UNIVERSITY OF CALIFORNIA, SAN DIEGO

Variational and Scale Mixture Representations of Non-Gaussian Densities for Estimation in the Bayesian Linear Model: Sparse Coding, Independent Component Analysis, and Minimum Entropy Segmentation

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy
in
Electrical Engineering (Intelligent Systems, Robotics, and Control)

by

Jason Allan Palmer

Committee in charge:

Professor Kenneth Kreutz-Delgado, Chair
Professor Charles Elkan
Professor Philip E. Gill
Professor Bhaskar D. Rao
Professor Nuno Vasconcelos

2006

The dissertation of Jason Allan Palmer is approved, and it is acceptable in quality and form for publication on microfilm:

Chair

University of California, San Diego

2006

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Professor Ken Kreutz-Delgado, for his guidance and for the invaluable intuitive understanding of problems that he provided. I am also grateful to Professors Bhaskar D. Rao and Philip E. Gill for the example of excellence in research.

I would also like to thank Dr. Scott Makeig for his support, and for his assistance in development of applications to brain imaging. These applications have benefitted from discussion and collaboration with others at the Swartz Center for Computational Neuroscience, particularly Julie Onton, Rey Ramirez, and Nima Bigdely Shamlo.

I am especially grateful to my colleague and friend, David Wipf, for his collaboration, and for many useful discussions regarding the topics considered in this thesis.

Finally, I am grateful to my family and friends, and to my mom in particular for her constant encouragement and support.

Chapter 3, in part, is a reprint of material contained in the paper, Palmer, J. A., Wipf, D. P., Kreutz-Delgado, K., Rao, B. D., "Variational EM Algorithms for Non-Gaussian Latent Variable Models," Advances in Neural Information Processing Systems, MIT Press, 2005. The dissertation author is the primary author of this paper.

VITA

1974	Born, Tallahassee, FL
1996	B.A. University of Chicago
1999	B.S. Illinois Institute of Technology
2000–2004	Teaching Assistant, University of California, San Diego
2001	M.S., University of California, San Diego
2001–2006	Research Assistant, University of California, San Diego
2006	Ph.D., University of California, San Diego

PUBLICATIONS

“A Globally Convergent Algorithm for MAP Estimation with Non-Gaussian Priors,” Proceedings of the 36th Asilomar Conference on Signals and Systems, 2002.

“A General Framework for Component Estimation,” Proceedings of the 4th International Symposium on Independent Component Analysis, 2003.

“Variational EM Algorithms for Non-Gaussian Latent Variable Models,” Advances in Neural Information Processing Systems, 2005.

“Super-Gaussian Mixture Source Model for ICA,” Proceedings of the 6th International Symposium on Independent Component Analysis, 2006.

FIELDS OF STUDY

Major Field: Electrical and Computer Engineering

Studies in Parameter Estimation and Signal Processing.
Professors Kenneth Kreutz-Delgado and Bhaskar D. Rao

Studies in Machine Learning and Statistical Learning Theory.
Professors Charles Elkan and Nuno Vasconcelos

Studies in Numerical Computation.
Professor Philip E. Gill

ABSTRACT OF THE DISSERTATION

Variational and Scale Mixture Representations of Non-Gaussian Densities for
Estimation in the Bayesian Linear Model:
Sparse Coding, Independent Component Analysis, and
Minimum Entropy Segmentation

by

Jason Allan Palmer
Doctor of Philosophy in
Electrical Engineering (Intelligent Systems, Robotics, and Control)
University of California, San Diego, 2006
Professor Kenneth Kreutz-Delgado, Chair

This thesis considers representations of non-Gaussian probability densities for use in various estimation problems associated with the Bayesian Linear Model. We define a class of densities that we call Strongly Super-Gaussian, and show the relationship of these densities to Gaussian Scale Mixtures, and densities with positive kurtosis. Such densities have been used to model "sparse" random variables, with densities that are sharply peaked with heavy tails. We show that strongly super-Gaussian densities are natural generalizations of Gaussian densities, and permit the derivation of monotonic iterative algorithms for parameter estimation in sparse coding in overcomplete signal dictionaries, blind source separation, independent component analysis, and blind multichannel deconvolution. Mixtures of strongly super-Gaussian densities can be used to model arbitrary densities with greater economy that a Gaussian mixture model. The framework is extended to multivariate dependency models for independent subspace analysis. We apply the methods to the estimation of neural electro-magnetic sources from electro-encephalogram recordings, and to sparse coding of images.

1

Introduction

This thesis is concerned with representations of non-Gaussian probability densities that lead to monotonic, closed-form update algorithms for parameter estimation, with particular application to estimation in the context of the discrete-time Bayesian Linear Model,

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \nu(t) \tag{1.1}$$

Here $\mathbf{x}(t) \in \mathbb{R}^m$, $t = 1, \ldots, T$, is an observed vector process, $\mathbf{A} \in \mathbb{R}^{m \times n}$ is a matrix that may be thought of as a *basis* for the signal $\mathbf{x}(t)$, driven by the non-Gaussian, and temporally independent and identically distributed *source* process $\mathbf{s}(t) \in \mathbb{R}^n$ and *noise* process, $\nu(t)$, which will generally be taken to be Gaussian.

A particular case where the non-Gaussian character of the source $\mathbf{s}(t)$ is important is when $\mathbf{s}(t)$ is *sparse*, meaning that it has only a relatively small number of non-zero elements. The non-zero elements in this case correspond to vectors in the *dictionary* $\mathbf{A}$, which may be thought of as features present in the observation $\mathbf{x}(t)$. Sparse random variables are those with densities with high probability of being close to zero, but with relatively large *tails*, allowing the random variable to take on relatively large magnitudes as well. The density models we consider are, in their basic form, particularly suited to the representation of sparse random variables.

Sparsity is associated with the term *super-Gaussian*, which essentially refers to their having a sparse character relative to the Gaussian density, which, for fixed variance, is less sharply peaked at zero with more negligible probability of taking on magnitudes far from zero. *Sub-Gaussian* densities are those which are flatter, or more uniform at zero, with less probability of being outside this region, relative to Gaussian. The most commonly used measure of this relationship to the Gaussian density is the *kurtosis*, which is a function of the fourth moment, and is zero for the Gaussian density, positive of super-Gaussian densities, and negative for sub-Gaussian densities. The density representation we develop can also be thought of as characterizing super-Gaussianity in a form that is useful for the derivation of globally convergent algorithms for parameter estimation with these densities.

We consider in particular two forms of super-Gaussian density representations: Gaussian scale mixtures, and a more general representation based on variational representation of concave functions. Scale mixtures have the general form,

$$p(s) = \int_0^\infty K(s/\xi)\, d\mu(\xi)/\xi$$

where μ is a non-decreasing and bounded function on $(0, \infty)$ [59] and K is a kernel density. Gaussian scale mixtures are represented in the form,

$$p(s) = \int_0^\infty \mathcal{N}(s; 0, \xi^{-1})\, d\mu(\xi) \tag{1.2}$$

where $\mathcal{N}(s; \mu, \sigma^2)$ denotes the Gaussian density with mean μ and variance σ^2. In the form (1.2), $p(s)$ is represented as an integral over the inverse variance ξ^{-1}. In fact, a random variable with a Gaussian scale mixture density can be represented as the product of a unit variance Gaussian random variable Z, and a non-negative random variable Ξ,

$$S = Z\Xi^{-1/2}$$

The representation (1.2) is a type of scale convolution for the density of products as ordinary convolution is used in the computation of the density of sums of random variables. We shall concentrate in this thesis on the case were $\mu(\xi)$ is differentiable so that $d\mu(\xi) = p(\xi)d\xi$.[1]

A more general type of representation for super-Gaussian densities is given by the pointwise supremum over Gaussian densities of varying scale,

$$p(s) = \sup_{\xi>0} \mathcal{N}(s; 0, \xi^{-1})\, \varphi(\xi) \tag{1.3}$$

Such a representation is the basis for many algorithms that have been proposed recently. The analysis and application of such densities is the main topic of this thesis.

Another case involving non-Gaussian sources is the problem of *blind* separation of source signals $\mathbf{s}(t) = [s_1(t) \cdots s_n(t)]^T$, which are observed in linear superposition at a set of m sensors as the processes $x_i(t) = \sum_{j=1}^n a_{ij} s_j(t)$, $i = 1, \ldots, m$, or $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$. Here the source signals may have arbitrary probability densities, and we are given only the assumption that the sources $s_i(t)$, $i = 1, \ldots, n$ are mutually *independent* with arbitrary and not necessarily identically distributed densities $p_i(s_i)$. The representations we develop, while of sparse character in their basic form, may be combined in *mixture models* to represent essentially arbitrary densities, while still being amenable to the derivation of closed-form, monotonic parameter estimation algorithms. This generalizes the Gaussian mixture model, yielding a source model with greater flexibility than the Gaussian mixture model with similar model and computational complexity.

In addition to the density representation theory, we develop a general framework for estimation and modeling with linear mixture models which we call the *linear process mixture model.* This model encompasses and generalizes current state-of-the-art algorithms for multichannel, multidimensional blind source separation and deconvolution, with *non-stationary*, *non-minimum phase* sources, including models with dependent subspaces. The mixture model is derived by approximating the probability of a signal segment using a known Toeplitz determinant formula, and modeling the signal as stationary over individual segments,

[1] We shall often denote probability densities by $p(\,\cdot\,)$, distinguishing them only by their argument when the context is clear. Thus densities $p(s)$ and $p(\xi)$ here are entirely different probability densities.

with different linear models active in different segments. The methods used here are not new, but the application of the approximate segment probability to the derivation of mixture models for representing non-stationary signals seems not to have been proposed before.

In the following sections we further describe the problems considered in this thesis with reference to previous developments and related models in the literature.

1.1 Super-Gaussianity and Sparse coding

In this case we shall be interested in the problem of estimation of the source vector $\mathbf{s}(t)$ for a given observation $\mathbf{x}(t)$ and a given signal dictionary $\mathbf{A}$, and in the problem of the estimation or *learning* of a signal dictionary $\mathbf{A}$ that yields sparse representations of an observation process $\mathbf{x}(t)$, given the set of observations $\mathbf{x}(t)$, $t = 1, \ldots, T$.

1.1.1 Finding Sparse Representations in a Given Dictionary

The problem of finding sparse solutions to an underdetermined linear system has been the investigated recently, largely as a result of research into signal bases other than the Fourier basis, including wavelets, and more general dictionaries of signal "atoms" [25, 21, 75]. In the sparse representation problem, for a given observation $\mathbf{x}$, we may think of the number of non-zero elements in the representation $\mathbf{s}$ as a cost function that is to be minimized subject to $\mathbf{As} \approx \mathbf{x}$. However, representation of this problem by continuous functions such as $\sum_i \log |s_i|$, or $\sum_i |s_i|^p$, $p < 1$, which approximate the number of non-zero elements measure, leads to non-convex optimization problems, which may not be differentiable or convex at the optimum.

Various algorithms have been developed to solve these problems, including sequential methods [75, 25, 27, 28], which sequentially select vectors from the dictionary, and global optimization methods such as Basis Pursuit [21], which optimizes the convex function $\sum_i |s_i|$, which is concave in $|s_i|$, subject to the constraint $\mathbf{As} = \mathbf{x}$.

In a series of papers [89, 90, 88], Gorodnitsky, Rao, and Kreutz-Delgado developed an iteratively reweighted least squares (IRLS) algorithm called FOCUSS for minimizing the concave functions,

$$\sum_i |s_i|^p, \; p < 1$$

and the limiting function

$$\lim_{p \to 0} (1/p) \sum_i (|s_i|^p - 1) = \sum_i \log |s_i|$$

subject to the constraint $\mathbf{As} = \mathbf{x}$, or with an additional error term,

$$\sum_i |s_i|^p + \tfrac{1}{2} \left\| \mathbf{x} - \mathbf{As} \right\|^2_{\mathbf{\Sigma}^{-1}}$$

In [61], Kreutz-Delgado extended this proof to general concave functions. A contribution of the research presented in this thesis is the further generalization of this set of algorithms to its most natural context of functions that are concave in s_i^2, and the corresponding probability densities which can be represented in the form (1.3). We present in §4.2 proof of global convergence of this algorithm, which we shall refer to as Generalized FOCUSS, using Zangwill's global convergence theorem [104], as in [90, 88]. We prove a general theorem on iteratively reweighted least squares algorithms, including a novel convergence rate analysis.

An alternative sparse coding method called Sparse Bayesian Learning (SBL) was developed by Tipping [98] using ideas of Type-II Maximum Likelihood and *automatic relevance determination* from Mackay [72]. A particular form of this algorithm involving an improper scale density was shown to be superior to the FOCUSS algorithm with respect to its ability to avoid local optima by Wipf [102]. The SBL type methods generally use a form of Gaussian scale mixture. Gaussian scale mixtures (GSMs) were discussed in [59] and [3], and in the examples of Dempster, Laird, and Rubin's original EM paper [31]. GSMs were treated more extensively and applied to the analysis of IRLS algorithms in [32]. Another contribution of the research presented here is the analysis of the relationship between the Generalized FOCUSS algorithm and the SBL type algorithms, and the super-Gaussian density representations on which they rely.

1.1.2 Dictionary Learning

The dictionary learning problem is inspired partly by the development of wavelets and more general signal dictionaries, as well as neuro-biological considerations of coding strategies used in the brains to code sensory information by Barlow [6] and Field [36]. Particular emphasis was placed on the case of *overcomplete* dictionaries, i.e. matrices **A** with more columns than rows, i.e. more basis vectors than the dimension of the observation space, which is the maximum number of linearly independent vectors in that space. Having large number of possible representative vectors allows a more efficient, or sparse, representation of particular observations, and seems to accord with the brains strategy of selective response of neurons to a particular *receptive field*, found in the visual cortex by Hubel and Wiesel [48, 49].

Algorithms for dictionary learning have proposed by Olshausen and Field [79], and Lewicki and Olshausen [67] and Lewicki and Sejnowski [68] for image coding and denoising. The FOCUSS framework was applied by Kreutz-Delgado and Rao to the dictionary learning problem in [63], and Kreutz-Delgado and Murray [62]. Overcomplete dictionary learning algorithms were also developed using an by Girolami [43]. Another contribution of this thesis is the generalization and analysis of the relationships among the dictionary learning methods using the density representation framework developed.

1.2 Blind Signal Processing and Independent Component Analysis

Typically, in signal processing with a linear model, either the input signal is assumed known and the linear system is to be estimated, as in equalization, or the system is assumed known, and the input signal is to be estimated, as in some communications models. In the blind context, both the input signal and the linear system are assumed to be unknown, with either or both to be estimated. Of course some assumptions must be made to guarantee identifiability of the model. The assumption made here is that each input signal to the linear system is a linear random process field, or field, which is defined as a linearly filtered independent identically distributed (i.i.d.) random field [93, 47, 17]

Much of traditional signal processing has been based on the assumption of Gaussianity of input signals. This makes the analysis tractable, and largely serves the purposes of correlation and spectral analysis. However it has been noted that in the case of Gaussian input, the actual linear mixing filters cannot be determined, and thus cannot be inverted to determine the Gaussian input signals themselves. If uncorrelated or "white" noise is used as a model for the input signal, then the spectrum of the mixing filter can be determined, and white noise can be generated by linearly filtering the mixed signal (the Wold representation [17],) but the generated white noise signal is not unique, and thus is not guaranteed to be identical to the input signal.

In the case of non-Gaussian inputs, however, the situation is different. It has been shown that non-Gaussian signals *can* be recovered from linearly mixed versions of them. The new operating assumption concerns the *independence* of the generating signal. Each input signal is modeled as a non-Gaussian, i.i.d. sequence. Such a signal can be recovered from a linearly filtered version of it, and n such signals mixed by a multivariate linear filter can be recovered if observed by n different sensors.

In its most general statement, the model we consider has the form,

$$\mathbf{x}(t_1,\ldots,t_d) = \sum_{\tau_1}\cdots\sum_{\tau_d}\mathbf{A}(\tau_1,\ldots,\tau_d)\mathbf{s}(t_1-\tau_1,\ldots,t_d-\tau_d) \tag{1.4}$$

where $\mathbf{x}$ is an m dimensional array of sensors, or channels, observed over a d dimensional field with coordinates $t_1,\ldots,t_d$, $\mathbf{s}$ is the set of n sources defined over the same field and mixed by linear convolution with the field of matrices $\mathbf{A}(t_1,\ldots,t_n)$. The field in general is taken to be infinite, but it will be approximated by a finite field in applications.

1.2.1 Non-identifiability of Gaussian Linear Process and Identifiability of Non-Gaussian Processes

Consider the model (1.1) in the case of i.i.d. vectors $\mathbf{s}(t) \sim \mathcal{N}(\mathbf{s};\mathbf{0};\mathbf{\Sigma})$, and a complete non-singular basis $\mathbf{A} = \mathbf{W}^{-1}$. The the log probability density, or *log likelihood*, of the set of observations $\mathbf{x}(t)$, $t = 1,\ldots,T$ is,

$$\log p\big(\{\mathbf{x}(t)\}_{t=1}^{T}\big) \;=\; \sum_{t=1}^{T}\log|\det\mathbf{W}| - \tfrac{1}{2}\mathbf{x}^T\mathbf{A}^T\mathbf{\Sigma}^{-1}\mathbf{A}\mathbf{x}$$

The gradient of this function with respect to $\mathbf{W}$ is,

$$T\mathbf{W}^{-T} - \mathbf{\Sigma}^{-1}\mathbf{W}\mathbf{x}\mathbf{x}^T$$

Thus, critical points of the likelihood must satisfy,

$$\mathbf{A}\mathbf{\Sigma}\mathbf{A}^T = \frac{1}{T}\sum_{t=1}^{T}\mathbf{x}\mathbf{x}^T \triangleq \mathbf{S}$$

This equation is satisfied by,

$$\mathbf{A} = \mathbf{S}^{1/2}\mathbf{\Sigma}^{-1/2}$$

where $\mathbf{S}^{1/2}$ and $\mathbf{\Sigma}^{-1/2}$ are the unique symmetric square root matrices of the postive definite symmetric matrices $\mathbf{S}$ and $\mathbf{\Sigma}$ respectively. However it is also satisfied by,

$$\mathbf{A} = \mathbf{Q}\,\mathbf{S}^{1/2}\mathbf{\Sigma}^{-1/2} \tag{1.5}$$

for all orthonormal matrices $\mathbf{Q}$. In fact, in the case of i.i.d. $\mathbf{s} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, we have $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{A\Sigma A}^T)$. The Gaussian density is completely determined by its mean and covariance. Thus in the Gaussian case, without further constraints, we cannot hope to identify $\mathbf{A}$ or $\mathbf{s}(t)$, even if we can determine exactly the density of the observations. We can only identify the covariance of the observations, $\mathbf{S}$.

Fortunately, however, this situation is unique to the Gaussian density, and a basic theorem of Cramér shows that the only source density that produces Gaussian observations is another Gaussian. The following theorem [22] shows that non-Gaussian sources can be identified. Consider the scalar fields $a(t_1, \ldots, t_d) = a_{\mathbf{t}}$ and $\xi(t_1, \ldots, t_d) = \xi_{\mathbf{t}}$, and denote,

$$(a * \xi)_{\mathbf{t}} = \sum_{\tau_1} \cdots \sum_{\tau_d} a(\tau_1, \ldots, \tau_d) \xi(t_1 - \tau_1, \ldots, t_d - \tau_d) \tag{1.6}$$

The following theorem [93, Thm. 1.3.1], [22], gives a basic result on the identifiability of non-Gaussian linear processes.

Theorem 1. *Let,*

$$x_{\mathbf{t}} = (a * \xi)_{\mathbf{t}} = (a' * \xi')_{\mathbf{t}}, \;\; \mathbf{t} \in Z^d, \; 0 < E\xi_0^2, E\xi_0'^2 < \infty$$

with $\{\xi_{\mathbf{t}}\}$ and $\{\xi'_{\mathbf{t}}\}$ each independent and identically distributed and $\{a_{\mathbf{t}}\}$, $\{a'_{\mathbf{t}}\}$ each square summable with almost everywhere non-zero Fourier transforms. If $x_{\mathbf{t}}$ is non-Gaussian, it follows that

$$\xi'_{\mathbf{t}} = \alpha\, \xi_{\mathbf{t}-\mathbf{t}_0}, \;\; a'_{\mathbf{t}} = \beta\, a_{\mathbf{t}+\mathbf{t}_0}$$

for non-zero constants α and β, and $\mathbf{t}_0 \in Z^d$.

This theorem shows that the generating field of a non-Gaussian linear random field is unique up to scaling and translation.

1.2.2 Blind Source Separation and Blind Deconvolution

Several approaches have been developed to identify the mixing matrices $\mathbf{A}$, or their inverses $\mathbf{W} = \mathbf{A}^{-1}$. Early approaches of Comon and others used a truncated Edgeworth or related expansion of the unknown source densities $p_i(s_i)$ to approximate the density using only a finite number of moments of the observations [26]. Another approach based on diagonalization of the fourth order cumulant tensor, was proposed and developed by Cardoso [18].

A third approach, based on Maximum Likelihood estimation, we developed by Pham et al. [87, 86]. A similar approach based on entropy maximization, called Infomax, was proposed by Bell and Sejnowski [9]. A more efficient and better conditioned algorithm based on the same likelihood cost function was given by Amari [1] based on what Amari calls the *natural gradient*, and a related algorithm was proposed earlier by Cardoso and Laheld for the case of estimating an orthogonal basis for whitened observations, using what they call the relative gradient [19]. They show that this algorithm is equivariant, i.e. its convergence rate is independent of the solution to which it is converging. This approach, employing the *global system*, **WA**, whose optimum is always identity, may be traced from the work of Benveniste, Ruget, and Goursat [10] for single channel blind deconvolution. The Maximum Likelihood approach was also considered by Hyvärinen [52].

The ML approach was extended to multichannel deconvolution by Amari, Cichocki, and Douglas [34], and to mixture models by Lee [66]. More recently, *Variational Bayes* [4], or *ensemble learning* [64] methods have been proposed based on recent developments in Bayesian estimation algorithms [4, 71]. Related work on variational mixture models was proposed by [42], Choudrey and Roberts [23], and Chan and Lee [20].

ICA methods were recently extended to handle mutually independent subspaces, with *variance dependency* inducing non-Gaussian radially symmetric densities. Independent Subspace Analysis (ISA) and Topographic ICA algorithms were proposed by Hyvärinen et al. [51], and Kim et al. [60]. Other algorithms exploiting variance dependency were proposed by Park and Lee [84].

The model we propose is a general multichannel linear process mixture model, encompassing the recent variational algorithms, as well as the dependent subspace algorithms. The existing variational algorithms either use mixtures of Gaussian sources [4, 23], or they depend on particular cases of the super-Gaussian representations (1.2) or (1.3). The dependent subspace models [60, 35] likewise depend on a particular multivariate Gaussian scale mixture. A contribution of this thesis in this area is the derivation in §3.7 of general forms of multivariate Gaussian scale mixtures and their moments based on properties of Gaussian scale mixtures used by Dempster, Laird, and Rubin in [32], but which have not been exploited in the recent developments in independent subspace analysis, Topographic ICA [51], or Indpendent Vector Analysis [60].

1.3 Linear Process Mixture Model Examples

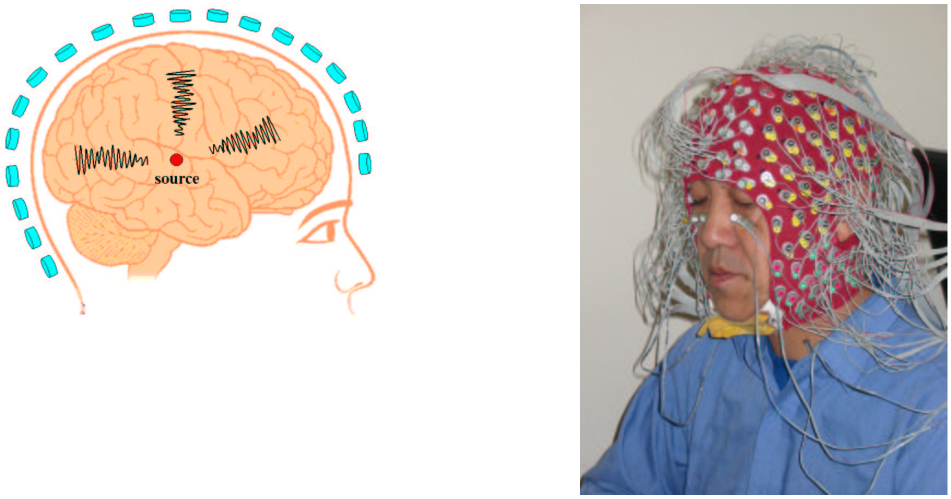
The linear process mixture model is very general and applicable in many signal processing areas. The difference between the model presented here and traditional linear process models in signal processing is that the source densities are presumed unknown, along with the filters, and the segmentation. All are adapted to maximize the independence of the estimated sources. In the following, we give some examples of real systems and possible linear process models. In each case, each observation is modeled locally as a linearly filtered i.i.d. driving field.

1.3.1 Electro-encephalogram

EEG is as an example of the "instantaneous" model,

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

wherein there is no convolution of the sources. Current sources in the brain and elsewhere in the body, e.g. heart and muscles, emit electromagnetic waves that travel at the speed of light to the sensors (see Figure 1.1). The sensor sampling rate cannot detect any delays so the mixing is instantaneous. The EEG source signals themselves however will not generally be i.i.d., though using the i.i.d. model is often sufficient for separation of temporally correlated sources.



(a) (b)

Figure 1.1: (a) EEG sources cast characteristic voltage distributions on the sensors. (b) A subject wearing an EEG sensor cap with 256 electrodes.

Voice and microphone

A speech signal $s(t)$ recorded by a single microphone, as an i.i.d. process $\eta(t)$ passed through a linear filter with impulse response $\theta(\tau)$, $\tau = 0, \ldots, \infty$,

$$s(t) = \sum_{\tau=0}^{\infty} \theta(\tau)\eta(t-\tau) \tag{1.7}$$

This is a causal moving average (MA) representation. Long enough filters can represent any causal stable linear process, including the commonly used autoregressive (AR) model,

$$s(t) = \sum_{\tau=1}^{p} \phi(\tau)s(t-\tau) + \eta(t) \tag{1.8}$$

Likewise, the room and microphone can be modeled by linear filters, as in Figure 1.2.

Blind deconvolution of audio sources

Suppose two speakers are talking in a room, and sensors, or microphones, record the sound as in Figure 1.3. If they speak at the same time, the speakers can

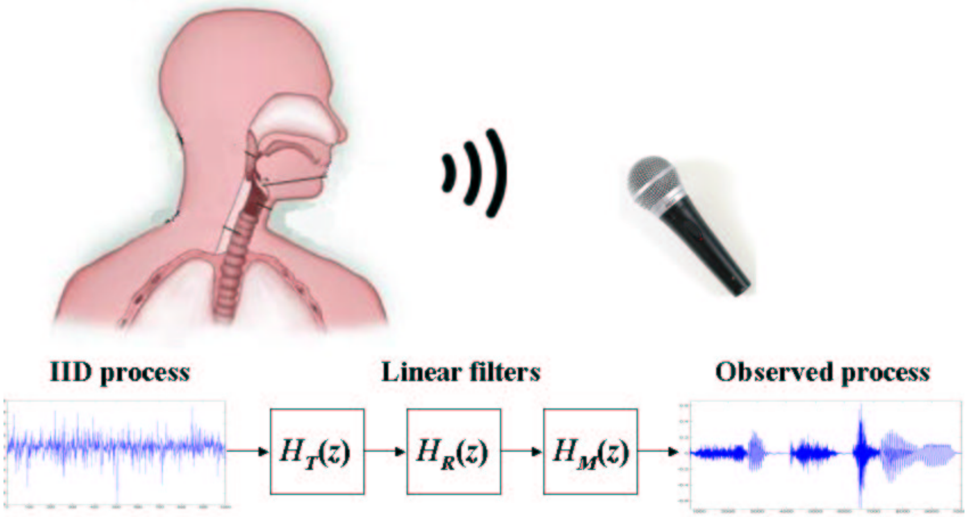


Figure 1.2: Speech as temporally independent driving signal passed through linear filter representing the parts of the larynx and throat, the room, and the microphone interface.

be distinguished by the independence of their speech signals. The observed vector, $\mathbf{x}(t)$, will be the amplitude distribution on the sensors. The problem of separating the speech is decoupled from the problem of locating the speaker given the amplitude distribution associated with the location. Orientation of the speaker's head will also change the distribution on the sensors. The blind signal processing problem can generally be decomposed into a separation problem, and an inversion, or localization problem. Recorded speech may also be convolved due to delays in arrival of echoes bouncing off different walls as in Figure 1.4

1.3.2 Speech signal as mixture of linear processes

A speech signal is an example of a non-stationary process that exhibits local stationarity. The unvoiced 's' sound at the start and end of the syllable "six" exhibit the same statistics. The voiced regions "i-i-i", "e-e-e", and "n-n-n" have a characteristic stationary structure. Such models are used in speech

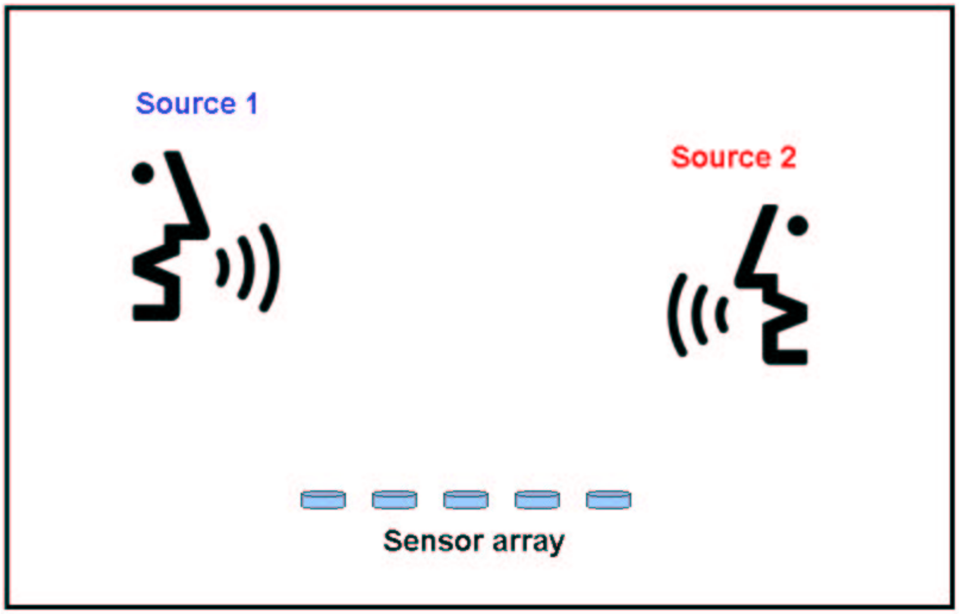


Figure 1.3: Two speakers cast different amplitude distributions on the sensors depending on where they are in relation to the sensors, and which way they are facing.

processing, but the phoneme building blocks are constructed by experts. Blind signal processing using a mixture of linear processes allows the automatic learning of locally stationary signal features without segmenting and sorting segments by hand.

1.3.3 Binocular Color Images

Binocular color images are an example of multichannel data, consisting of a two dimensional field of six dimensional vectors. The field can be blocked and modeled with learned image bases, or it can be modeled as a mixture of spatial filters, which represent the correlation structure of homogeneous regions in the image.

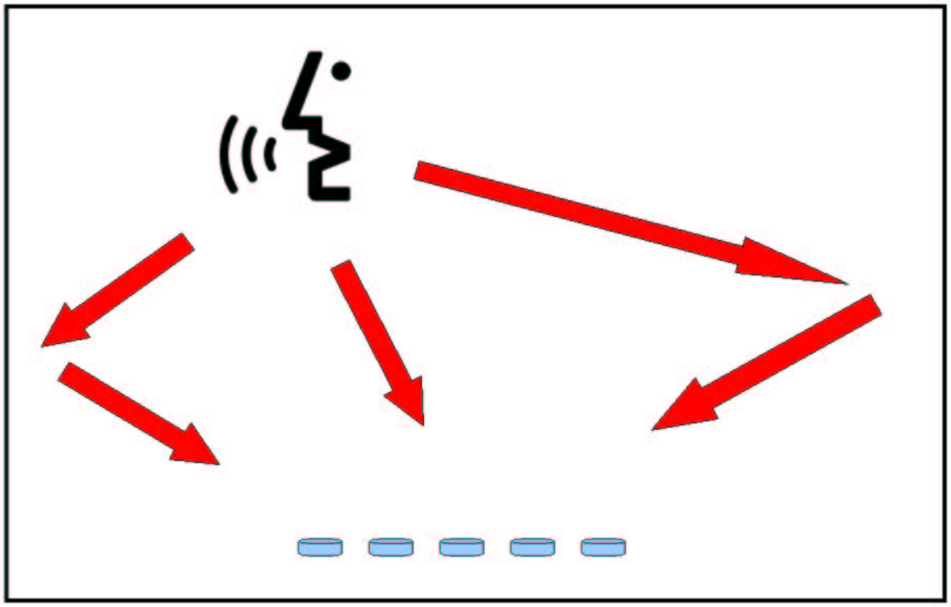


Figure 1.4: In a room, the speech signal travels slow enough that there are delays in the time of arrival of echoes bouncing off different walls. In this case, the source signal is convolved as it arrives at the sensors at different times with different amplitudes.

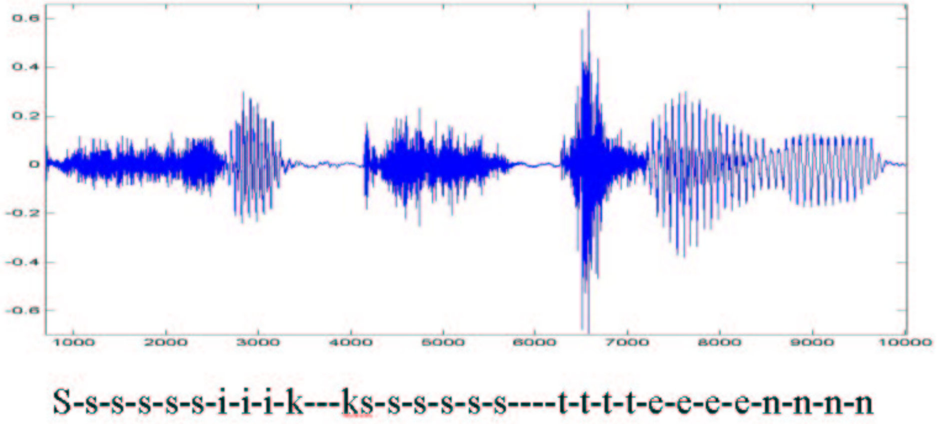


Figure 1.5: Speech signal at 16,000 Hz of a woman saying the word “sixteen”. The signal is non-stationary, but localized regions are homogeneous.

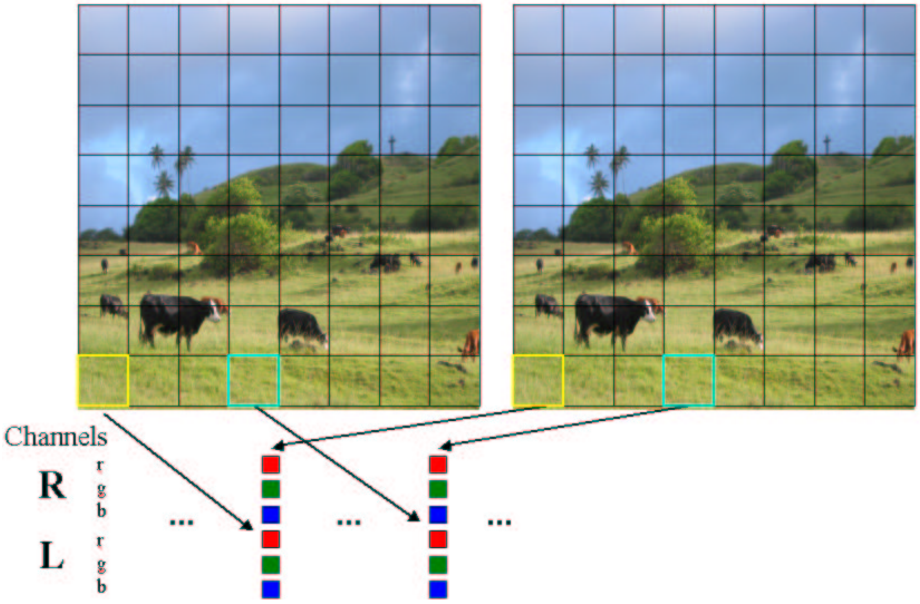


Figure 1.6: Binocular color images can be represented as a 2D field of 6D vectors, three color components, e.g. red, green, and blue, for right and left channels.

