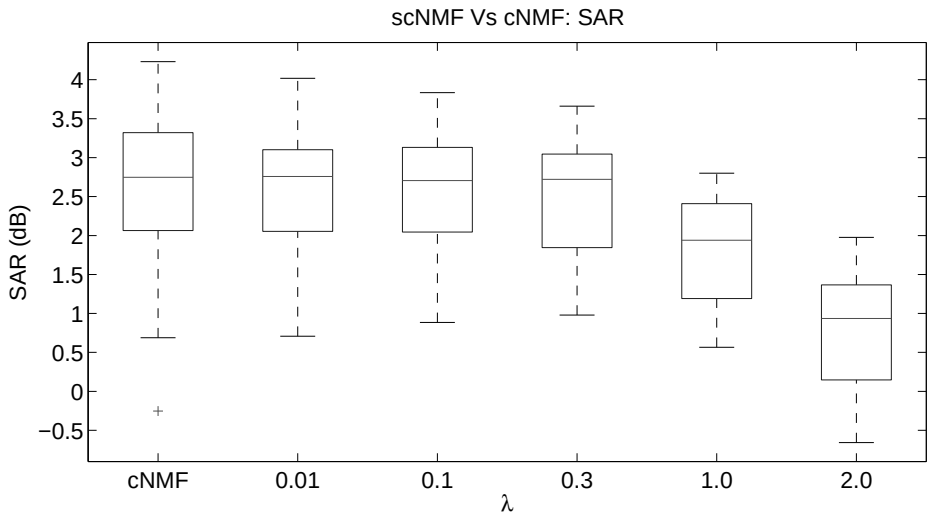

Figure 4.15: A comparison of the SDR results obtained by convolutive and sparse convolutive NMF: Box plots are used to illustrate the performance results, with each box representing the median and the interquartile range of the results, the whiskers represent the spread of the results. Here, convolutive NMF achieves best results. This may reflect the fact that sparse phone sets exhibit phones that are rich in features, which may produce artifacts in the resultant source estimates.

sion of the results. The top and bottom of each box represents the upper and lower quartiles, while the length between them is the interquartile range; the whiskers represent the extent of the rest of the data, and outliers are represented by +. Box plots for SDR, SIR and SAR are presented in Figure 4.13, Figure 4.14 and Figure 4.15 respectively.

The SDR results indicate that for $\lambda = \{0.1, 0.3\}$, the median performance obtained (0.66 dB, 0.62 dB) exceeds convolutive NMF (0.44 dB) for our given algorithm parameters. It is also evident that a better spread of results with reduced variability is produced for sparse convolutive NMF; demonstrating that when λ is chosen appropriately, sparse convolutive NMF achieves superior overall performance. For SIR, $\lambda = 0.3$ produces the best spread of results, which indicates that sparse convolutive NMF is more resilient to interference from other sources. However, for SAR, convolutive NMF produces the best results, this may reflect the fact that each sparse phone set exhibits phones that are rich in features, which may manifest as artifacts in the resultant source estimates. It is also evident that the performance of the sparse convolutive algorithm degrades significantly for large λ values, so much so, that it renders the results useless, this is especially evident for $\lambda > 1$.

4.4 Coding Efficiency of Learned Bases

Here, we demonstrate the utility of sparse convolutive NMF in the information coding of speech data, we use a simple scheme whereby the K largest coefficients in each column of $\mathbf{H}$, along with their positions, are used to reconstruct the data:

$$\mathbf{\Lambda}_K = \sum_{t=0}^{T_o-1} \mathbf{W}_t \ \text{maxcol}(K, \overset{t\rightarrow}{\mathbf{H}}) \qquad 0 < K \leq R, \tag{4.20}$$

where maxcol creates a matrix the same size as $\mathbf{H}$, with all but the K largest coefficients in each column being zeroed. Here, we consider only the reconstruction of the magnitude spectrogram and do not address how to encode phase information.

We use an experimental procedure similar to that used in our separation experiments, whereby we fix $\mathbf{W}$ to the basis for our speaker and fit an unknown sentence to it by updating $\mathbf{H}$. We then reconstruct $\mathbf{\Lambda}_K$ by using its K largest coefficients, over $K = 1, \ldots, R$, and measure reconstruction quality using the *Signal-to-Noise Ratio* (SNR); each experiment is repeated for 10 Monte Carlo runs. We select a male (`ABC0`) and female speaker (`EXM0`), and use the 220 basis set learned for the experiments in the previous section. The reconstruction quality for a range of λ values is investigated, and the results are presented in Figure 4.16.

The curves in Figure 4.16 illustrate the trade-off between the fidelity of the reconstruction and the coding cost, expressed in coefficients. We are interested in the transitional phase leading to quiescent value for SNR; the quicker the convergence the fewer coefficients needed to reconstruct $\mathbf{\Lambda}$. The coding efficiency for convolutive NMF ($\circ$) can be easily compared with the other curves, which represent the results for sparse convolutive NMF.

For both speakers, it is evident that sparse convolutive NMF needs fewer coefficients to reach a quiescent SNR value, the SNR achieved is very dependent on λ, which is indicative of the trade-off between the sparseness of $\mathbf{H}$ and accuracy of reconstruction, this effect is particularly evident for large λ values. For the male speaker, $\lambda = 0.001$ provides superior SNR over convolutive NMF when $K < 15$. Furthermore, the SNR achieved produces a level of reconstruction quality such that the encoded sentence is intelligible. For the female speaker $\lambda = 0.01$ produces superior quality when $K < 5$,

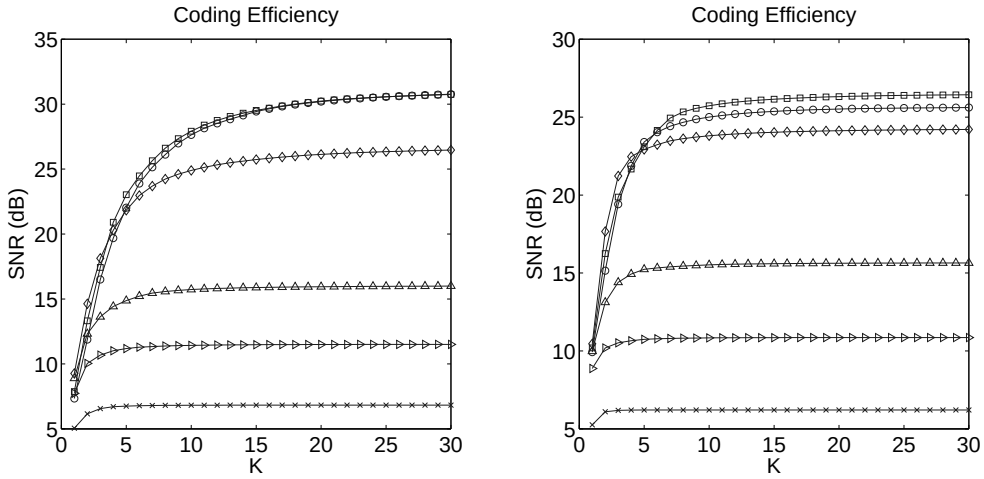

Figure 4.16: Coding efficiency curves for sparse convolutive NMF, for both a male (left: ABC0) and female (right: EXM0) speaker. The curve for convolutive NMF ($\circ$) can be contrasted with sparse convolutive NMF ($\square$: $\lambda = 0.001$, $\diamond$: $\lambda = 0.01$, $\triangle$: $\lambda = 0.1$, $\triangleright$: $\lambda = 0.3$, $\times$: $\lambda = 1$). It is evident that sparse convolutive NMF provides a faster rate of convergence to a quiescent SNR. Furthermore, for large λ values, a degenerative effect on reconstruction quality is evident, which is indicative of the tradeoff between sparseness and reconstruction quality.

while $\lambda = 0.001$ produces superior quality for all K. For both speakers $\lambda = \{0.1, 0.3, 1\}$ never exceeds the performance of convolutive NMF at any point along the curve. Therefore, these values are an inappropriate choice for our data and produce results that are of no use. The faster convergence rate illustrated for the female speaker's curves, may be due to the larger training set (32 sec., 250 phonemes) used in the phone extraction phase (Table 4.1).

4.5 Discussion

The benefit obtained by combining convolutive NMF with a sparseness constraint on the activations, is due to the requirement that a parsimonious representation must be found in order to satisfy sparseness. Such representations extract bases that are rich in phonetic structure, and exhibit superior separation properties. Although, improved overall separation performance is at the expense of additional artifacts in the estimates.

In contrast to previously proposed algorithms, which have additive updates (Virtanen, 2003; O'Grady and Pearlmutter, 2006), our algorithm re-

tains its advantages of parameter-independent gradient descent and fast convergence. Moreover, multiplicative updates ensure that the algorithm arrives at some solution, which from our experience, has not always been the case for additive update algorithms. An additional benefit to our algorithm is that it utilises the beta divergence, which enables different reconstruction penalties to be selected depending on some additional requirement, *e.g.*, perceptual quality, as discussed in Chapter 3.

Normalisation of the objects in $\mathbf{W}$ introduces an asymmetry between $\mathbf{W}$ and $\mathbf{H}$, which makes it difficult to prove convergence properties of Eq. 4.18 as indicated in Lee and Seung (2001) (Eggert and Körner, 2004). Nonetheless, we have performed many experiments with our algorithm and it converges to sensible solutions every time. Eggert and Körner (2004) propose that convergence can be explained by the fact that the rescaling of the gradient introduced by the multiplicative update rule, results in a gradient step that has a positive projection on the true gradient, due to the non-negativity constraint. Furthermore, as long as the gradient step size is sufficiently small (this is true when $\mathbf{\Lambda}$ approaches $\mathbf{V}$), convergence is achieved; we believe this to be true in our case also.

Finally, due to the fact that our algorithm is implemented using column-wise updates for $\mathbf{W}_t$ (because of the normalisation of the objects, $\mathbf{W}_j$, contained in $\mathbf{W}$), the run time of the algorithm increases greatly: Consider speaker `ABC0` from Table 4.1, to extract 40 bases (as per our experiments) on a 2.53 GHz Intel Pentium 4 computer with 256Mb of RAM, takes 4 minutes for convolutive NMF, while the same experiment takes 50 minutes for sparse convolutive NMF. Furthermore, sparse convolutive NMF algorithms with additive updates may run faster. However, our multiplicative algorithm will always arrive at a solution with better quality results, and removes the requirement to select both an appropriate learning rate and λ, which can sometimes be painfully difficult to achieve.

4.6 Conclusion

In this chapter, we presented a sparse convolutive NMF algorithm, which effectively discovers a sparse parts-based representation for non-negative data. This method extends the convolutive NMF objective by including a sparseness constraint on the activation patterns, enabling the discovery of over-

complete representations. Moreover, in contrast to previously proposed algorithms, normalisation of the basis vectors is explicitly included in the reconstruction objective, resulting in multiplicative updates and more stable convergence properties. We have applied the algorithm to speech data, and have demonstrated its superiority to convolutive NMF, when applied to the separation of speakers from a monophonic mixture and speech coding.

CHAPTER 5

Conclusions and Future Work

In this thesis, we presented the principles that make blind source separation of speech possible. For the even-determined case, we overviewed a number of independent component analysis algorithms, each utilising a different notion of statistical independence. For the under-determined case, where identification of the mixing matrix is not possible using ICA as it has no inverse, we discussed how the mixing matrix, $\mathbf{A}$, is discovered using some clustering procedure that identifies corresponding clusters in some, typically sparse, transform domain. Furthermore, we discussed how to estimate the sources by using a minimum L_1-norm constraint, which regularises the inverse and provides good results as long as the mixtures are in a sparse domain. We also discussed Non-negative Matrix Factorisation, which is a contrasting parts-based approach to separation, where monophonic mixtures of components are separated using a non-negativity constraint. Furthermore, we demonstrated the utility of a convolutive generative model when the components are time-varying.

We built on the previous, and introduced two algorithms that utilise sparseness in the separation of instantaneously mixed speech sources. First, we explored the case where there are two or more observations and introduced the LOST (*Line Orientation Separation Technique*) algorithm, which identifies linear subspaces in a sparse domain that correspond to the columns of $\mathbf{A}$. Furthermore, once $\mathbf{A}$ is found, source estimates are calculated using

L_1-norm minimisation. Second, we introduced a convolutive NMF algorithm that places a sparseness constraint on the activations. The algorithm has multiplicative updates and extracts phones from a magnitude spectrogram representation of speech, which can be subsequently used in a supervised separation scheme.

Prior to the introduction of sparse convolutive NMF, we considered the recently proposed beta divergence reconstruction objective, and demonstrated the use of this objective for an NMF algorithm that is applied to speech. The criterion we apply to qualify the objectives utility is perceptual performance, which for the most part is dependent on signal detection. In this respect, we demonstrated that the parameterisable beta divergence provides greater flexibility than the originally proposed Squared Euclidean Distance or Kullback-Leibler Divergence.

5.1 Summary

To complete this thesis we summarise the work presented and discuss future directions for the presented work.

The LOST Algorithm: We introduced a blind source separation algorithm for instantaneous mixtures that estimates $\mathbf{A}$ by identifying corresponding linear subspaces in a scatter plot. The mixing process is expressed as a Laplacian mixture model where an EM procedure is used to estimate the model parameters, which identify the lines. Such an approach enables the identification of an arbitrary number of sources from an arbitrary number of mixtures, and together with an L_1-norm minimisation provides a solution for the under-determined case. We named this algorithm the LOST algorithm, and applied it to the separation of speech mixtures. The results demonstrate that the LOST algorithm is an effective algorithm for the separation of speech. Furthermore, we provided an empirical assessment of the robustness of the algorithm in the presence of Gaussian noise, which also yielded useful results.