1.4 Inverse modeling

Our goal is to derive a general system and an unsupervised adaptive strategy that can be used to learn salient features in piecewise stationary environments. This system is intended to be useful for signal processing applications such as EEG, audio separation, image and video processing. It is also intended to be a model of neural processing systems for the purposes of investigations in neuroscience and adaptive learning systems. As such, the algorithm derived models the inverse system and avoids more complicated operations like matrix inverses. The observed signal passes through a set of filters that are adapted to make the output as independent as possible.

It is also possible to consider learning to reconstruct signals for the purpose of planning and simulation in artificial intelligence systems.

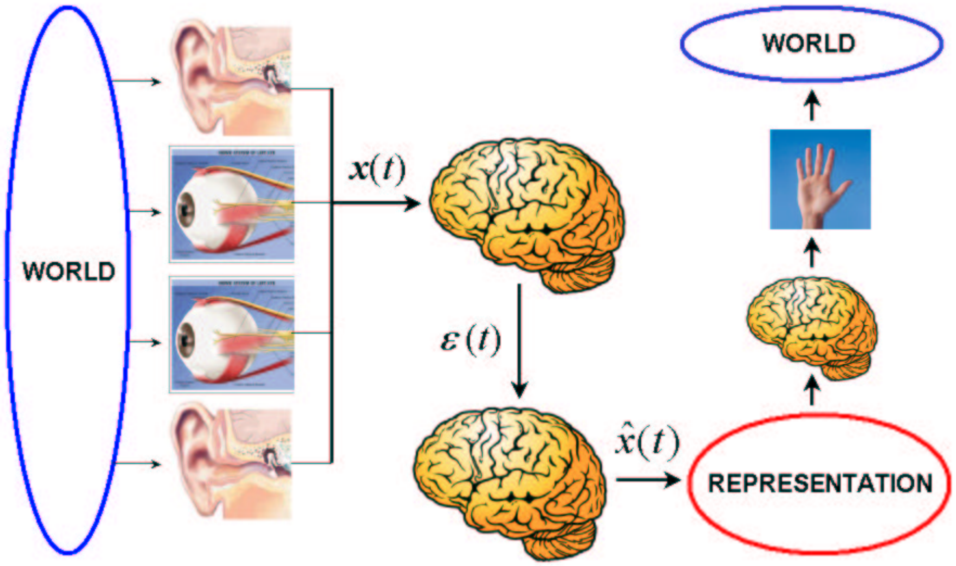


Figure 1.7: Sense organs are like sensors, channelling information to the brain, where it is integrated, and mapped to representations and actions.

1.5 Rate Distortion Theory

Processing signals into "independent components" can also be motivated on the basis of rate distortion theory using a theorem of R. Gray [46]. The rate of a code is the average number of bits used per symbol encoded. For discrete random variables, the minimum rate code is the minimum entropy code, and it can be determined by the Huffman encoding procedure. For continuous random variables, specifying values directly would take an infinite number of bits. Rate distortion theory [96] provides a framework for the analysis of encoding systems for continuous amplitude signals.

However, a difficulty arises now as to how to characterize the "distortion, or the "differences that make a difference," accurately. For example, sparse random variables, or "on/off", "active/inactive variables have high probability mass around zero, the "off" state. If we were assigning code vectors based only on probability mass, then we would spend the majority of our vectors coding the off state. Rate distortion theory addresses this by taking into account the metric structure of the observation space as well as the probability distribution.

When coding a sequence of data samples, the average rate of the code will decrease if we use *vector quantization* to encode blocks of samples by individual symbols, rather than encoding each sample separately. A code is actually a set of codebooks, or mappings of observations to code vectors, for each block length N, with N being arbitrarily large. A code is said to achieve the rate distortion pair (R, D) if, for all $\epsilon > 0$, there is a block length N such that the expected distortion between an observation sample block and its code vector in the N-block codebook, is less than $D + \epsilon$. The rate distortion function $R(D)$ is the infimum of the rates R such that the pair (R, D) is achievable.

The following theorem [12, Thm. 6.3.6] provides a basic result concerning the rate distortion function of linear processes with difference distortion measures, i.e. distortion measures of the form $d(x - y)$.

Theorem 2. *Let $\{X_t,\, t = 1, \ldots\}$, be a real autoregressive source with autoregression coefficients $(a_1, \ldots, a_m)$, zero initial state, and i.i.d. generating sequence $\{Z_t,\, t = 1, 2, \ldots\}$. Let d be a difference distortion measure, and let $R_X(D)$ and $R_Z(D)$ denote the rate distortion functions of $\{X_t\}$ and $\{Z_t\}$, respectively, relative to the single-letter fidelity criterion $N^{-1} \sum_{t=1}^{N} d(X_t - \hat{X}_t)$. Then,*

$$R_X(D) \geq R_Z(D)$$

for all D.

This theorem shows that when the source is an autoregressively filtered i.i.d. sequence, then the best linear transform of the observed sequence $\{X_t\}$ from the rate distortion theory standpoint, is the inverse of the autoregressive filter which reproduces the generating sequence $\{Z_t\}$ when the Z_t are non-Gaussian.

1.6 Summary of main contributions of this thesis

The main contribution of this thesis is the analysis of the structure of certain classes of non-Gaussian signals, and the use of this analysis to derive new algorithms for optimization of linear process mixture models non-Gaussian source signals. Our analysis of non-Gaussian signals has led to the development of a very flexible non-Gaussian mixture model that can be optimized in largely the same way as the Gaussian model, but with improved flexibility due to the inclusion of non-Gaussian components.

Chapter 2 develops the parameter estimation methods used in this thesis, including Maximum Likelihood (ML), Maximum á posteriori (MAP), and Ensemble Learning, or Variational Bayes (VB). These methods are illustrated in the derivation of the basic mixture model equations for the various methods, which are used in the main linear process mixture model algorithm.

Chapter 3 analyzes the Gaussian Scale Mixture and variational concave representation of super-Gaussian Densities. We derive the criteria for these representations and illuminate their relationship, which seems not to have been noted previously. We also prove in §3.6 that the variational concave super-Gaussian representation implies positive kurtosis. We use results of Karlin [58] in the proof. In §3.7 we derive general forms of multivariate Gaussian scale mixture densities and their moments in terms of derivatives of given univariate Gaussian scale mixture.

In Chapter 4 we investigate sparse coding algorithms. §4.1 describes the representations and develops their relationships. In §4.2, we prove a theorem on the convergence of Generalized FOCUSS algorithm, and its convergence rates. We discuss the properties of dual programs involving function convex and concave in x^2, and give a Newton method to solve the dual problem when all sources are strongly super-Gaussian. In §4.4 we show how these algorithms for given dictionaries may be applied to derive monotonic algorithms for kernel regression.

Chapter 5 surveys dictionary learning algorithms, and proposes a Generalized FOCUSS type algorithm based on a partial Newton method to find a Lagrangian stationary point. We compare performance of the various algorithms in a Monte Carlo experiment.

In Chapter 6, we apply the representations to propose a unifying framework for Blind Source Separation (BSS) and Independent Component Analysis (ICA). §6.1 develops the basic ML approach to ICA, and §6.2 derives the Hessian, or Fisher Information matrix, as in [87, 2], which is used to derive the convexity conditions, the Cramér-Rao lower bound for ML estimation, as well as a Newton type method [87, 2] under the conditions of convexity. §6.3 applies the density representation theory to derive an EM based algorithm for ICA which is stable for a wider class of source densities than those that are stable for the direct optimization of the cost function.

Chapter 7 develops the linear process mixture model. §7.1 describes the convolutive ICA model and, §7.2 derives the approximate probability density of signal segments. In §7.3 we derive the gradient and natural gradient of the segment likelihood. §7.4 extends the model to mixtures of linear processes.

In Chapter 8, we provide and in depth application of the methods to analysis of EEG signals.

2

Parameter Estimation

Our approach is to define a probabilistic model and optimize the parameters within an Expectation Maximization (EM) framework. The model is defined by a probability density with parameters θ, over sets of independent and identically distributed (i.i.d.) observations $\{\mathbf{X}_1, \ldots, \mathbf{X}_t, \ldots, \mathbf{X}_T\}$,

$$p(\mathbf{X}_1, \ldots, \mathbf{X}_T; \theta) \;=\; \prod_{k=1}^{T} p(\mathbf{X}_t; \theta) \tag{2.1}$$

For the estimation of time series or random fields, each $\mathbf{X}_t$ will be a segment or a block, over which the time series is supposed to be stationary. Non-stationarity of the complete time series is modeled by taking $p(\mathbf{X}_t; \theta)$ to be a mixture model,

$$p(\mathbf{X}_t; \theta) = \sum_{h=1}^{M} \gamma_h \, p(\mathbf{X}_t; \theta_h)$$

2.1 Estimation methods

We consider three types of estimation: Maximum Likelihood, Maximum à Posteriori (MAP), and Ensemble Learning (also known as Variational Bayes.)

2.1.1 Maximum Likelihood and Kullback-Leibler Divergence

In the Maximum Likelihood (ML) approach, we simply maximize (2.1), called the "likelihood" of the data, with respect to θ, or equivalently, since p is non-negative, and the $\mathbf{X}_t$ are i.i.d.,

$$\begin{aligned}\hat{\theta}_{ML} &= \arg\max_{\theta}\ \log p(\mathbf{X}_1,\ldots,\mathbf{X}_T;\theta) \\ &= \arg\min_{\theta}\ -\frac{1}{T}\sum_{t=1}^{T}\log p(\mathbf{X}_t;\theta) \qquad (2.2)\end{aligned}$$

The model we use will not be identical to the true generating distribution of $\mathbf{X}$, say p^*. To see the relationship between optimizing the model density and the true density, we can use the Law of Large Numbers, by which the expression in (2.2) becomes,

$$\lim_{T\to\infty}\ -\frac{1}{T}\sum_{t=1}^{T}\ \log p(\mathbf{X}_t;\theta)\ =\ -E\log p(\mathbf{X})\ =\ H(\mathbf{X}) + D\big(p(\mathbf{X})||\,p^*(\mathbf{X})\big)$$

where $H(\mathbf{X}) = -\int p^*(\mathbf{X})\log p^*(\mathbf{X})\,d\mathbf{X}$ is the entropy of the true distribution p^*, and $D(p||p^*)$ is the Kullback-Leibler divergence (KL divergence) between p and p^*. Since $H(\mathbf{X})$ is constant, we have,

$$\hat{\theta}_{ML}\ =\ \arg\min_{\theta}\ D\big(p(\mathbf{X};\theta)||\,p^*(\mathbf{X})\big)$$

The KL divergence is non-negative and zero if and only if $p = p^*$ almost surely, so it acts like a distance measure, although it is non-symmetric.

When $\mathbf{X}$ is a discrete random variable, the KL divergence of p from p^*, is a measure of how many extra bits one must use when encoding $\mathbf{X}$ using p rather than the optimal p^* [29]. For continuous random variables, this intuitive picture breaks down and the situation becomes more complicated, but it gives a rough idea of what minimizing the KL divergence does.

2.1.2 MAP Estimation

In the MAP approach, the parameters θ are considered random variables with prior distribution $p(\theta)$, and we attempt to maximize $p(\theta\,|\,\mathbf{X}_1,\ldots,\mathbf{X}_T)$,

$$\begin{aligned}\hat{\theta}_{MAP} &= \arg\max_{\theta}\ \log p(\theta\,|\,\mathbf{X}_1,\ldots,\mathbf{X}_T) \qquad (2.3) \\ &= \arg\max_{\theta}\ \log p(\mathbf{X}_1,\ldots,\mathbf{X}_T\,|\,\theta)\ +\ \log p(\theta) \qquad (2.4)\end{aligned}$$

where we use Bayes' rule: $p(a|b) = p(b|a)p(a)/p(b)$. So we see that MAP estimation is equivalent to ML estimation, with the addition of the prior term. The prior term can help to "regularize" the estimation, preventing θ from taking arbitrary values when the likelihood does not well determine θ, for example when there are few observations.

2.1.3 The EM Algorithm

The EM algorithm is a method of solving a difficult ML or MAP problem by replacing it with a sequence of easy problems, leading, it is hoped, to the solution. The algorithm does not guarantee that the global optimum is found, nor in general even that a local optimum is found. It does guarantee, however, that the objective function, i.e. the likelihood of the data, is increased at each iteration. This is accomplished by introducing auxiliary random variables.

Variational Free Energy

The EM algorithm can be derived from the a mean field perspective as follows. We introduce the random variables $\mathbf{z}$, and write the log likelihood as follows,

$$\begin{aligned} \log p(\mathbf{x};\theta) &= \int q(\mathbf{z}|\mathbf{x}) \log \frac{p(\mathbf{z},\mathbf{x};\theta)}{q(\mathbf{z}|\mathbf{x})}\, d\mathbf{z} \,+\, D\big(q(\mathbf{z}|\mathbf{x}) \big\|\, p(\mathbf{z}|\mathbf{x};\theta)\big) \\ &= -F(q;\theta) \,+\, D(q\|p_\theta) \qquad (2.5) \end{aligned}$$

where the integration is performed over the support[1] of $p(\mathbf{z}|\mathbf{x})$, and $q(\mathbf{z}|\mathbf{x})$ is an arbitrary density over the auxiliary random variables $\mathbf{z}$ having the same support as $p(\mathbf{z}|\mathbf{x})$. The term $F(q;\theta)$ is commonly referred to as the *variational free energy* [95, 78]. This representation is useful if $F(q;\theta)$ can be easily optimized with respect to θ, whereas $\log p(\mathbf{x};\theta)$ cannot.

Since the KL divergence is non-negative, and equal to 0 if $q = p_\theta$, and

[1] The support is the set of non-zero measure.

the left hand side of (2.5) is constant with respect to the density q, it follows that,

$$-\log p(\mathbf{x};\theta) = \min_q F(q;\theta)$$

where equality is obtained if and only if $q(\mathbf{z}|\mathbf{x}) = p(\mathbf{z}|\mathbf{x};\theta)$ almost everywhere[2].

The EM algorithm, at the lth iteration, given θ^l, proceeds as follows,

$$q^l = p\big(\mathbf{z}|\mathbf{x};\theta^l\big)\ , \quad \theta^{l+1} = \arg\min_\theta F\big(q^l;\theta\big) \tag{2.6}$$

This algorithm is guaranteed to increase the likelihood since,

$$-\log p(\mathbf{x};\theta^{l+1}) = F(q^{l+1};\theta^{l+1}) \leq F(q^l;\theta^{l+1}) \leq F(q^l;\theta^l) = -\log p(\mathbf{x};\theta^l)$$

Note that it is not necessary to find the actual minimum of F with respect to θ in order to guarantee that the likelihood increases. It is enough to guarantee that $F(q^l;\theta^{l+1}) \leq F(q^l;\theta^l)$, i.e. that F decreases as a result of updating θ. This leads to the Generalized EM (GEM) algorithm [31].

If the optimization of F is also a difficult problem, it may still be possible to guarantee a decrease in $F(q;\theta)$ with respect to θ. In this thesis we shall use a convexity-based inequality to define a function $\tilde{F}(q;\theta)$ that is easy to minimize with respect to θ, and which satisfies, for all θ, θ',

$$F(q;\theta') - F(q;\theta) \ \leq \ \tilde{F}(q;\theta') - \tilde{F}(q;\theta)$$

Setting θ^{l+1} to minimize $\tilde{F}(q^l;\theta)$ over θ then guarantees that,

$$F(q^l;\theta^{l+1}) - F(q^l;\theta^l) \ \leq \ \tilde{F}(q^l;\theta^{l+1}) - \tilde{F}(q^l;\theta^l) \ \leq \ 0$$

and thus that $F(q^l;\theta)$ is decreased as required by the GEM algorithm.

The EM algorithm can be used for MAP estimation as well [31], since

$$\arg\max_\theta p(\theta|\mathbf{x}) = \arg\max_\theta p(\mathbf{x}|\theta)p(\theta)$$

We can decompose $p(\mathbf{x}|\theta)$ as before, so that we have,

$$\begin{aligned}\log p(\mathbf{x}|\theta)p(\theta) \ &= \ \int q(\mathbf{z}|\mathbf{x}) \log \frac{p(\mathbf{z},\mathbf{x}|\theta)}{q(\mathbf{z}|\mathbf{x})}\, d\mathbf{z} \ + \ \log p(\theta) + D\big(q(\mathbf{z}|\mathbf{x}) \big\| \, p(\mathbf{z}|\mathbf{x},\theta)\big) \\ &= \ -F(q,\theta) \ + \ D(q\|p_\theta)\end{aligned}$$

[2] I.e., except on a set of measure zero.

where the free energy now contains an additional term depending on the prior density $p(\theta)$. The EM algorithm for MAP estimation of θ is then,

$$q^l = p\big(\mathbf{z}|\mathbf{x}, \theta^l\big)\,, \quad \theta^{l+1} = \arg\min_{\theta} F\big(q^l, \theta\big) \tag{2.7}$$

In general, the GEM algorithm is only guaranteed to converge to a stationary point of the likelihood with respect to the parameters θ. More assumptions are needed to guarantee that the algorithm converges to a local maximum, and further assumptions to guarantee that the sequence θ^l converges to a point θ^*.

Q and H functions

The EM/GEM algorithm was originally formulated in terms of the functions,

$$Q(\theta|\theta') = \int p(\mathbf{z}|\mathbf{x};\theta') \log p(\mathbf{z},\mathbf{x};\theta) d\mathbf{z} \tag{2.8}$$

and,

$$H(\theta|\theta') = \int p(\mathbf{z}|\mathbf{x};\theta') \log p(\mathbf{z}|\mathbf{x};\theta) d\mathbf{z}$$

The function $\log p(\mathbf{z},\mathbf{x};\theta)$ is called the complete log likelihood, since it is the likelihood of the "complete" data $(\mathbf{x},\mathbf{z})$ which includes the auxiliary random variables $\mathbf{z}$. The variables $\mathbf{z}$ are called the "hidden" data, and we iteratively integrate out the hidden data using (2.8) with the parameters θ', optimize (2.8) over the parameters θ to get the new θ', and repeat.

Denote the log likelihood by,

$$L(\theta) = \log p(\mathbf{X}_1, \ldots, \mathbf{X}_T; \theta)$$

The EM, or GEM, iterations are denoted $\theta^{l+1} \in \mathbf{M}(\theta^l)$, where $\mathbf{M}(\theta)$ is a point-to-set mapping such that $L(\theta^{l+1}) \geq L(\theta^l)$. Let $\mathcal{S}$ be the set of stationary points, and $\mathcal{M}$ the set of local maxima in the interior of the domain of θ. Then we have the following theorem [103].

Theorem 3. *Let $\{\theta^l\}$ be a GEM sequence generated by $\theta^{l+1} \in \mathbf{M}(\theta^l)$, and suppose that (i) $\mathbf{M}$ is closed over the complement of $\mathcal{S}$ (respectively $\mathcal{M}$,) (ii) $L(\theta^{l+1}) > L(\theta)$.*

Then all limit points of $\{\theta^l\}$ are stationary points (resp. local maxima) of L, and $L(\theta^l)$ converges monotonically to $L^* = L(\theta^*)$ for some theta$^* \in \mathcal{S}$ (resp. $\mathcal{M}$.)

A sufficient condition for $\mathbf{M}(\theta)$ to be a closed mapping in the case of an EM algorithm is that $Q(\theta'|\theta)$ be continuous in θ' and θ.

Theorem 4. *Suppose $Q(\theta'|\theta)$ is continuous in θ and θ'. Then all limit points of an EM sequence $\{\theta^l\}$ are stationary points of L, and $L(\theta^l)$ converges monotonically to $L(\theta^*)$ for some point θ^*.*

2.1.4 Ensemble learning and Variational Bayes

In the ensemble learning approach (also Variational Bayes [8, 4, 13]), rather than finding a point estimate of θ, we attempt to find the separable, or factorial, posterior density that minimizes the KL divergence from the true posterior,

$$\hat{q}_{VB}(\theta_1, \ldots, \theta_n|\mathbf{x}) = \arg \min_{q_1,\ldots,q_n} D\big(q_1(\theta_1|\mathbf{x}) \cdots q_n(\theta_n|\mathbf{x}) \,||\, p(\theta_1, \ldots, \theta_n|\mathbf{x})\big)$$

For simplicity of exposition, suppose θ contains only two (random) parameters, ϕ and ξ. Again we use the following decomposition of the log likelihood,

$$\begin{aligned} \log p(\mathbf{x}) &= \int q(\phi, \xi|\mathbf{x}) \log \frac{p(\phi, \xi, \mathbf{x})}{q(\phi, \xi|\mathbf{x})}\, d\mathbf{x} + D\big(q(\phi, \xi|\mathbf{x})\big|\big|\, p(\phi, \xi|\mathbf{x})\big) \\ &= -F(q) \,+\, D(q||p) \end{aligned}$$

The approximating posterior distribution is factorial,

$$q(\phi, \xi|\mathbf{x}) = q(\phi|\mathbf{x})\, q(\xi|\mathbf{x})$$

The VB algorithm consists of alternately updating each approximating marginal distribution, keeping the other approximating marginals fixed. For fixed $q(\xi|\mathbf{x})$, the free energy F is given by,

$$-\iint q(\phi|\mathbf{x})q(\xi|\mathbf{x})\log\frac{p(\phi,\xi,\mathbf{x})}{q(\phi|\mathbf{x})q(\xi|\mathbf{x})}\,d\xi\,d\phi \;=\; D\big(q(\phi|\mathbf{x})\,||\,e^{\langle\log p(\phi,\xi,\mathbf{x})\rangle_\xi}\big) + \text{const.}$$

where $\langle\cdot\rangle_\xi$ denotes expectation with respect to $q(\xi|\mathbf{x})$, and the constant is the entropy, $H\big(q(\xi|\mathbf{x})\big)$. The minimum of the KL divergence, and thus of F, is attained if and only if

$$q(\phi|\mathbf{x}) \;\propto\; \exp\big\langle\log p(\phi,\xi,\mathbf{x})\big\rangle_\xi$$

almost everywhere. An identical derivation yields the optimal $q(\xi|\mathbf{x})$,

$$q(\xi|\mathbf{x}) \;\propto\; \exp\big\langle\log p(\phi,\xi,\mathbf{x})\big\rangle_\phi$$

when $q(\phi|\mathbf{x})$ is fixed. The generalization to more than two parameters is obvious.

2.2 Mixture Model Estimation

As an example of the use of the three estimation methods, ML, MAP, and VB, consider the estimation of the mixing proportions α in a mixture model,

$$p(\mathbf{x}\,|\,\alpha_1,\ldots,\alpha_m) = \sum_{j=1}^{m}\alpha_j\,p(\mathbf{x};\theta_j)$$

where $\alpha_j \geq 0$, $\sum_j \alpha_j = 1$.

Suppose we are given data, $\mathbf{x}_1,\ldots,\mathbf{x}_T$. To use the EM algorithm, we define the discrete random vectors $\mathbf{z}_t$, $t = 1,\ldots,T$, ranging over the set $\{\mathbf{e}_1,\ldots,\mathbf{e}_m\}$, where $\mathbf{e}_j$ is the vector with 1 in the jth component and zeros elsewhere, such that,

$$\text{Prob}(\mathbf{z}_t = \mathbf{e}_j) = \alpha_j$$

Then we can write,

$$p\big(\mathbf{x}_t,\mathbf{z}_t\big|\alpha) \;=\; \prod_{j=1}^{m}\alpha_j^{z_{jt}}p(\mathbf{x}_t;\theta_j)^{z_{jt}}$$

on the support of $(\mathbf{x}_t,\mathbf{z}_t)$.

For ML estimation, given the current estimate α^l, we find $p(\mathbf{z}_t|\mathbf{x}_t;\alpha^l)$,

$$\begin{aligned}\mathrm{Prob}(\mathbf{z}_t=\mathbf{e}_j|\mathbf{x}_t;\alpha^l) &= \frac{p(\mathbf{x}_t|\mathbf{z}_t=\mathbf{e}_j;\alpha^l)\,\mathrm{Prob}(\mathbf{z}_t=\mathbf{e}_j;\alpha^l)}{p(\mathbf{x}_t;\alpha^l)}\\ &= \frac{\alpha_j^l\,p(\mathbf{x}_t;\theta_j)}{\sum_{j'=1}^m \alpha_{j'}^l\,p(\mathbf{x}_t;\theta_{j'})}\end{aligned}$$

We then minimize,

$$\begin{aligned}&-\prod_{t=1}^T \int p(\mathbf{z}_t|\mathbf{x}_t;\alpha^l)\log\left(\prod_{\tau=1}^T p(\mathbf{x}_\tau,\mathbf{z}_\tau;\alpha)\right)d\mathbf{z}_t\\ &\qquad = -\sum_{t=1}^T\sum_{j=1}^m \hat{z}_{jt}^l\log\alpha_j+\hat{z}_{jt}^l\log p(\mathbf{x}_t;\theta_j) \qquad (2.9)\end{aligned}$$

over α on the positive simplex, where $\hat{z}_{jt}^l$ is given by,

$$\hat{z}_{jt}^l = \int z_{jt}\,p(\mathbf{z}_t|\mathbf{x}_t;\alpha^l)\,d\mathbf{z}_t = \frac{\alpha_j^l\,p(\mathbf{x}_t;\theta_j)}{\sum_{j'=1}^m \alpha_{j'}^l\,p(\mathbf{x}_t;\theta_{j'})} \qquad (2.10)$$

Using Lagrange multipliers, the minimum is readily found to be,

$$\text{ML:} \qquad \alpha_j^{l+1} = \frac{1}{T}\sum_{t=1}^T \hat{z}_{jt}^l \qquad (2.11)$$

For MAP and VB we need to specify a prior distribution over α. The distribution commonly used for this purpose is the Dirichlet distribution, defined over the simplex $\alpha_j \ge 0$, $\sum_j \alpha_j = 1$, by,

$$\mathcal{D}\big(\alpha_1,\ldots,\alpha_m;\bar{\alpha},\bar{N}\big) = \frac{\Gamma(\bar{N})}{\prod_j\Gamma\big(\bar{\alpha}_j\bar{N}\big)}\prod_{j=1}^m \alpha_j^{\bar{\alpha}_j\bar{N}-1}$$

where $\Gamma(\,\cdot\,)$ is the Gamma function. The Dirichlet distribution has mean $\bar{\alpha}$, and "concentration" parameter $\bar{N}>0$. With two parameters α_1 and $\alpha_2 = 1-\alpha_1$, the Dirichlet is equivalent to the Beta distribution. We define the symmetric Dirichlet distribution by taking $\bar{\alpha}_j = 1/m$, $j=1,\ldots,m$. The distribution then takes the form,

$$\mathcal{SD}\big(\alpha_1,\ldots,\alpha_m;\tilde{N}\big) = \frac{\Gamma(\bar{N})}{\prod_j\Gamma\big(\bar{N}/m\big)}\prod_{j=1}^m \alpha_j^{\bar{N}/m-1} = \frac{\Gamma(m\tilde{N}+1)}{\prod_j\Gamma\big(\tilde{N}+1\big)}\prod_{j=1}^m \alpha_j^{\tilde{N}}$$

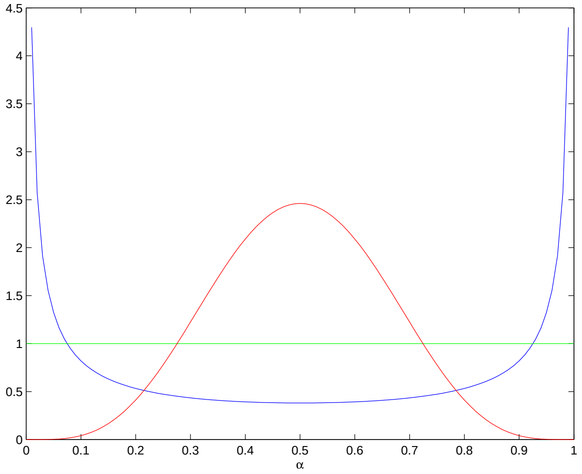


Figure 2.1: The Dirichlet distribution for $m = 2$, $\bar{\alpha}_1 = \bar{\alpha}_2 = 1/2$, and $\bar{N} = 1$, 2, and 10, or $\tilde{N} = -1/2$, 0, and 4.

where we define $\tilde{N} = \bar{N}/m - 1 > -1$.

The MAP estimate is found by maximizing,

$$p\left(\{\mathbf{x}_t\}_{t=1}^{T} | \alpha\right) p(\alpha; \bar{\alpha}, \bar{N})$$

The only difference in the EM algorithm for MAP estimation of α is in the free energy minimization step, wherein there are now additional terms due to the prior. The expression to be minimized over α in the positive simplex, similar to (2.9), is,

$$-\sum_{t=1}^{T}\left(\sum_{j=1}^{m} \hat{z}_{jt}^{l} \log \alpha_j + \hat{z}_{jt}^{l} \log p(\mathbf{x}_t; \theta_j)\right) - \sum_{j=1}^{m} (\bar{\alpha}_j \bar{N} - 1) \log \alpha_j$$

where $\hat{z}_{jt}^{l}$ is again given by (2.10). For the optimum, we find,

$$\text{MAP:} \qquad \alpha_j^{l+1} = \frac{\sum_{t=1}^{T} \hat{z}_{jt}^{l} + \bar{\alpha}_j \bar{N} - 1}{T + \bar{N} - m} \tag{2.12}$$

or, for the symmetric Dirichlet distribution,

$$\text{MAP:} \qquad \alpha_j^{l+1} = \frac{\sum_{t=1}^{T} \hat{z}_{jt}^{l} + \tilde{N}}{T + m\tilde{N}} \tag{2.13}$$

Finally, for the VB method, we find the factorial posterior over $\mathbf{z}$ and α that minimizes the KL divergence from the true posterior. This involves taking expectations of,

$$\log p\big(\{\mathbf{x}_t, \mathbf{z}_t\}_{t=1}^T, \alpha\big) = \sum_{t=1}^T \left(\sum_{j=1}^m z_{jt} \log \alpha_j + z_{jt} \log p(\mathbf{x}_t; \theta_j) \right) + \sum_{j=1}^m (\bar{\alpha}_j \bar{N} - 1) \log \alpha_j \tag{2.14}$$

Suppose we are given $q^l\big(\{\mathbf{z}_t\}_{t=1}^T | \{\mathbf{x}_t\}_{t=1}^T\big)$, producing the expectations $\hat{z}_{jt}^l$, for $j = 1, \ldots, m$, and $t = 1, \ldots, T$. Then according to the VB update, we exponentiate the expected value of (2.14). This produces another Dirichlet distribution for the approximating posterior,

$$\text{VB:} \qquad q^{l+1}\big(\alpha \,|\, \{\mathbf{x}_t\}_{t=1}^T\big) \;\sim\; \mathcal{D}\big(\alpha; \bar{\alpha}^l, \bar{N}^l\big)$$

where,

$$\bar{\alpha}_j^{l+1} = \frac{\sum_t \hat{z}_{jt}^l + \bar{\alpha}_j \bar{N}}{T + \bar{N}} \;, \qquad \bar{N}^{l+1} = T + \bar{N}$$

Now, we fix $q^{l+1}\big(\alpha \,|\, \{\mathbf{x}_t\}_{t=1}^T\big)$ and find $q^{l+1}\big(\{\mathbf{z}_t\}_{t=1}^T \,|\, \{\mathbf{x}_t\}_{t=1}^T\big)$. For this, we need the fact that, for the Dirichlet distribution, we have,

$$E \log \alpha_j \;=\; \Psi(\bar{\alpha}_j \bar{N}) - \Psi(\bar{N})$$

where Ψ is the digamma function[3]. If we define,

$$\psi_j^{l+1} \;\triangleq\; \exp\big(\Psi(\bar{\alpha}_j^{l+1} \bar{N}^l) - \Psi(\bar{N}^{l+1})\big)$$

then the posterior over $\{\mathbf{z}_t\}_{t=1}^T$ is determined by the expectations,

$$\text{VB:} \qquad \hat{z}_{jt}^{l+1} \;=\; \frac{\psi_j^{l+1}\, p(\mathbf{x}_t; \theta_j)}{\sum_{j'=1}^m \psi_{j'}^{l+1}\, p(\mathbf{x}_t; \theta_{j'})} \tag{2.15}$$

2.3 Bayesian Linear Model – Gaussian case

Let $\mathbf{A}$ be an $m \times n$ real matrix. Consider the linear model,

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \nu$$

where $\nu \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_\nu)$, and $\mathbf{s} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Lambda})$ with $\mathbf{\Lambda} = \text{diag}(\xi)^{-1}$ is diagonal with diagonal components ξ_i^{-1}, $i = 1, \ldots, n$.

[3] $\Psi(x) = (\partial/\partial x)\, \Gamma(x)$.

The density of $\mathbf{x}$ is Gaussian,

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{0}, \mathbf{A}\mathbf{\Lambda}\mathbf{A}^T + \mathbf{\Sigma}_\nu)$$

and the posterior density of $\mathbf{s}$ given $\mathbf{x}$ is also Gaussian,

$$p(\mathbf{s}|\mathbf{x}) = \mathcal{N}(\mu_\mathbf{s}, \mathbf{\Sigma}_\mathbf{s})$$

where,

$$\mu_\mathbf{s} = \mathbf{\Lambda}\mathbf{A}^T(\mathbf{A}\mathbf{\Lambda}\mathbf{A}^T + \mathbf{\Sigma}_\nu)^{-1}\mathbf{y}$$

and,

$$\mathbf{\Sigma}_\mathbf{s} = \mathbf{\Lambda} - \mathbf{\Lambda}\mathbf{A}^T(\mathbf{A}\mathbf{\Lambda}\mathbf{A}^T + \mathbf{\Sigma}_\nu)^{-1}\mathbf{A}\mathbf{\Lambda}$$

Also, given data $\mathbf{x}_k$ and sources $\mathbf{s}_k$, $k = 1, \ldots, N$, the Maximum Likelihood estimate of $\mathbf{A}$ is,

$$\hat{\mathbf{A}}_{\text{ML}} = \sum_{k=1}^{N} \mathbf{x}_k \mathbf{s}_k^T \left(\sum_{k=1}^{N} \mathbf{s}_k \mathbf{s}_k^T \right)^{-1}$$

3

Density Representations

In the linear process and piecewise linear process models, a field of vector observations is modeled locally as linear convolutive mixture of a discrete i.i.d. generating vector field. The scalar components of the vectors in the generating field will generally be taken to be independent as well, but not identically distributed. In adapting or optimizing the model, we adapt the component densities as well as the linear filters. In this chapter we consider theory and methods of representing densities the component densities. We shall also consider generalizations to vectors with dependent components based on the univariate theory.

First we shall limit consideration to symmetric unimodal densities. Subsequently we show how the theory can be applied to mixtures of such densities.

3.1 Super-Gaussianity and Sub-Gaussianity

The Gaussian density is fundamental in probability and statistics for many reasons.

$$\mathcal{N}\big(x\,;\mu,\sigma^2\big) \;=\; \tfrac{1}{\sqrt{2\pi}\sigma}\, e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Linear functions of Gaussians are Gaussian, and the Central Limit Theorem states that limits of sums of random variables with finite variance are Gaussian distributed. In fact, the Gaussian density is an extreme point in the set of α-Stable

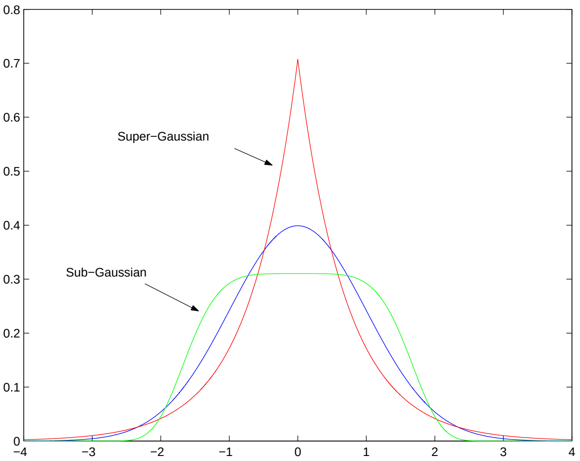


Figure 3.1: With unit variance, super-Gaussian densities have a sharper peak and heavier tails, and sub-Gaussian densities have a flatter peak and shorter tails.

densities, which are the distributions of limits of sums of random variables with finite moments of order α for $0 < \alpha \leq 2$, and which also have the property of being closed under linear operations [94]. The Gaussian density is also the density corresponding to quadratic optimization and least squares, which are important in practice due to their tractability.