NMF Reconstruction Objective: We investigated the utility of the beta divergence as a reconstruction objective for NMF, where the perceptual quality of the NMF reconstruction is evaluated using the noise-to-mask ratio over a wide selection of algorithm parameters. The results indi-

cate that optimal β selection is very dependent on algorithm parameters. Although, the general trend is that as the number of objects R increases, β tends toward 0. We also provided a comparison between NMF utilising the beta divergence, and a perceptually weighted NMF algorithm that utilises noise-to-mask ratio; the results indicate that NMF utilising beta divergence, and an appropriately selected β, results in faster convergence and comparable NMR_{tot} performance to the perceptually weighted algorithm for the same number of iterations. Although, the perceptually weighted algorithm does not require the discovery of an optimal divergence parameter such as β, and will provide better estimates as long as the algorithm is run for enough iterations. In this way, we demonstrated the usefulness of the beta divergence when used in NMF algorithms that are applied to speech spectrograms.

Convolutive NMF with a Sparseness Constraint: We introduced a convolutive NMF algorithm that enforces a sparseness constraint on the activations. The proposed method overcomes the previously discussed restriction leading to an additive update for the basis by including the required basis normalisation step directly in the reconstruction objective, resulting in a multiplicative update. We applied the algorithm to synthetic data and demonstrated that it achieves sparse activation patterns by identifying over-complete dictionary elements. Moreover, we extracted sparse phone sets from speakers in the TIMIT database, and used the extracted phones in a supervised separation scheme for monophonic mixtures, which produces better separation performance than convolutive NMF.

5.2 Future Work

Although the presented algorithms achieve the separation of speech from under-determined instantaneous mixtures with some success, as with all scientific endeavours improvements and extensions can be made.

The LOST Algorithm

The separation performance of the LOST algorithm is heavily influenced by how well defined the linear subspaces are in the scatter plot, which in turn is dependent on sparsity of mixture coefficients in the transform domain. Currently, the LOST algorithm exploits the sparseness of speech in the STFT transform domain. However, further performance improvements may be achieved by using alternative transformations such as the Gabor or Wavelet transforms. Furthermore, the LOST algorithm need not be restricted to speech; by selecting an appropriate sparse transformation for the data at hand, the LOST algorithm may be applied to other mixture types, such as image or music data.

For the under-determined case, we estimate $\mathbf{S}$ using L_1-norm minimisation, where minimisation is achieved using a linear program, and the real and imaginary components of the STFT coefficients are treated separately. Alternatively, L_1-norm minimisation for complex data can be implemented using a second order conic program optimisation, which eliminates the approximation made by treating real and imaginary components separately.

For most BSS algorithms the number of mixtures and sources are provided beforehand. However, for the LOST algorithm it may be possible to estimate the number of sources directly from the mixtures. Such a scheme would involve initialising the algorithm with a sufficiently large number of orientation vectors, where line orientations that have principal eigenvalues smaller than some predefined threshold are removed as the algorithm runs. In this way, line orientation vectors that are additional to requirements are removed, and the number of sources is estimated.

The LOST algorithm currently separates sources where $\mathbf{A}$ is fixed for the duration of the mixing process. The algorithm may be extended to instantaneous mixtures that have a time-varying $\mathbf{A}$, *i.e.*, mixtures of moving sources. Such an extension may be achieved by estimating $\mathbf{A}$ on an observation-by-observation basis using a stochastic gradient algorithm to estimate the principal eigenvector of each linear subspace, which adapts to the line orientations as they move.

Convolutive NMF with a Sparseness Constraint

The experiments we presented for sparse convolutive NMF have been restricted to speech spectrograms. However, the algorithm will extract bases that vary along the horizontal for any non-negative data, *e.g.*, music, image data, *etc.* Furthermore, in the case of image data, a double convolutive algorithm may be derived, such that the image is composed of two-dimensional image patches that represent features that vary in both the vertical and horizontal direction, *i.e.*, a convolutive NMF algorithm that performs horizontal *and* vertical shifts.

It would be interesting to investigate the utility of the extracted sparse phone sets in a *compressive sampling* (Candes et al., 2006) framework: Nyquist sampling theory states that a signal must be sampled at a rate at least twice its highest frequency in order to be represented without error. However, in practice, signals are often compressed soon after sensing, trading off accurate reconstruction for some acceptable level of error. Clearly, this is a waste of valuable sensing resources. In recent years, a new and exciting theory of compressive sampling has emerged, in which the signal is sampled and compressed simultaneously at a greatly reduced rate using sparse representations and an L_1-norm minimisation.

The reconstruction objective used for our algorithm is the beta divergence; for our experiments we kept $\beta = 1$, which specifies the Kullback-Leibler Divergence. In Chapter 3 we discussed the perceptual properties of the beta divergence. Additionally, it would be interesting to see how the sparseness parameter λ and divergence parameter β interplay. Specifically, it would be interesting to investigate the effect of β selection on separation performance.

The presented experiments separated male and female speakers; it would also be interesting to evaluate the performance of the extracted sparse phone sets when applied to a speaker identification task. Furthermore, the extracted phones may be applied to a speech denoising task, where a monophonic recording of a known speaker is made in the presence of an unwanted sound, such as street noise or a musical instrument.

Currently, the sparseness function used in the presented algorithm quantifies the sparsity of the activation coefficients using the L_1-norm. Alternatively, other differentiable sparseness indicators may also be used, which may

bring added benefits such as improved rates of convergence.

5.3 Closing Comment

Finally, we hope that the algorithms presented and the results discussed will provide a worthwhile contribution to the area, and go some way towards the ultimate goal of a practical and accurate speech separation machine.

APPENDIX A

LOST Algorithm Source Signals

The source signals are taken from a commercial audio CD of poems read by their authors (Paschen et al., 2001). The data is recorded as raw 44.1 kHz 16-bit stereo waveforms. Prior to further processing, ten-second clips are extracted, the two signal channels are averaged, and the data is down-sampled to 8 kHz. The scale of the audio data is arbitrary, leading to the arbitrary units on auditory waveform samples throughout Chapter 2. The sources are extracted from the following poems:

- *Coole Park and Ballylee*, by William Butler Yeats.
- *The Lake Isle of Innisfree*, by William Butler Yeats.
- *Among Those Killed in the Dawn Raid Was a Man Aged a Hundred*, by Dylan Thomas.
- *Fern Hill*, by Dylan Thomas.
- *Ave Maria*, by Frank O'Hara.
- *Lana Turner Has Collapsed*, by Frank O'Hara.

APPENDIX B

Psychoacoustic Model

In this appendix, we detail the psychoacoustic model that is used to create the masking thresholds used in Chapter 3. The model is based on the PEAQ algorithm (Perceptual Evaluation of Audio Quality), which is an internationally recognised standard for the measurement of perceived audio quality. The notation we use here corresponds to that in the PEAQ documentation (ITU, 1998).

B.1 Input Signal

The PEAQ algorithm assumes that the sample frequency, F_s, of the test signal, x, is 48 kHz, our model makes no such assumption. The test and reference signals are assumed to be aligned in time. Processing occurs on a frame-by-frame basis, where the length of each frame is N_f and contains samples $x[n]$ for $n = (0, \ldots, N_f - 1)$.