Given the centrality of the Gaussian density it is natural to categorize non-Gaussian densities by their properties relative to the Gaussian density. One such categorization that has been found to be useful is that of sub-Gaussian and super-Gaussian. These notions are commonly used without a formal definition, but rather to refer to the qualitative characteristics of "peakedness" and "heaviness" of the tail of the density, relative to Gaussian. Informally, super-Gaussian densities are those with a sharper peak and heavier tails than Gaussian, and sub-Gaussian densities are flatter at the mode, with shorter tails, or faster decay, than Gaussian.

One quantitative measure that has been used to characterize sub- and super-Gaussianity if kurtosis. The kurtosis of a zero mean random variable X with finite fourth moment can be defined as the difference between the fourth moment of X and the fourth moment of a Gaussian random variable of equal variance, or $E(X^4) - 3E(X^2)^2$. If a density has positive kurtosis, then it is likely to be more

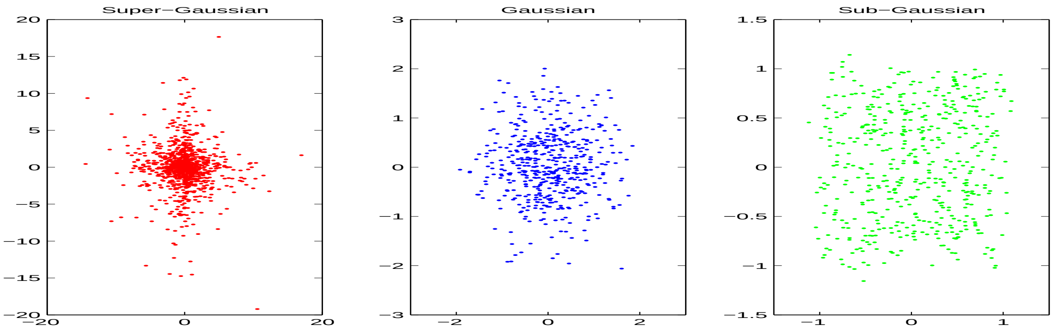


Figure 3.2: Super-Gaussian densities tend to be closer to zero, with occasional large magnitudes, while sub-Gaussian densities tend to be more uniform.

peaked about the mean, and have heavier tails than the Gaussian density.

In [76] the heaviness of tail criterion is addressed specifically, and the concept of "over-gaussianity" is defined as a density's having a tail that is asymptotically heavier than than the Gaussian tail, with sub-gaussianity defined similarly. A theorem is given that for a unimodal density having two points of intersection with the normalized Gaussian density, the density is over-gaussian if and only if the density has positive kurtosis. A similar theorem is given in Finucan [38] which assumes four density crossings (both sharper peak and heavier tails) rather than two crossings as in [76] (heavier tails only).

A random variable with a density that is more peaked with heavier tails than Gaussian also has the property of being "sparse", meaning that it is close to zero most of the time, but occasionally takes relatively large values. This is in contrast to the more uniform character of sub-Gaussian densities. These properties can be seen in scatter plots as shown in Figure 3.2.

3.2 Variational representations of super-Gaussian densities

Given the eminent tractabilility of the Gaussian density, it is natural to consider densities and random variables that are related to Gaussian in hopes of exploiting some of its nice characteristics. Two such representations are described in this section: Gaussian scale mixture representations, and convexity-based extremum representations which we shall call *strong* super-Gaussian.

A Gaussian scale mixture density is represented as an integral over the scale parameter of the density,

$$p(x) = \int_0^\infty \mathcal{N}(x; 0, \xi^{-1})\, d\mu(\xi)\,. \tag{3.1}$$

where μ is a non-decreasing and bounded function on $(0, \infty)$. Such representations with a general kernel are referred to as scale mixtures [59]. Gaussian scale mixtures were discussed in the examples of Dempster, Laird, and Rubin's original EM paper [31], and treated more extensively in [32].

In the convex type of variational representation, the non-Gaussian density is represented as a supremum over Gaussian functions of varying scale,

$$p(x) = \sup_{\xi > 0} \mathcal{N}(x; 0, \xi^{-1})\, \varphi(\xi)\,. \tag{3.2}$$

In the following sections we determine criteria that densities must satisfy in order to be represented in these forms.

3.2.1 Convex representation of strong super-Gaussians

A convex function can be represented as the pointwise supremum over a set of affine functions.

$$f(x) = \sup_{\phi} \phi x - b(\phi)$$

The function $b(\phi)$ which is the intercept corresponding to the slope ϕ is called the convex conjugate of f, and is denoted $f^*(\phi)$. The convex conjugate satisfies the dual relationship,

$$f^*(\phi) = \sup_{x} x\phi - f(x)$$

Now we wish to determine when a symmetric, unimodal density $p(x)$ can be represented in the form (3.2) for some function $\varphi(\xi)$. Equivalently, when,

$$-\log p(x) \;=\; -\sup_{\xi>0}\; \log \mathcal{N}\big(x\,; 0, \xi^{-1}\big)\varphi(\xi) \;=\; \inf_{\xi>0}\; \frac{1}{2}x^2\xi - \log \xi^{\frac{1}{2}}\varphi(\xi)$$

for all $x > 0$. The last formula says that $-\log p(\sqrt{x})$ is the concave conjugate of (the closure of the convex hull of) the function, $\log \xi^{\frac{1}{2}} \varphi(\xi)$ [92, §12]. This is possible if and only if $-\log p(\sqrt{x})$ is closed, increasing and concave on $(0, \infty)$. Thus we have the following.

Theorem 5. *A symmetric probability density* $p(x) = \exp(-g(x^2))$ *can be represented in the convex variational form,*

$$p(x) = \sup_{\xi > 0} \mathcal{N}(x; 0, \xi^{-1})\, \varphi(\xi)$$

if and only if $g(x) \equiv -\log p(\sqrt{x})$ *is increasing and concave on* $(0, \infty)$. *In this case we can use the function,*

$$\varphi(\xi) = \sqrt{2\pi/\xi}\, \exp\big(g^*(\xi/2)\big)\,,$$

where g^* *is the concave conjugate of* g.

Examples of densities satisfying this criterion include: (i) Generalized Gaussian $\propto \exp(-\gamma |x|^p)\,, p \leq 2$, (ii) Logistic, $\frac{d}{dx}(1 + \exp(-x))^{-1}$, (iii) Student's t, and (iv) symmetric α-stable densities (with characteristic function $\exp(-|\omega|^\alpha)$, $0 < \alpha \leq 2$).

The convex variational representation motivates the following definition.

Definition 1. *A symmetric probability density* $p(x)$ *is* ***strongly super-gaussian*** *if* $p(\sqrt{x})$ *is log-convex on* $(0, \infty)$, *and* ***strongly sub-gaussian*** *if* $p(\sqrt{x})$ *is log-concave on* $(0, \infty)$.

An equivalent definition is given in [11, pp. 60-61], which defines $p(x) = \exp(-f(x))$ to be sub-gaussian (super-gaussian) if $f'(x)/x$ is increasing (decreasing) on $(0, \infty)$. This condition is equivalent to $f(x) = g(x^2)$ with g concave, i.e. g' decreasing. The property of being strongly sub- or super-gaussian is independent of scale.

The essential property of "concavity in x^2" leading to this representation was used in [95, 55, 53, 56, 13] to represent the Logistic link function. A convex type representation of the Laplace density was applied to learning overcomplete representations in [43].

3.2.2 Gaussian scale mixtures

We now wish to determine when a probability density $p(x)$ can be represented in the form (3.1) for some $\mu(\xi)$ non-decreasing on $(0, \infty)$. A fundamental result dealing with integral representations was given by Bernstein and Widder (see [101]). It uses the following definition.

Definition 1. *A function $f(x)$ is **completely monotonic** on (a, b) if,*

$$(-1)^n f^{(n)}(x) \geq 0\,, \quad n = 0, 1, \ldots$$

for every $x \in (a, b)$.

That is, $f(x)$ is completely monotonic if it is positive, decreasing, convex, and so on. Bernstein's theorem [101, Thm. 12b] states:

Theorem 6. *A necessary and sufficient condition that $p(x)$ should be completely monotonic on $(0, \infty)$ is that,*

$$p(x) = \int_0^\infty e^{-tx} d\alpha(t)\,,$$

where $\alpha(t)$ is non-decreasing on $(0, \infty)$.

Thus for $p(x)$ to be a Gaussian scale mixture,

$$p(x) = e^{-f(x)} = e^{-g(x^2)} = \int_0^\infty e^{-\frac{1}{2}tx^2} d\alpha(t)\,,$$

a necessary and sufficient condition is that $p(\sqrt{x}) = e^{-g(x)}$ be completely monotonic for $0 < x < \infty$, and we have the following (see also [59, 3]),

Theorem 7. *A function $p(x)$ can be represented as a Gaussian scale mixture if and only if $p(\sqrt{x})$ is completely monotonic on $(0, \infty)$.*

3.2.3 Relationship between strong super-Gaussians and Gaussian scale mixtures

We now consider the relationship between the convexity-based strong super-Gaussianity and the integral-based Gaussian scale mixtures. Let $p(x) =$

$\exp(-g(x^2))$. We have seen that $p(x)$ can be represented in the form (3.2) if and only if $g(x)$ is symmetric and concave on $(0, \infty)$. And we have seen that $p(x)$ can be represented in the form (3.1) if and only if $p(\sqrt{x}) = \exp(-g(x))$ is completely monotonic. We now consider whether or not complete monotonicity of $p(\sqrt{x})$ implies the concavity of $g(x) = -\log p(\sqrt{x})$, that is whether the class of Gaussian scale mixtures is a subset of the class of strong super-Gaussians.

Complete monotonicity of a function $q(x)$ implies that $q \geq 0$, $q' \leq 0$, $q'' \geq 0$, etc. For example, if $p(\sqrt{x})$ is completely monotonic, then,

$$\frac{d^2}{dx^2}\, p(\sqrt{x}) = \frac{d^2}{dx^2}\, e^{-g(x)} = e^{-g(x)}\big(g'(x)^2 - g''(x)\big) \geq 0\,.$$

Thus if $g'' \leq 0$, then $p(\sqrt{x})$ is convex, but the converse does not necessarily hold. That is, concavity of g does not follow from convexity of $p(\sqrt{x})$, as the latter only requires that $g'' \leq g'^2$.

Concavity of g does follow however from the complete monotonicity of $p(\sqrt{x})$. For example, we can use the following result [16, §3.5.2].

Theorem 8. *If the functions $f_t(x)$, $t \in \mathcal{D}$, are convex, then $\int_{\mathcal{D}} e^{f_t(x)} dt$ is convex.*

Thus, completely monotonic functions, being scale mixtures of the log convex function e^{-x} by Theorem 6, are also log convex. We thus see that *any function that can be represented in the scale mixture form* (3.1) *can also be represented in the convex variational form* (3.2).

In fact, a stronger result holds. The following theorem [14, Thm. 4.1.5] establishes the equivalence between $q(x)$ and $g'(x) = d/dx - \log q(x)$ in terms of complete monotonicity.

Theorem 9. *If $g(x) > 0$, then $e^{-ug(x)}$ is completely monotonic for every $u > 0$, if and only if $g'(x)$ is completely monotonic.*

In particular, it holds that $q(x) \equiv p(\sqrt{x}) = \exp(-g(x))$ is convex only if $g''(x) \leq 0$.

To summarize, let $p(x) = e^{-g(x^2)}$. If g is increasing and concave for $x > 0$, i.e. p is strongly super-Gaussian, then $p(x)$ can be represented in the form (3.2).

If, in addition, the higher derivatives satisfy $g^{(3)}(x) \geq 0$, $g^{(4)}(x) \leq 0$, $g^{(5)}(x) \geq 0$, etc., then $p(x)$ also admits the Gaussian scale mixture representation (3.1).

3.3 Posterior moments of Gaussian Scale Mixtures

Following [32], differentiating under the (absolutely convergent) integral we get,

$$\begin{aligned} p'(s) &= \frac{d}{ds}\int_0^\infty p(s|\xi)p(\xi)d\xi \;=\; -\int_0^\infty \xi\, sp(s,\xi)\, d\xi \\ &= -sp(s)\int_0^\infty \xi p(\xi|s)\, d\xi \end{aligned}$$

Thus, with $p(s) = \exp(-f(s))$, we see that,

$$E(\xi_i|s_i) \;=\; \int_0^\infty \xi_i p(\xi_i|s_i)\, d\xi_i \;=\; -\frac{p'(s_i)}{s_i p(s_i)} \;=\; \frac{f'(s_i)}{s_i}\,. \tag{3.3}$$

3.4 Example densities

In this section we give some examples of densities used in practice that are Gaussian scale mixtures, and thus from the results of the previous section, necessarily also strong super-Gaussians. All densities will be given in a standard form. The form used for adaptation of the model will include location and scale parameters as well, so the density $p(x)$ refers to the family,

$$\frac{1}{\sigma}p\left(\frac{x-\mu}{\sigma}\right)$$

Generalized Gaussian

The Generalized Gaussian density has the form,

$$\mathcal{G}(x;\rho) = \frac{1}{2\Gamma(1+\frac{1}{\rho})}\, e^{-|x|^\rho}$$

It is strongly super-Gaussian for $0 < \rho \leq 2$. The scale mixing density is related to a positive alpha stable density of order $\rho/2$. Alpha stable densities are described below.

Generalized Cauchy

The generalized Cauchy density has the form,

$$\frac{\alpha\Gamma(\nu+1/\alpha)}{2\Gamma(1/\alpha)\Gamma(\nu)}\,\frac{1}{(1+|x|^{\alpha})^{\nu+1/\alpha}}$$

The Generalized Cauchy is strongly super-Gaussian for $\nu > 0$ and $0 < \alpha < 2$. The scale mixing density is related to the Gamma density.

Generalized Logistic

Also called the symmetric Fisher's z distribution [7]. The generalized logistic density has the form,

$$\frac{\Gamma(2\alpha)}{\Gamma(\alpha)^2}\,\frac{e^{-\alpha x}}{(1+e^{-x})^{2\alpha}}$$

The Generalized Logistic is strongly super-Gaussian for all $\alpha > 0$. The scale mixing density is related to the Kolmogorov-Smirnov distance statistic [44].

Symmetric α-stable

α-Stable densities have characteristic function

$$\varphi(t) = e^{-|t|^{\alpha}}$$

They are known to be Gaussian scale mixtures [94, 44] with scale density

$$p(\xi) = \frac{1}{2}\, s_{\alpha/2}\!\left(\frac{\xi}{2}\right)$$

where $s_{\alpha/2}$ is again a (one-sided) stable distribution of order $\alpha/2$. Since the Gaussian kernel is its own Fourier transform, taking Fourier transforms of both sides of the scale mixture equation shows that the mixing density of a Gaussian scale mixture has the same form as the mixing density of characteristic function, but over the inverse scale. Thus the mixing density of the generalized Gaussian density is also α-stable of order $\alpha/2$.

Other Gaussian scale mixtures are discussed in [44] involving Bessel functions and the incomplete gamma function.

3.5 Duality of sub- and super-Gaussianity

In analysis and geometry, there is a duality between vectors and linear functions, points and hyperplanes, and vertices and faces of polyhedra. In the basic linear model,

$$\mathbf{x} = \sum_i s_i \mathbf{a}_i$$

A useful intuitive understanding of sub- and super-Gaussianity can be gained from the association of sub-Gaussian s_i densities with hyperplanes, or faces of the distribution of $\mathbf{x}$ in the direction $\mathbf{a}_i$, and super-Gaussian s_i densities with vertices in the direction $\mathbf{a}_i$. The Gaussian density is distinguished by its having no inherent "directional" structure modulo linear transformations.

The strong super-Gaussian and Gaussian scale mixture representation theorems show that variational representations in terms of the Gaussian density apply only to super-Gaussian densities, ana cannot be used to represent sub-gaussian densities. Thus, for example, there is no variational Gaussian representation for the density $p(x) \propto \exp(-x^4)$. One can formulate representations of sub-gaussian densities in terms of other functions, but the usefulness of variational representations derives mainly from the properties of the Gaussian density.

There is, however, and interesting relationship between the convexity of strongly sub- and super-gaussian functions which we describe in this section. Consider the MAP estimation problem,

$$\begin{aligned} \hat{x} &= \arg\max_x \; p(x|y) \;=\; \arg\max_x \; p(y|x)p(x) \\ &= \arg\max_x \; \log p(y|x) + \log p(x) \end{aligned}$$

Now define $d(x) \equiv -\log p(y|x)$ and $f(x) \equiv -\log p(x)$. If f and d are convex, then the dual problem can be formulated involving $d^*(\phi)$ and $f^*(\phi)$. It turns out that if $p(x) = \exp(-f(x))$ is strongly sub-gaussian, i.e. $f(x) = g(x^2)$ with $g(x)$ convex on $(0, \infty)$, then we have the non-trivial result that $\exp(-f^*(\phi))$ is strongly super-gaussian, i.e. $f^*(\phi) = h(\phi^2)$ with $h(\phi)$ concave on $(0, \infty)$. We prove this in the following.

Theorem 10. *If f is convex, then f is strictly concave in x^2 on $(0, \infty)$ if and only if f^* is strictly convex in x^2 on $(0, \infty)$.*

Proof. If f and f^* are differentiable on $(0, \infty)$. Then f is strictly square-concave if and only if $f'(x)/x$ is strictly decreasing, i.e. if and only if $x < y$ implies,

$$\frac{f'(x)}{x} < \frac{f'(y)}{y} \tag{3.4}$$

Let $\phi_x = f'(x)$ and $\phi_y = f'(y)$. Then $\phi_x < \phi_y$ since f is strictly convex, and $x = f^{*\prime}(\phi_x)$ and $y = f^{*\prime}(\phi_y)$, since f' and $f{*}'$ are inverse functions. Substituting these into (3.4), we get,

$$\frac{\phi_x}{f^{*\prime}(\phi_x)} = \frac{f'(x)}{x} < \frac{f'(y)}{y} = \frac{\phi_y}{f^{*\prime}(\phi_y)}$$

or, $f^{*\prime}(\phi_x)/\phi_x > f^{*\prime}(\phi_y)/\phi_y$, which implies that $f^*(\phi)/\phi$ is strictly increasing and f^* is strictly convex in x^2.

More generally, we have,

$$\begin{aligned} f^*(\phi) &= \sup_x \phi x - f(x) \\ &= \sup_x \phi x - \inf_\xi \xi x^2/2 - g^*(\xi/2) \\ &= \sup_\xi g^*(\xi/2) + \sup_x \phi x - \xi x^2/2 \\ &= \sup_\xi g^*(\xi/2) + \tfrac{1}{2}\,\phi^2/\xi \end{aligned}$$

where $g(x) = f(\sqrt{|x|})$. This shows that $f^*(\sqrt{\phi})$ is the pointwise supremum of linear functions on $(0, \infty)$, and thus f^* is convex in ϕ^2. □

As $f^*(\phi)$ is convex in ϕ^2 on $(0, \infty)$, and this is the defining property of strongly sub-Gaussians, we have that $\exp(-f^*(\phi))$ is strongly sub-Gaussian. It is also apparent from the proof that $\exp(-f^*(\phi))$ can be represented as the pointwise infimum of functions proportional to Gaussian densities, in contrast to strongly super-Gaussian densities, which are pointwise supremums of Gaussians.

The functions that are concave in x^2, or strongly super-Gaussian, are useful in deriving monotonic algorithms for optimization since the family of bounds is on the "right" side of the objective function. For example, minimization of $f(x)$ becomes minimization of $F(x, \xi)$ over x and ξ. For f convex in x^2, we would have $\min_x \max_\xi F(x, \xi)$, resulting in a more difficult minimax problem.

It should be noted that while strongly sub-Gaussian densities are not supremums of Gaussians, mixtures of strong super-Gaussians are pointwise supremums of mixtures of Gaussians, and mixtures of strong super-Gaussians may be strongly sub-Gaussian. This is the case with a mixture of two unit variance Normal densities with means -1 and 1, which can be written in the form $\exp(-x^2/2)\cosh(x)$. This density is strongly sub-Gaussian, and is the density used as the sub-Gaussian model by Lee in [65] in the extended Infomax algorithm for separating sub- and super-Gaussian sources. Strong super-Gaussian mixtures are discussed in §6.3.3.

3.6 Kurtosis and strong super-Gaussianity

In this section we show that all strong super-Gaussian densities have positive kurtosis. For the proof, we use the following version of a result of Karlin [57, Lemmas A,B].

Theorem 11. *If $\int_a^b p(x)\,dx = \int_a^b q(x)\,dx$, $\int_a^b x^2 p(x)\,dx = \int_a^b x^2 q(x)\,dx$, $p(x)$ intersects $q(x)$ exactly two times on (a, b), and $p(x) \geq q(x)$ in a neighborhood of b, then*

$$\int_a^b \varphi(x)\,p(x)\,dx \geq \int_a^b \varphi(x)\,q(x)\,dx$$

for all φ such that $\varphi(\sqrt{x})$ is convex.

Theorem 12. *If $-\log p(\sqrt{x})$ is concave on $(0, \infty)$, (i.e. $p(x)$ is strongly super-gaussian,) then $p(x)$ has positive kurtosis.*

Proof. Let $p(x)$ be strongly super-gaussian, so that $g(x) = -\log p(\sqrt{x})$ is concave and increasing on $(0, \infty)$. Since $g(x)$ is concave, it can intersect a linear function a maximum of two times on $(0, \infty)$ [58, §1.4]. Thus $g(x^2)$ can intersect a quadratic function a maximum of two times, and $e^{-g(x^2)}$ can intersect $e^{-\beta x^2}$ a maximum of two times on $(0, \infty)$.

Now let X be distributed according to $p(x)$, and let Y be Gaussian, with $EX = EY = 0$, and $EX^2 = EY^2$. Then $p(x)$ must intersect the Gaussian density exactly four times (twice on $(0, \infty)$, and Karlin's theorem applies. In particular, since x^4 is convex with respect to x^2, if $X \sim p(x)$, with $p(x) = \exp(-g(x^2))$ with g concave, and Y is Gaussian, $EX = EY$, $EX^2 = EY^2$, then $EX^4 \geq EY^4$. In other words, X has a greater fourth moment than a Gaussian density with the same variance, which is to say that the density $p(x)$ has positive kurtosis. □

3.7 Multivariate densities and representations of dependent subspaces

The Gaussian scale mixture representation can be extended to vector subspaces to yield a model of non-affine dependency. This has been used recently by [60, 35] in a special case for independent component analysis. In this section we show how more general dependent multivariate densities can be derived using scale mixtures.

Suppose that we have a Gaussian scale mixture

$$x = \xi^{-1/2} z$$

where z is a standard Normal random variable. We can construct a random vector by multiplying the same scalar random variable $\xi^{-1/2}$ by a Gaussian random vector,

$$\mathbf{x} = \xi^{-1/2} \mathbf{z}$$

where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. For the density of $\mathbf{x}$ we then have,

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}} \int_0^\infty \xi^{d/2} e^{-\frac{1}{2}\xi \|\mathbf{x}\|^2} p(\xi) d\xi$$

If ξ is a Gamma random variable, then the density of $\mathbf{x}$ can be written in terms of the modified Bessel function of the second kind [35].

Now, if x is a Gaussian scale mixture with density $p(x)$, then,

$$p(\sqrt{x}) = \frac{1}{(2\pi)^{1/2}} \int_0^\infty \xi^{1/2} e^{-\frac{1}{2}\,\xi\, x} p(\xi) d\xi$$

Taking the kth derivative of both sides, we find,

$$\frac{d^k}{dx^k} p(\sqrt{x}) = \frac{(-2)^{-k}}{(2\pi)^{1/2}} \int_0^\infty \xi^{k+1/2} e^{-\frac{1}{2}\,\xi\, x} p(\xi) d\xi$$

Thus, if d is odd, then,

$$\pi^{-(d-1)/2} D^{(d-1)/2} p(\sqrt{x}) = \frac{1}{(2\pi)^{d/2}} \int_0^\infty \xi^{d/2} e^{-\frac{1}{2}\,\xi\, x} p(\xi) d\xi$$

and we can write the density of $p(\mathbf{x})$

$$d\ \textbf{odd}: \qquad p(\mathbf{x}) = \pi^{-(d-1)/2} (-D)^{(d-1)/2} p(\sqrt{x}) \big|_{x=\|\mathbf{x}\|^2} \tag{3.5}$$

For even d, the density of $p(\mathbf{x})$ can be written formally in terms of the Weyl fractional derivative of order $(d+1)/2$. However as the fractional derivative is is not generally obtainable in closed form, we consider a modification of the original univariate scale density $p(\xi)$,

$$\tilde{p}(\xi) = \frac{\xi^{-1/2} p(\xi)}{\int_0^\infty \xi^{-1/2} p(\xi)}$$

If $p(\xi)$ is finite at 0, then $E\,\xi^{-1/2}$ is finite. With this modified scale density, the density of x evaluated at $\sqrt{x}$ becomes,

$$p(\sqrt{x}) = \frac{1}{(2\pi)^{1/2} \mathcal{Z}} \int_0^\infty e^{-\frac{1}{2}\,\xi\, x} \tilde{p}(\xi) d\xi$$

where,

$$\mathcal{Z} = \int_0^\infty \xi^{-1/2} p(\xi) d\xi$$

Proceeding as we did for odd d, we find,

$$d\ \textbf{even}: \qquad p(\mathbf{x}) = \mathcal{Z} \sqrt{2} \pi^{-(d-1)/2} (-D)^{d/2} p(\sqrt{x}) \big|_{x=\|\mathbf{x}\|^2} \tag{3.6}$$

Example: 3D Dependent Logistic

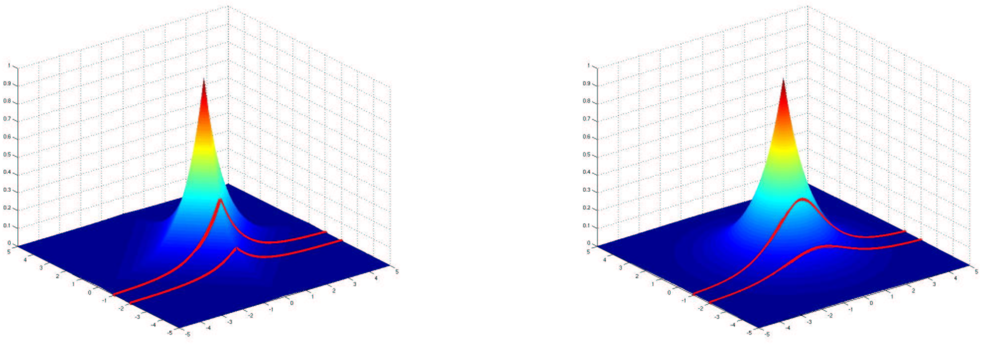
We consider an example. Suppose we wish to formulate a dependent Logistic type density on $\mathbb{R}^3$. The scale mixing density in the Gaussian scale mixture representation for the Logistic density has the density of the Kolmogorov-Smirnov distance statistic [3], which only expressible in series form. However, we may determine the multivariate density produced from the product,

$$\mathbf{s} = \xi^{-1/2}\mathbf{z}$$

where $\mathbf{s}, \mathbf{z} \in \mathbb{R}^3$, and $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Using the formula (3.5) for $d = 3$, we get,

$$p(\mathbf{s}) = \frac{1}{8\pi} \frac{\sinh\left(\frac{1}{2}\|\mathbf{s}\|\right)}{\|\mathbf{s}\| \cosh^3\left(\frac{1}{2}\|\mathbf{s}\|\right)}$$

Figure 3.3 illustrates the difference between the dependent and independent models for the Laplacian density.



(a)

(b)

Figure 3.3: (a) A two dimensional independent (factorial) Laplacian density. The marginal densities are constant. (b) A two dimensional dependent Laplacian density. The marginal densities are not constant.

The material in this chapter, in part, was published in, Palmer, J. A., Wipf, D. P., Kreutz-Delgado, K., Rao, B. D., "Variational EM Algorithms for Non-Gaussian Latent Variable Models," Advances in Neural Information Processing Systems, MIT Press, 2005.

4

Sparse Coding

In this chapter and the next, we consider the problem of "sparse coding", or representation of data vectors $\mathbf{x} \in \mathbb{R}^m$ in terms of a small number of "basis" vectors from a set of n vectors in a "dictionary" $\mathbf{A} = [\mathbf{a}_1 \cdots \mathbf{a}_n]$. If $n > m$, the dictionary is called overcomplete. As noted in chapter 2, strongly super-Gaussian densities are suitable for representing sparse random variables, i.e. random variables that are close to zero in most instances, but occasionally take on relatively large magnitudes.

The model is given by,

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \nu \tag{4.1}$$

We will take the noise to be Gaussian, $\nu \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_\nu)$, or in the overcomplete case we may take $\nu = \mathbf{0}$ and enforce the constraint $\mathbf{x} = \mathbf{A}\mathbf{s}$.

In this chapter, we consider the problem in the context of a given dictionary, where we are given $\mathbf{A}$ and $\mathbf{x}$ and we wish to find a sparse representation $\mathbf{s}$ such that $\mathbf{x} \approx \mathbf{A}\mathbf{s}$. The next chapter considers dictionary learning, where we are given a set of data $\mathbf{x}_1, \ldots, \mathbf{x}_T$, and we wish to find a dictionary $\mathbf{A}$ such that the data vectors are expected to have sparse representations in $\mathbf{A}$.

4.1 Estimation with a given dictionary

We consider two basic approaches to the sparse estimation problem, by which we mean specifically estimation of strongly super-Gaussian $\mathbf{s}$ in the model (4.1).

The first approach is to find the MAP estimate of $\mathbf{s}$, i.e.,

$$\hat{\mathbf{s}} = \arg\max_{\mathbf{s}} p(\mathbf{s}|\mathbf{x};\mathbf{A})$$

This corresponds to a generalized version of the FOCUSS algorithm [89, 90], and to a version of an EM algorithm derived by Dempster, Laird, and Rubin [32]. The generalized FOCUSS derivation exploits the convexity-based representation of strongly super-Gaussian priors, using an inequality of the form,

$$f(t) - f(s) \leq \frac{1}{2}\frac{f'(s)}{s}(t^2 - s^2) \tag{4.2}$$

which holds for f concave in s^2.

The EM algorithm uses the scale mixture representation, but leads to the same algorithm as the strong super-Gaussian representation when the prior density $p(\mathbf{s})$ is a Gaussian scale mixture. This is due to the fact that for Gaussian scale mixtures, $p(s) = \int \mathcal{N}(s; 0, \xi^{-1})p(\xi)d\xi = \exp(-f(s))$, we have,

$$E(\xi|s) = \frac{f'(s)}{s}$$

as shown in §3.3. The expression on the right is the same as that which appears in the convexity based inequality (4.2). A close relationship between monotonic algorithms for strong super-Gaussians and Gaussian scale mixtures is not unexpected given that the former class includes the latter as shown in Chapter 3.

The second approach attempts in a certain sense to model the posterior source density itself, as opposed to finding a point estimate of its maximum as in the MAP approach. Specifically the algorithm attempts to minimize an upper bound on the Kullback-Leibler divergence between $p(\mathbf{s}|\mathbf{x})$ and an approximating

Gaussian density. Note that the prior on $\mathbf{s}$ is *not* assumed to be Gaussian in the approximation. Rather a Gaussian approximation is used for the posterior $p(\mathbf{s}|\mathbf{x})$.

We consider first the MAP EM algorithms for strong super-Gaussian and the Gaussian scale mixture, showing that they are equivalent. We then consider the VB and MAP estimate of hyperparameter algorithms, showing that the strong super-Gaussian based convex bounding method is equivalent to the Gaussian scale mixture VB method.

4.1.1 MAP Estimation of Sources: Generalized FOCUSS

Consider first the MAP estimate of the sources.

Gaussian scale mixture case

Consider an EM algorithm to estimate $\mathbf{s}$ when the s_i are independent Gaussian scale mixtures. The EM algorithm alternates setting $\hat{\xi}_i$ to the posterior mean, $E(\xi_i|s_i^l) = f_i'(s_i^l)/s_i^l$, and setting $\mathbf{s}$ to minimize,

$$Q(\mathbf{s}|\mathbf{s}^l) = -\log p(\mathbf{x}|\mathbf{s})p(\mathbf{s}|\hat{\xi}) \;=\; \tfrac{1}{2}\,\mathbf{s}^T\mathbf{A}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A}\mathbf{s} - \mathbf{x}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A}\mathbf{s} + \tfrac{1}{2}\,\mathbf{s}^T\boldsymbol{\Lambda}\mathbf{s} + \text{const.} \tag{4.3}$$

where $\boldsymbol{\Lambda} = \text{diag}(\hat{\xi})^{-1}$. At iteration l, we put $\lambda_i^l = s_i^l/f'(s_i^l)$, and $\boldsymbol{\Lambda}^l = \text{diag}(\lambda^l)$, and

$$\mathbf{s}^{l+1} = \boldsymbol{\Lambda}^l\mathbf{A}^T(\mathbf{A}\boldsymbol{\Lambda}^l\mathbf{A}^T + \boldsymbol{\Sigma}_\nu)^{-1}\mathbf{x} \tag{4.4}$$

In [32], the s_i in are estimated as non-random parameters, with the noise ν being non-gaussian, but the derivation of the algorithm is the same.

Strong super-Gaussian case

Now consider the MAP estimate of $\mathbf{s}$ given $\mathbf{x}$, assuming only strong super-gaussianity of the s_i. Then we have,

$$\arg\max_{\mathbf{s}} p(\mathbf{s}|\mathbf{x}) \;=\; \arg\max_{\mathbf{s}} p(\mathbf{x}|\mathbf{s})p(\mathbf{s}) \;=\; \arg\max_{\mathbf{s}}\max_{\xi} p(\mathbf{x}|\mathbf{s})p(\mathbf{s};\xi)h(\xi)$$

Now since,

$$-\log p(\mathbf{x}|\mathbf{s})p(\mathbf{s};\xi)h(\xi) \;=\; \tfrac{1}{2}\,\mathbf{s}^T\mathbf{A}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A}\mathbf{s} \;-\; \mathbf{x}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A}\mathbf{s} \;+\; \sum_{i=1}^{n} s_i^2\xi_i/2 - g_i^*(\xi_i/2)$$

the MAP estimate can be improved iteratively by alternately maximizing $\mathbf{s}$ and ξ,

$$\xi_i^l \;=\; 2\,g_i^{*\prime\,-1}\Big({s_i^l}^2\Big) \;=\; 2\,g_i'\Big({s_i^l}^2\Big) \;=\; \frac{f_i'(s_i^l)}{s_i^l} \tag{4.5}$$

with $\mathbf{s}$ updated as in (4.4). We thus see that this algorithm is equivalent to the MAP algorithm derived in the previous section for Gaussian scale mixtures. That is, for direct MAP estimation of latent variable $\mathbf{s}$, the EM Gaussian scale mixture method and the variational bounding method yield the same algorithm.

This algorithm has also been derived in the image restoration literature [41] as the "half-quadratic" algorithm, and it is the basis for the FOCUSS algorithms derived in [89, 90, 88]. The regression algorithm given in [37] for the particular cases of Laplacian and Jeffrey's priors is based on the derivation in §4.1.1, and is in fact equivalent to the FOCUSS algorithm derived in [89].

4.1.2 Minimizing KL Divergence

There are three ways to derive this family of algorithms: (1) a variational bounding approach using strong super-Gaussianity, which finds the Gaussian density that minimizes and upper bound on the KL divergence from the true posterior [43, 13], (2) a MAP estimate of the "hyperparameters", ξ, in the Gaussian scale mixture representation of the sources, $s = z\,\xi^{-\frac{1}{2}}$, where z is standard Normal, and (3) a Variational Bayes approach relying on the Gaussian scale mixture representation [4].

Convex Bounding with Strong Super-Gaussians

We again use the following variational free energy formulation,

$$\log p(\mathbf{x}) = \int q(\mathbf{s})\log\frac{p(\mathbf{x},\mathbf{s})}{q(\mathbf{s})}d\mathbf{s} + D\big(q(\mathbf{s})\|p(\mathbf{s}|\mathbf{x})\big) \tag{4.6}$$

Since the left hand side does not depend on $q(\mathbf{s})$, we have,

$$\arg\min_{q(\mathbf{s})\in\mathcal{C}} D\big(q(\mathbf{s})\|p(\mathbf{s}|\mathbf{x})\big) = \arg\max_{q(\mathbf{s})\in\mathcal{C}} \int q(\mathbf{s}) \log \frac{p(\mathbf{x},\mathbf{s})}{q(\mathbf{s})} d\mathbf{s}$$

where $\mathcal{C}$ is some class of probability densities. So minimizing the KL divergence with respect to $q(\mathbf{s}) \in \mathcal{C}$ is equivalent to maximizing the negative free energy.