A rudimentary but important step in the process is test signal calibration. For most purposes the test data will originate from a standard computer sound card, and will be stored in a raw audio format such as 16-bit PCM. Therefore, because the model is level dependent, it is necessary to fix the amplitude range of the test signal in relation to a real acoustical signal. The following expression can be used to scale the test signal such that it corresponds to L_p $\mathrm{dB_{SPL}}$, where a root-mean-squared test signal amplitude

of 1 corresponds to 0-dB_{SPL},

$$x[n] = \frac{x[n]}{\sqrt{\sum_i^{N_f} x[i]^2/N_f}} 10^{(L_p/20)} \ (\text{dB}_{\text{SPL}}). \tag{B.1}$$

It is important to select an appropriate L_p for the signal under consideration, *e.g.*, it would make no sense to set L_p to 30-dB_{SPL}(sound level of a whisper) for a recording of a jet engine.

B.2 Frequency Transformation

As discussed in Section 3.1, the psychoacoustic phenomena of interest are observed in the frequency domain. Therefore, the constituent operations of the perceptual model operate in this domain. Each frame of data is first windowed by a Hamming window, and subsequently transformed to the frequency domain by a N_f-point Discrete Fourier Transform,

$$X[k] = \frac{1}{N_f} \sum_{n=0}^{N_f-1} h_w[n] \ x[n] \ e^{-j2\pi nk/N_f}.$$

Only coefficients for $0 \leq k \leq N_f/2$, corresponding to frequencies from 0 to $F_s/2$, are retained, and their magnitude $|X[k]|$ will be used in subsequent stages.

B.3 Outer and Middle Ear Weighting

The absolute threshold of hearing in quiet (Eq. 3.1), is modelled as a combination of internal noise and the outer and middle ear transfer function,

$$A_{dB}(f) = w \cdot 3.64 \left(\frac{f}{1000}\right)^{-0.8} - 6.5e^{-0.6\left(\frac{f}{1000-3.3}\right)^2} + k \cdot 10^{-3} \left(\frac{f}{1000}\right)^4. \tag{B.2}$$

The internal noise is considered to be low frequency, and is modelled by a portion of the energy in the first term as specified by w, while the weighting parameter k determines the upper frequency cutoff, which is age dependent. It can be seen from inspection of Figure 3.1 that the upper frequency cut-off starts early, indicating that the subjects were rather old and may not be

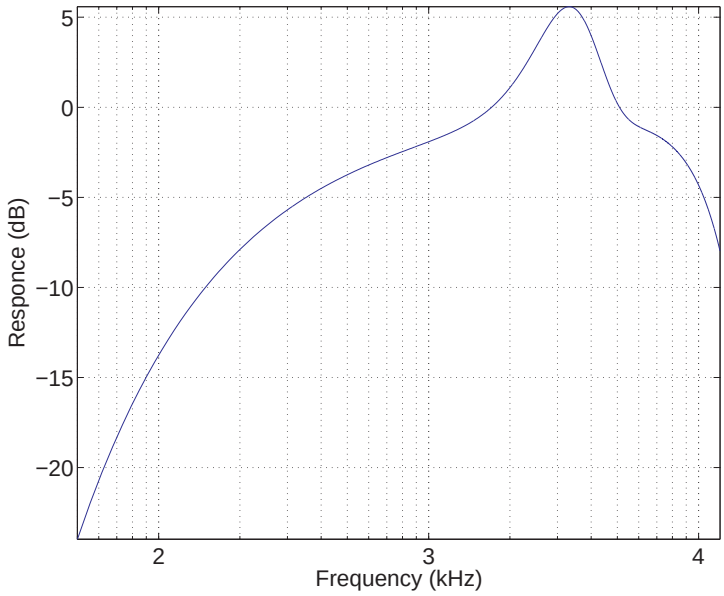

Figure B.1: Outer and middle ear frequency response.

considered as *normal hearing listeners*. The weighting parameter $w = 0.6$ and $k = 2\ 2$ (shifts upper frequency cut-off to higher frequencies) resulting in following outer and middle ear weighting,

$$A_{dB}(f) = -2.184\left(\frac{f}{1000}\right)^{-0.8} + 6.5e^{-0.6\left(\frac{f}{1000-3.3}\right)^2} - 10^{-3}\left(\frac{f}{1000}\right)^{3.6} \quad \text{(B.3)}$$

$$W(f) = 10^{A_{dB}(f)/20}. \quad \text{(B.4)}$$

A plot of the weighting function is presented in Figure B.1. Using the outer and middle ear weighting the weighted DFT is

$$|X_w[k]| = W(f)\ |X[k]|. \quad \text{(B.5)}$$

In order to save computational effort, the internal noise is added after the bark domain transformation.

B.4 Frequency Grouping

As discussed in Section 3.1.3, the Cochlea can be characterised by a bank of bandpass filters, which represent critical bands. It is necessary to group the elements of $|X_w[k]|$ to replicate this behaviour. This grouping is achieved by a frequency (Hz) to Bark scale conversion using a simplified approximation

of the critical band conversion of Eq. 3.3 (Schroder et al., 1979),

$$z = B(f) = 7 \operatorname{arcsinh}\left(\frac{f}{650}\right) \text{ (Bark)}. \tag{B.6}$$

This conversion maps the frequencies 80 Hz to 18,000 Hz to 27 non-overlapping critical bands, which have a bandwidth of 1 Bark each. For the PEAQ algorithm, a fractional critical band representation is used, where the widths and spacing of the critical band groupings are defined as 1/4 Bark, corresponding to 109 critical bands. Each band i has a lower frequency edge, $f_l[i]$, a centre frequency, $f_c[i]$, and an upper frequency edge, $f_u[i]$. The frequency of the k-th bin of $|X_w[k]|$ is kF_s/N_f Hz. The bins that fall within the range of $f_l[i]$ and $f_u[i]$ are grouped together ($k_l[i] \leq k \leq k_u[i]$) and the energy contribution from each bin is calculated as

$$U[i,k] = \frac{\max[0, \min(f_u[i], \frac{2k+1}{2}\frac{F_s}{N_f}) - \max(f_l[i], \frac{2k-1}{2}\frac{F_s}{N_f})]}{\frac{F_s}{N_f}}. \tag{B.7}$$

The energy contributions are summed to give the energy in the i-th critical band,

$$E_b[i] = U_l[i]\ |X_w[k_l[i]]| + \sum_{k=k_l[i]+1}^{k_u[i]-1} |X_w[k]| + U_u[i]\ |X_w[k_u[i]]|, \tag{B.8}$$

where $U_l[i] = U[i, k_l[i]]$ and $U_u[i] = U[i, k_u[i]]$.