Also, maximizing a lower bound on the negative free energy is equivalent to minimizing an upper bound on the KL divergence. If we write (4.6) for simplicity as, $L = -F + D$, and suppose that $\tilde{F}$ is an upper bound on F. Then $\tilde{D} = L + \tilde{F}$ is an upper bound on D, and minimizing $\tilde{F}$ is equivalent to minimizing $\tilde{D}$.

Now, we have,

$$\log p(\mathbf{x},\mathbf{s}) = -\tfrac{1}{2} \left\|\mathbf{x} - \mathbf{A}\mathbf{s}\right\|^2_{\boldsymbol{\Sigma}_\nu^{-1}} + \log p(\mathbf{s}) + \text{const.}$$

And since the s_i are assumed to be independent and symmetric about 0,

$$p(\mathbf{s}) = \sum_{i=1}^{n} \log p_i(s_i) = -\sum_{i=1}^{n} f_i(s_i) = -\sum_{i=1}^{n} g_i(s_i^2)$$

where $g_i(s_i) = f_i(\sqrt{(s_i)}) = -\log p_i(\sqrt{|s_i|})$. Strong super-Gaussianity of $p_i(s_i)$ means that g_i is concave on $(0, \infty)$, which then implies,

$$E g_i(s_i^2) \leq g_i(E s_i^2)$$

where E is the expectation operator. Let E_q denote expectation with respect to the density $q(\mathbf{s})$. Then

$$\begin{aligned} -E_q \log p(\mathbf{x},\mathbf{s}) &= \tfrac{1}{2} E_q \left\|\mathbf{x} - \mathbf{A}\mathbf{s}\right\|^2_{\boldsymbol{\Sigma}_\nu^{-1}} + \sum_i E_q\, g_i(s_i^2) + \text{const.} \\ &\leq \tfrac{1}{2} E_q \left\|\mathbf{x} - \mathbf{A}\mathbf{s}\right\|_{\boldsymbol{\Sigma}_\nu^{-1}} + \sum_i g_i(E_q s_i^2) + \text{const.} \end{aligned}$$

Now if $q(\mathbf{s})$ is Gaussian with mean $\mu_{\mathbf{s}|\mathbf{x}}$ and covariance $\boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}$, then

$$\begin{aligned} F = -\int q(\mathbf{s}) \log \frac{p(\mathbf{x},\mathbf{s})}{q(\mathbf{s})} d\mathbf{s} \;&\leq\; \mathbf{x}^T \boldsymbol{\Sigma}_\nu^{-1} \mu_{\mathbf{s}|\mathbf{x}} + \tfrac{1}{2} \operatorname{tr}\big(\mathbf{A}^T \boldsymbol{\Sigma}_\nu^{-1} \mathbf{A} \big(\mu_{\mathbf{s}|\mathbf{x}} \mu_{\mathbf{s}|\mathbf{x}}^T + \boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}\big)\big) \\ &\quad + \sum_{i=1}^{n} g_i\big(\mu^2_{s_i|\mathbf{x}} + [\boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}]_{ii}\big) + \tfrac{1}{2} \log\det \boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}} + \text{const.} = \tilde{F} \end{aligned}$$

Now since the g_i are concave on $(0, \infty)$,

$$g(t^2) = \min_{\xi} \xi t^2/2 - g^*(\xi/2)$$

and the infimum is attained at $\xi = 2g'(t^2)$, where

$$g'(t^2) = \frac{f'(t)}{2t}$$

Thus we have

$$\begin{aligned} \tilde{F} \;&=\; \min_{\xi} \; \mathbf{x}^T \boldsymbol{\Sigma}_\nu^{-1} \mu_{\mathbf{s}|\mathbf{x}} + \tfrac{1}{2}\mathrm{tr}\big(\mathbf{A}^T \boldsymbol{\Sigma}_\nu^{-1} \mathbf{A}\big(\mu_{\mathbf{s}|\mathbf{x}} \mu_{\mathbf{s}|\mathbf{x}}^T + \boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}\big)\big) \\ &\quad + \tfrac{1}{2}\mu_{\mathbf{s}|\mathbf{x}}^T \mathrm{diag}(\xi) \mu_{\mathbf{s}|\mathbf{x}} + \tfrac{1}{2}\sum_{i=1}^{n} \xi_i [\boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}]_{ii} - g^*(\xi_i/2) + \tfrac{1}{2}\log\det \boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}} + \mathrm{const.} \end{aligned}$$

This can be minimized with respect to $\mu_{\mathbf{s}|\mathbf{x}}$, $\boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}$, and ξ by coordinate descent. Let $\boldsymbol{\Lambda}^l = \mathrm{diag}(\xi^l)^{-1}$. Then the algorithm,

$$\begin{aligned} \mu_{\mathbf{s}|\mathbf{x}}^{l+1} \;&=\; \boldsymbol{\Lambda}^l \mathbf{A}^T (\mathbf{A}\boldsymbol{\Lambda}^l \mathbf{A} + \boldsymbol{\Sigma}_\nu)^{-1} \mathbf{x} \\ \boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}^{l+1} \;&\leftarrow\; \boldsymbol{\Lambda}^l - \boldsymbol{\Lambda}^l \mathbf{A}^T (\mathbf{A}\boldsymbol{\Lambda}^l \mathbf{A} + \boldsymbol{\Sigma}_\nu)^{-1} \mathbf{A}\boldsymbol{\Lambda}^l \\ \xi_i^{l+1} \;&\leftarrow\; \frac{f'(\sigma_i^l)}{\sigma_i^l}, \quad \sigma_i^l = \sqrt{\mu_{s_i|\mathbf{x}}^2 + [\boldsymbol{\Sigma}_{\mathbf{s}|\mathbf{x}}]_{ii}}, \quad i = 1, \ldots, n \end{aligned}$$

monotonically decreases $\tilde{F}$.

The algorithm is thus equivalent to that in §4.1.1 except that the expectation is taken of s^2 and ξ is minimized, rather than taking the expectation of ξ and maximizing $\mathbf{s}$ as in §4.1.1. Here, instead of $f'(s^l)/s^l$, the diagonal weighting matrix becomes,

$$\xi_i \;=\; \frac{f'(\sigma_i^l)}{\sigma_i^l}$$

where $\sigma_i = \sqrt{E\,(s_i^2|\mathbf{y};\xi_i)}$. Although $\tilde{p}$ is not a probability density function, the proof of convergence for EM does not assume unit normalization. This theory is the basis for the algorithm presented in [43] for the particular case of a Laplacian prior (where in addition $\mathbf{A}$ in the model (5.2) is updated according to the standard EM update.)

MAP Estimation of Hyperparameters

If the density of s is a Gaussian scale mixture, then s can be represented as a product of a standard Normal, z, and a non-negative random variable ξ, as $s = z\xi^{-\frac{1}{2}}$. This leads to the Gaussian scale mixture representation (3.1) of the density of s. Furthermore, $\mathbf{x}$ is non-Gaussian, but conditionally Gaussian given ξ,

$$p(\mathbf{x}|\xi) = \mathcal{N}\big(\mathbf{0}, \mathrm{diag}(\xi)^{-1}\big)$$

In this representation, the ξ_i are sometimes referred to as "hyperparameters" [98].

Now consider an EM algorithm to find the MAP estimate of ξ,

$$\hat{\xi} = \arg\max_{\xi} p(\xi|\mathbf{x})$$

For the complete likelihood, we have,

$$p(\xi, \mathbf{s}|\mathbf{x}) \;\propto\; p(\mathbf{x}|\mathbf{s}, \xi)p(\mathbf{s}|\xi)p(\xi) \;=\; p(\mathbf{x}|\mathbf{s})p(\mathbf{s}|\xi)p(\xi)$$

The function to be minimized over ξ is then,

$$\big\langle -\log p(\mathbf{s}|\xi)p(\xi)\big\rangle_{\mathbf{s}} \;=\; \sum_i \tfrac{1}{2}\langle s_i^2\rangle\,\xi_i - \log\sqrt{\xi_i}\,p(\xi_i) + \text{const.} \tag{4.7}$$

If we define $h(\xi) = \log\sqrt{\xi_i}\,p(\xi_i)$, and assume that this function is concave, then the optimal value of ξ is given by,

$$\xi_i = h^{*\prime}\big(\tfrac{1}{2}\langle s_i^2\rangle\big)$$

This algorithm monotonically increases of $p(\xi|\mathbf{s})$, yielding $\hat{\xi}$, which then yields an estimate of $\mathbf{s}$ by taking $\hat{\mathbf{s}} = E(\mathbf{s}|\mathbf{x}, \hat{\xi})$. Alternative algorithms result from using this method to find the MAP estimate of different functions of the scale random variable ξ.

Variational Bayes Algorithm

Now consider using the VB method as in §2.1.4. In the linear model with Gaussian scale mixture latent variables, the complete likelihood is again,

$$p(\mathbf{x}, \mathbf{s}, \xi) \;=\; p(\mathbf{x}|\mathbf{s})p(\mathbf{s}|\xi)p(\xi)$$

The optimal approximate posteriors are given by,

$$q(\mathbf{s}|\mathbf{x}) \;=\; \mathcal{N}(\mathbf{s};\mu_{\mathbf{s}|\mathbf{x}},\Sigma_{\mathbf{s}|\mathbf{x}})\,, \qquad q(\xi_i|\mathbf{x}) \;=\; p\left(\xi_i \,\middle|\, s_i=\langle s_i^2\rangle^{1/2}\right)$$

where, letting $\mathbf{\Lambda}=\mathrm{diag}(\langle\xi\rangle)^{-1}$, the posterior moments are given by,

$$\begin{aligned}\mu_{\mathbf{s}|\mathbf{x}} &= \mathbf{\Lambda}\mathbf{A}^T(\mathbf{A}\mathbf{\Lambda}\mathbf{A}^T+\mathbf{\Sigma}_\nu)^{-1}\mathbf{x}\\ \Sigma_{\mathbf{s}|\mathbf{x}} &= (\mathbf{A}^T\mathbf{\Sigma}_\nu^{-1}\mathbf{A}+\mathbf{\Lambda}^{-1})^{-1} \;=\; \mathbf{\Lambda}-\mathbf{\Lambda}\mathbf{A}^T(\mathbf{A}\mathbf{\Lambda}\mathbf{A}^T+\mathbf{\Sigma}_\nu)^{-1}\mathbf{A}\mathbf{\Lambda}\,.\end{aligned}$$

The only relevant fact about $q(\xi|\mathbf{y})$ that we need is $\langle\xi\rangle$, for which we have, using (3.3),

$$\langle\xi_i\rangle \;=\; \int \xi_i q(\xi_i|\mathbf{y})\,d\xi_i \;=\; \int \xi_i p\left(\xi_i\,|\,s_i=\langle s_i^2\rangle^{1/2}\right)\,d\xi_i \;=\; \frac{f'(\sigma_i)}{\sigma_i}$$

where $\sigma_i=\sqrt{E\left(s_i^2|\mathbf{x};\xi_i\right)}$. We thus see that the VB algorithm is equivalent to the algorithm which found the Gaussian posterior minimizing a convexity based upper bound on the KL divergence. Here we find the factorial density minimizing the KL divergence assuming that $p(\mathbf{s})$ is a Gaussian scale mixture.

The algorithms of this section differ from those in §4.1.1 in that they all use the conditional expectation of s_i^2 given $\mathbf{x}$ and ξ as the diagonal weighting matrix, whereas the algorithms in §4.1.1 use the posterior maximum over $\mathbf{s}$. The diagonal weighting matrix in §4.1.1 is a function of $\mathbf{s}^l$ only, and each diagonal component ξ_i is a function of s_i only. In this section, the elements of the diagonal weighting matrix are functions the entire vector $\mathbf{s}^l$ through the term $[\mathbf{\Sigma}_{\mathbf{s}|\mathbf{x}}]_i i$. In fact, $[\mathbf{\Sigma}_{\mathbf{s}|\mathbf{x}}]_i i$ cannot be written as a function of any $\mathbf{s}^{l-1}$ either, and is actually a function of all of the previous $\mathbf{s}_l$.

In the MAP algorithms of the previous section, and the VB type algorithms of this section, the functional form of the diagonal weighting matrix involves the function $f'(t)/t$. This function is given in Table 4.1 for some commonly used strongly super-Gaussian densities.

Table 4.1: Variational weight parameter for common strongly super-gaussian densities.

Density Name	Density Form	$\xi = f'(y)/y$
Generalized Gaussian, $0 < \rho \leq 2$	$\exp(-\lvert y\rvert^{\rho})$	$\rho \, \lvert y\rvert^{\rho-2}$
Student's t, $\nu > 0$	$(1 + y^2/\nu)^{-(\nu+1)/2}$	$(\nu + 1)/(\nu + y^2)$
Jeffrey's prior	$1/y$	$1/y^2$
Logistic	$1/\cosh^2(y/2)$	$\tanh(y/2)/y$
Symmetric α-stable	no closed form	no closed form

4.1.3 Illustration of the Difference between MAP and VB

We illustrate the difference intuitively with an example that shows that the VB approach does not necessarily lead to sparse solutions when "proper" priors, i.e. densities $p(\mathbf{s})$ that are integrable, are used. This is due to the whole posterior density modeling nature of the VB approach. Only when an improper prior is used is the VB approximating density able to concentrate all of its mass on a sparse solution with zero variance estimates for some dimensions.

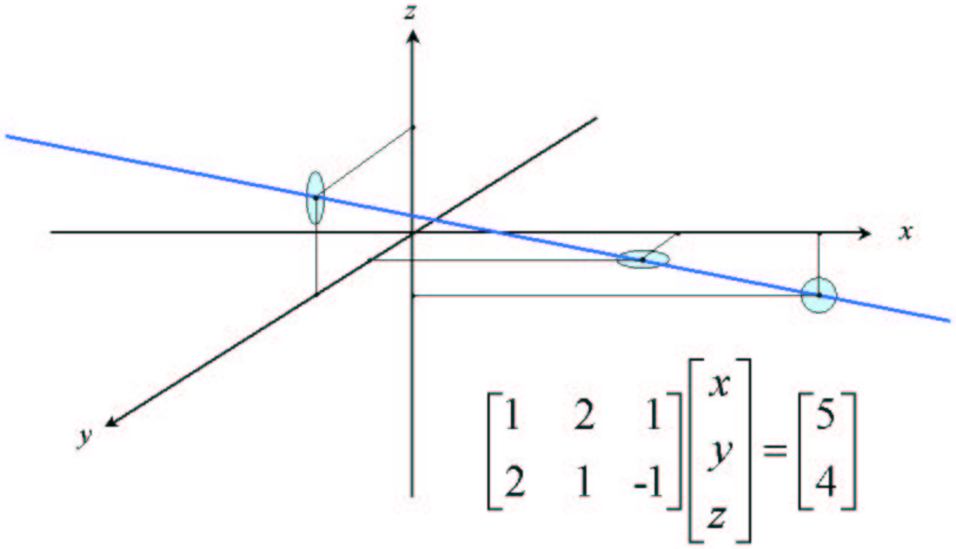

z
x
y
$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}$

Figure 4.1: Example of a sparse coding problem. The solution space is for the given linear system is plotted. The sparse solutions occur where the solution space breaks the orthant boundaries.

The example problem is depicted graphically in Figure 4.1. The null space of the 2 × 3 matrix is one dimensional and passes through the minimum norm solution. The line in the figure is a plot of the one dimensional space of solutions. The sparse solutions are those with one or more zeros. In the figure, the sparse solutions are found where the solution line breaks the orthant boundaries. For the system shown, there are three sparse solutions. If the solution line passed through one of the axes, then there would be two sparse solutions, one of which would have only one non-zero element.

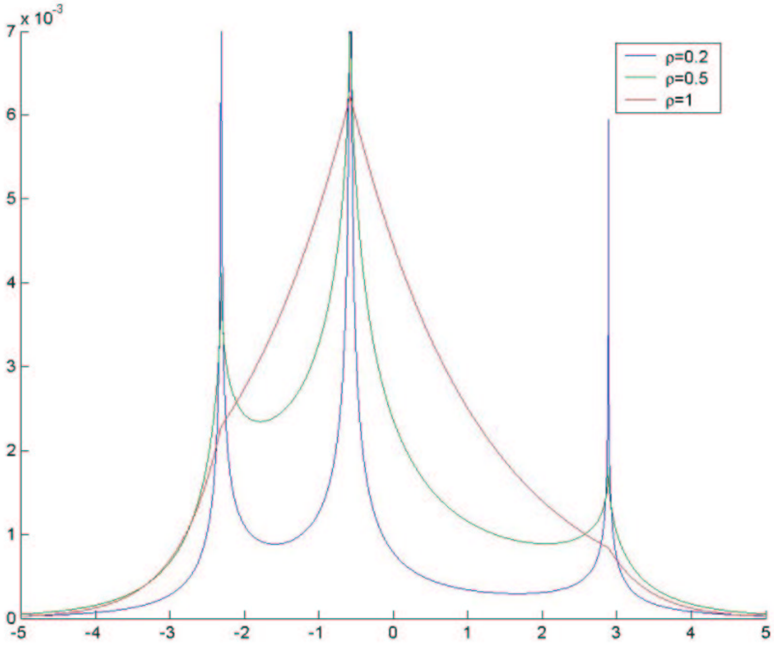
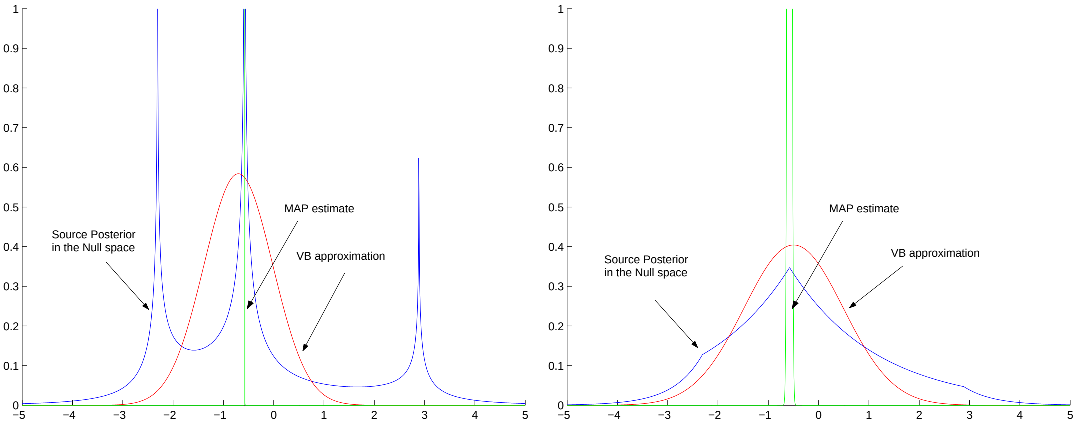


Figure 4.2: Plot of the posterior source density on the line solution space in Figure 4.1 for three values of Generalized Gaussian shape parameter. For the Laplacian prior ($\rho = 1.0$), the maximum of the posterior is the global maximum. For $\rho < 1$ there is a maximum for each sparse solution. In the latter case, the MAP estimate may converge to any of the sparse solutions, depending on starting point. The VB method has fewer local optima, but is only sparse as $\rho \rightarrow 0$, and even for $\rho = 0.2$ the posterior maximum is not sparse.

Since the null space in this example is one dimensional, we can make a two dimensional plot of the posterior density in the low noise limit by evaluating $p(\mathbf{s})$ as t varies in $\mathbf{s} = \bar{\mathbf{s}} + t\mathbf{v}$, where $\bar{\mathbf{s}}$ is the minimum norm solution and vbf is the vector of the null space. This is shown in Figure 4.1.



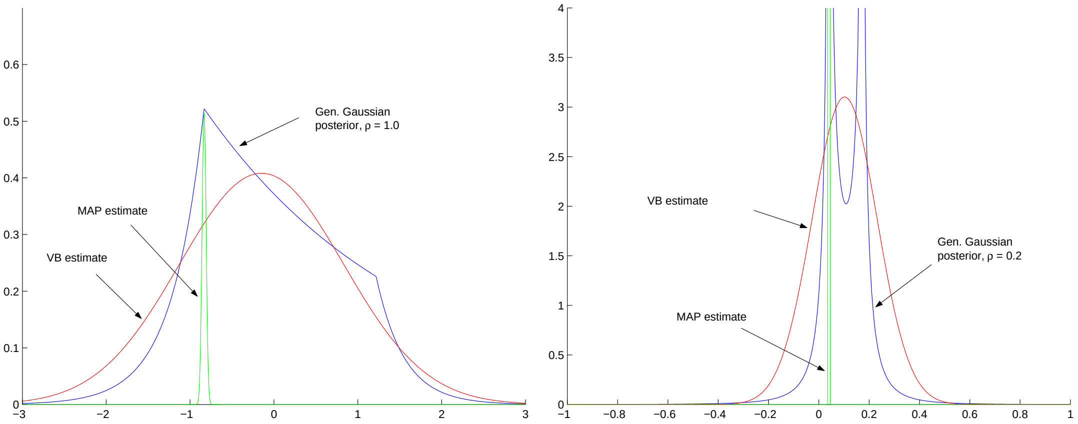
(a)

(b)

Figure 4.3: Plot of the posterior in the null space for the example in Figure 4.1 for (a) Laplacian prior and (b) Generalized Gaussian with $\rho = 0.2$. The MAP estimate goes to the maximum of the posterior, while the VB finds the mean *and* variance of the Gaussian that minimizes an upper bound on the KL divergence. In this case, the Laplacian VB algorithm converges to the sparse solution.

In Figure 4.4 another null space plot is shown with $\mathbf{A}$ of dimension 1×2, with two sparse solutions. It is clear in this example that the VB solution tries to match the entire posterior density, not necessarily leading to a sparse estimate for the source, even for negative log concave priors.

The distinction between MAP estimation of components and estimation of hyperparameters has been discussed in [72] and [98] for the case of Gamma distributed inverse variance.



(a) (b)

Figure 4.4: Plot of the posterior in the null space for $\mathbf{A}$ of dimension 1×2 for (a) Laplacian prior and (b) Generalized Gaussian with $\rho = 0.1$. The MAP estimate goes to the maximum of the posterior, while the VB finds the mean *and* variance of the Gaussian that minimizes an upper bound on the KL divergence. Since the Generalized Gaussian has finite mass, even though for $\rho = 0.2$ it is considered highly sparse, the mean of the VB approximation does not necessarily converge to a sparse solution.

4.2 Iteratively Re-weighted Least Squares and Convergence of MAP estimate

To prove convergence, we use the Global Convergence Theorem of Zangwill [104, 70], which is stated in the following theorem.

Theorem 13 (Global Convergence Theorem [70]). *Let* $\mathbf{M}$ *be an algorithm on* Ω*, and suppose that, given* x_0*, the sequence* $\left\{x^l\right\}, l = 0, 1, ...$ *is generated satisfying*

$$x^{l+1} \in \mathbf{M}(x^l) \tag{4.8}$$

Let a solution set $\Gamma \subset \Omega$ *be given, and suppose*

1. *All points* x^l *are contained in a compact subset of* Ω*.*

2. *There is a continuous function* f *on* Ω *such that*

(a) *if* $x \notin \Gamma$, *then* $f(y) < f(x)$ *for all* $y \in \mathbf{M}(x)$

(b) *if* $x \in \Gamma$, *then* $f(y) \leq f(x)$ *for all* $y \in \mathbf{M}(x)$

3. *The mapping* $\mathbf{M}$ *is closed at points outside* Γ

Then the limit of any convergent subsequence of $\left\{x^l\right\}$ *is a solution.*

Lemma 1. *Let* $\mathcal{W} \subset \mathbb{R}^{n\times n}$ *be the set of diagonal and positive definite matrices. Let* $\mathcal{S}$ *be the set of all* $\mathbf{s} \in \mathbb{R}^n$ *such that* $\mathbf{s} = \arg\min_{\mathbf{s}} \frac{1}{2}\mathbf{s}^T W \mathbf{s}$ *subject to* $\mathbf{A}\mathbf{s} = \mathbf{x}$, *for some* $W \in \mathcal{W}$. *Then* $\mathcal{S}$ *is bounded.*

Proof. Denote the constraint space $\mathcal{C} = \{\mathbf{s} : \mathbf{A}\mathbf{s} = \mathbf{x}\}$, and let $\mathcal{O}_j$, $j = 1, \ldots, 2^n$, be the jth orthant in $\mathbb{R}^n$. The intersections $\mathcal{C}_j = \mathcal{C} \bigcap \mathcal{O}_j$ are either bounded or unbounded. Clearly each $\mathcal{C}_j$ is the intersection of a finite number of half-spaces, and thus by the theorem of Minkowski and Weyl [92, Thm. 19.1], $\mathcal{C}_j$ is a finitely generated convex set. This means that for a finite set of points $\mathbf{v}_j$, $j = 1, \ldots, n_1$, and directions $\mathbf{d}_k$, $k = 1, \ldots, n_2$, we have

$$\mathcal{C}_j = \left\{\mathbf{s} \;:\; \mathbf{s} = \sum_{r=1}^{n_1} \alpha_r \mathbf{v}_r + \sum_{k=1}^{n_2} \lambda_k \mathbf{d}_k, \quad \sum_{r=1}^{n_1} \alpha_r = 1,\ \alpha_r \geq 0,\ \lambda_k \geq 0\right\}.$$

Now suppose that $\mathbf{s}^l$ is the minimizer of $\frac{1}{2}\mathbf{s}^T W^{-1}\mathbf{s}$ subject to $\mathbf{A}\mathbf{s} = \mathbf{x}$, and suppose $\mathbf{s}^l \in \mathcal{C}_j$ with $\mathcal{C}_j$ unbounded. Clearly the directions $\mathbf{d}_k$ are in $\mathcal{O}_j$. Now, if $\lambda_k > 0$ for some k then,

$$\mathbf{s}^l = \tilde{\mathbf{s}}^l + \lambda_k \mathbf{d}_k$$

where $\tilde{\mathbf{s}}^l \in \mathcal{O}_j$, and,

$$\tfrac{1}{2}\mathbf{s}^l W \mathbf{s}_l \;=\; \tfrac{1}{2}(\tilde{\mathbf{s}}^l)^T W \tilde{\mathbf{s}}^l + \lambda_k \mathbf{d}_k^T W \tilde{\mathbf{s}}^l + \tfrac{1}{2}\lambda_k \mathbf{d}_k^T W \mathbf{d}_k \;>\; \tfrac{1}{2}(\tilde{\mathbf{s}}^l)^T W \tilde{\mathbf{s}}^l$$

where $\mathbf{d}_k^T W \tilde{\mathbf{s}}^l$ is positive because $\tilde{\mathbf{s}}^l$ and $\mathbf{d}_k$ are in the same orthant $\mathcal{O}_j$ and W is diagonal and positive definite. But this contradicts the assumption that $\mathbf{s}^l = \arg\min_{\mathbf{s}\in\mathcal{C}} \frac{1}{2}\mathbf{s}^T W^{-1}\mathbf{s}$. Thus $\mathbf{s}^l$ must lie in the bounded convex hull of the points $\mathbf{v}_j$, $j = 1, \ldots, n_1$. Thus $\mathbf{s}_l$ either lies in a bounded $\mathcal{C}_j = \mathcal{C} \bigcap \mathcal{O}_j$, or if $\mathcal{C}_j$ is unbounded, then $\mathbf{s}^l$ lies in a bounded subset of $\mathcal{C}_j$. Hence $\mathbf{s}^l$ lies in the union of all the "bounded parts" of the orthant intersections $\mathcal{C}_j$, which is bounded. □

Lemma 2. *Let* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *be such that every subset of* m *columns of* $\mathbf{A}$ *is linearly independent, and let* $\mathbf{x} \in \mathbb{R}^m$ *be linearly independent of every subset of* $m-1$ *columns of* $\mathbf{A}$. *Let* $w : \mathbb{R}^n \to \mathbb{R}^n$ *be continuous on* $\mathbb{R}^n$. *Then the mapping,*

$$\mathbf{M}(\mathbf{s}) = \operatorname{diag}(w(\mathbf{s}))\mathbf{A}^T(\mathbf{A}\operatorname{diag}(w(\mathbf{s}))\mathbf{A}^T)^{-1}\mathbf{x}$$

is continuous.

Proof. We show that for any positive definite matrix $\mathbf{B}$,

$$(\mathbf{B}+\mathbf{E})\mathbf{A}^T\big(\mathbf{A}(\mathbf{B}+\mathbf{E})\mathbf{A}^T\big)^{-1}\mathbf{x} = \mathbf{B}\mathbf{A}^T(\mathbf{A}\mathbf{B}\mathbf{A}^T)^{-1}\mathbf{x} + o(\|\mathbf{E}\|)$$

where $o(\|\mathbf{E}\|)$ is a matrix whose elements tend to 0 as $\|\mathbf{E}\|$ tends to $\mathbf{0}$.

We have,

$$\begin{aligned}
&(\mathbf{B}+\mathbf{E})\mathbf{A}^T\big(\mathbf{A}(\mathbf{B}+\mathbf{E})\mathbf{A}^T\big)^{-1}\mathbf{x} - \mathbf{B}\mathbf{A}^T\big(\mathbf{A}\mathbf{B}\mathbf{A}^T\big)^{-1}\mathbf{x} \\
&\qquad = \big(\mathbf{B}\mathbf{A}^T\big)\Big(\big(\mathbf{A}\mathbf{B}\mathbf{A}^T + \tilde{\mathbf{E}}\big)^{-1} - \big(\mathbf{A}\mathbf{B}\mathbf{A}^T\big)^{-1}\Big)\mathbf{x} + o(\|\mathbf{E}\|)
\end{aligned}$$

and thus,

$$\begin{aligned}
&\Big\|(\mathbf{B}+\mathbf{E})\mathbf{A}^T\big(\mathbf{A}(\mathbf{B}+\mathbf{E})\mathbf{A}^T\big)^{-1}\mathbf{x} - \mathbf{B}\mathbf{A}^T\big(\mathbf{A}\mathbf{B}\mathbf{A}^T\big)^{-1}\mathbf{x}\Big\| \\
&\leq \ \big\|\mathbf{B}\mathbf{A}^T\big\| \ \Big\|\big(\mathbf{A}\mathbf{B}\mathbf{A}^T + \tilde{\mathbf{E}}\big)^{-1} - \big(\mathbf{A}\mathbf{B}\mathbf{A}^T\big)^{-1}\Big\| \ \|\mathbf{x}\| + o(\|\mathbf{E}\|) \\
&= \ o(\|\mathbf{E}\|)
\end{aligned}$$

where $\tilde{\mathbf{E}} = \mathbf{A}\mathbf{E}\mathbf{A}^T$. The last step follows from the continuity of the matrix inverse [45, p. 58]. For the lemma we use the particular case where $\mathbf{E}$ is diagonal. □

Lemma 3. *Let* $\mathbf{A}$ *and* $\mathbf{x}$ *satisfy the assumptions of Lemma 2, let* $\mathbf{s} \in \mathbb{R}^n$, *and let* W^l *be the diagonal, positive semidefinite matrix with diagonal elements* $s_i^l / f_i'(s_i^l)$, *where* $f_i'(s_i)$ *is a function that is anti-symmetric and positive on* $(0, \infty)$. *If,*

$$\lim_{s_i \to 0} s_i / f_i'(s_i) > 0$$

then $\mathbf{s}^l$ *is a fixed point of the mapping,*

$$\mathbf{s}^{l+1} = W^l \mathbf{A}^T(\mathbf{A}W^l\mathbf{A}^T)^{-1}\mathbf{x} \tag{4.9}$$

if and only if,

$$\nabla f(\mathbf{s}^l) \in \mathcal{R}\big(\mathbf{A}^T\big).$$

If

$$\lim_{s_i \to 0} s_i / f_i'(s_i) = 0$$

then $s_i^{l+1} = 0$ *if* $s_i^l = 0$*. Let* $\tilde{\mathbf{s}}^l$ *be the vector containing the non-zero elements of* $\mathbf{s}^l$*, let* $\tilde{\nabla} f(\tilde{\mathbf{s}}^l)$ *be the corresponding sub-vector of* $\nabla f(\mathbf{s}^l)$*, and let* $\tilde{\mathbf{A}}^l$ *be the matrix containing the columns of* $\mathbf{A}$ *corresponding to the non-zero elements of* $\mathbf{s}^l$*, so that* $\tilde{\mathbf{A}}^l \tilde{\mathbf{s}}^l = \mathbf{x}$*. Then* $\mathbf{s}^l$ *is a fixed point of the mapping* (4.9) *if and only if,*

$$\tilde{\nabla} f(\tilde{\mathbf{s}}^l) \in \mathcal{R}\big((\tilde{\mathbf{A}}^l)^T\big).$$

Proof. If $\lim_{s_i \to 0} s_i / f_i'(s_i) > 0$, then $[W^l]_{ii} > 0, \forall\, i$. Note that $\mathbf{A} W^l \nabla f(\mathbf{s}^l) = \mathbf{x}$, and,

$$\mathbf{s}^{l+1} = W^l \mathbf{A}^T \big(\mathbf{A} W^l \mathbf{A}^T\big)^{-1} \mathbf{A} W^l \nabla f(\mathbf{s}^l)$$

Suppose $\nabla f(\mathbf{s}^l) \in \mathcal{R}(\mathbf{A}^T)$ so that $\nabla f(\mathbf{s}^l) = \mathbf{A}^T \lambda$ for some $\lambda \in \mathbb{R}^m$. Then,

$$\mathbf{s}^{l+1} \;=\; W^l \mathbf{A}^T \lambda \;=\; \mathbf{s}^l \tag{4.10}$$

and $\mathbf{s}^l$ is a fixed point of the mapping (4.9). Now suppose $\mathbf{s}^l$ is a fixed point of (4.9). Then,

$$\nabla f(\mathbf{s}^l) = (W^l)^{-1} \mathbf{s}^{l+1} = \mathbf{A}^T (\mathbf{A} W^l \mathbf{A}^T)^{-1} \mathbf{x}$$

and thus $\nabla f(\mathbf{s}^l) \in \mathcal{R}(\mathbf{A}^T)$.

If $\lim_{s_i \to 0} s_i / f_i'(s_i) = 0$, let $\tilde{W}^l$ be the diagonal matrix with diagonal elements $\tilde{s}_i^l / f_i'(\tilde{s}_i^l)$. Note that $\tilde{\mathbf{A}}^l \tilde{W}^l \tilde{\nabla} f(\tilde{\mathbf{s}}^l) = \mathbf{x}$, and,

$$\tilde{\mathbf{s}}^{l+1} = \tilde{W}^l (\tilde{\mathbf{A}}^l)^T \big(\tilde{\mathbf{A}}^l \tilde{W}^l (\tilde{\mathbf{A}}^l)^T\big)^{-1} \tilde{\mathbf{A}}^l \tilde{W}^l \tilde{\nabla} f(\tilde{\mathbf{s}}^l)$$

Suppose $\tilde{\nabla} f(\tilde{\mathbf{s}}^l) \in \mathcal{R}\big((\tilde{\mathbf{A}}^l)^T\big)$ so that $\tilde{\nabla} f(\tilde{\mathbf{s}}^l) = (\tilde{\mathbf{A}}^l)^T \lambda$ for some $\lambda \in \mathbb{R}^m$. Then,

$$\tilde{\mathbf{s}}^{l+1} \;=\; \tilde{W}^l (\tilde{\mathbf{A}}^l)^T \lambda \;=\; \tilde{\mathbf{s}}^l \tag{4.11}$$

Since the zeros of $\mathbf{s}^l$ are also fixed, we have that $\mathbf{s}^l$ is a fixed point of the mapping (4.9). Now suppose $\mathbf{s}^l$ is a fixed point of (4.9). Then,

$$\tilde{\nabla} f(\tilde{\mathbf{s}}^l) = (\tilde{W}^l)^{-1}\tilde{\mathbf{s}}^{l+1} = (\tilde{\mathbf{A}}^l)^T(\tilde{\mathbf{A}}^l\tilde{W}^l(\tilde{\mathbf{A}}^l)^T)^{-1}\mathbf{x}$$

and thus $\tilde{\nabla} f(\tilde{\mathbf{s}}^l) \in \mathcal{R}\big((\tilde{\mathbf{A}}^l)^T\big)$. □

We now state the main theorem on iteratively reweighted least squares.

Theorem 14. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ be such that every submatrix of m columns of $\mathbf{A}$ is full rank, and let $\mathbf{x} \in \mathbb{R}^m$ be linearly independent of every subset of $m-1$ columns of $\mathbf{A}$. Let $w_i(s_i)$, $i = 1, \ldots, n$, be symmetric, continuous, non-negative, and strictly increasing on $[0, \infty)$, and let W^l be the diagonal matrix with diagonal elements $w_i(s_i^l)$. Then starting from any point $\mathbf{s}^0 \in \mathbb{R}^n$, the algorithm,*

$$\mathbf{s}^{l+1} = W^l\mathbf{A}^T(\mathbf{A}W^l\mathbf{A}^T)^{-1}\mathbf{x} \tag{4.12}$$

converges to a fixed point.

If for all i, $w_i(s_i)/s_i$ is strictly decreasing on $(0, \infty)$, then the algorithm converges to the unique minimizer, $\mathbf{s}^*$, of the function, $\sum_i f_i(s_i)$, subject to the constraint $\mathbf{As} = \mathbf{x}$, where,

$$f_i(s_i) = \int_0^{s_i} t/w_i(t)\,dt \tag{4.13}$$

The convergence rate is at least linear with rate, $r = \max_i s_i^* w'(s_i^*)/w(s_i^*)$.

If for all i, $w_i(s_i)/s_i$ is strictly increasing, then the algorithm converges to a discrete, finite set of stationary points, and only those points with $n-m$ zeros are stable. In this case the algorithm monotonically decreases the function $\sum_i f_i^{(\epsilon)}(s_i)$, where

$$f_i^{(\epsilon)}(s_i) = \int_\epsilon^{s_i} t/w_i(t)\,dt \tag{4.14}$$

for all $0 < \epsilon < s_i$, $i = 1, \ldots, n$. If $\lim_{s_i\to 0} w_i(s_i)/s_i = 0$, then when converging to a stable fixed point, the convergence is superlinear. If $\lim_{s_i\to 0} w_i(s_i)/|s_i|^q < \infty$, then the convergence rate is at least Q-order q.

Proof. To use Zangwill's theorem, we define the solution set to be the set of fixed points of the mapping, and show that the mapping is continuous and bounded, and provide a descent function that is strictly decreased outside the solution set. The boundedness and continuity of the mapping follow from Lemmas 1 and 2 respectively.

To show that $\sum_i f_i(s_i)$ is a descent function for (4.12), with the f_i defined by (4.13), we first show that the function $g_i(t) = f_i(\sqrt{t})$ is strictly concave for $t > 0$. Since $f_i(t) = g_i(t^2)$, we have

$$f_i'(t) = \frac{t}{w_i(t)} = 2t\, g_i'(t^2)$$

so that,

$$g_i'(t^2) = \frac{1}{2w_i(t)}$$

Since $w_i(t)$ is strictly increasing for $t > 0$ by assumption, $g'(t)$ is strictly decreasing for $t > 0$ and thus g is strictly concave for $t > 0$. This implies that,

$$g_i(t^2) - g_i(s^2) \;=\; f_i(t) - f_i(s) \;<\; g_i'(s^2)(t^2 - s^2) \;=\; \frac{1}{2w_i(s)}\,(t^2 - s^2), \quad \forall\, i$$

for all $t \neq s$, $t \neq 0$, $s \neq 0$. Thus, with $W = \operatorname{diag}(w(\mathbf{s}))$,

$$\sum_i f_i(t_i) - \sum_i f_i(s_i) \;<\; \tfrac{1}{2}\mathbf{t}^T W^{-1}\mathbf{t} - \tfrac{1}{2}\mathbf{s}^T W^{-1}\mathbf{s}$$

for all $\mathbf{t} \neq \mathbf{s}$, $t_i \neq 0$, $s_i \neq 0$. If $\mathbf{s}^l$ is not a fixed point, then,

$$f(\mathbf{s}^{l+1}) - f(\mathbf{s}^l) \;<\; \tfrac{1}{2}\mathbf{s}^{l+1}(W^l)^{-1}\mathbf{s}^{l+1} - \tfrac{1}{2}(\mathbf{s}^l)^T(W^l)^{-1}\mathbf{s}^l$$

By Lemma 3, we have that $\mathbf{s}^l$ is a fixed point if and only if $\nabla f(\mathbf{s}^l) \in \mathcal{R}\big(\mathbf{A}^T\big)$. If $w_i(s_i)/s_i = 1/f_i'(s_i)$ is strictly decreasing on $(0, \infty)$, then $f_i(s_i)$ is strictly convex on $(0, \infty)$, and the condition for optimality is that $\nabla f(\mathbf{s}) \in \mathcal{R}(\mathbf{A}^T)$, so we see that the only fixed point is the minimum of $\sum_i f_i(s_i)$ subject to $\mathbf{A}\mathbf{s} = \mathbf{x}$.

To determine the convergence rate in the case where $w_i(s_i)/s_i$ is strictly decreasing for every i, consider the function,

$$Q(\mathbf{s}, \mathbf{s}^l) \;=\; \tfrac{1}{2}\,\mathbf{s}^T(W^l)^{-1}\mathbf{s} \;+\; \tfrac{1}{2\sigma^2}\,\|\mathbf{x} - \mathbf{A}\mathbf{s}\|^2$$

which is a function of two vectors $\mathbf{s}$ and $\mathbf{s}^l$. The gradient of this function with respect to $\mathbf{s}$ is,

$$\nabla_{\mathbf{s}} Q(\mathbf{s}, \mathbf{s}^l) = \big((W^l)^{-1} + \sigma^{-2}\mathbf{A}^T\mathbf{A}\big)\mathbf{s} - \sigma^{-2}\mathbf{A}^T\mathbf{x}$$

Now we perform a Taylor series expansion of this vector valued function about the point $(\mathbf{s}^*, \mathbf{s}^*)$,

$$\begin{aligned} &\nabla_{\mathbf{s}} Q(\mathbf{s}, \mathbf{s}^l) - \nabla_{\mathbf{s}} Q(\mathbf{s}^*, \mathbf{s}^*) = \\ &\qquad \big((W^*)^{-1} + \sigma^{-2}\mathbf{A}\mathbf{A}^T\big)(\mathbf{s} - \mathbf{s}^*) + V^*(\mathbf{s}^l - \mathbf{s}^*) + o(\big\|\mathbf{s}^l - s^*\big\|) \end{aligned} \tag{4.15}$$

Where V^* is the diagonal matrix with diagonal elements,

$$[V^*]_{ii} = -\frac{s_i^* w_i'(s_i^*)}{w_i(s_i^*)^2}$$

If we put,

$$\mathbf{s} \;=\; \mathbf{s}^{l+1} \;=\; W^l\mathbf{A}^T(\mathbf{A}W^l\mathbf{A}^T + \sigma^2\mathbf{I})^{-1}\mathbf{x}$$

then the left hand side of (4.15) is zero, since,

$$\nabla_{\mathbf{s}} Q(\mathbf{s}^{l+1}, \mathbf{s}^l) = 0$$

and

$$\nabla_{\mathbf{s}} Q(\mathbf{s}^*, \mathbf{s}^*) = 0$$

Thus we have,

$$\begin{aligned} \mathbf{s}^{l+1} - \mathbf{s}^* &= \big((W^*)^{-1} + \sigma^{-2}\mathbf{A}\mathbf{A}^T\big)^{-1}V^*(\mathbf{s}^l - \mathbf{s}^*) + o(\big\|\mathbf{s}^l - \mathbf{s}^*\big\|) \\ &= \Big(W^* - W^*\mathbf{A}^T(\mathbf{A}W^*\mathbf{A}^T + \sigma^2\mathbf{I})^{-1}\mathbf{A}W^*\Big)V^*(\mathbf{s}^l - \mathbf{s}^*) + o(\big\|\mathbf{s}^l - \mathbf{s}^*\big\|) \\ &= (W^*)^{\frac{1}{2}}\mathbf{P}^*(W^*)^{-\frac{1}{2}}W^*V^*(\mathbf{s}^l - \mathbf{s}^*) + o(\big\|\mathbf{s}^l - \mathbf{s}^*\big\|) \end{aligned} \tag{4.16}$$

where,

$$\mathbf{P}^* = \mathbf{I} - (W^*)^{\frac{1}{2}}\mathbf{A}^T(\mathbf{A}W^*\mathbf{A}^T + \sigma^2\mathbf{I})^{-1}\mathbf{A}(W^*)^{\frac{1}{2}}$$

If we let $\sigma \longrightarrow 0$, then $\mathbf{P}^*$ is a projection matrix onto the null space of $\mathbf{A}(W^*)^{\frac{1}{2}}$, and thus $\lim_{\sigma\to 0}\|\mathbf{P}^*\| = 1$. Thus, taking norms of both sides of (4.16) and dividing by

$\left\|\mathbf{s}^l - \mathbf{s}^*\right\|$, we get,

$$\lim_{l\to\infty} \frac{\left\|\mathbf{s}^{l+1} - \mathbf{s}^*\right\|}{\left\|\mathbf{s}^l - \mathbf{s}^*\right\|} \le \left\|(W^*)^{\frac{1}{2}}\right\| \left\|(W^*)^{-\frac{1}{2}}\right\| \left\|W^*V^*\right\| = \left\|W^*V^*\right\| = \max_i \frac{s_i^* w_i'(s_i^*)}{w_i(s_i^*)}$$

If $w_i(s_i)/s_i$ is strictly increasing for every i, then f_i defined by (4.14) is strictly concave on (ϵ, ∞) for all $\epsilon > 0$, and thus $\sum f_i(\tilde{s}_i)$ subject to $\mathbf{A}\tilde{\mathbf{s}} = \mathbf{x}$ and $\tilde{\mathbf{s}} \in \tilde{\mathcal{O}}_j$ for some reduced orthant $\tilde{\mathcal{O}}_j$ has a unique maximum on the bounded part of $\tilde{\mathcal{C}}_j$, the intersection of $\tilde{\mathcal{O}}_j$ and $\tilde{\mathcal{C}}$ defined in Lemma 1. This point is characterized by the condition, $\tilde{\nabla} f(\tilde{\mathbf{s}}^l) \in \mathcal{R}\big((\tilde{\mathbf{A}}^l)^T\big)$. Since f is strictly concave on the interior of each orthant, the stationary point is a local maximum in the reduced space, and perturbations show that it is unstable under the mapping (4.12), which causes f to strictly decrease. If $\mathbf{s}$ has only m non-zero elements, then $\tilde{\nabla} f(\tilde{\mathbf{s}}^l) \in \mathcal{R}\big((\tilde{\mathbf{A}}^l)^T\big)$ since $(\tilde{\mathbf{A}}^l)^T$ is square and non-singular by assumption. In the case of concave f_i, all other stationary points are in the strict interior of their reduced orthant, and a "basic feasible" solution with only m non-zero values cannot jump into the reduced orthant interior, but must follow a continuous path. Hence, the basic feasible solutions are local minima of the

To determine the convergence rate in the case where $w_i(s_i)/s_i$ is strictly increasing, first note that when $n - m$ elements of $\mathbf{s}$ converge to zero, the convergence rate is determined by the convergence rate of the elements converging to zero. Specifically, if $\mathbf{s}_1$ denotes the elements of $\mathbf{s}$ converging to non-zero values, and $\mathbf{s}_2$ denotes the elements converging to zero, then

$$\mathbf{A}_1\mathbf{s}_1 + \mathbf{A}_2\mathbf{s}_2 = \mathbf{x}$$

and,

$$\mathbf{s}_1 = \mathbf{A}_1^{-1}(\mathbf{x} - \mathbf{A}_2\mathbf{s}_2) = \mathbf{s}_1^* - \mathbf{A}_1^{-1}\mathbf{A}_2\mathbf{s}_2$$

so that,

$$\left\|\mathbf{s}_1 - \mathbf{s}_1^*\right\| = \left\|\mathbf{A}_1^{-1}\mathbf{A}_2\mathbf{s}_2\right\| \le \left\|\mathbf{A}_1^{-1}\mathbf{A}_2\right\| \left\|\mathbf{s}_2\right\| = \left\|\mathbf{A}_1^{-1}\mathbf{A}_2\right\| \left\|\mathbf{s}_2 - \mathbf{s}_2^*\right\|$$

Thus s_1 converges to s_1^* at the same rate that $\mathbf{s}_2$ converges to $\mathbf{s}_2^* = 0$. Now note that,

$$s_i^{l+1} = w_i(s_i^l)\, \mathbf{a}_i^T (\mathbf{A} W^l \mathbf{A}^T)^{-1} \mathbf{x}$$

where $\mathbf{a}_i$ is the ith column of $\mathbf{A}$. If $s_i^* = 0$, then,

$$|s_i^{l+1} - s_i^*| = w_i(s_i^l - s_i^*) \left| \mathbf{a}_i^T (\mathbf{A} W^l \mathbf{A}^T)^{-1} \mathbf{x} \right|$$

Since $\lim_{s \to 0} w_i(s_i)/s_i = 0$ in the case we are considering, we have,

$$\lim_{l \to \infty} \frac{|s_i^{l+1} - s_i^*|}{|s_i^l - s_i^*|} = 0$$

and the zeros converge superlinearly. If $w_i(s)$ is $o(|s|^{q_i})$, $q_i > 1$, then

$$\lim_{l \to \infty} \frac{|s_i^{l+1} - s_i^*|}{|s_i^l - s_i^*|^{q_i}} = 0$$

for all s_i converging to 0, and the zeros converge superlinearly with Q-order at least $\min_i q_i$. □

4.2.1 Examples

1. Generalized Gaussian: For $w_i(s_i) = |s_i|^{2-p_i}$ with $1 < p_i < 2$, $i = 1, \ldots, n$, we have $w_i(s_i)/s_i$ strictly decreasing on $(0, \infty)$. Thus the convergence is linear with convergence rate at least $\max_i (2 - p_i)$.

   If $0 < p_i < 1$, $i = 1, \ldots, n$, then $w_i(s_i)/s_i$ is strictly increasing on $(0, \infty)$, and $w_i(s_i)$ is at least $o(|s_i|^{2-p_{\max}})$, so the convergence is superlinear with Q-order at least $\max_i (2 - p_i)$, as determined in [89].

   Note that in both cases the convergence rate is independent of the particular solution to which the algorithm is converging.

2. Logistic: For $w_i(s_i) = s_i / \tanh(s_i)$, we have $w_i(s_i)/s_i$ strictly decreasing on $(0, \infty)$, so the convergence is linear. The convergence rate is at least $\max_i s_i^* / \sinh(s_i^*)$.

Dempster, Laird, and Rubin [31] also developed the theory of convergence rate of EM, and applied this to the ML estimation of the regression vector in the linear model with Gaussian scale mixture errors [32].

4.3 Newton's Method and the Dual MAP Problem

We have seen that when the descent function $f(s)$ is convex as well as concave in s^2, the convergence rate ot the IRLS algorithm of §4.2 is linear. In the case of the ρ-norms, $1 < \rho < 2$, which are associated with the Generalized Gaussian density, the convergence rate is $2-\rho$. The motivation for developing this IRLS algorithm was the fact that the ρ-norms are not twice differentiable at the origin, and thus Newton's method is unstable when the solution has components near zero.

It is possible, however, in the case of convex descent functions $f(s)$, to formulate the dual optimization problem, which involves the conjugate function,

$$f^*(\phi) = \sup_s \phi s - f(s)$$

Assume the $f(s)$ is differentiable at the origin, and $f''(s)$ tends continuously toward infinity at $s \to 0$. Since,

$$\frac{d^2}{ds^2} f(s) = \left(\frac{d^2}{d\phi^2} f^*(\phi(s)) \right)^{-1}$$

where $\phi(s) = f'(s)$, the conjugate function will have a bounded continuous second derivative at the origin, and thus will be amenable to Newton's method. This idea was proposed for the ρ-norms by Fischer in [39].

Newton's theorem is given in the following, taken from Ortega and Rheinboldt [80].

Theorem 15. *Assume that ∇f is Gateaux differentiable in an open neighborhood of $\mathbf{s}^*$, where $\nabla f(\mathbf{s}^*) = 0$, and that the Hessian matrix $H_f(\mathbf{s})$ is nonsingular and continuous at $\mathbf{s}^*$. Then $\mathbf{s}^*$ is a point of attraction for the Newton iteration,*

$$\mathbf{s}^{l+1} = \mathbf{s}^l - H_f(\mathbf{s}^l)^{-1} \nabla f(\mathbf{s}^l)$$

and the convergence is superlinear. If in addition there are constants $\alpha < \infty$ and $p \in (0, 1]$ such that,

$$\| H_f(\mathbf{s}) - H_f(\mathbf{s}^*) \| \leq \alpha \, \| \mathbf{s} - \mathbf{s}^* \|^p$$

the the convergence rate is at least Q-order $1+p$*. If* ∇F *is continuously differentiable in an open neighborhood of* $\mathbf{s}^*$*, and the second Frechet derivative of* ∇f *exists at* $\mathbf{s}^*$ *and the Hessians of the components of* ∇f *are non-singular at* $\mathbf{s}^*$*, then the convergence is Q-order 2 (quadratic).*

The dual problem to the MAP estimation problem,

$$\min_{\mathbf{s}} f(\mathbf{s}) \quad s.t. \quad \mathbf{A}\mathbf{s} = \mathbf{x}$$

is readily formulated using Fenchel duality [92, 15]. Specifically, we have,

$$\min_{\mathbf{A}\mathbf{s}=\mathbf{x}} f(\mathbf{s}) = \max_{\lambda} \lambda^T \mathbf{x} - f^*(\mathbf{A}^T \lambda)$$

The Newton iteration for the dual problem is,

$$\lambda^{l+1} = \lambda^l - \big(\mathbf{A} H_{f^*}(\mathbf{A}^T \lambda)\mathbf{A}^T\big)^{-1}\big(\mathbf{A}\nabla f^*(\mathbf{A}^T \lambda^l - \mathbf{x}\big)$$

where we have $H_{f^*}(\phi) = H_f^{-1}(\phi)$, and $f^{*\prime}(\phi) = f'^{-1}(\phi)$, i.e. the Hessians of conjugate functions are inverse matrices, and the gradients are inverse functions (they are univalent since f and f^* are convex.) Given the solution to the dual problem λ^*, the primal solution $\mathbf{s}^*$ is given by,

$$\mathbf{s}^* = \nabla f^*\big(\mathbf{A}^T \lambda^*\big)$$

The dual algorithm will have superlinear convergence, which is certainly more desirable than the linear convergence of the IRLS algorithm of §4.2. However, our ultimate goal is to derive algorithms for optimizing bases $\mathbf{A}$ given a set of observations $\mathbf{x}_1, \ldots, \mathbf{x}_N$. In this case, if we want, say, the MAP estimate of $\mathbf{A}$ and $\mathbf{s}_1, \ldots, \mathbf{s}_N$ given $\mathbf{x}_1, \ldots, \mathbf{x}_N$, then formulating the dual with respect to the $\mathbf{s}_k$ makes the problem minimization over the $\mathbf{s}_k$ and maximization over $\mathbf{A}$, and there is no descent function that can be used to control convergence. Ideally we could formulate the dual problem with respect to $\mathbf{A}$ as well, but this does not seem to be expressible in a simple form for general non-Gaussian priors on $\mathbf{s}$. Hence, despite its relatively slow convergence rate, the IRLS iteration on §4.2 will prove useful in dictionary learning and Independent Component Analysis in subsequent chapters.

4.4 Kernel Regression

In this section we show how the algorithm for estimation in the linear model can be used for estimation in the kernel non-linear regression. The development is similar to that in [54], where generalized kernel machines are defined, and an algorithm is developed to solve the primal classification problem for the case of the logistic link function.

Let $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$ be point and function value pairs observed from a nonlinear function $y(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$. Consider a linear approximation of y given by,

$$\mathbf{w}^T\mathbf{x} + b = y\,, \qquad \mathbf{w}^T\mathbf{w} \le A^2$$

Let $\mathbf{\Phi}$ be the matrix with $\mathbf{x}_i$ in column i, and let $z_i = \mathbf{w}^T\mathbf{x}_i + b - y_i$ denote the residuals. In kernel methods, the $\mathbf{x}_i$ are possibly infinite dimensional "feature" vectors residing in feature space, but whose inner products with each other can be calculated in observation space using the kernel.

We define a probabilistic model in which z is random with symmetric density $p(z)$, and attempt to find parameters $\mathbf{w}$ and b that minimize the negative log likelihood $-\sum \log p(\mathbf{z}_i) \equiv d(\mathbf{z})$ of the samples,

$$\min_{\mathbf{z},\mathbf{w},b} \; d(\mathbf{z}) \quad s.t. \quad \mathbf{z} = \mathbf{\Phi}^T\mathbf{w} + b\,\mathbf{e} - \mathbf{y}\,, \quad \mathbf{w}^T\mathbf{w} \le A^2$$

where $\mathbf{e}$ is the vector of all 1's. This is a concave program. For the Lagrangian, we have,

$$L(\mathbf{z}, \mathbf{w}, b, \lambda, \mu) \;=\; d(\mathbf{z}) \;+\; \lambda^T\big(\mathbf{\Phi}^T\mathbf{w} + b\,\mathbf{e} - \mathbf{y} - \mathbf{z}\big) \;+\; \mu\,(\mathbf{w}^T\mathbf{w} - A^2)$$

Minimizing over $\mathbf{w}$ and $\mathbf{z}$ and treating b as a Lagrange multiplier, we get the dual problem,

$$\min_{\lambda \in \partial f} \min_{\mu \ge 0} \; \frac{1}{4\mu}\,\lambda^T\mathbf{K}\,\lambda \;-\; \mathbf{y}^T\lambda \;+\; d^*(\lambda) \;+\; A^2\mu \quad s.t. \quad \mathbf{e}^T\lambda = 0 \tag{4.17}$$

where $\mathbf{K} = \mathbf{\Phi}^T\mathbf{\Phi}$. Substituting the optimal $\mu_{\min} = \frac{1}{2A}\,\|\lambda\|_{\mathbf{K}}$, we get,

$$\min_{\lambda \in \partial f} \; A\,\|\lambda\|_{\mathbf{K}} \;-\; \mathbf{y}^T\lambda \;+\; d^*(\lambda) \quad s.t. \quad \mathbf{e}^T\lambda = 0$$

The requirement for formulation of the dual problem is that $d(z)$ be convex.

4.4.1 General algorithm for kernel regression.

The dual of the general regression problem in terms of λ and μ is,

$$\min_{\lambda\in\partial f}\min_{\mu\geq 0}\ \frac{1}{4\mu}\lambda^T\mathbf{K}\,\lambda\,-\,\mathbf{y}^T\lambda\,+\,d^*(\lambda)\,+\,A^2\mu\quad s.t.\quad \mathbf{e}^T\lambda=0 \tag{4.18}$$

From §3.5, we know that when $d(z)$ is convex in z^2, $d^*(\lambda)$ is concave in λ^2, and the following algorithm can be used to minimize (4.18). Given μ^l, let $\mathbf{Q}^l=\frac{1}{2\mu^l}\mathbf{K}+\mathbf{\Pi}(\lambda^l)$, and set,

$$\lambda^{l+1}\ =\ \arg\min_{\lambda\in\partial f}\ \tfrac{1}{2}\,\lambda^T\mathbf{Q}^l\,\lambda\,+\,\mathbf{y}^T\lambda\quad s.t.\quad \mathbf{e}^T\lambda=0$$

and,

$$\mu^l=\tfrac{1}{2}\,\|\lambda_k\|_{\mathbf{K}}$$

When range $\partial f=\mathbb{R}$, the algorithm reduces to,

$$\lambda^l=\left((\mathbf{Q}^l)^{-1}-\frac{(\mathbf{Q}^l)^{-1}\mathbf{e}\mathbf{e}^T(\mathbf{Q}^l)^{-1}}{\mathbf{e}^T(\mathbf{Q}^l)^{-1}\mathbf{e}}\right)\mathbf{y}\ \ =\ \ \mathbf{r}-\frac{\mathbf{e}^T\mathbf{r}}{\mathbf{e}^T\mathbf{p}}\,\mathbf{r}$$

where $\mathbf{r}$ and $\mathbf{p}$ are defined by the equations $\mathbf{Q}^l\mathbf{r}=\mathbf{y}$ and $\mathbf{Q}^l\mathbf{p}=\mathbf{e}$ respectively.

For robust regression (in the primal space), however, we want $d(z)$ to concave in z^2, so that $d^*(\lambda)$ is convex in λ^2. In this case we can update λ with a standard Newton step, replacing $\mathbf{\Pi}(\lambda^l)$ by $\mathbf{H}_{f^*}(\lambda^l)$, however we must in principle impose safeguards in the optimization to ensure decrease of the objective.

4.4.2 Examples

1. The indicator function $d(z)=0$, $|z|\leq\delta,\ d(z)=\infty$, $|z|>\delta$, has,

$$d^*(\lambda)=\delta|\lambda|\quad \lambda\in\mathbb{R}$$

2. Vapnik's ϵ-insensitive loss function $d(z)=0$, $|z|<=\epsilon,\ d(z)=C(|z|-\epsilon)$, $|z|>\epsilon$, has,

$$d^*(\lambda)=\begin{cases}\delta\lambda & |\lambda|\leq C\\ \infty & |\lambda|>C\end{cases}$$

3. The ϵ-insensitive quadratic loss function $d(z) = 0$, $|z| <= \epsilon$, $d(z) = \frac{1}{2\sigma^2}(|z| - \epsilon)^2$, $|z| > \epsilon$, has,

$$d^*(\lambda) = \frac{1}{2}\sigma^2\lambda^2 + \delta\,|\lambda| \quad \lambda \in \mathbb{R}$$

   For this function, range $\partial f = \mathrm{dom} f^* = \mathbb{R}$, and (4.17) can be optimized for λ using a relatively simple iterative reweighted least squares algorithm. As in the similar classification example, this function is less robust to outliers than the asymptotically linear functions.

4. In general, if $d(z)$ is symmetric with $d(0) = 0$, and $\tilde{d}(z)$ is defined by,

$$\tilde{d}(z) = \begin{cases} 0 & |z| \leq \epsilon \\ d(|z| - \epsilon) & |z| > \epsilon \end{cases}$$

   then the conjugate is given by,

$$\tilde{d}^*(\lambda) = d^*(\lambda) + \epsilon|\lambda|$$

5. The loss function $d(z)$ corresponding to the negative logarithm of the logistic derivate, given by $d(z) = -\log s - \log(1 - s) - \log 4$, where $s = (1 + \exp(-z))^{-1}$, has,

$$d^*(\lambda) = \begin{cases} -\log(1 - |\lambda|) & |\lambda| \leq 1 \\ \infty & |\lambda| > 1 \end{cases}$$

   Since $d(z)$ is asymptotically linear, it is robust to outliers.

6. Huber's loss function, $d(z) = \frac{1}{2}z^2$, $|z| \leq c$, $d(z) = c|z| - c^2/2$, $|z| > c$, has,

$$d^*(\lambda) = \begin{cases} \frac{1}{2}\lambda^2 & |\lambda| \leq c \\ \infty & |\lambda| > c \end{cases}$$

   This function is also robust to outliers.

5

Dictionary Learning

This chapter considers learning overcomplete data representations based on a linear generative model [36, 79, 67, 68, 50, 24, 63, 43]. Given observations $\mathbf{X} = [\mathbf{x}_1 \ldots \mathbf{x}_N]$, the problem is to estimate the parameters $A \in \mathbb{R}^{m\times n}$ and $\mathbf{S} = [\mathbf{s}_1 \ldots \mathbf{s}_N]$ in the Bayesian linear model,

$$\mathbf{x}_k = \mathbf{A}\mathbf{s}_k + \nu_k, \quad k = 1, \ldots, N \tag{5.1}$$

assuming that the sources $\mathbf{x}_k$ are independent. The low noise limit is equivalent to the case in which the noise random variables ν_k are not present. Since Field [36], much consideration has been given to representations that assume sparse and distributed sources, i.e. many source components with relatively few of the components having significant magnitude, or "active", at any given time. One way to ensure a sparse representation is to take A to be "overcomplete", or have more columns than rows. However, sparse coding can also be carried out when the matrix A is not overcomplete, for example when the data is high dimensional but occupies a relatively low dimensional manifold [69].

5.1 The model

We assume the standard Bayesian linear model, with a set of observations $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_N]$, $\mathbf{x}_k \in \mathbb{R}^m$, generated according to,

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \nu \tag{5.2}$$

where $\mathbf{A} = [\mathbf{a}_1 \cdots \mathbf{a}_n]$ contains the basis vectors, $\mathbf{s}$ is the random vector of coefficients, and ν is a random noise vector. Results from the standard Gaussian case should be familiar and will be used repeatedly in the development of the non-gaussian algorithms. In particular, if $\mathbf{s} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{s}})$ and $\nu \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_\nu)$ are uncorrelated, then for the likelihood of $\mathbf{A}$ we have,

$$p(\mathbf{x}; \mathbf{A}) \;=\; \int p(\mathbf{x}|\mathbf{s}; \mathbf{A})\, p(\mathbf{s})\, d\mathbf{s} \;=\; \mathcal{N}\big(\mathbf{0}, \mathbf{\Sigma}_\nu + \mathbf{A}\mathbf{\Sigma}_{\mathbf{s}}\mathbf{A}^T\big) \tag{5.3}$$

and for the posterior of $\mathbf{s}$ given $\mathbf{x}$, we have

$$p(\mathbf{s}|\mathbf{x}; \mathbf{A}) = \frac{p(\mathbf{x}|\mathbf{s}; \mathbf{A})\, p(\mathbf{s})}{p(\mathbf{x}; \mathbf{A})} = \mathcal{N}(\mu_{\mathbf{s}|\mathbf{x}}, \mathbf{\Sigma}_{\mathbf{s}|\mathbf{x}}) \tag{5.4}$$

where,

$$\mu_{\mathbf{s}|\mathbf{x}} = \mathbf{\Sigma}_{\mathbf{s}}\mathbf{A}^T(\mathbf{A}\mathbf{\Sigma}_{\mathbf{s}}\mathbf{A}^T + \Sigma_\nu)^{-1}\mathbf{x}\,, \qquad \Sigma_{\mathbf{s}|\mathbf{x}} = (\mathbf{A}^T\mathbf{\Sigma}_\nu^{-1}\mathbf{A} + \mathbf{\Sigma}_{\mathbf{s}}^{-1})^{-1} \tag{5.5}$$

The standard EM update for $\mathbf{A}$ is,

$$\mathbf{A} = \left(\sum_{k=1}^{N} \mathbf{y}\mu_{\mathbf{s}_k|\mathbf{x}_k}^T\right)\left(\sum_{k=1}^{N} \mu_{\mathbf{s}_k|\mathbf{x}_k}\mu_{\mathbf{s}_k|\mathbf{x}_k}^T + \mathbf{\Sigma}_{\mathbf{s}_k|\mathbf{x}_k}\right)^{-1} \tag{5.6}$$

We shall be interested primarily in the batch estimation problem: given N independent observations $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_N]$ from the model (5.2), determine the Maximum Likelihood estimate of $\mathbf{A}$. We take $\mathbf{A}$ to be non-random, possibly constrained to a compact set. Letting $\mathbf{S} = [\mathbf{s}_1 \cdots \mathbf{s}_N]$, the likelihood can be written,

$$p(\mathbf{X}; \mathbf{A}) = \int p(\mathbf{X}|\mathbf{S}; \mathbf{A})\, p(\mathbf{S})\, d\mathbf{S} = \prod_{k=1}^{N} \int p(\mathbf{x}_k|\mathbf{s}_k; \mathbf{A})\, p(\mathbf{s}_k)\, d\mathbf{s}_k \tag{5.7}$$

For non-gaussian component or noise priors, the integral does not exist in closed form. The algorithms described in this paper approach this problem in different ways. We shall assume for simplicity of exposition that the noise is distributed $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_\nu)$, and only the component priors are non-gaussian, though the more general case is easily handled.

5.2 Laplace approximations and super-gaussian priors

The first order Laplace approximation [68], [91, p. 103] of an integral $\int F(\mathbf{x})\, d\mathbf{x}$ on $\mathbb{R}^n$ is given by,

$$\int F(\mathbf{s})\, d\mathbf{s} \;\approx\; (2\pi)^{\frac{n}{2}} F(\hat{\mathbf{s}}) \left| \det H_f(\hat{\mathbf{s}}) \right|^{-\frac{1}{2}}$$

where $f(\mathbf{s}) \equiv -\log F(\mathbf{s})$, $H_f(\mathbf{s})$ is the Hessian of $f(\mathbf{s})$, and $\hat{\mathbf{s}}$ is a maximum of $F(\mathbf{s})$ such that $\nabla F(\hat{\mathbf{s}}) = \mathbf{0}$. This approximation is derived by expanding $-\log F(\mathbf{s}) = f(\mathbf{s})$ in a second order Taylor series about the mode of $F(\mathbf{x})$.

Now, for a particular integral in (5.7), if we define $f(\mathbf{s}) = -\log p(\mathbf{s})$, and define $H_f(\mathbf{s})$ to be the Hessian of f, then,

$$-\log p(\mathbf{x}|\mathbf{s};\mathbf{A})\, p(\mathbf{s}) \;=\; \frac{1}{2} \left\| \mathbf{x} - \mathbf{A}\mathbf{s} \right\|^2_{\mathbf{\Sigma}_\nu^{-1}} + f(\mathbf{s}) + \text{const.} \tag{5.8}$$

and the Hessian is $\mathbf{A}^T \mathbf{\Sigma}_\nu^{-1} \mathbf{A} + H_f(\mathbf{s})$. The Laplace approximation is then,

$$p(\mathbf{x};\mathbf{A}) \;\approx\; (2\pi)^{\frac{n}{2}} \cdot p(\mathbf{x}|\hat{\mathbf{s}};\mathbf{A}) \cdot p(\hat{\mathbf{s}}) \cdot \left| \det\bigl(\mathbf{A}^T \mathbf{\Sigma}_\nu^{-1} \mathbf{A} + H_f(\hat{\mathbf{s}}) \bigr) \right|^{-\frac{1}{2}}$$

The idea of the Lewicki-Sejnowski algorithm is to minimize the negative logarithm of this approximation, which is proportional to,

$$\frac{1}{2} \left\| \mathbf{x} - \mathbf{A}\hat{\mathbf{s}} \right\|^2_{\mathbf{\Sigma}_\nu^{-1}} + f(\hat{\mathbf{s}}) + \frac{1}{2} \log \left| \det\bigl(\mathbf{A}^T \mathbf{\Sigma}_\nu^{-1} \mathbf{A} + H_f(\hat{\mathbf{s}}) \bigr) \right| \tag{5.9}$$

by natural gradient descent [1] in $\mathbf{A}$, concurrently updating $\hat{\mathbf{s}}$ to minimize (5.8).

However, Strongly super-Gaussian densities may not be twice differentiable on $\mathbb{R}^n$. For example, Generalized Gaussian densities with shape parameter $\rho < 2$ are not twice differentiable at the origin. Thus the Laplace approximation is may be unstable when used with these densities.

5.3 Lewicki-Sejnowski algorithm

The Laplace approximation is used in [68] to derive and algorithm for estimating $\mathbf{A}$. The derivation given here differs somewhat from that given in [68, app. A]. For the natural gradient (see [1], also §6.2.3 below) of (5.9) with respect to $\mathbf{A}$, we have,

$$\mathbf{A}\mathbf{A}^T \frac{\partial}{\partial \mathbf{A}}(\cdot) = \mathbf{A}\mathbf{A}^T \Big(-\boldsymbol{\Sigma}_\nu^{-1}(\mathbf{x} - \mathbf{A}\hat{\mathbf{s}})\, \hat{\mathbf{s}}^T - \boldsymbol{\Sigma}_\nu^{-1}\mathbf{A}\big(\mathbf{A}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A} + H_f(\hat{\mathbf{s}})\big)^{-1}\Big) \tag{5.10}$$

Note that at a stationary point of (5.8), we have,

$$\nabla f(\hat{\mathbf{s}}) = \mathbf{A}^T\boldsymbol{\Sigma}_\nu^{-1}(\mathbf{x} - \mathbf{A}\hat{\mathbf{s}}) \tag{5.11}$$

Substituting this into (5.10), we get,

$$\begin{aligned} \mathbf{A}\mathbf{A}^T \frac{\partial}{\partial \mathbf{A}}(\cdot) &= -\mathbf{A}\nabla f(\hat{\mathbf{s}})\, \hat{\mathbf{s}}^T - \mathbf{A}\mathbf{A}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A}\big(\mathbf{A}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A} + H_f(\hat{\mathbf{s}})\big)^{-1} \\ &= -\mathbf{A}\big(\nabla f(\hat{\mathbf{s}})\, \hat{\mathbf{s}}^T + \mathbf{I}\big) \; + \; \mathbf{A}H_f(\hat{\mathbf{s}})\big(\mathbf{A}^T\boldsymbol{\Sigma}_\nu^{-1}\mathbf{A} + H_f(\hat{\mathbf{s}})\big)^{-1} \end{aligned} \tag{5.12}$$

Arguments are made in [68] to the effect that in the low noise case, the second term in (5.12) tends to zero. Neglecting the second term in (5.12), we have the following algorithm, where we update $\hat{\mathbf{s}}$ using the generalized FOCUSS iteration, which is stable when the Newton update (suggested by [68]) is not.

```
Lewicki-Sejnowski algorithm
Choose $\mathbf{A}^0 \in \mathbb{R}^{m\times n}$, $\alpha \in (0,1)$. Set $\mathbf{s}_k^0 = (\mathbf{A}^0)^+\mathbf{y}_k\,,\ k = 1,\ldots,N$.
for $l = 0,1,2,\ldots,$
    for $k = 1,\ldots,N$
        $[W_k^l]_{ii} = [\mathbf{s}_k^l]_i/[\nabla f(\mathbf{s}_k^l)]_i, \quad i = 1,\ldots,n$
        $\mathbf{s}_k^l = W_k^l(\mathbf{A}^l)^T(\mathbf{A}^l W_k^l(\mathbf{A}^l)^T + \boldsymbol{\Sigma}_\nu)^{-1}\mathbf{y}_k\,,\ k = 1,\ldots,N$
    end
    $\mathbf{A}^{l+1} = (1-\alpha)\mathbf{A}_l + \alpha\sum_{k=1}^N \nabla f(\mathbf{s}_k^l)\,(\mathbf{s}_k^l)^T$
end
```

The rule is derived for the case of one observation, and details are left open as to the best way to implement the algorithm in batch or adaptive mode. There is also some freedom in the implementation of finding $\hat{\mathbf{s}}$, where the optimization may not be performed completely, but only one or two iterations made.