B.5 Internal Noise

Internal noise is generated in the ear, and can be caused by blood flow, muscle activity, shot noise and neuronal noise. An offset is added to the critical band energies to compensate for this noise,

$$E[i] = E_b[i] + E_{IN}[i]. \tag{B.9}$$

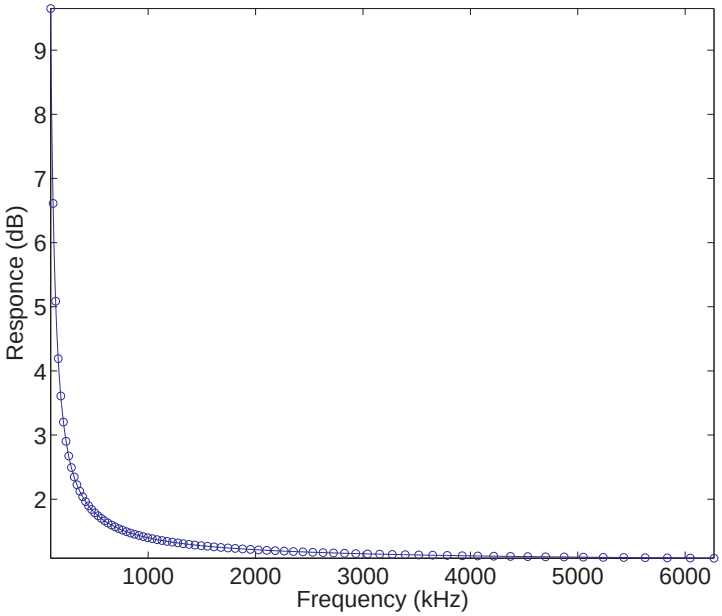

Figure B.2: Internal noise contribution for the first 80 critical bands, where ∘ indicates the centre frequencies of each band.

As previously discussed, the internal noise, $E_{IN}[i]$, is modelled by the first term of Eq. B.2,

$$E_{INdB}(f) = (1-w)\cdot 3.64\left(\frac{f}{1000}\right)^{-0.8} = 1.456\left(\frac{f}{1000}\right)^{-0.8}, \quad \text{(B.10)}$$

$$E_{IN}(i) = 10^{E_{INdB}(f)/20}. \quad \text{(B.11)}$$

The factor $w = 0.6$ (as specified in Section B.3) and the response is plotted in Figure B.2. The energy bands $E[i]$ are known as *pitch patterns*.

B.6 Frequency Spreading

The spread of masking effects that are evident during simultaneous masking are modelled by an energy spreading function that is level and frequency dependant. The bark domain energy spread response is

$$E_s[i] = \frac{1}{B_s[i]}\left(\sum_{l=0}^{N_c-1}\left(E[l]\ S(i,l,E[l])\right)\right), \quad \text{(B.12)}$$

where N_c is the number of critical bands. The normalising factor is calculated for a reference level of 0 dB for each band,

$$B_s[i] = \left(\sum_{l=0}^{N_c-1} (S(i,l,E_0)) \right), \tag{B.13}$$

where $E_0 = 1$ (0 dB). On the dB scale, the spreading function, $S(i,l,E_0)$, is triangular, with the peak of the triangle at $i = l$

$$S_{dB}(i,l,E) = \begin{cases} 27(i-l)\frac{1}{4}, & i \leq l, \\ [-24 - \frac{230}{f_c[l]} + 2\log_{10}(E)](i-l)\frac{1}{4}, & i \geq l. \end{cases} \tag{B.14}$$

The slope for bark values less than l is fixed, whereas the slope for i larger than l is frequency and level dependant. The spreading function is defined in terms of the spreading function expressed in dB,

$$S(i,l,E[l]) = \frac{1}{A(l,E)} 10^{S_{dB}(i,l,E)/20}, \tag{B.15}$$

where the normalising factor, $A(l,E)$, is chosen to give unit area to the spreading function in the magnitude domain. The patterns derived from this process are referred to as *unsmeared excitation patterns* (unsmeared in time).

B.7 Time Domain Spreading

In order to replicate the effects of temporal masking, a frequency dependant temporal filtering is performed on the unsmeared excitation patterns. In contrast to the previous steps this operation is performed on multiple frames that are contiguous in time. Consequently, a frame index n in introduced. Frames are updated every N_f/o samples and the frame rate is

$$F_{ss} = \frac{F_s}{N_f/o}, \tag{B.16}$$

where o is the reciprocal of the frame overlap. Time domain spreading is performed by a first order smoothing filter,

$$E_f[i,n] = \alpha[i]E_f[i,n-1] + (1-\alpha[i])E_s[i,n], \tag{B.17}$$

$$\tilde{E}_s[i,n] = \max\left(E_f[i,n], E_s[i,n]\right), \tag{B.18}$$

where $E_f[i,n]$ is the spread energy in band i at frame n and $\tilde{E}_s[i,n]$ is the maximum of the spread energy band or the unsmeared energy band. The maximum operation ensures that $\tilde{E}_s[i,n]$ follows increases in energy instantaneously *i.e.* the filter delay is not evident at the output. The parameter $\alpha[i]$ is a frequency dependant decaying coefficient for band i and is controlled by a time constant, $\tau[i]$,

$$\tau[i] = \tau_{\min} + \frac{100}{f_c[i]}\left(\tau_{100} - \tau_{\min}\right), \tag{B.19}$$

$$\alpha[i] = \exp\left(-\frac{1}{F_{ss}\tau[i]}\right), \tag{B.20}$$

where $\tau_{\min} = 8$ ms and $\tau_{100} = 30$ ms. The resultant patterns, $\tilde{E}_s[i,n]$, are known as *excitation patterns* and represent the physical activity of the hair cells along the basilar membrane.

B.8 Masking Threshold

The masking threshold lies below the level of the excitation and is obtained by introducing a frequency dependant offset to the excitation patterns. For the PEAQ algorithm the offset is defined as

$$m_{dB}[k] = \begin{cases} 3, & k \leq 48, \\ 0.25k\frac{1}{4}, & k \geq 48 \end{cases} \tag{B.21}$$

$$m[k] = 10^{M_{dB}[k]/20}, \tag{B.22}$$

and the masking threshold is calculated as

$$M[k,n] = \frac{1}{m[k]}\tilde{E}_s[i,n]. \tag{B.23}$$

The intermediary representations used in creating the masking threshold are presented in Figure B.3.