The Hessian of super-gaussian priors

When optimizing an arbitrary function $f(x)$ using a sequential quadratic algorithm, one naturally thinks first to try Newton's method, using the Hessian of f as the quadratic weighting matrix, at least in a neighborhood of the optimum. However, the application Newton's method requires the stability of second derivatives. As it happens, the peakedness characteristic of super-gaussian distributions may well be at odds with the boundedness of the Hessian. For example, consider the Generalized Gaussian, or Exponential Power family, in which $p(\mathbf{s}) \propto \exp(-\sum |s|^{\rho})$. When $\rho < 2$, this density becomes super-gaussian, but it also loses twice-differentiability at the origin.

5.4 Lagrangian MAP

This approach can be seen as forming a joint MAP estimate of $\mathbf{A}$ and $\mathbf{S}$. The approximation is then,

$$\arg\max_{\mathbf{A}} p(\mathbf{X}|\mathbf{A}) \;\approx\; \arg\max_{\mathbf{A}} \max_{\mathbf{S}} p(\mathbf{S}|\mathbf{X};\mathbf{A})$$

With the constraint $\|\mathbf{a}_i\| \leq 1$ for $i = 1, \dots, n$, the Lagrangian is,

$$L(\mathbf{A}, \mathbf{S}, \lambda, \mu) = \sum_{k=1}^{N} \Big[f(\mathbf{s}_k) + \lambda_k^T (\mathbf{x}_k - \mathbf{A}\mathbf{s}_k) \Big] + \sum_{i=1}^{n} \mu_i (\mathbf{a}_i^T \mathbf{a}_i - 1) \tag{5.13}$$

where $\mathbf{A} = [\mathbf{a}_1 \dots \mathbf{a}_n]$. The algorithm can also be seen as a Newton method for finding a stationary point of the Lagrangian using sequentially updated estimates of the Lagrange multipliers. The gradient of (5.13) with respect to $\mathbf{A}$ is,

$$-\sum_{k=1}^{N} \lambda_k \mathbf{s}_k^T + \mathbf{A}\,\mathrm{diag}(\mu) \tag{5.14}$$

The Hessian operator is $\mathrm{diag}(\mu)$ multiplied on the right, and the inverse Hessian operator is $\mathrm{diag}(\mu)^{-1}$ multiplied on the right. Thus for the Newton direction, given the estimates $\hat{\lambda}_k, \hat{\mathbf{s}}_k$, $k = 1, \ldots, N$, and $\hat{\mu}$, we have,

$$\Delta\mathbf{A} = -\left(\sum_{k=1}^{N} \hat{\lambda}_k \hat{\mathbf{s}}_k^T\right) \mathrm{diag}(\hat{\mu})^{-1} + \mathbf{A} \tag{5.15}$$

A general approximate Newton algorithm for finding a stationary point of the Lagrangian with respect to $\mathbf{A}$ is thus defined by the following.

```
Lagrangian Newton algorithm
Choose $\mathbf{A}^0 \in \mathbb{R}^{m \times n}$, $\alpha \in (0,1)$. Set $\mathbf{s}_k^0 = (\mathbf{A}^0)^+ \mathbf{x}_k\,,\ k = 1, \ldots, N$.
for $l = 0, 1, 2, \ldots,$
        for $k = 1, \ldots, N$
                $\left[W_k^l\right]_{ii} = \left[\mathbf{s}_k^l\right]_i / \left[\nabla f(\mathbf{s}_k^l)\right]_i, \ \ i = 1, \ldots, n$
                $\lambda_k^l = (\mathbf{A}^l W_k^l (\mathbf{A}^l)^T + \mathbf{\Sigma}_\nu)^{-1} \mathbf{x}_k$
        end
        $\mu^l = \sum_{k=1}^{N} \mathbf{s}_k^l \odot (\mathbf{A}^l)^T \lambda_k^l$
        $\mathbf{A}^{l+1} = (1 - \alpha)\mathbf{A}^l + \alpha \left(\sum_{k=1}^{N} \lambda_k^l (\lambda_k^l)^T \mathbf{A}^l W_k^l\right) \mathrm{diag}(\mu^l)^{-1}$
        $\mathbf{s}_k^{l+1} = W_k^l (\mathbf{A}^{l+1})^T (\mathbf{A}^{l+1} W_k^l (\mathbf{A}^{l+1})^T + \mathbf{\Sigma}_\nu)^{-1} \mathbf{x}_k\,, \ \ k = 1, \ldots, N$
end
```

The symbol $\odot$ indicates component-wise multiplication.

The Lagrangian Newton algorithm was derived in [81] with the goal of formulating a joint iteration over $\mathbf{A}$ and $\mathbf{S}$ to increase the joint posterior likelihood. The difficulty encountered involves the attempt to update $\mathbf{A}$ and $\mathbf{S}$ such that $\mathbf{AS} = \mathbf{X}$, and the joint posterior likelihood is increased. This leads to the sort of double computation of λ each iteration, where in a sense, source predictions are computed using the current $\mathbf{A}$, then $\mathbf{A}$ is updated using these predictor sources, and finally the sources are updated using the new $\mathbf{A}$ to guarantee feasibility. In fact the algorithm seems to work when the algorithm is modified to update the sources only once, despite the lack of guarantee that the new feasible **A** and **S** will increase the likelihood. In our comparison tests, we also make two iterations of the source updates in the Lewicki-Sejnowski algorithm and the Kreutz-Delgado FOCUSS based algorithms to ensure uniformity in the comparison.

5.5 Variational Bayes

For a strongly super-gaussian prior $p(s) = \exp(-f(s))$, we have $f(s) = g(s^2)$ with g concave and increasing on $(0, \infty)$. By definition of the concave conjugate [92], we have $f(s) = g(s^2) = \inf_\xi \; \xi s^2/2 - g^*(\xi/2)$, and thus,

$$p(s) \;=\; \exp(-f(s)) \;=\; \sup_\xi \; \exp\Big(-\frac{\xi}{2}\,s^2\Big)\exp\Big(g^*\Big(\frac{\xi}{2}\Big)\Big) \;=\; \sup_\xi \; \mathcal{N}(s\,; 0, \xi^{-1})\,h(\xi)$$

Using this to perform the integration as in (5.3), we have the approximation,

$$p(\mathbf{x}; \mathbf{A}) \;\approx\; \sup_\xi \mathcal{N}\big(\mathbf{x}; \mathbf{0}, \mathbf{\Sigma} + \mathbf{A}\mathbf{\Lambda}^{-1}\mathbf{A}^T\big)\prod_{i=1}^n \varphi(\xi_i) \;\equiv\; \sup_\xi \; L(\mathbf{x}; \mathbf{A}, \xi)$$

An EM-type iteration is performed to maximize this lower bound with respect to the parameter vector ξ, or equivalently minimize an upper bound on the free energy,

$$-\log L(\mathbf{x}, \mathbf{s}; \xi) \propto \frac{1}{2}\,\|\mathbf{x} - \mathbf{A}\mathbf{s}\|^2_{\Sigma^{-1}} + \sum_{i=1}^n \left[\frac{\xi_i}{2}s_i^2 - g^*\Big(\frac{\xi_i}{2}\Big)\right] \tag{5.16}$$

Taking the expected value with respect to the approximate posterior $\mathcal{N}(\mathbf{s}; \mu_{\mathbf{x}}^{(k)}, \Sigma_{\mathbf{x}}^{(k)})$ and minimizing with respect to ξ, we have,

$$\frac{\xi_i}{2} \;=\; g^{*\prime -1}\big(E_{Q_k}[s_i^2]\big) \;=\; g'\big(E_{Q_k}[\,s_i^2]\big) \;=\; \frac{f'(\sigma_i)}{2\sigma_i}$$

where $\sigma_i^2 = E_{Q_k}[\,s_i^2]$. Thus we can write the weight matrix as,

$$[\mathbf{\Lambda}]_{i,i} = \frac{f'(\sigma_i)}{\sigma_i} \tag{5.17}$$

As in [43], we generate an approximate Maximum Likelihood estimate of **A** by minimizing (5.16) for **A** along with $\{\xi_k\}$. This leads to the standard EM update for **A** (5.6).

```
Variational EM algorithm
Choose $\mathbf{A}^0 \in \mathbb{R}^{m \times n}$. Set $\mathbf{\Lambda}_k^0 = \mathbf{I}\,,\ \ k = 1, \ldots, N$.
for $l = 0, 1, 2, \ldots,$
        for $k = 1, \ldots, N$
                $\mathbf{B}_k^l = \mathbf{\Lambda}_k^l - \mathbf{\Lambda}_k^l \big(\mathbf{A}^l \mathbf{\Lambda}^l (\mathbf{A}^l)^T + \mathbf{\Sigma}_\nu\big) \mathbf{\Lambda}_k^l \mathbf{A}^l$
                $\mathbf{s}_k^l = \mathbf{\Lambda}_k^l (\mathbf{A}^l)^T \big(\mathbf{A}^l \mathbf{\Lambda}_k^l (\mathbf{A}^l)^T + \mathbf{\Sigma}_\nu\big)^{-1} \mathbf{x}_k$
                $\sigma_i^{l+1} = \sqrt{[\mathbf{s}_k^l]_i^2 + \big[\mathbf{B}_k^l\big]_{ii}}\,, \quad \big[\mathbf{\Lambda}_k^{l+1}\big]_{ii} = \sigma_i^{l+1} / f'(\sigma_i^{l+1})$
        end
        $\mathbf{A}^{l+1} = \left(\sum_{k=1}^N \mathbf{x}_k (\mathbf{s}_k^l)^T\right) \left(\sum_{k=1}^N \mathbf{B}_k^l + \mathbf{s}_k^l (\mathbf{s}_k^l)^T\right)^{-1}$
end
```

In particular, for the Laplacian prior, $p(s) \propto \exp(-|s|)$, and $f(s) = |s|$. Thus the update of the weight matrix is,

$$\big[\mathbf{\Lambda}_{k,l}\big]_{i,i} = \frac{1}{\sigma_i}$$

This update differs from the case of assuming a Gaussian prior for $p(\mathbf{x})$ with unknown variance, and estimating the variance using an EM algorithm, only by a square, as in the latter case we have,

$$\big[\mathbf{\Lambda}_{k,l}\big]_{i,i} = \frac{1}{\sigma_i^2}$$

5.6 Monte Carlo Experimental Comparison

We performed a Monte Carlo experiment to assess the ability of the various algorithms to learn overcomplete generating matrices.

First, the following experiment was performed fifty times. We generated a 2×3 $\mathbf{A}$ matrix, and 200 data points by choosing one of the three columns with equal probability, multiplying by a random scalar uniform on $(-1, 1)$ and adding zero mean Gaussian noise with standard deviation 0.005. We ran the Girolami [43] algorithm, the VB algorithm with Jeffrey's prior, the Lagrangian algorithms for $\rho = 1.0$ and $\rho = 1.1$, the FOCUSS-CNDL algorithm [62], the FOCUSS-CNDL algorithm with scaled gradient, and the Lewick-Sejnowski algorithm with Generalized Gaussian $\rho = 1.1$ and Logistic priors. We started each from the same initial point and ran until convergence, or a maximum of 500 iterations. We calculated the best matching assignment of the matrix solution by finding the pair with highest normalized inner product, then finding the pair from the remaining vectors with the highest normalized inner product, and so on. We did this for the solution generated by each algorithm comparing to the known generating matrix, and stored the normalized inner products.

We made Box-Whiskers plots from the resulting inner product data, which had $50 \cdot 200 = 10000$ points for each algorithm. The column numbers correspond to the algorithms as follows:

1. VB algorithm with Laplacian prior [43]

2. VB algorithm with Jeffrey's prior

3. Lagrangian Newton algorithm described in this thesis and [81], using Generalized Gaussian prior with $\rho = 1.0$

4. Lagrangian Newton algorithm with $\rho = 1.1$

5. The algorithm proposed in [62]

6. The algorithm in [62] but scaled gradient columns as in the Lagrangian Newton algorithm

7. Lewicki-Sejnowski algorithm [68] with Generalized Gaussian prior $\rho = 1.1$

8. Lewicki-Sejnowski algorithm with Logistic prior.

The red line marks the median, and the box encloses the central two quartiles. The lines (i.e. "whiskers") mark the extent almost all of the data, with outliers plotted as red crosses.

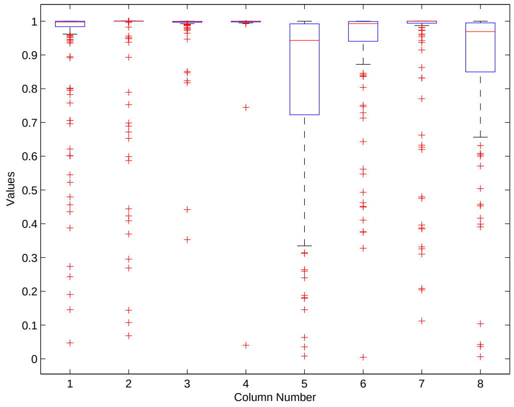


Figure 5.1: Box plot of inner product of best matching columns with true generating matrix for the algorithms with parameters as enumerated in the text. 2×3 $\mathbf{A}$ case with sparsity 1, i.e. 1 non-zero element in $\mathbf{s}$.

This experiment was repeated with 4×8 matrices, multiplying by random source vectors with 2 non-zero elements distributed $U(-1, 1)$, with Gaussian noise $\sigma = 0.005$ added to the resulting $\mathbf{x}$.

The experiment was repeated a third time with 10×20 $\mathbf{A}$ matrices, multiplying by random source vectors with 1 to 5 (equal probability) non-zero elements distributed $U(-1, 1)$, with Gaussian noise $\sigma = 0.005$ added to $\mathbf{x}$.

In each of the experiments, the Lagrangian Newton algorithm (4) performs best, in terms of greatest median, or largest number of correct vectors, with success indicated by normalized inner product (angle cosine) greater than some threshold such as 0.99. The similarly derived algorithm (6) performs similarly to the Lagrangian Newton algorithm. The algorithm (5) without column re-normalization essentially stops immediately because it is based on the error in the representation without scaling as in the Lagrangian type of normalization.

It is also apparent that the VB algorithm with the Laplacian bound performs worse than the Jeffrey's bound, as suggested by the VB algorithm's goal of minimization of a KL divergence bound.

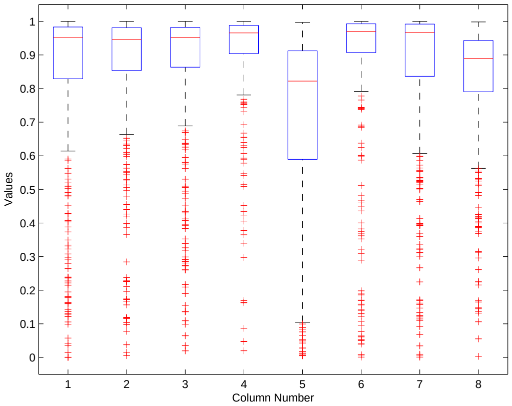


Figure 5.2: Box plot of inner product of best matching columns with true generating matrix, for the algorithms with parameters as enumerated in the text. **A** 4 × 8 and **s** with sparsity 2.

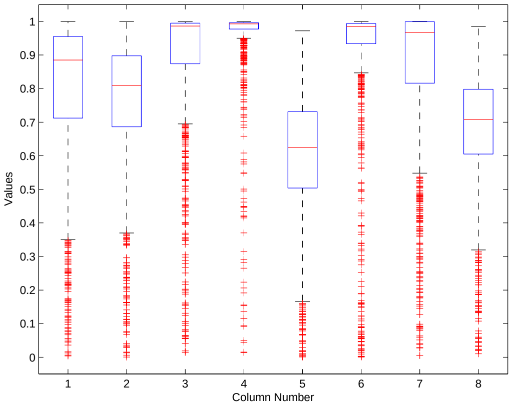


Figure 5.3: Box plot of inner product of best matching columns with true generating matrix,for the algorithms with parameters as enumerated in the text. **A** 10 × 20 and **s** with sparsity 1 to 5.

The Lewicki-Sejnowski (LS) algorithms can be seen as approximations of the Lagrangian Newton algorithm (without column normalization.) In the complete case of square $\mathbf{A}$, the Lagrangian Newton algorithm reduces to the LS algorithm and standard ICA Maximum Likelihood algorithm. The LS algorithm is unstable when used with the Laplacian prior ($\rho = 1.0$), but is stable with $\rho = 1.1$, and performs better with this prior at finding sparse generating bases than with the Logistic prior. The LS algorithm seems to avoid matrix updates, but iterations to find the source estimates, e.g. Newton or other IRLS steps, invariably involve computation of matrix inverses.

6

Independent Component Analysis

This chapter considers the basic ICA model,

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

where $\mathbf{A} \in \mathbb{R}^{n \times n}$ and there is no noise present, so that $\mathbf{s} = \mathbf{W}\mathbf{x}$ where $\mathbf{W} = \mathbf{A}^{-1}$. In this context, the emphasis is not necessarily on sparse estimation (though speech signals for example do tend to have sparse densities) but rather on modeling arbitrary densities of the sources $p_i(s_i)$, only assuming that they are mutually independent. We shall show that the theory developed for estimation with strong super-Gaussians and Gaussian scale mixtures can be extended to mixture models of these densities. This constitutes a generalization of the Gaussian mixture model, offering greater flexibility in the adaptive source density with the same "complexity" in the structural risk minimization sense [100].

We first review the basic Infomax algorithm of Bell and Sejnowski, and the Natural Gradient algorithm of Amari.

6.1 Maximum Likelihood Estimation

Given i.i.d. data $\mathbf{x}_1, \ldots, \mathbf{x}_T$, we consider the ML estimate of $\mathbf{W} = \mathbf{A}^{-1}$. For the density of $\mathbf{x}$, we have,

$$p_{\mathbf{x}}(\mathbf{x}) = \prod_{t=1}^{T} |\,\det \mathbf{W}|\, p_{\mathbf{s}}(\mathbf{W}\mathbf{x}_t)$$

Let $\mathbf{y}_t = \mathbf{W}\mathbf{x}_t$ be the estimate of the sources $\mathbf{s}_t$, and let $q_i(y_i)$ be the density model for the ith source. For the log likelihood of the data then, we have,

$$L(\mathbf{W}) = \sum_{t=1}^{T} \log |\det \mathbf{W}| + \sum_{i=1}^{n} \log q_i(y_{it}) \tag{6.1}$$

The gradient of this function is proportional to,

$$\mathbf{W}^{-T} + \frac{1}{T} \sum_{t=1}^{T} \varphi(\mathbf{y}) \mathbf{x}_t^T \tag{6.2}$$

where we define the score function,

$$\varphi(\mathbf{y}) = \nabla \log q(\mathbf{y}_t)$$

and scale by $1/T$.

Note that if we multiply (6.2) by $\mathbf{W}^T\mathbf{W}$ on the right, we get,

$$\Delta \mathbf{W} = \Big(\mathbf{I} + \big\langle \varphi(\mathbf{y}_t) \mathbf{y}_t^T \big\rangle_T \Big) \mathbf{W} \tag{6.3}$$

where $\langle \cdot \rangle_T$ denotes the average over the T data points. This transformation is in fact a positive definite linear transformation of the matrix gradient. Specifically, using the matrix inner product, for arbitrary $\mathbf{V} \in \mathbb{R}^{n \times n}$, we have,

$$\Big\langle \mathbf{V}, \mathbf{V}\mathbf{W}\mathbf{W}^T \Big\rangle = \Big\langle \mathbf{V}\mathbf{W}, \mathbf{V}\mathbf{W} \Big\rangle > 0 \tag{6.4}$$

when $\mathbf{W}$ is full rank. The direction (6.3) is known as the “natural gradient” [1].

6.2 Stability of ICA and Asymptotic Newton's method

The stability of the ICA algorithm is determined by the positive definiteness of the function (6.1). We can determine asymptotic, or large sample size, stability, by replacing the averages $\langle\cdot\rangle_T$ by the expectation of these quantities. The posit!ive definiteness of (6.1) depends on the positivity of the eigenvalues of the Hessian, i.e. the matrix of partial second derivatives with respect to the elements of $\mathbf{W}$. A Newton method will be derived by applying the inverse of the Hessian linear operator to the gradient (6.2).

6.2.1 Stability

Denote the gradient (6.2) by $\mathbf{G}(\mathbf{W})$ with elements $g_{ij}(\mathbf{W})$. Taking the derivative of the gradient (6.2), we find,

$$\frac{\partial g_{ij}(\mathbf{W})}{\partial w_{kl}} = -[\mathbf{W}^{-1}]_{li}[\mathbf{W}^{-1}]_{jk} + \varphi'(\mathbf{w}_k^T\mathbf{x})x_j x_l \delta_{ik}$$

where $\mathbf{w}_k^T$ is the kth row of $\mathbf{W}$, and δ_{ik} is the Kronecker delta symbol. To see how this linear Hessian operator transforms an argument $\mathbf{B}$, let $\mathbf{C} = \mathcal{H}(\mathbf{B})$ be the transformed matrix. Then we calculate,

$$c_{ij} \quad = \quad -\sum_k \sum_l [\mathbf{W}^{-1}]_{li}[\mathbf{W}^{-1}]_{jk} b_{kl} + \varphi'(y_i)x_j \sum_l b_{il} x_l$$

The first term of c_{ij} can be written,

$$\sum_l [\mathbf{W}^{-1}]_{li}[\mathbf{W}^{-1}\mathbf{B}]_{jl} = \sum_l [\mathbf{W}^{-T}]_{il}[\mathbf{B}^T\mathbf{W}^{-T}]_{lj} = \mathbf{W}^{-T}\mathbf{B}^T\mathbf{W}^{-T}$$

Writing the second term in matrix form as well, we have

$$\mathbf{C} = \mathcal{H}(\mathbf{B}) = -\mathbf{W}^{-T}\mathbf{B}^T\mathbf{W}^{-T} + \mathrm{diag}(\varphi')\mathbf{B}\mathbf{x}\mathbf{x}^T \tag{6.5}$$

The asymptotic stability of the algorithm is determined by the positivity of the eigenvalues of the expected value of this transformation evaluated at the optimum [2]. Assuming that the model holds, the source estimates at the optimal $\mathbf{W}$ will be independent. We also assume that the mean of the data has been removed, so that the sources are zero mean as well.

It will be easier to calculate the expected value of the Hessian if we rewrite the transformation (6.5) in terms of the source estimates $\mathbf{y}$ since the sources are assumed to be independent and zero mean. At the optimum, we may assume that the source density models $q_i(y_i)$ are equivalent to the true source densities $p_i(s_i)$. We first write,

$$\mathbf{C} = -(\mathbf{B}\mathbf{W}^{-1})^T\mathbf{W}^{-T} + \operatorname{diag}(\varphi')\mathbf{B}\mathbf{W}^{-1}\mathbf{W}\mathbf{y}\mathbf{y}^T\mathbf{W}^{-T}$$

where $\operatorname{diag}(\varphi')$ is the diagonal matrix with diagonal elements $\varphi'(y_i)$. Now if we define $\tilde{\mathbf{C}} = \mathbf{C}\mathbf{W}^T$ and $\tilde{\mathbf{B}} = \mathbf{B}\mathbf{W}^{-1}$, then we have,

$$\tilde{\mathbf{C}} = \tilde{\mathbf{B}}^T + \operatorname{diag}(\varphi')\tilde{\mathbf{B}}\mathbf{y}\mathbf{y}^T \tag{6.6}$$

Writing this equation in component form and taking the expected value we find for the diagonal elements,

$$E[\tilde{c}_{ii}] = \tilde{b}_{ii} + E\left[\varphi'(y_i)\sum_k \tilde{b}_{ik}y_k y_i\right] = \tilde{b}_{ii}(1+\gamma_i^2) \tag{6.7}$$

where we define $\gamma_i = E\varphi'(y_i)y_i^2$. The cross terms drop out since the expected value of $\alpha_i y_i y_k$ is zero for $k \neq i$ by the independence and zero mean assumption on the sources. Now we note [2, 19] that the off-diagonal elements of the equation (6.6) can be paired as follows,

$$\begin{aligned}
E[\tilde{c}_{ij}] &= \tilde{b}_{ji} + E\left[\varphi'(y_i)\sum_k \tilde{b}_{ik}y_k y_j\right] = \tilde{b}_{ji} + \alpha_i\tilde{b}_{ij}\sigma_j^2 \\
E[\tilde{c}_{ji}] &= \tilde{b}_{ij} + E\left[\varphi'(y_j)\sum_k \tilde{b}_{jk}y_k y_i\right] = \tilde{b}_{ij} + \alpha_j\tilde{b}_{ji}\sigma_i^2
\end{aligned}$$

where we define $\alpha_i = E\varphi'(y_i)$ and $\sigma_i^2 = Ey_i^2$. Again the cross terms drop out from the expectation of independent zero mean random variable. Putting these equations in matrix form, we have,

$$\begin{bmatrix} E[\tilde{c}_{ij}] \\ E[\tilde{c}_{ji}] \end{bmatrix} = \begin{bmatrix} \alpha_i\sigma_j^2 & 1 \\ 1 & \alpha_j\sigma_i^2 \end{bmatrix} \begin{bmatrix} \tilde{b}_{ij} \\ \tilde{b}_{ji} \end{bmatrix} \tag{6.8}$$

If we denote the linear transformation defined by equations (6.7) and (6.8) by $\tilde{\mathbf{C}} = \tilde{\mathcal{H}}(\tilde{\mathbf{B}})$, then we have,

$$\mathbf{C} = \mathcal{H}(\mathbf{B}) = \tilde{\mathcal{H}}\Big(\mathbf{B}\mathbf{W}^{-1}\Big)\mathbf{W}^{-T}$$

Thus by reasoning similar to (6.4), we see that the expected value of $\mathcal{H}$ is a positive definite transformation if and only if the expected value of $\tilde{\mathcal{H}}$ is positive definite and $\mathbf{W}$ is full rank.

The conditions for positive definiteness of $\tilde{\mathcal{H}}$ can be found by inspection of equations (6.7) and (6.8). With our definitions,

$$\gamma_i = E[\varphi_i'(y_i)y_i^2], \quad \alpha_i = E[\varphi_i'(y_i)], \quad \sigma_i^2 = E[y_i^2]$$

the conditions can be stated [2] as,

1. $1 + \gamma_i > 0,\ \forall\, i$
2. $\alpha_i > 0,\ \forall\, i,$ and,
3. $\alpha_i\alpha_j\sigma_i^2\sigma_j^2 - 1 > 0, \quad \forall\, i \neq j$

6.2.2 Newton Method

The inverse of the Hessian operator will be given by,

$$\mathbf{B} = \mathcal{H}^{-1}(\mathbf{C}) = \tilde{\mathcal{H}}^{-1}\Big(\mathbf{C}\mathbf{W}^{T}\Big)\mathbf{W} \tag{6.9}$$

The calculation of $\tilde{\mathbf{B}} = \tilde{\mathcal{H}}^{-1}(\tilde{\mathbf{C}})$ can again be found by inspection of (6.7) and (6.8),

$$\tilde{b}_{ii} = \frac{\tilde{c}_{ii}}{1 + \gamma_i},\ \forall\, i \tag{6.10}$$

$$\tilde{b}_{ij} = \frac{\alpha_j\sigma_i^2\tilde{c}_{ij} - \tilde{c}_{ji}}{\alpha_i\alpha_j\sigma_i^2\sigma_j^2 - 1},\ \forall\, i \neq j \tag{6.11}$$

The Newton direction is given by taking $\mathbf{C} = \mathbf{G}(\mathbf{W})$, the gradient (6.2),

$$\Delta\mathbf{W} = \tilde{\mathcal{H}}^{-1}\Big(\mathbf{G}\mathbf{W}^{T}\Big)\mathbf{W} \tag{6.12}$$

6.2.3 Natural Gradient

First, note that if we take $\tilde{\mathcal{H}}$ to be identity in (6.12), then we get,

$$\Delta\mathbf{W} = \mathbf{G}\mathbf{W}^T\mathbf{W} \tag{6.13}$$

which is the natural gradient, shown previously to be a positive definite transformation of the gradient.

The natural gradient transformation can also be derived by a sort of affine scaling transformation. In affine scaling, one makes a linear change of variable, computes the gradient with respect to the new variable in the transform space, then maps the transform gradient back into the original space.

In linear programming, the linear transformation is chosen to map the current iterate of the solution vector to the point $[1, 1, \ldots, 1]^T$. This has the effect of scaling the gradient so that it does not get "bogged down" as the iterations get closer to the positive orthant boundary.

The natural gradient transformation can be seen as mapping the current iterate of $\mathbf{W}$ to the identity matrix. Specifically, given a matrix function, $f(\mathbf{W})$, suppose we make the linear change of variable $\mathbf{Z} = \mathbf{W}\mathbf{V}^{-1}$. Then

$$\nabla\tilde{f}(\mathbf{Z}) = \nabla f(\mathbf{Z}\mathbf{V})\mathbf{V}^T$$

Then transform back into the $\mathbf{W}$ space,

$$\nabla\tilde{f}(\mathbf{Z})\mathbf{V} = \nabla f(\mathbf{W})\mathbf{V}^T\mathbf{V}$$

Taking $\mathbf{V} = \mathbf{W}^l$, the current iterate of $\mathbf{W}$ is equivalent to mapping each current iterate to the identity matrix in the $\mathbf{Z}$ space, calculating the transform gradient, and transforming back.

This heuristic derivation cannot completely account for the success of the natural gradient, however, since we may just as well consider the transformation $\mathbf{Z} = \mathbf{V}^{-1}\mathbf{W}$, multiplying on the left instead of the right. This also maps to identity, and produces the positive definite transformation of the gradient, $\mathbf{W}\mathbf{W}^T\mathbf{G}$. Experience shows however that this direction does not have good scaling properties, while the natural gradient is found to perform well.

The explanation may be seen from the fact that multiplying by the inverse on the right is equivalent to mapping to the $\mathbf{W}$ to the "tilde" space, as when we defined $\tilde{\mathbf{B}} = \mathbf{B}\mathbf{W}^{-1}$ to get (6.6). This is a sort of partial Newton method which forgoes inverting $\tilde{\mathcal{H}}$.

The natural gradient has also been derived by Amari from the standpoint of Lie Algebras, and by Cardoso and Laheld as the "relative gradient" [19], where it is shown that the relative gradient has the property of equivariance, i.e. the variance of the estimate is independent of the value of the true parameter being estimated.

6.3 ICA with Strongly Super-Gaussian Sources

The properties of strong super-Gaussians (and Gaussian scale mixtures) can be used to derive an EM-based ICA algorithm for estimation with sharply peaked priors whose Hessian is unbounded, making Newton's method unstable. When the $q_i(y_i)$ densities are strongly super-Gaussian, the Q function of the EM algorithm for the likelihood (6.1),

$$Q(\mathbf{W}|\mathbf{W}^l) \;=\; \log|\det\mathbf{W}| - \tfrac{1}{2}\big\langle \mathbf{x}_t^T\mathbf{W}^T\boldsymbol{\Lambda}_t^{l\,-1}\mathbf{W}\mathbf{x}_t\big\rangle_T \tag{6.14}$$

where $\boldsymbol{\Lambda}_t^{l\,-1}$ has diagonal elements,

$$\lambda_{it}^{l\;-1} \;=\; \xi_{it}^l \;=\; \frac{f'(y_{it}^l)}{y_{it}^l}$$

where $\mathbf{y}_t^l = \mathbf{W}^l\mathbf{x}_t$, and $f_i(y_i) = -\log q_i(y_i)$.

6.3.1 Stability of EM iteration

In this section we consider the stability of the algorithm using analysis similar to [2]. We show that when each source is strongly super-Gaussian, the Q

function is convex in $\mathbf{W}$.

Taking second derivatives as we did for the stability analysis and Newton method, we find for the Hessian operator of (6.14),

$$\mathcal{H}(\mathbf{B}) = -\mathbf{W}^{-T}\mathbf{B}^T\mathbf{W}^{-T} + \mathbf{\Lambda}_t^{l^{-1}}\mathbf{B}\mathbf{x}\mathbf{x}^T \tag{6.15}$$

which is equivalent to the Hessian of actual likelihood except that $\varphi_i'(y_i) = f_i''(y_i)$ is replaced by $\varphi(y_i)/y_i = f_i'(y_i)/y_i$. With the new definitions,

$$\gamma_i = E[\varphi(y_i)y_i], \quad \alpha_i = E[\varphi(y_i)/y_i], \quad \sigma_i^2 = E[y_i^2]$$

the conditions for the positive definiteness of the function (6.14) have the same form as before,

1. $1 + \gamma_i > 0, \ \forall\, i$
2. $\alpha_i > 0, \ \forall\, i,$ and,
3. $\alpha_i\alpha_j\sigma_i^2\sigma_j^2 - 1 > 0, \quad \forall\, i \neq j$

If we assume that $\varphi_i(y_i)$ is derived from the true density, i.e. $q_i(y_i) = p_i(y_i)$, then the stability conditions are always satisfied, provided the moments are finite. The first condition is satisfied since $E[\varphi(y)y] = 1$ by the well-known property of the score function, which is readily derived by integrating by parts.

The symmetry and unimodatlity of strongly super-Gaussian $q_i(y_i)$ implies that $\varphi_i(y_i)$ has the same sign as y_i, and $\varphi_i(y_i)/y_i > 0$ for all y_i, so that the second condition will always be satisfied as well when the moment is finite.

The last condition is satisfied by the Cauchy-Schwartz inequality,

$$E[\varphi_i(y_i)/y_i]E[y_i^2] > E[\varphi_i(y_i)y_i]^2 = 1$$

unless $\varphi_i(y_i) \propto y_i^3$, but the latter is only the case for the Generalized Gaussian density with $\rho = 4$, which is strongly sub-Gaussian, and thus not strongly super-Gaussian.

6.3.2 Cramer-Rao Lower Bound on ICA Variance

We calculated the expected Hessian, or Fisher Information matrix, of the ICA log likelihood near an optimal separating solution $\mathbf{W}^*$ with and source model in section §6.2.1. Since ML estimates are asymptotically unbiased, efficient, and Normal with covariance given by the inverse Fisher Information matrix we can use the results of §6.2.1 to calculate bounds on the covariance of the ML estimate for different source densities.

We use the global system $\mathbf{C} = \mathbf{WA}$, whose optimal solution is always identity, $\mathbf{I}$. In §6.2.1, we calculated the expected Hessian in the global system space and found that it depended only on the source densities. The inverse of this matrix gives a lower bound on the variance of elements estimate of the identity given by a ML estimate $\mathbf{W}^*$.

For the covariance of the diagonal elements, we have,

$$E(\hat{c}_{ii} - 1)^2 \;\geq\; \frac{1}{Ef_i''(s_i)s_i^2}$$

Let $\alpha_i = Ef_i''(s_i)$ and $\sigma^2 = Es_i^2$. For the off-diagonal estimates whose "true" parameter value is 0, we have,

$$E\hat{c}_{ij}^2 \;\geq\; \frac{\alpha_j \sigma_i^2}{\alpha_i \alpha_j \sigma_i^2 \sigma_j^2 - 1}$$

Suppose for simplicity that the $f_i = \log p_i$ are Generalized Gaussian with the same shape parameter p. Then the required moments can be calculated in closed form. For the off diagonal elements, the minimum variance of the global systems $\mathbf{C} = \mathbf{WA}$ of an unbiased estimator is plotted in Figure 6.1.

6.3.3 Strongly Super-Gaussian Mixtures

To model arbitrary densities using the global convergence properties of the EM algorithm, we can consider mixture models with strongly super-Gaussian densities. This constitutes a generalization of the basic Gaussian mixture model while maintaining the simplicity and monotonicity of the EM update [82].

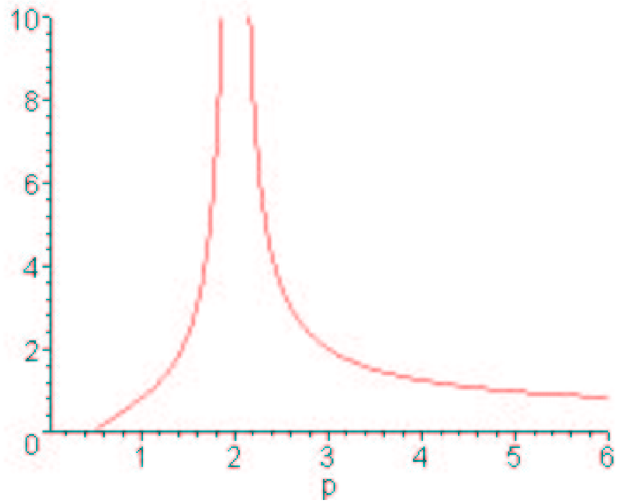


Figure 6.1: Plot of Cramer-Rao lower bound on standard deviation (also asymptotic standard deviation of ML estimator) for off-diagonal elements of the global system $\mathbf{C} = \mathbf{WA}$, per sample. For N samples, the values are divided by N. The magnitude of the off-diagonal c_{ij} of the global system determines the interference between the estimates of sources s_i and s_j. The variance becomes infinite for the Gaussian $p = 2$ as expected from the non-identifiability of Gaussian bases shown in §1.2.1. The expected second derivative of f (or first derivative of the score function φ,) and the variance, are actually positive and finite for $p > 0.5$, going to zero at $p = 0.5$. The variance also tends to zero as $p \to \infty$.

Mixture densities have the form,

$$p(s) = \sum_{j=1}^{m} \alpha_j \sqrt{\beta_j}\, p_j\Big(\sqrt{\beta_j}\,(s - \mu_j)\Big)\,, \quad \sum_j \alpha_j = 1,\ \alpha_j \geq 0,\ \beta_j > 0$$

We first consider a single square basis, or mixing matrix, $\mathbf{A}$ with super-gaussian mixture sources, so that the $p_j(s)$ are assumed to be strongly super-gaussian. Note that $p(s)$ is not necessarily super-gaussian, only the mixture components densities $p_j(s)$. Later we extend the model to mixtures over mixing or basis matrices. Initially, the jth source mixture component density of the ith source will be denoted $p_{ij}(s_{ij})$ with mode (location) μ_{ij} and inverse square scale β_{ij}. In the Gaussian case $p_{ij}(s) = \mathcal{N}(s\,; \mu_{ij}, \beta_{ij}^{-1})$, μ_{ij} is the mean and β_{ij} is the inverse variance. For general

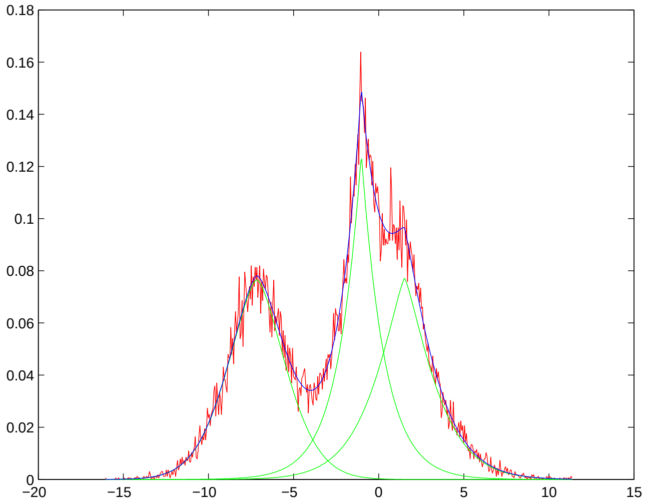


Figure 6.2: Example of adaptive convergence of the Generalized Gaussian mixture model. The histogram is plotted in red, the converged mixture components are plotted in green, and their sum, which is the approximating mixture density, is plotted in blue.

strongly super-gaussian densities, μ_{ij} is the mean only if the mean exists, and β_{ij} is the inverse variance divided by $\int s^2 p_{ij}(s)ds$ only when the latter exists.

6.3.4 ICA with Strongly Super-Gaussian Mixture Sources

Let the data $\mathbf{x}_k$, $k = 1, \ldots, N$ be given, and consider the instantaneous model,

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

where $\mathbf{A} \in \mathbb{R}^{n \times n}$ is non-singular, and the sources s_i, $i = 1, \ldots, n$, are independent with strongly super-gaussian mixture densities. We allow the number of source mixture components m_i to differ for different sources.

We wish to estimate the parameter $\mathbf{W} = \mathbf{A}^{-1}$ and the parameters of the source mixtures,

$$\theta = \{\mathbf{w}_i, \alpha_{ij}, \mu_{ij}, \beta_{ij}\}, \quad i = 1, \ldots, n\,,\; j = 1, \ldots, m_i$$

where the vector $\mathbf{w}_i$ is the ith column of $\mathbf{W}^T$. We define $\mathbf{X} \equiv [\mathbf{x}_1 \cdots \mathbf{x}_N]$.

The source mixture model is equivalent to a scenario in which for each source s_i, a mixture component j_i is drawn from the discrete probability distribution $P[j_i = j] = \alpha_{ij}$, $1 \leq j \leq m_i$, then s_i is drawn from the mixture component density p_{ij_i}. We define j_{ik} to be the index chosen for the ith source in the kth sample.

To use the EM algorithm, we define the random variables z_{ijk} as follows,

$$z_{ijk} \equiv \begin{cases} 1, & j_{ik} = j \\ 0, & \text{otherwise} \end{cases}$$

Let $\mathbf{Z} = \{z_{ijk}\}$. Then we have,

$$p(\mathbf{X};\theta) = \sum_{\mathbf{Z}} \prod_{k=1}^{N} |\det \mathbf{W}| \prod_{i=1}^{n} \prod_{j=1}^{m_i} \Big[\alpha_{ij} \sqrt{\beta_{ij}}\, p_{ij}\Big(\sqrt{\beta_{ij}}\big(\mathbf{w}_i^T \mathbf{x}_k - \mu_{ij}\big)\Big)\Big]^{z_{ijk}}$$

For the variational free energy, we have $F(q^l;\theta) = F^l(\theta) + H(\mathbf{Z};\theta^l)$, where $H(\mathbf{Z};\theta^l)$ is the entropy of the $\mathbf{Z}$ evaluated for $\theta = \theta^l$, and $F^l(\theta)$ is given by,

$$-N \log|\det \mathbf{W}| \;+\; \sum_{k=1}^{N} \sum_{i=1}^{n} \sum_{j=1}^{m_i} \hat{z}_{ijk}^l \Big[-\log \alpha_{ij} - \tfrac{1}{2}\log \beta_{ij} + f_{ij}\Big(\sqrt{\beta_{ij}}\big(\mathbf{w}_i^T \mathbf{x}_k - \mu_{ij}\big)\Big)\Big]$$

where we define $f_{ij} \equiv -\log p_{ij}$ and $\hat{z}_{ijk}^l \equiv E[z_{ijk}|\mathbf{x}_k;\theta^l]$. We also define $y_{ijk} \equiv \sqrt{\beta_{ij}}\big(\mathbf{w}_i^T \mathbf{x}_k - \mu_{ij}\big)$, and,

$$y_{ijk}^l \equiv \sqrt{\beta_{ij}^l}\left(\mathbf{w}_i^{l\,T} \mathbf{x}_k - \mu_{ij}^l\right) \tag{6.16}$$

The $\hat{z}_{ijk}^l = P[z_{ijk} = 1|\mathbf{x}_k;\theta^l]$ are determined as in the usual Gaussian EM algorithm,

$$\hat{z}_{ijk}^l = \frac{p(\mathbf{x}_k|z_{ijk} = 1;\theta^l)P[z_{ijk} = 1;\theta^l]}{\sum_{j'=1}^{m_i} p(\mathbf{x}_k|z_{ij'k} = 1;\theta^l)P[z_{ij'k} = 1;\theta^l]} = \frac{\alpha_{ij}^l \sqrt{\beta_{ij}^l}\, p_{ij}\big(y_{ijk}^l\big)}{\sum_{j'=1}^{m_i} \alpha_{ij'}^l \sqrt{\beta_{ij'}^l}\, p_{ij'}\big(y_{ij'k}^l\big)} \tag{6.17}$$

The new α_{ij} are found by maximizing $F^l(\theta)$ such that $\sum_{j=1}^{m_i} \alpha_{ij} = 1$, $\alpha_{ij} > 0$, yielding,

$$\alpha_{ij}^{l+1} = \frac{\sum_{k=1}^{N} \hat{z}_{ijk}^l}{\sum_{j'=1}^{m_i} \sum_{k=1}^{N} \hat{z}_{ij'k}^l} = \frac{1}{N} \sum_{k=1}^{N} \hat{z}_{ijk}^l \tag{6.18}$$

which is equivalent to the update in the ordinary Gaussian mixture model EM algorithm.

To update the source mixture component parameters, we define,

$$\xi^l_{ijk} \;\equiv\; \frac{f'_{ij}(y^l_{ijk})}{y^l_{ijk}} \tag{6.19}$$

and use the strong super-Gaussianity inequality to replace $f_{ij}(y_{ijk})$ in $F^l(\theta)$ by $\frac{1}{2}\xi^l_{ijk} y^2_{ijk}$ to get,

$$-N \log|\det \mathbf{W}| \;+\; \sum_{k=1}^{N}\sum_{i=1}^{n}\sum_{j=1}^{m_i} \hat{z}^l_{ijk}\left[-\log\alpha_{ij} - \tfrac{1}{2}\log\beta_{ij} + \tfrac{1}{2}\xi^l_{ijk}\beta_{ij}\left(\mathbf{w}_i^T\mathbf{x}_k - \mu_{ij}\right)^2\right]$$

Minimizing $\tilde{F}^l$ with respect to μ_{ij} and β_{ij} guarantees, using the strong super-Gaussianity inequality, that,

$$F(q^l;\theta^{l+1}) - F(q^l;\theta^l) \;\leq\; \tilde{F}(q^l;\theta^{l+1}) - \tilde{F}(q^l;\theta^l) \;\leq\; 0$$

and thus that $F(q^l;\theta)$ is decreased as required by the EM algorithm.

As in the Gaussian mixture case, the optimal value of μ_{ij} does not depend on β_{ij}. The updates, using the definitions (6.16), (6.17) and (6.19), are found to be,

$$\mu^{l+1}_{ij} \;=\; \frac{\sum_{k=1}^N \hat{z}^l_{ijk}\xi^l_{ijk}\mathbf{w}_i^{l^T}\mathbf{x}_k}{\sum_{k=1}^N \hat{z}^l_{ijk}\xi^l_{ijk}} \;=\; \mu^l_{ij} + \frac{\sum_{k=1}^N \hat{z}^l_{ijk} f'_{ij}(y^l_{ijk})}{\sqrt{\beta^l_{ij}}\sum_{k=1}^N \hat{z}^l_{ijk}\xi^l_{ijk}} \tag{6.20}$$

and,

$$\beta^{l+1}_{ij} \;=\; \frac{\sum_{k=1}^N \hat{z}^l_{ijk}}{\sum_{k=1}^N \hat{z}^l_{ijk}\xi^l_{ijk}\left(\mathbf{w}_i^{l^T}\mathbf{x}_k - \mu^l_{ij}\right)^2} \;=\; \frac{\beta^l_{ij}\sum_{k=1}^N \hat{z}^l_{ijk}}{\sum_{k=1}^N \hat{z}^l_{ijk} f'_{ij}(y^l_{ijk}) y^l_{ijk}} \tag{6.21}$$

We adapt $\mathbf{W}$ according to the natural gradient of F. Defining the vector $\mathbf{u}^l_k$ such that,

$$\left[\mathbf{u}^l_k\right]_i \equiv \sum_{j=1}^{m_i} \hat{z}^l_{ijk}\sqrt{\beta^l_{ij}}\, f'_{ij}(y^l_{ijk}) \tag{6.22}$$

we have,

$$\Delta\mathbf{W} = \left(\mathbf{I} - \frac{1}{N}\sum_{k=1}^{N}\mathbf{u}^l_k\mathbf{x}^T_k\mathbf{W}^{l\,T}\right)\mathbf{W}^l \tag{6.23}$$

ICA Mixture Model with Strongly Super-Gaussian Mixture Sources

We now consider the case in which the data is generated by a mixture over a set of mixing matrices, $\mathbf{A}_h = {\mathbf{W}_h}^{-1}$, $h = 1, \ldots, M$,

$$p(\mathbf{x}_k; \theta) = \sum_{h=1}^{M} \gamma_h p_h(\mathbf{x}_k; \theta)\,, \;\; \gamma_h \geq 0, \;\; \sum_{h=1}^{M} \gamma_h = 1$$

The parameters to be estimated are,

$$\theta = \left\{\gamma_h, \mathbf{W}_h, \alpha_{hij}, \mu_{hij}, \beta_{hij}\right\}, \; h = 1, \ldots, M, \; i = 1, \ldots, n_h, \; j = 1, \ldots, m_{hi}$$

The EM algorithm for the full mixture model is derived similarly to the case of source mixtures. In this model, each (independent) sample $\mathbf{x}_k$ is generated by drawing a mixture component h' from the discrete probability distribution $P[h' = h] = \gamma_h$, $1 \leq h \leq M$, then drawing $\mathbf{x}$ from $p_{h'}(\mathbf{x}; \theta)$.

We define h_k to be the index chosen for the kth sample, and we define the random variable,

$$v_{hk} \equiv \begin{cases} 1, & h_k = h \\ 0, & \text{otherwise} \end{cases}$$

Let $\mathbf{V} \equiv \{v_{hk}\}$. We define j_{hik} to be the source mixture component index chosen (independently of h_k) for the ith source of the hth model in the kth sample, and we define the random variables z_{hijk} by,

$$z_{hijk} \equiv \begin{cases} 1, & j_{hik} = j \\ 0, & \text{otherwise} \end{cases}$$

with $\mathbf{Z} \equiv \{z_{hijk}\}$. Now, for the likelihood of θ, we can write,

$$p(\mathbf{X}; \theta) =$$
$$\sum_{\mathbf{V},\mathbf{Z}} \prod_{k=1}^{N} \prod_{h=1}^{M} \gamma_h^{v_{hk}} |\det \mathbf{W}_h|^{v_{hk}} \prod_{i=1}^{n_h} \prod_{j=1}^{m_{hi}} \left[\alpha_{hij} \sqrt{\beta_{hij}}\, p_{hij}\Big(\sqrt{\beta_{hij}}\big(\mathbf{w}_{hi}^T \mathbf{x}_k - \mu_{hij}\big)\Big)\right]^{v_{hk} z_{hijk}}$$

For the variational free energy we have $F(q^l;\theta) = F^l(\theta) + H(\mathbf{V};\theta^l) + H(\mathbf{Z};\theta^l)$, where $H(\mathbf{V};\theta^l)$ and $H(\mathbf{Z};\theta^l)$ are the entropies of $\mathbf{V}$ and $\mathbf{Z}$ with the parameters set to θ^l. We now have,

$$\sum_{k=1}^{N}\sum_{h=1}^{M}\left[\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}} E\big[v_{hk}z_{hijk}|\mathbf{x}_k;\theta^l\big]\Big(-\log\alpha_{hij} - \tfrac{1}{2}\log\beta_{hij} + f_{hij}\Big(\sqrt{\beta_{hij}}\,\big(\mathbf{w}_{hi}^T\mathbf{x}_k - \mu_{hij}\big)\Big)\Big)\right]$$
$$+\ E\big[v_{hk}|\mathbf{x}_k;\theta^l\big]\Big(-\log\gamma_h - \log|\det\mathbf{W}_h|\Big)$$

where we define $f_{hij} \equiv -\log p_{hij}$. We define y^l_{hijk} and ξ^l_{hijk} as in (6.16) and (6.19), and we define $\hat{z}^l_{hijk}$ to be the conditional expectation of z_{hijk},

$$\hat{z}^l_{hijk} \;\equiv\; E\big[z_{hijk}\,\big|\,v_{hk}=1, \mathbf{x}_k, ; \theta^l\big] \;=\; \frac{\alpha^l_{hij}\sqrt{\beta^l_{hij}}\,p_{hij}\big(y^l_{hijk}\big)}{\sum_{j'=1}^{m_{hi}}\alpha^l_{hij'}\sqrt{\beta^l_{hij'}}\,p_{hij'}\big(y^l_{hij'k}\big)} \tag{6.24}$$

The $\hat{v}^l_{hk} \equiv E[v_{hk}|\mathbf{x}_k;\theta^l]$ are given by,

$$\begin{aligned}\hat{v}^l_{hk} &= \frac{p(\mathbf{x}_k|v_{hk}=1;\theta^l)P[v_{hk}=1;\theta^l]}{\sum_{h'=1}^{M}p(\mathbf{x}_k|v_{h'k}=1;\theta^l)P[v_{h'k}=1;\theta^l]}\\ &= \frac{\gamma^l_h\,|\det\mathbf{W}^l_h|\prod_{i=1}^{n_h}\sum_{j=1}^{m_{hi}}\alpha^l_{hij}\sqrt{\beta^l_{hij}}\,p_{hij}\big(y^l_{hijk}\big)}{\sum_{h'=1}^{M}\gamma^l_{h'}\,|\det\mathbf{W}^l_{h'}|\prod_{i=1}^{n_{h'}}\sum_{j=1}^{m_{h'i}}\alpha^l_{h'ij}\sqrt{\beta^l_{h'ij}}\,p_{h'ij}\big(y^l_{h'ijk}\big)}\end{aligned}$$

Defining $\hat{r}^l_{hijk} \equiv E[v_{hk}z_{hijk}|\mathbf{x}_k;\theta^l]$, we have,

$$\begin{aligned}\hat{r}^l_{hijk} &= P\big[v_{hk}=1, z_{hijk}=1\,|\,\mathbf{x}_k;\theta^l\big]\\ &= P\big[z_{hijk}=1\,|\,v_{hk}=1,\mathbf{x}_k;\theta^l\big]P\big[v_{hk}=1\,|\,\mathbf{x}_k;\theta^l\big]\\ &= \hat{z}^l_{hijk}\hat{v}^l_{hk}\end{aligned} \tag{6.25}$$

Minimizing F over γ_h and α_{hij}, we get,

$$\gamma^{l+1}_h = \frac{1}{N}\sum_{k=1}^{N}\hat{v}^l_{hk}\,,\quad \alpha^{l+1}_{hij} = \frac{1}{N\gamma^{l+1}_h}\sum_{k=1}^{N}\hat{r}^l_{hijk} \tag{6.26}$$

The remaining parameters are updated as before,

$$\mu^{l+1}_{hij} \;=\; \frac{\sum_{k=1}^{N}\hat{r}^l_{hijk}\xi^l_{hijk}\mathbf{w}^{l\,T}_{hi}\mathbf{x}_k}{\sum_{k=1}^{N}\hat{r}^l_{hijk}\xi^l_{hijk}} \;=\; \mu^l_{hij} + \frac{\sum_{k=1}^{N}\hat{r}^l_{hijk}f'_{hij}\big(y^l_{hijk}\big)}{\sqrt{\beta^l_{hij}}\sum_{k=1}^{N}\hat{r}^l_{hijk}\xi^l_{ijk}} \tag{6.27}$$

and,

$$\beta_{hij}^{l+1} \;=\; \frac{\sum_{k=1}^{N} \hat{r}_{hijk}^{l}}{\sum_{k=1}^{N} \hat{r}_{hijk}^{l} \xi_{hijk}^{l} \big(\mathbf{w}_{hi}^{l\,T}\mathbf{x}_k - \mu_{hij}^{l}\big)^2} \;=\; \frac{\beta_{hij}^{l} \sum_{k=1}^{N} \hat{r}_{hijk}^{l}}{\sum_{k=1}^{N} \hat{r}_{hijk}^{l} f_{hij}^{\prime}\big(y_{hijk}^{l}\big) y_{hijk}^{l}} \tag{6.28}$$

Defining the vector $\mathbf{u}_{hk}^{l}$ such that,

$$\big[\mathbf{u}_{hk}^{l}\big]_i \equiv \sum_{j=1}^{m_{hi}} \hat{r}_{hijk}^{l} \sqrt{\beta_{hij}^{l}}\, f_{hij}^{\prime}\big(y_{hijk}^{l}\big) \tag{6.29}$$

we have,

$$\Delta \mathbf{W}_h = \left(\gamma_h^{l+1} \mathbf{I} - \frac{1}{N} \sum_{k=1}^{N} \mathbf{u}_{hk}^{l} \mathbf{x}_k^{T} \mathbf{W}_h^{l\,T} \right) \mathbf{W}_h^{l} \tag{6.30}$$

If we make the definitions,

$$C_{hijk}^{l} \;\equiv\; \alpha_{hij}^{l} \sqrt{\beta_{hij}^{l}}\, p_{hij}\big(y_{hijk}^{l}\big), \qquad L_{hk}^{l} \;\equiv\; \gamma_h^{l} \left|\det \mathbf{W}_h^{l}\right| \prod_{i=1}^{n_h} \sum_{j=1}^{m_{hi}} C_{hijk}^{l} \tag{6.31}$$

then the $\hat{z}_{hijk}$ and $\hat{v}_{hk}$ updates become,

$$\hat{z}_{hijk}^{l} = \frac{C_{hijk}^{l}}{\sum_{j'=1}^{m_{hi}} C_{hij'k}^{l}}, \qquad \hat{v}_{hk}^{l} = \frac{L_{hk}^{l}}{\sum_{h'=1}^{M} L_{h'k}^{l}} \tag{6.32}$$

The log likelihood of θ^l given $\mathbf{X}$, which we denote by $\bar{L}^l$, is calculated as,

$$\bar{L}^{l} \;=\; \sum_{k=1}^{N} \log \left(\sum_{h=1}^{M} L_{hk}^{l} \right) \tag{6.33}$$

$\bar{L}^l$ increases monotonically with iteration l.

6.3.5 Adaptive Strong Super-Gaussians

We can obtain further flexibility in the source model by adapting the mixture component densities within a parameterized family of strongly super-gaussian densities.

Generalized Gaussians with adaptive shape parameter, ρ

In this section we consider the case of Generalized Gaussian mixtures, with source mixture component densities,

$$p(s_{hij}; \mu_{hij}, \beta_{hij}, \rho_{hij}) = \frac{\sqrt{\beta_{hij}}}{2\,\Gamma\Big(1 + \frac{1}{\rho_{hij}}\Big)} \exp\left(- \left| \sqrt{\beta_{hij}} \big(s_{hij} - \mu_{hij}\big) \right|^{\rho_{hij}} \right)$$

The parameters ρ_{hij} are adapted by scaled gradient descent. The gradient of F with respect to ρ_{hij} is,

$$\frac{\partial F}{\partial \rho_{hij}} \;=\; \sum_{k=1}^{N} \hat{r}_{hijk}\left[|y_{hijk}|^{\rho_{hij}} \log|y_{hijk}| - \frac{1}{\rho_{hij}^{2}} \Psi\left(1 + \frac{1}{\rho_{hij}}\right)\right]$$

We have found that scaling this by $\rho_{hij}^{2} / \left(\Psi\left(1 + \frac{1}{\rho_{hij}}\right) \sum_{k=1}^{N} \hat{r}_{hijk}\right)$, which is positive for $0 < \rho_{hij} \leq 2$, leads to faster convergence. The update then becomes,

$$\Delta\rho_{hij} \;=\; 1 \;-\; \frac{{\rho_{hij}^{l}}^{2} \sum_{k=1}^{N} \hat{r}_{hijk}^{l}\, |y_{hijk}^{l}|^{\rho_{hij}^{l}} \log|y_{hijk}^{l}|}{\Psi\left(1 + \frac{1}{\rho_{hij}^{l}}\right) \sum_{k=1}^{N} \hat{r}_{hijk}^{l}} \tag{6.34}$$

Student's t densities with adaptive degrees of freedom parameter, ν

Student's t densities have the form,

$$p(s_{hij};\mu_{hij},\beta_{hij},\nu_{hij}) = \frac{\sqrt{\beta_{hij}}\;\Gamma\left(\frac{\nu_{hij}+1}{2}\right)}{\sqrt{\pi\nu_{hij}}\;\Gamma\left(\frac{\nu_{hij}}{2}\right)} \left(1 + \frac{\beta_{hij}}{\nu_{hij}} s_{hij}^{2}\right)^{-\frac{\nu_{hij}+1}{2}}$$

The parameters ν_{hij} are adapted by scaled gradient descent. The gradient of F with respect to ν_{hij} is,

$$\frac{\partial F}{\partial \nu_{hij}} \;=\; \frac{1}{2}\sum_{k=1}^{N} \hat{r}_{hijk}\left[\Psi\left(\frac{\nu_{hij}}{2}\right) - \Psi\left(\frac{\nu_{hij}+1}{2}\right) + \frac{\nu_{hij}+1}{\nu_{hij} + y_{hijk}^{2}} + \log\left(1 + \frac{y_{hijk}^{2}}{\nu_{hij}}\right) - 1\right]$$

Dividing this by $\frac{1}{2}\left(1 + \Psi\left(\frac{\nu_{hij}^{l}+1}{2}\right) - \Psi\left(\frac{\nu_{hij}^{l}}{2}\right)\right) \sum_{k=1}^{N} \hat{r}_{hijk}$, which is positive for $\nu_{hij} > 0$, the update becomes,

$$\Delta\nu_{hij} \;=\; 1 \;-\; \frac{\sum_{k=1}^{N} \hat{r}_{hijk}^{l}\left[\frac{\nu_{hij}^{l}+1}{\nu_{hij}^{l} + {y_{hijk}^{l}}^{2}} + \log\left(1 + \frac{{y_{hijk}^{l}}^{2}}{\nu_{hij}^{l}}\right)\right]}{\left(1 + \Psi\left(\frac{\nu_{hij}^{l}+1}{2}\right) - \Psi\left(\frac{\nu_{hij}^{l}}{2}\right)\right) \sum_{k=1}^{N} \hat{r}_{hijk}^{l}} \tag{6.35}$$

6.3.6 Example: Image Segmentation

We used the image data used in [68], taking as data 12 × 12 image blocks with lag 2, creating around 350,000 data vectors of length 144. We used the Generalized Gaussian shape adaptive mixture source model (initialized to $\rho = 1.5$), and learned two basis sets using the ICA mixture model. The bases were started at small perturbations of the square root of the covariance matrix, shown in Figure 6.4. We first ran ICA with only one (complete model), and the result is show in Figure 6.5. We then ran a two model mixture model, with the results shown in Figures 6.6 and 6.7.

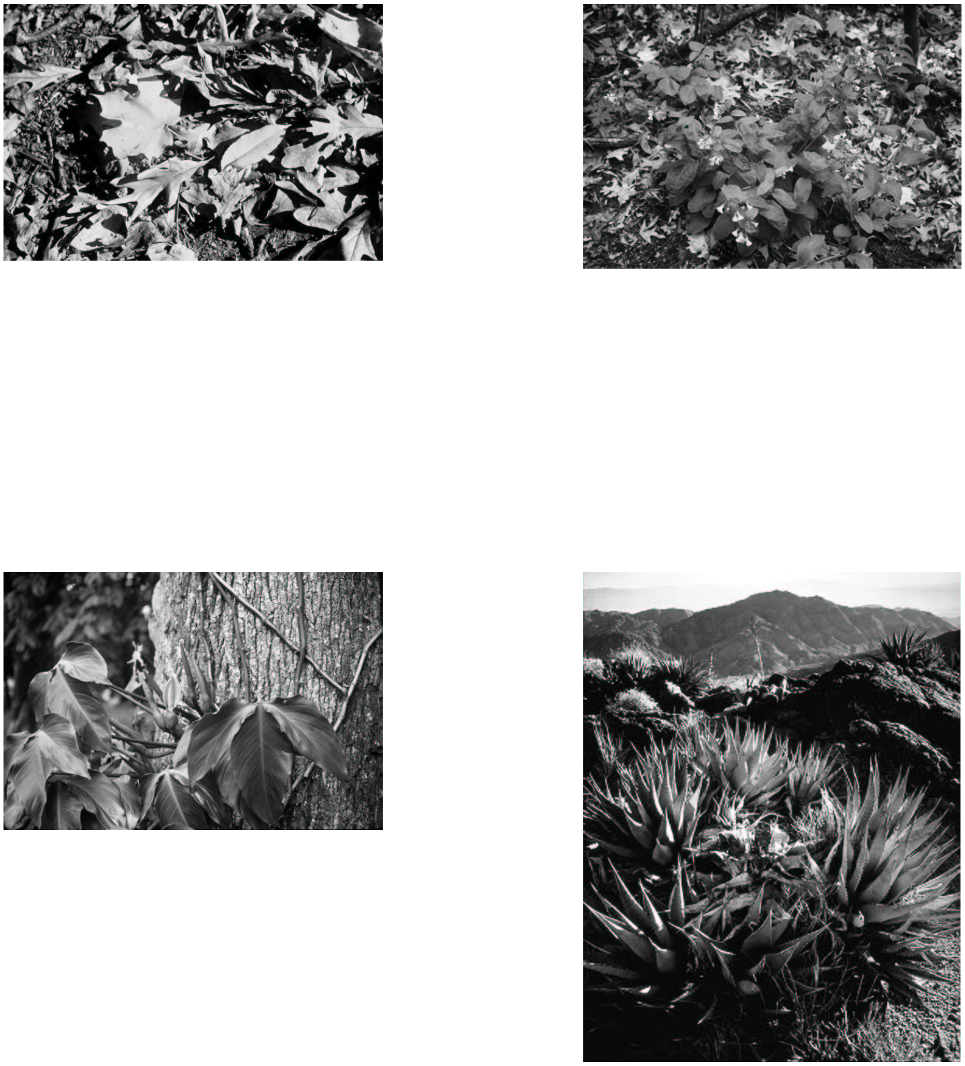


Figure 6.3: Some images used in the segmentation experiment.

We subsequently computed the likelihood of each image block under each model, and classified the pixels according to which model was more likely for

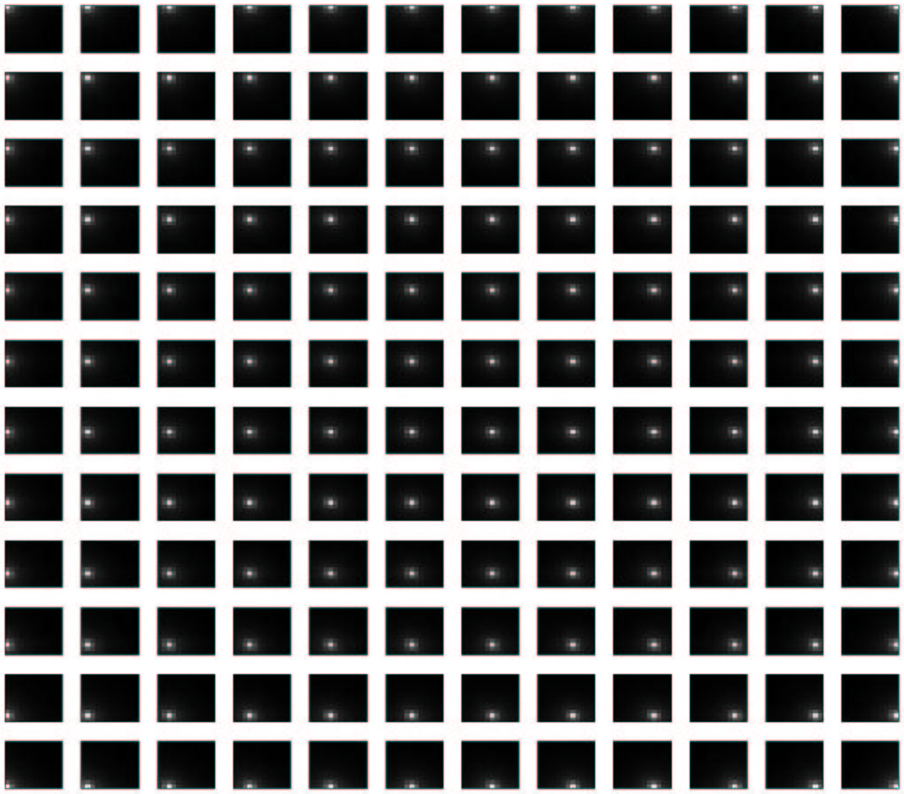


Figure 6.4: Components from the symmetric square root of the covariance matrix of the data blocks.

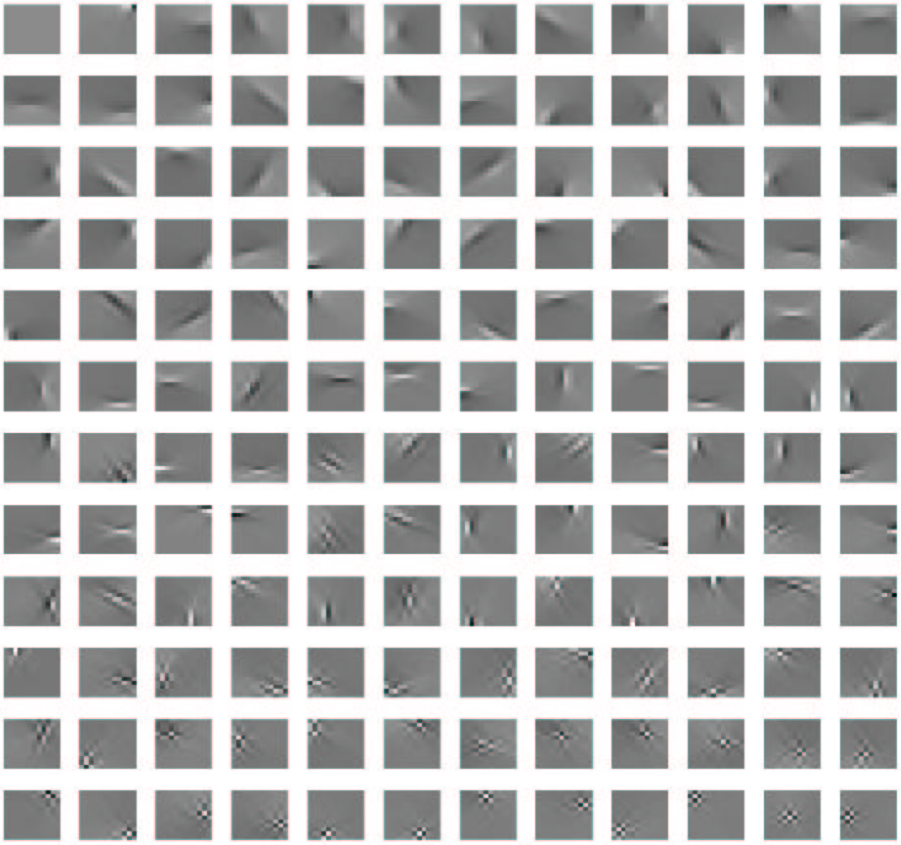


Figure 6.5: Single model learned using ICA with Generalized Gaussian adaptive shape mixture source priors.

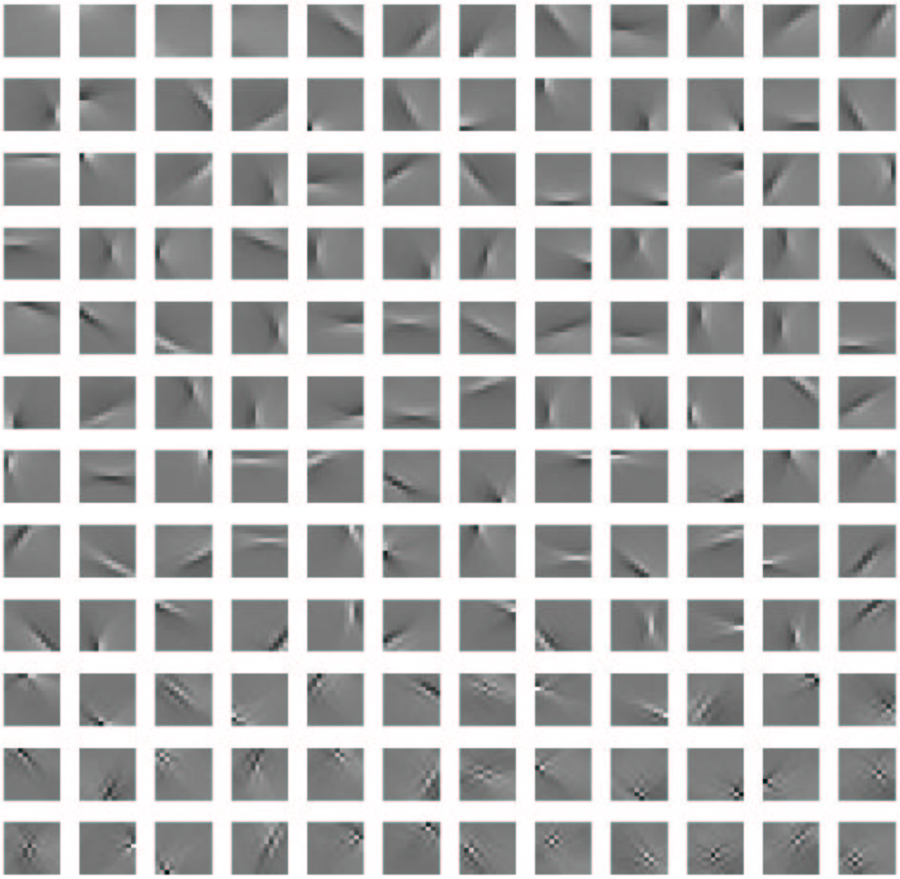


Figure 6.6: Model 1 learned by the two model ICA mixture model on the 12×12 block image data, with Generalized Gaussian adaptive shape mixture source priors.