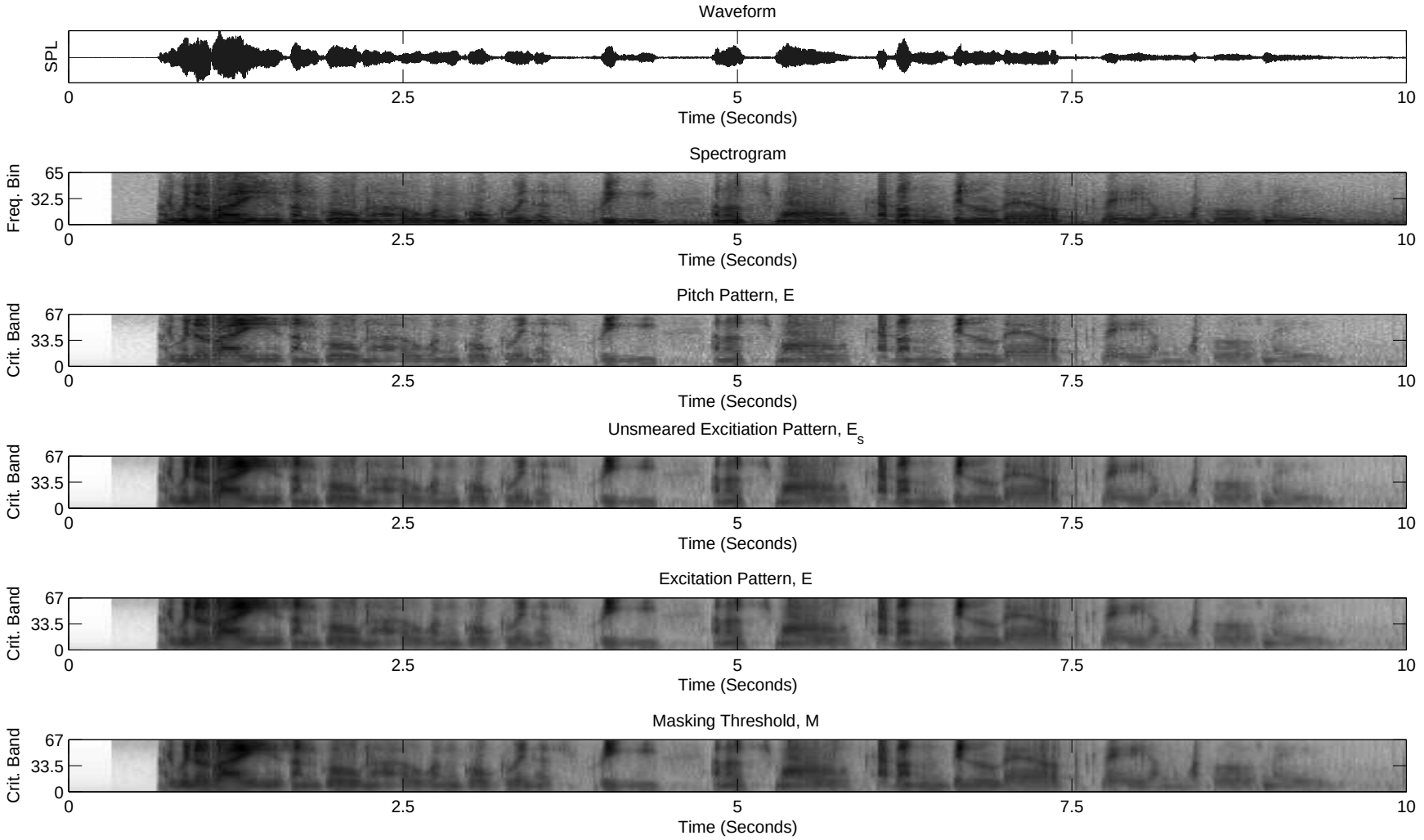

Figure B.3: Excitation patterns and masking threshold for 10 seconds of speech taken from the poem *The Lake Isle of Innisfree* by William Butler Yeats. The recording was taken from a commercial audio CD of poems read by their authors (Paschen et al., 2001)

bibliography

S. A. Abdallah and M. D. Plumbley. Polyphonic transcription by non-negative sparse coding of power spectra. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR 2004)*, pages 318–25, 2004.

M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, November 2006.

S. Amari, A. Cichocki, and H. H. Yang. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems 8*. MIT Press, 1996.

J. Anemüller and B. Kollmeier. Adaptive separation of acoustic sources for anechoic conditions: A constrained frequency domain approach. *Speech Commun.*, 39(1-2):79–95, 2003. ISSN 0167-6393.

M. Babaie-Zadeh, A. Mansour, C. Jutten, and F. Marvasti. A geometric approach for separating several speech signals. In C. G. Puntonet and A. Prieto, editors, *ICA*, volume 3195 of *Lecture Notes in Computer Science*, pages 798–806. Springer, 2004. ISBN 3-540-23056-4.

A. D. Back and A. S. Weigend. A first application of independent component analysis to extracting structure from stock returns. *Int. J. on Neural Systems*, 8(4):473–484, 1997.

A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–59, 1995.

A. Belouchrani and J.-F. Cardoso. Maximum likelihood source separation for discrete sources. In *Proc. EUSIPCO*, pages 768–71, 1994.

A. Belouchrani, K. Abed Meraim, J.-F. Cardoso, and É. Moulines. A blind source separation technique based on second order statistics. *IEEE Trans on Signal Proc*, 45(2):434–44, Feb. 1997.

P. Bofill and M. Zibulevsky. Underdetermined blind source separation using sparse representations. *Signal Processing*, 81(11):2353–62, 2001.

P. Bofill and M. Zibulevsky. Blind separation of more sources than mixtures using the sparsity of the short-time fourier transform. In *2nd International Workshop on Independent Component Analysis and Blind Signal Separation*, pages 87–92, Helsinki, Finland, June 19–20 2000.

K. Brandenburg. OCF - a new coding algorithm for high quality sound signals. In *Proceedings, International Conference on Acoustics, Speech, and Signal Processing*, pages 141–144. IEEE Press, 1987.

A. S. Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, Cambridge, Massachusetts, 1990. ISBN 0-262-02297-4.

E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *Information Theory, IEEE Transactions on*, 52(2):489–509, 2006. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1580791`.

J.-F. Cardoso. Eigen-structure of the fourth-order cumulant tensor with application to the blind source separation problem. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'90)*, pages 2655–2658, Albuquerque, New Mexico, 1990.

J. F. Cardoso. Infomax and maximum likelihood for blind source separation. *IEEE Signal Processing Letters*, 4(4):112–114, Apr. 1997.

J.-F. Cardoso and A. Souloumiac. Blind beamforming for non gaussian signals. *IEE Proceedings-F*, 140(6):362–70, Dec. 1993. URL `ftp://tsi.enst.fr/pub/jfc/Papers/iee.ps.gz`.

J.-F. Cardoso, J. Delabrouille, and G. Patanchon. Independent component analysis of the cosmic microwave background. In *Fourth International Symposium on Independent Component Analysis and Blind Signal Separation*, pages 1111–6, Nara, Japan, Apr. 1–4 2003.

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

C. E. Cherry. Some experiments in the recognition of speech, with one and two ears. *Journal of the Acoustical Society of America*, 25:975–9, 1953.

A. Cichocki, R. Zdunek, and S.-i. Amari. Csiszár's divergences for non-negative matrix factorization: Family of new algorithms. In J. P. Rosca, D. Erdogmus, J. C. Príncipe, and S. Haykin, editors, *Independent Component Analysis and Blind Signal Separation, 6th International Conference, ICA 2006, Charleston, SC, USA, March 5-8, 2006, Proceedings*, volume 3889 of *Lecture Notes in Computer Science*, pages 32–39. Springer, 2006. ISBN 3-540-32630-8. URL `http://dx.doi.org/10.1007/11679363_5`.

P. Comon. Independent component analysis: A new concept. *Signal Processing*, 36:287–314, 1994.

G. Darmois. Analyse générale des liaisons stochastiques. *Rev. Inst. Internat. Stat.*, 21:2–8, 1953.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1976.

M. R. DeWeese, M. Wehr, and A. M. Zador. Binary spiking in auditory cortex. *J. Neurosci.*, 23(21):7940–9, 2003.

D. Donoho and V. Stodden. When does non-negative matrix factorization give a correct decomposition into parts? In *Advances in Neural Information Processing Systems 16*. MIT Press, 2004. URL `http://books.nips.cc/papers/files/nips16/NIPS2003_LT10.pdf`.

J. Du, C.-H. Lee, H.-K. Lee, and Y.-H. Suh. BSS: A new approach for watermark attack. In *Proceedings of the 4th International Symposium on Multimedia Software Engineering (ISMSE 2002)*, pages 182–7, 2002.

J. Egan and H. Hake. On the masking pattern of a simple auditory stimulus. *Journal of the Acoustical Society of America*, 22:622–630, 1950.

J. Eggert and E. Körner. Sparse coding and NMF. In *IEEE International Joint Conference on Neural Networks, 2004. Proceedings*, volume 4, pages 2529–2533. IEEE, July 2004.

C. Févotte, R. Gribonval, and E. Vincent. BSS_EVAL toolbox user guide. Technical Report 1706, IRISA, 2005.

D. J. Field. What is the goal of sensory coding? *Neural Computation*, 6: 559–601, 1994.

D. FitzGerald, M. Cranitch, and E. Coyle. Sound source separation using shifted non-negative tensor factorisation. In *Proceedings, IEEE International Conference on Acoustics, Speech and Signal Processing*, 2006. URL `http://homepage.eircom.net/~derryfitzgerald/ICASSP06.pdf`.

P. Földiák and M. Young. Sparse coding in the primate cortex. In *The Handbook of Brain Theory and Neural Networks*, pages 895–8. MIT Press, 1995.

M. Gaeta and J.-L. Lacoume. Source separation without prior knowledge: the maximum likelihood solution. In *Proc. EUSIPCO'90*, pages 621–4, 1990.

J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. S. Pallett, and N. L. Dahlgren. DARPA TIMIT acoustic-phonetic continuous speech corpus. Feb. 1993.

J. Herault and C. Jutten. Space or time adaptive signal processing by neural models. In *Proceedings AIP Conference: Neural Networks for Computing*, pages 206–11. American Institute of Physics, 1986.

P. O. Hoyer. Non-negative sparse coding. In *IEEE Workshop on Neural Networks for Signal Processing*, 2002.

A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–92, Oct. 1997.

F. Itakura and S. Saito. An analysis-synthesis telephony based on maximum likelihood method. In *6th Int. Conf. Acoustics*, pages 17–20, 1968.

ITU. Recommendation BS.1387, method for objective measurements of perceived audio quality, Dec. 1998.

A. Jourjine, S. Rickard, and O. Yilmaz. Blind separation of disjoint orthogonal signals: Demixing N sources from 2 mixtures. In *IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP2000)*, volume 5, pages 2985–2988, June 2000.

T.-P. Jung, S. Makeig, M. Westerfield, J. Townsend, E. Courchesne, and T. J. Sejnowski. Analyzing and visualizing single-trial event-related potentials. In *Advances in Neural Information Processing Systems 11*, pages 118–24. MIT Press, 1999.

T.-P. Jung, C. Humphries, T.-W. Lee, M. J. McKeown, V. Iragui, S. Makeig, and T. J. Sejnowski. Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, 37:163–78, 2000.

C. Jutten and J. Herault. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10, 1991.

J. Karvanen and A. Cichoki. Measuring sparseness of noisy signals. In *ICA03*, 2003.

R. Kompass. A generalized divergence measure for non-negative matrix factorization. In *Neuroinformatics workshop*, Torun, Poland, Sept. 2005.

K. P. Körding, P. König, and D. J. Klein. Learning of sparse auditory receptive fields. In *International Joint Conference on Neural Networks*, 2002. URL `http://www.koerding.com/pubs/KKKijcnn01.pdf`.

S. Kullback. Information theory and statistics. 1959.

R. H. Lambert. A new method for source separation. In *Proceedings, IEEE Conference on Acoustics, Speech, and Signal Processing*, pages 2116–9, Detroit, MI, 1995.

R. H. Lambert. *Multichannel Blind Deconvolution: FIR Matrix Algebra and Separation of Multipath Mixtures*. PhD thesis, Univ. of Southern California, 1996.

D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13*, pages 556–62. MIT Press, 2001. URL `citeseer.ist.psu.edu/lee00algorithms.html`.

K. F. Lee and H. W. Hon. Speaker-independent phone recognition using hidden Markov models. *IEEE Trans. Acoustics, Speech and Signal Processing*, ASSP-37(11):1641–1648, 1989.

T.-W. Lee, M. S. Lewicki, M. Girolami, and T. J. Sejnowski. Blind source separation of more sources than mixtures using overcomplete representations. *IEEE Signal Processing Letters*, 4(5):87–90, 1999.

M. Lewicki and T. J. Sejnowski. Learing overcomplete representations. In *Advances in Neural Information Processing Systems 10*, pages 556–562. MIT Press, 1998.

J. K. Lin, D. G. Grier, and J. D. Cowan. Feature extraction approach to blind source separation. In *IEEE Workshop on Neural Networks for Signal Processing (NNSP)*, pages 398–405, 1997.

R. Linsker. An application of the principle of maximum information preservation to linear systems. In *Advances in Neural Information Processing Systems*, pages 186–94. Morgan Kaufmann, 1989.

M. J. McKeown, S. Makeig, G. G. Brown, T. P. Jung, S. S. Kindermann, A. J. Bell, and T. J. Sejnowski. Analysis of fMRI data by blind separation into independent spatial components. *Human Brain Mapping*, 6(3):160–88, 1998.

N. Mitianoudis and T. Stathaki. Overcomplete source separation using laplacian mixture models. *IEEE Signal Processing Letters*, 12(4):277–280, 2005.

K. Nakadai, H. G. Okuno, and H. Kitano. Real-time sound source localization and separation for robot audition. In *Proceedings of 2002 International Conference on Spoken Language Processing (ICSLP-2002)*, pages 193–6, 2002.

P. D. O'Grady and B. A. Pearlmutter. Hard-LOST: Modified k-means for oriented lines. In *Proceedings of the Irish Signals and Systems Conference*, pages 247–52, Belfast, June 30–July 2 2004a.

P. D. O'Grady and B. A. Pearlmutter. Soft-LOST: EM on a mixture of oriented lines. In *Fifth International Conference on Independent Component Analysis*, LNCS 3195, pages 430–6, Granada, Spain, Sept. 22–24 2004b. Springer-Verlag.

P. D. O'Grady and B. A. Pearlmutter. Convolutive non-negative matrix factorisation with sparseness constraint. In *International Workshop on Machine Learning for Signal Processing*, pages 427–432, Maynooth, Ireland, Sept. 6–8 2006. IEEE Press.

P. D. O'Grady, B. A. Pearlmutter, and S. T. Rickard. Survey of sparse and non-sparse methods in source separation. *International Journal of Imaging Systems and Technology (IJIST)*, 15:18–33, 2005. Special issue on blind source separation and deconvolution in imaging and image processing.

B. A. Olshausen and D. J. Field. Sparse coding of sensory inputs. *Curr Opin Neurobiol*, 14(4):481–7, 2004.

P. Paatero and U. Tapper. Positive matrix factorization: A nonnegative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5:111–26, 1994.

T. Painter and A. Spanias. Perceptual coding of digital audio. *Proceedings of the IEEE*, 88:451–513, 2000.

L. Parra and P. Sajda. Blind source separation via generalized eigenvalue decomposition. *Journal of Machine Learning Research*, 4:1261–9, 2003.

E. Paschen, C. Osgood, and R. P. Mosby, editors. *Poetry Speaks: Hear Great Poets Read Their Work from Tennyson to Plath*. Sourcebooks Incorporated, 2001. ISBN 1570717206.

B. A. Pearlmutter and S. Jaramillo. Progress in blind separation of magnetoencephalographic data. In *Independent Component Analyses, Wavelets, and Neural Networks*, volume 5102 of *Procceding of the SPIE*, pages 129–34, Apr. 2003.

B. A. Pearlmutter and L. C. Parra. A context-sensitive generalization of ICA. In *International Conference on Neural Information Processing*, pages 151–7, Hong Kong, Sept. 24–27 1996. Springer-Verlag. URL `http://www.bcl.hamilton.ie/~barak/papers/iconip-96-cica.ps.gz`.

B. A. Pearlmutter and A. M. Zador. Monaural source separation using spectral cues. In *Fifth International Conference on Independent Component Analysis*, LNCS 3195, pages 478–85, Granada, Spain, Sept. 22–24 2004. Springer-Verlag.

K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2:559–572, 1901.

D. T. Pham, P. Garrat, and C. Jutten. Separation of a mixture of independent sources through a maximum likelihood approach. In *European Signal Processing Conference*, pages 771–4, 1992.

R. K. Potter, G. A. Kopp, and H. C. Green. *Visible Speech*. D. Van Nostrand Company, 1947.

S. R. Quackenbush, T. P. Barnwell, and M. A. Clements. *Objective Measures of Speech Quality*. Prentice-Hall, Englewood Cliffs, NJ, 1988.

S. Rickard and M. Fallon. The Gini index of speech. In *Conference on Information Sciences and Systems (CISS2004)*, March 2004.

S. Rickard, R. Balan, and J. Rosca. Real-time time-frequency based blind source separation. In *3rd International Conference on Independent Component Analysis and Blind Source Separation (ICA2001)*, December 2001.

S. T. Rickard and F. Dietrich. DOA estimation of many W-disjoint orthogonal sources from two mixtures using DUET. In *Proceedings of the 10th IEEE Workshop on Statistical Signal and Array Processing (SSAP2000)*, pages 311–4, Pocono Manor, PA, Aug. 2000.

Z. Roth and Y. Baram. Multi-dimensional density shaping by sigmoids. *IEEE Transactions on Neural Networks*, 7(5):1291–8, 1996.

S. T. Roweis. One microphone source separation. In *Advances in Neural Information Processing Systems 13*, pages 793–9. MIT Press, 2001.

B. Scharf. Critical bands. In J. V. Tobias, editor, *Foundations of Modern Auditory Theory*, chapter 5, pages 159–202. Academic Press, 1970.

M. R. Schroder, B. S. Atal, and J. L. Hall. Optimizing digital speech coders by exploiting masking properties of the human ear. *Journal of the Acoustical Society of America*, 66:1647–1652, 1979.

P. Smaragdis. Convolutive speech bases and their application to supervised speech separation. *IEEE Transaction on Audio, Speech and Language Processing*, 2007.

P. Smaragdis. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In *Fifth International Conference on Independent Component Analysis*, LNCS 3195, pages 494–9, Granada, Spain, Sept. 22–24 2004. Springer-Verlag.

A. C. Tang, B. A. Pearlmutter, M. Zibulevsky, and S. A. Carter. Blind separation of multichannel neuromagnetic responses. *Neurocomputing*, 32–33:1115–20, 2000.

E. Terhardt. Calculating virtual pitch. *Hearing Research*, 1:155–182, 1979.

F. Theis, E. Lang, and C. Puntonet. A geometric algorithm for overcomplete linear ica. *Neurocomputing*, 5:381–398, 2004.

F. J. Theis, A. Jung, C. G. Puntonet, and E. W. Lang. Linear geometric ICA: Fundamentals and algorithms. *Neural Computation*, 15:419–439, 2003. URL `http://www.biologie.uni-regensburg.de/Biophysik/Theis/publications/theis03linearGeoICA_NC.pdf`.

T. Thiede. *Perceptual Audio Quality Assessment using a Non-Linear Filter Bank*. PhD thesis, Technische Universität Berlin, 1999.

K. Torkkola. Blind separation of delayed sources based on information maximization. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'96)*, pages 3509–3512, Atlanta, Georgia, 1996.

M. van Hulle. Clustering approach to square and non-square blind source separation. In *IEEE Workshop on Neural Networks for Signal Processing (NNSP)*, pages 315–23, 1999.

L. Vielva, D. Erdogmus, and J. C. Principe. Underdetermined blind source separation using a probabilistic source sparsity model. In *2nd International Workshop on Independent Component Analysis and Blind Signal Separation*, pages 675–9, Helsinki, Finland, June 19–20 2000.

L. Vielva, D. Erdogmus, C. Pantaleon, I. Santamaria, J. Pereda, and J. C. Principe. Underdetermined blind source separation in a time-varying environment. In J. Principe and H. Bourlard, editors, *Proceedings, IEEE*

International Conference on Acoustics, Speech and Signal Processing, volume 3, pages 3049–52, 2002.

R. Vigário, J. Särelä, V. Jousmäki, M. Hämäläinen, and E. Oja. Independent component approach to the analysis of EEG and MEG recordings. *IEEE Transactions on Biomedical Engineering*, 47(5):589–93, 2000.

T. Virtanen. Sound source separation using sparse coding with temporal continuity objective. In *in Proceedings of the International Computer Music Conference (ICMC 2003)*, 2003.

N. Wiener. *Extrapolation, Interpolation and Smoothing of Stationary Time Series*. MIT Press, 1949.

G. Wübbeler, A. Ziehe, B.-M. Mackert, K.-R. Müller, L. Trahms, and G. Curio. Independent component analysis of non-invasively recorded cortical magnetic DC-fields in humans. *IEEE Transactions on Biomedical Engineering*, 47(5):594–9, 2000.

O. Yilmaz and S. Rickard. Blind separation of speech mixtures via time-frequency masking. *IEEE Transactions on Signal Processing*, 52(7):1830–1847, July 2004.

M. Zibulevsky and B. A. Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. *Neural Computation*, 13(4):863–82, Apr. 2001.

M. Zibulevsky, P. Kisilev, Y. Y. Zeevi, and B. A. Pearlmutter. Blind source separation via multinode sparse representation. In *Advances in Neural Information Processing Systems 14*, pages 1049–56. MIT Press, 2002.

A. Ziehe, K.-R. Müller, G. Nolte, B.-M. Mackert, and G. Curio. Artifact reduction in magnetoneurography based on time-delayed second order correlations. *IEEE Transactions on Biomedical Engineering*, 47(1):75–87, Jan. 2000.

E. Zwicker and H. Fastl. *Psychoacoustics: Facts and Models*. Springer, 1999.