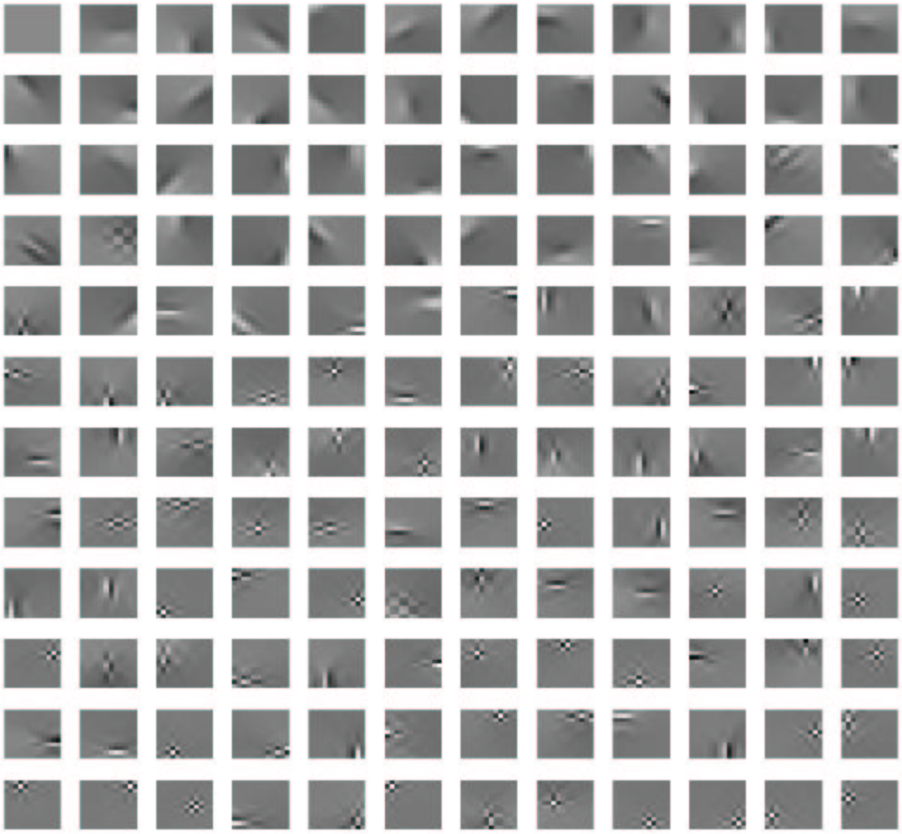
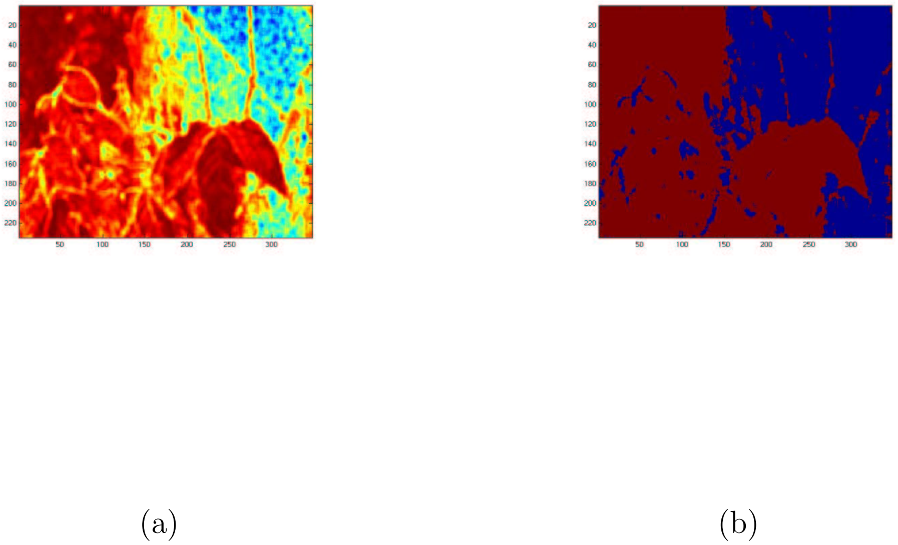


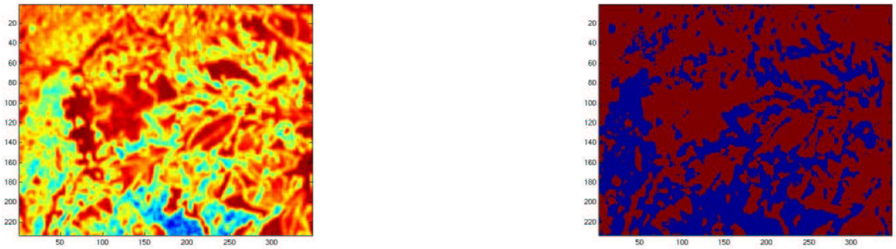
Figure 6.7: Model 2 learned by the two model ICA mixture model on the 12 × 12 block image data, with Generalized Gaussian adaptive shape mixture source priors.

the surrounding block. We performed the classification for two images. The log likelihood for the pixels under one of the models is shown on the left, and the segmented image is shown on the right.



(a) (b)

Figure 6.8: Segmentation of image based on model likelihood. (a) Raw log likelihood under model 1. (b) segmentation of the image by assigning pixels to the more likely of the two models. Apparently one of the models is more likely for the high frequency forest floor, while the other model is more likely of the lower frequency leaves, etc.



(a) (b)

Figure 6.9: Segmentation of image based on model likelihood. (a) Raw log likelihood under model 1. (b) segmentation of the image by assigning pixels to the more likely of the two models. On model seems to favor the high frequency tree bark, while the other models the low frequency leaves.

7

Linear Process Mixture Model

Since multichannel deconvolution is a linear operation, the output can be expressed as the product of a (possibly infinite sized) matrix and an input vector, just as in the instantaneous linear mixing case. The essential difference between the convolutive case and the instantaneous case is that the matrix in the convolutive linear operation has a particular structure, specifically a block Toeplitz structure, and thus resides in a particular subspace of matrices. Thus derivatives of functions of the block Toeplitz matrix $\overline{\mathbf{W}}$, in particular the derivative of $\log\det\overline{\mathbf{W}}$ in the likelihood, will differ from derivatives of unconstrained demixing linear operators. However, the block Toeplitz structure allows us to approximately calculate the determinant in terms of the blocks in a single row, using Szegö's limit formula concerning Toeplitz matrices, or by a similar argument that takes Toeplitz matrices as limits of circuilants [34, 85]. We can then calculate derivatives with respect to the individual blocks rather than the entire matrix. The block generalization of the Szegö theorem also allows us to efficiently calculate the likelihood using Fourier transforms, which we then use in an EM algorithm to adapt a mixture model involving multiple multichannel deconvolution filters. The natural gradient transformation $\Delta\overline{\mathbf{W}}\,\overline{\mathbf{W}}^T\overline{\mathbf{W}}$, and Newton algorithms can also be used as in instantaneous ICA.

7.1 The Convolutive Component Model

We define a multivariate linear process [47, 85].

Definition 2. *A real-valued discrete-time L^p* ***linear process****, $\mathbf{x}(t)$, is defined as a multivariate random process of the form,*

$$\mathbf{x}(t) = \sum_{k=-\infty}^{\infty} \mathbf{A}_k \, \mathbf{s}(t-k) \tag{7.1}$$

where $\mathbf{A}_k \in \mathbb{R}^{n \times n}$ with $\sum_k |[\mathbf{A}_k]_{ij}|^p < \infty$, and $\mathbf{s}(t)$ is i.i.d. for all t, and the components $s_i(t)$ of $\mathbf{s}(t)$ are independent (not necessarily identically distributed).

Note that the independent time series $\mathbf{s}(t)$ is distinct from the innovations representation of a second-order process. The innovation sequence is merely uncorrelated, and is only unique if it is causal. The independent series $\mathbf{s}(t)$ does not exist for every process, and is unique for all non-Gaussian linear processes [33, 93].

Equivalently, $\mathbf{x}(t)$ is seen as the sum of *convolutive components* $\mathbf{a}_i(t)$,

$$\mathbf{x}(t) = \sum_{i=1}^{n} s_i(t) * \mathbf{a}_i(t)$$

where the processes $s_i(t)$ are temporally i.i.d., and independent of each other. The convolution is interpreted as acting componentwise, i.e. $x_j(t) = \sum_{i=1}^{n} s_i(t) * a_{ij}(t)$, $j = 1, \ldots, n$. The components $\mathbf{a}_i(t)$ may be interpreted as independent "features" of the data $\mathbf{x}(t)$ which may be physically meaningful.

Given a multivariate discrete-time time series $\mathbf{x}(t)$, $t = 1, 2, \ldots, T$, we divide the length T series into a set of time series segments of length $2N + 1 \ll T$. In the model, each segment is generated independently by one of M linear process models. Dependency among the segments (for example among overlapping segments) can be accounted for by imposing a Markov dependence structure on the segments, but we shall assume here for simplicity that the segments are generated independently of one another, with model prior probabilities γ_h, $h = 1, \ldots, M$.

7.2 Asymptotic Likelihood

Given a finite dimensional random vector $\mathbf{s}$ with density $p_{\mathbf{s}}(\mathbf{s})$ and an invertible linear transformation $\mathbf{A} = \mathbf{W}^{-1}$, the density of the random vector $\mathbf{x} = \mathbf{A}\mathbf{s}$ is given by,

$$p_{\mathbf{x}}(\mathbf{x}) \;=\; \frac{1}{|\det \mathbf{A}|}\, p_{\mathbf{s}}(\mathbf{A}^{-1}\mathbf{x}) \;=\; |\det \mathbf{W}|\, p_{\mathbf{s}}(\mathbf{W}\mathbf{x})$$

The convolutive model (7.1) is a linear transformation of the process $\mathbf{s}(t)$. We suppose that the matrix filter is of finite duration, i.e. $\mathbf{A}_k = \mathbf{0}$ for $|k| > L$, where $L \ll N$. Let $N_0 = 2N + 1$. If we form the $nN_0 \times nN_0$ block Toeplitz matrix,

$$\overline{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_0 & \mathbf{A}_{-1} & \cdots & \\ \mathbf{A}_1 & \mathbf{A}_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{A}_{-1} \\ & \cdots & \mathbf{A}_1 & \mathbf{A}_0 \end{bmatrix}$$

and define the matrix,

$$\mathbf{X}_t \equiv \begin{bmatrix} \mathbf{x}(t-N) \; \cdots \; \mathbf{x}(t) \; \cdots \; \mathbf{x}(t+N) \end{bmatrix}$$

and define $\mathbf{S}_t$ similarly, then we have,

$$\mathrm{vec}(\mathbf{X}_t) \approx \overline{\mathbf{A}}\mathrm{vec}(\mathbf{S}_t)$$

where the equation is only approximate for the first L and last L vectors in $\mathbf{X}_t$ since $\overline{\mathbf{A}}$ is block banded. We suppose that the (two-sided) inverse matrix filter exists,

$$\mathbf{W}_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\sum_{l=-\infty}^{\infty} \mathbf{A}_l e^{-i\omega l} \right)^{-1} e^{i\omega k}\, d\omega$$

and can be approximated by the truncated filter with $\mathbf{W}_k = \mathbf{0}$ for $|k| > L$. Then we have,

$$\mathrm{vec}(\mathbf{S}_t) \approx \bar{\mathbf{W}}_N \mathrm{vec}(\mathbf{X}_t)$$

where we define,

$$\overline{\mathbf{W}}_N \equiv \overbrace{\begin{bmatrix} \mathbf{W}_0 & \mathbf{W}_{-1} & \cdots & \mathbf{W}_{-L} & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{W}_1 & \mathbf{W}_0 & \mathbf{W}_{-1} & \ddots & \mathbf{W}_{-L} & \mathbf{0} & \ddots \\ \vdots & \mathbf{W}_1 & \mathbf{W}_0 & \ddots & & \ddots & \ddots \\ \mathbf{W}_L & \ddots & \ddots & \ddots & \mathbf{W}_{-1} & & \\ \mathbf{0} & \mathbf{W}_L & & \mathbf{W}_1 & \mathbf{W}_0 & \ddots & \\ \mathbf{0} & \mathbf{0} & \ddots & & \ddots & \ddots & \\ \vdots & \ddots & \ddots & & & & \end{bmatrix}}^{2N+1 \text{ blocks}}$$

Then we have,

$$p_{\mathbf{x}}\big(\mathrm{vec}(\mathbf{X}_t)\big) \;\approx\; \big|\det\big(\overline{\mathbf{W}}_N \overline{\mathbf{W}}_N^T\big)\big|^{\frac{1}{2}}\, p_{\mathbf{s}}\big(\overline{\mathbf{W}}_N \mathrm{vec}(\mathbf{X}_t)\big)$$

For large N, the matrix $\overline{\mathbf{W}}_N \overline{\mathbf{W}}_N{}^T$ tends to the symmetric block Toeplitz matrix $\overline{\mathbf{R}}_N$ with blocks $\mathbf{R}_k$ given by,

$$\mathbf{R}_k = \sum_{l=-L}^{L} \mathbf{W}_l \mathbf{W}_{k+l}^T$$

for $k = -2L, \ldots, 2L$. To evaluate the determinant of $\overline{\mathbf{R}}_N$ asymptotically, we use the following extension of the classical Szegö limit theorem for Toeplitz matrices [40, 77].

We use the following notation. Let $G(\omega)$ be a matrix valued function mapping the interval $(-\pi, \pi)$ to the set of $m \times m$ Hermitian matrices, $\mathcal{H}_m$, and define,

$$A_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} G(\omega) e^{-ik\omega} d\omega, \quad k = \ldots, -1, 0, 1, \ldots$$

We say that the function G "generates" the process A_k. Let T_n denote the block Toeplitz matrix with A_0 on the block diagonal, and side length n blocks. Let $\sigma(G(\omega))$ denote the set of eigenvalues of the Hermitian matrix $G(\omega)$, and let $\sigma(T_n)$ denote the set of (real) eigenvalues of the Hermitian block Toeplitz matrix T_n. Then we have the following theorem [77, Thm. 3.4].

Theorem 16. *Suppose that $G(\omega) : (-\pi, \pi) \to \mathcal{H}_m$ is square integrable on $(-\pi, \pi)$, and $\{T_n\}$ is the set of block Toeplitz matrices generated by G. Then for any continuous function $F(\lambda)$ with compact support in $\mathbb{R}$, it holds that,*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{\lambda\in\sigma(T_n)} F(\lambda) \;=\; \frac{1}{2\pi} \int_{-\pi}^{\pi} \sum_{\lambda\in\sigma(G(\omega))} F(\lambda)\, d\omega$$

If $G(\omega)$ is positive definite, then $F = \log$ is continuous on a positive, compact interval containing the eigenvalues of T_n and $G(\omega)$, and we have,

$$\sum_{\lambda\in\sigma(T_n)} \log\lambda \;=\; \log\det T_n$$

and,

$$\sum_{\lambda\in\sigma(G(\omega))} \log\lambda \;=\; \log\det G(\omega)$$

Hence, we have

$$\lim_{n\to\infty} \frac{1}{n} \log\det T_n \;=\; \frac{1}{2\pi} \int_{-\pi}^{\pi} \log\det G(\omega)\, d\omega$$

In the notation of our model, we have the following asymptotic form for the Toeplitz determinant:

$$\lim_{N\to\infty} \left(\det \overline{\mathbf{R}}_N\right)^{1/N_0} = \exp\left(\frac{1}{4\pi}\int_{-\pi}^{\pi} \log\det S_{\mathbf{W}}(\omega)\, d\omega\right)$$

where $N_0 = 2N + 1$, and

$$S_{\mathbf{W}}(\omega) = \sum_k \mathbf{R}_k e^{-i\omega k} = \left(\sum_k \mathbf{W}_k e^{-i\omega k}\right)\left(\sum_k \mathbf{W}_k^T e^{i\omega k}\right)$$

and we have,

$$\det S_{\mathbf{W}}(\omega) = \left|\det\left(\sum_k \mathbf{W}_k\, e^{-i\omega k}\right)\right|^2$$

Thus for the asymptotic approximation to the likelihood of the $n \times N_0$ sample $\mathbf{X}_t$ we have,

$$\frac{1}{N_0} \log p_{\mathbf{x}}(\mathbf{X}_t) \;\approx\; \frac{1}{2\pi}\int_{-\pi}^{\pi} \log\left|\det\left(\sum_k \mathbf{W}_k e^{-i\omega k}\right)\right| d\omega \;+\; \frac{1}{N_0}\log p_{\mathbf{s}}\big(\overline{\mathbf{W}}_N \mathrm{vec}(\mathbf{X}_t)\big)$$

For the implementation, we define the source estimates,

$$\mathbf{y}_\tau = \sum_{k=-L}^{L} \mathbf{W}_k \mathbf{x}_{t+\tau-k}, \quad \tau = -N+L, \ldots, N-L$$

The number of $\mathbf{y}$ vectors is less than the number of $\mathbf{x}$ vectors since we discard the edges. Thus, given the data segment $\mathbf{X}_t = [\mathbf{x}_{t-N} \cdots \mathbf{x}_{t+N}]$, we define the approximate likelihood $q_\mathbf{x}$ by,

$$\log q_\mathbf{x}(\mathbf{X}_t) \;=\; \frac{N_1}{2\pi}\int_{-\pi}^{\pi} \log\left|\det\left(\sum_k \mathbf{W}_k e^{-i\omega k}\right)\right| d\omega \;+\; \sum_{\tau=-N+L}^{N-L} \sum_{i=1}^{n} \log q_i(y_{i\tau}) \quad (7.2)$$

where $N_1 = 2(N-L)+1$ and $q_i(y)$ is the approximating density of the ith source.

7.3 Maximizing the Likelihood

Let $\mathbf{C} \in \mathbb{C}^{n\times n}$ be square and non-singular. We use the complex derivative defined by

$$\frac{\partial g(\mathbf{C})}{\partial \mathbf{C}} = \frac{1}{2}\left(\frac{\partial g(\mathbf{C})}{\partial \mathrm{Re}\mathbf{C}} - i\frac{\partial g(\mathbf{C})}{\partial \mathrm{Im}\mathbf{C}}\right)$$

$\partial g/\partial \mathbf{C}^*$ is defined similarly but as a sum rather than a difference. If $g : \mathbb{C}^{n\times n} \to \mathbb{R}$ is a real valued function of a complex matrix, and $\mathbf{H} : \mathbb{R}^{n\times n} \to \mathbb{C}^{n\times n}$ is a complex matrix valued function of a real matrix, then we have the following chain rule

$$\begin{aligned}\frac{\partial}{\partial \mathbf{B}_{ij}} g\big(\mathbf{H}(\mathbf{B})\big) &= \mathrm{tr}\left(\frac{\partial g}{\partial \mathbf{H}}\frac{\partial \mathbf{H}^T}{\partial \mathbf{B}_{ij}} + \frac{\partial g}{\partial \mathbf{H}^*}\frac{\partial \mathbf{H}^{*T}}{\partial \mathbf{B}_{ij}}\right) \\ &= 2\,\mathrm{Re}\;\mathrm{tr}\left(\frac{\partial g}{\partial \mathbf{H}}\frac{\partial \mathbf{H}^T}{\partial \mathbf{B}_{ij}}\right)\end{aligned}$$

since $\frac{\partial g}{\partial \mathbf{H}^*} = \left(\frac{\partial g}{\partial \mathbf{H}}\right)^*$ and $\frac{\partial \mathbf{H}^*}{\partial \mathbf{B}_{ij}} = \left(\frac{\partial \mathbf{H}}{\partial \mathbf{B}_{ij}}\right)^*$ according to our assumptions.

Now, using the fact that,

$$\frac{\partial}{\partial \mathbf{C}} \log\det \mathbf{C}\mathbf{C}^H = \mathbf{C}^{-T}$$

taking the derivative of the Toeplitz determinant term in (7.2) with respect to $\mathbf{W}_k$, we get,

$$\frac{\partial(1^{\text{st}}\,\text{term})}{\partial[\mathbf{W}_k]_{ij}} = \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{Re}\;\mathrm{tr}\left[\left(\sum_l \mathbf{W}_l e^{-i\omega l}\right)^{-T} E_{ij}^T\, e^{-i\omega k}\right] d\omega$$

so that for the matrix derivative, we have,

$$\frac{\partial(1^{\text{st}}\,\text{term})}{\partial \mathbf{W}_k} = \text{Re}\,\frac{1}{2\pi}\int_{-\pi}^{\pi}\left(\sum_l \mathbf{W}_l^T e^{-i\omega l}\right)^{-1} e^{-i\omega k} d\omega \;=\; \mathbf{A}_{-k}^T$$

where E_{ij} is the matrix with 1 in the (i,j)th element and 0 elsewhere, and $\mathbf{A}_k$, $k = \ldots,-1,0,1,\ldots$ is kth element in the inverse filter of $\mathbf{W}_k$, $k = \ldots,-1,0,1,\ldots$. Thus the gradient of the determinant term is the block Teoplitz matrix $\overline{\mathbf{A}}^T$ yielding the natural gradient with respect to the determinant term,

$$\overline{\mathbf{A}}^T\overline{\mathbf{W}}^T\overline{\mathbf{W}} = \overline{\mathbf{W}} \tag{7.3}$$

For the derivative of the second term in (7.2) with respect to $\mathbf{W}_k$, we have,

$$\frac{\partial(2^{\text{nd}}\,\text{term})}{\partial \mathbf{W}_k} = \sum_\tau \mathbf{g}_\tau \mathbf{x}_{t+\tau-k}^T \tag{7.4}$$

where $\mathbf{g}_\tau \equiv -\nabla_{\mathbf{y}} \log q(\mathbf{y}_\tau)$. Multiplying the block Toeplitz matrix with blocks given by (7.4) by $\overline{\mathbf{W}}^T$ on the right, we get the block Toeplitz matrix with blocks

$$\sum_\tau\sum_l \mathbf{g}_\tau \mathbf{x}_{t+\tau-l}^T \mathbf{W}_{l-k}^T \;=\; \sum_\tau \mathbf{g}_\tau \mathbf{y}_{\tau-k}^T$$

Then multiplying this on the right by $\overline{\mathbf{W}}$, we get the block Toeplitz matrix with blocks,

$$\sum_\tau\sum_l \mathbf{g}_\tau \mathbf{y}_{\tau-l}^T \mathbf{W}_{k-l} \;=\; \sum_\tau \mathbf{g}_\tau \mathbf{u}_{\tau-k}^T \tag{7.5}$$

where we define,

$$\mathbf{u}_\tau \equiv \sum_{l=-L}^{L} \mathbf{W}_l^T \mathbf{y}_{\tau+l},\;\; \tau = -N+2L,\ldots,N-2L$$

Again the number of $\mathbf{u}$ vectors is smaller than the number of $\mathbf{y}$ vectors since we discard the edges. The natural gradient, including both terms (7.3) and (7.5), is then,

$$\Delta\mathbf{W}_k = \mathbf{W}_k - \frac{1}{TN_2}\sum_{t=1}^{T}\sum_{\tau=-N+2L}^{N-2L} \mathbf{g}_{\tau+k}\mathbf{u}_\tau^T$$

7.4 Convolutive Mixture model

We extend the instantaneous model described in the previous section to include convolutive mixing. Assuming independent segments $\mathbf{X}_t$, $t = 1, \ldots, T$, we have for the likelihood,

$$p\big(\{\mathbf{X}_t\}\big) = \prod_{t=1}^{T} \sum_{h=1}^{M} \gamma_h p\big(\mathbf{X}_t \,|\, h\big)$$

The parameters to be estimated are,

$$\theta = \big\{\gamma_h, \mathbf{W}_{hk}, \alpha_{hij}, \mu_{hij}, \beta_{hij}\big\}, \quad h = 1, \ldots, M,$$
$$k = -L, \ldots, L, \;\; i = 1, \ldots, n_h, \;\; j = 1, \ldots, m_{hi}$$

In this model, each segment $\mathbf{X}_t$ is generated (independently) by drawing a mixture component h' from the discrete probability distribution $P[h' = h] = \gamma_h$, $1 \leq h \leq M$, then drawing $\mathbf{X}_t$ from $p_{h'}(\mathbf{X}; \theta)$.

We define h_t to be the index chosen for the tth segment, and we define the random variable,

$$v_{ht} \equiv \begin{cases} 1, & h_t = h \\ 0, & \text{otherwise} \end{cases}$$

Let $\mathbf{V} \equiv \{v_{ht}\}$. Now, for the complete log likelihood of $\{\mathbf{X}_t\}_{t=1}^{T}$ and $\mathbf{V}$, we can write,

$$p\big(\{\mathbf{X}_t\},\mathbf{V}; \theta\big) = \prod_{t=1}^{T} \prod_{h=1}^{M} \gamma_h^{v_{ht}} p(\mathbf{X}_t | h; \theta)^{v_{ht}}$$

We define $j_{hit\tau}$ to be the source mixture component index chosen (independently of h_t) for the ith source of the hth model in τth index of the tth segment, and we define the random variables $z_{hijt\tau}$ by,

$$z_{hijt\tau} \equiv \begin{cases} 1, & j_{hit\tau} = j \\ 0, & \text{otherwise} \end{cases}$$

with $\mathbf{Z} \equiv \{z_{hijt\tau}\}$. We define,

$$y_{hijt\tau} \equiv \sqrt{\beta_{hij}} \left(\sum_{k=-L}^{L} \mathbf{w}_{hik}^{T} \mathbf{x}_{t+\tau-k} - \mu_{hij} \right)$$

Then we have

$$p(\mathbf{X}_t, \mathbf{Z}|h;\theta) = \exp\left(\frac{N_1}{2\pi}\int_{-\pi}^{\pi} \log|\det \mathbf{W}_h(\omega)|\, d\omega\right) \times \prod_{\tau=-N+L}^{N-L} \prod_{i=1}^{n_h}\prod_{j=1}^{m_{hi}} \Big[\alpha_{hij}\sqrt{\beta_{hij}}\, q_{hij}\big(y_{hijt\tau}\big)\Big]^{z_{hijt\tau}}$$

where $N_1 = 2(N-L)+1$ and $\mathbf{W}_h(\omega) \equiv \sum_k \mathbf{W}_{hk}\, e^{-i\omega k}$. For the joint distribution, or "complete likelihood," we have then,

$$p\big(\{\mathbf{X}_t\}, \mathbf{V}, \mathbf{Z};\theta\big) = \prod_{t=1}^{T}\prod_{h=1}^{M} \gamma_h^{v_{ht}} p(\mathbf{X}_t, \mathbf{Z}|h;\theta)^{v_{ht}}$$

For the variational free energy we have $F(q^l;\theta) = F^l(\theta) + H(\mathbf{V};\theta^l) + H(\mathbf{Z};\theta^l)$, where $H(\mathbf{V};\theta^l)$ and $H(\mathbf{Z};\theta^l)$ are the entropies of $\mathbf{V}$ and $\mathbf{Z}$ with the parameters set to θ^l, and,

$$F^l(\theta) \equiv \sum_{t=1}^{T}\sum_{h=1}^{M}\Bigg[\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}}\sum_{\tau=-N+L}^{N-L} E\big[v_{ht}z_{hijt\tau}|\mathbf{X}_t;\theta^l\big] \times \bigg(-\log\alpha_{hij} - \tfrac{1}{2}\log\beta_{hij} + f_{hij}(y_{hijt\tau})\bigg)\Bigg] + E\big[v_{ht}|\mathbf{X}_t;\theta^l\big]\bigg(-\log\gamma_h - \frac{N_1}{2\pi}\int_{-\pi}^{\pi}\log\big|\det\mathbf{W}_h(\omega)\big|\,d\omega\bigg)$$

where we define $f_{hij} \equiv -\log q_{hij}$. We define $\hat{z}^l_{hijk}$ to be the conditional expectation,

$$\begin{aligned}\hat{z}^l_{hijt\tau} &\equiv E\big[z_{hijt\tau}\big|\, v_{ht}=1, \mathbf{X}_t, ;\theta^l\big] \\ &= \frac{\alpha^l_{hij}\sqrt{\beta^l_{hij}}\, q_{hij}\big(y^l_{hijt\tau}\big)}{\sum_{j'=1}^{m_{hi}} \alpha^l_{hij'}\sqrt{\beta^l_{hij'}}\, q_{hij'}\big(y^l_{hij't\tau}\big)}\end{aligned}$$

where we use Bayes' rule to evaluate the (discrete) posterior distribution. Simlarly, the $\hat{v}^l_{ht} \equiv E[v_{ht}|\mathbf{X}_t;\theta^l]$ are given by,

$$\begin{aligned}\hat{v}^l_{ht} &= \frac{p(\mathbf{X}_t|v_{ht}=1;\theta^l)P[v_{ht}=1;\theta^l]}{\sum_{h'=1}^{M} p(\mathbf{X}_t|v_{h't}=1;\theta^l)P[v_{h't}=1;\theta^l]} \\ &= \frac{\gamma^l_h p(\mathbf{X}_t|h;\theta^l)}{\sum_{h'=1}^{M}\gamma^l_{h'}p(\mathbf{X}_t|h';\theta^l)}\end{aligned}$$

and we have,

$$\begin{aligned} E\big[v_{ht}\, z_{hijt\tau}|\mathbf{X}_t;\theta^l\big] &= P\big[v_{ht}\!=\!1, z_{hijt\tau}\!=\!1\,|\,\mathbf{X}_t;\theta^l\big] \\ &= P\big[v_{ht}\!=\!1\,|\,\mathbf{X}_t;\theta^l\big]P\big[z_{hijt\tau}\!=\!1\,|\,v_{ht}\!=\!1,\mathbf{X}_t;\theta^l\big] \\ &= \hat{v}^l_{ht}\hat{z}^l_{hijt\tau} \end{aligned}$$

Minimizing F over γ_h and α_{hij} subject to the positivity and normalization constraints, we get,

$$\gamma^{l+1}_h = \frac{1}{T}\sum_{t=1}^{T}\hat{v}^l_{ht}\,, \quad \alpha^{l+1}_{hij} = \frac{1}{TN_1\gamma^{l+1}_h}\sum_{t=1}^{T}\hat{v}^l_{ht}\sum_{\tau=-N+L}^{N-L}\hat{z}^l_{hijt\tau}$$

where $N_1 = 2(N-L)+1$.

Now, to determine the updates for μ_{hij} and β_{hij}, we use the strong super-Gaussianity inequality to replace $f_{hij}(y_{hijt\tau})$ in $F^l(\theta)$ by $\frac{1}{2}\xi^l_{hijt\tau}y^2_{hijt\tau}$, where,

$$\xi^l_{hijt\tau} \equiv \frac{f'_{hij}(y^l_{hijt\tau})}{y^l_{hijt\tau}} \tag{7.6}$$

Our "surrogate" free energy is then,

$$\begin{aligned} \tilde{F}^l(\theta) = \sum_{t=1}^{T}\sum_{h=1}^{M}\hat{v}^l_{ht}\Bigg[\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}}\sum_{\tau=-N+L}^{N-L}\hat{z}^l_{hijt\tau}\,\times \\ \Big(-\log\alpha_{hij} - \tfrac{1}{2}\log\beta_{hij} + \tfrac{1}{2}\xi^l_{hijt\tau}y^2_{hijt\tau}\Big)\Bigg] + \\ \hat{v}^l_{ht}\Big(-\log\gamma_h - \frac{N_1}{2\pi}\int_{-\pi}^{\pi}\log\big|\det\mathbf{W}_h(\omega)\big|\,d\omega\Big) \end{aligned}$$

Minimizing $\tilde{F}^l$ with respect to μ_{hij} and β_{hij} guarantees, using the strong super-Gaussianity inequality, that,

$$F(q^l;\theta^{l+1}) - F(q^l;\theta^l) \;\le\; \tilde{F}(q^l;\theta^{l+1}) - \tilde{F}(q^l;\theta^l) \;\le\; 0$$

and thus that $F(q^l;\theta)$ is decreased as required by the EM algorithm. As in the Gaussian mixture case, the optimal value of μ_{hij} does not depend on β_{hij}. The updates are found to be,

$$\begin{aligned}\mu_{hij}^{l+1} &= \frac{\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}\xi_{hijt\tau}^{l}\Big(y_{hijt\tau}^{l}/\sqrt{\beta_{hij}^{l}}+\mu_{hij}^{l}\Big)}{\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}\xi_{hijt\tau}^{l}} \\ &= \mu_{hij}^{l} + \frac{\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}f_{hij}'\big(y_{hijt\tau}^{l}\big)}{\sqrt{\beta_{hij}^{l}}\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}\xi_{hijt\tau}^{l}}\end{aligned}$$

and,

$$\begin{aligned}\beta_{hij}^{l+1} &= \frac{\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}}{\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}\xi_{hijt\tau}^{l}{y_{hijt\tau}^{l}}^{2}/\beta_{hij}^{l}} \\ &= \frac{\beta_{hij}^{l}\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}}{\sum_{t=1}^{T}\hat{v}_{ht}^{l}\sum_{\tau=-N+L}^{N-L}\hat{z}_{hijt\tau}^{l}f_{hij}'\big(y_{hijt\tau}^{l}\big)y_{hijt\tau}^{l}}\end{aligned}$$

Since $\xi = f'(y)/y$ may go to infinity at $y = 0$ for strongly super-gaussian densities, we have eliminated it from the updates except in the denominator of the μ update, where ξ becoming infinite (with $f'(y)$ remaining bounded) has the effect of keeping μ constant.

Now we make the definitions,

$$\mathbf{b}_{ht\tau}^{l} \equiv \sum_{k=-L}^{L}\mathbf{W}_{hk}^{l}\mathbf{x}_{t+\tau-k}$$

for $\tau = -N+L, \ldots, N-L$, and

$$\mathbf{u}_{ht\tau}^{l} \equiv \sum_{k=-L}^{L}{\mathbf{W}_{hk}^{l}}^{T}\mathbf{b}_{ht(\tau+k)}^{l}$$

for $\tau = -N+2L, \ldots, N-2L$. We define

$$y_{hijt\tau}^{l} \equiv \sqrt{\beta_{hij}^{l}}\Big(b_{hit\tau}^{l} - \mu_{hij}\Big)$$

for $\tau = -N+L, \ldots, N-L$ and we define the vector $\mathbf{g}_{ht\tau}^{l}$ such that,

$$g_{hit\tau}^{l} \equiv \hat{v}_{ht}^{l}\sum_{j=1}^{m_{hi}}\hat{z}_{hijt\tau}^{l}\sqrt{\beta_{hij}^{l}}\,f_{hij}'\big(y_{hijt\tau}^{l}\big)$$

Then the natural gradient direction for $\mathbf{W}_{hk}$ is given by,

$$\Delta\mathbf{W}_{hk} = \gamma_{h}^{l+1}\mathbf{W}_{hk}^{l} - \frac{1}{TN_{2}}\sum_{t=1}^{T}\sum_{\tau=-N+2L}^{N-2L}\mathbf{g}_{ht(\tau+k)}^{l}{\mathbf{u}_{ht\tau}^{l}}^{T}$$

where we have replaced N_1 by N_2 to reduce the bias since we are discarding the edges. To express this in a form more efficient for computation, we define the subset of $\mathbf{X}_t$,

$$\tilde{\mathbf{X}}_{t,k} \equiv [\mathbf{x}_{t-k-N+L} \cdots \mathbf{x}_{t-k+N-L}]$$

Then, for $\mathbf{B}_t \equiv [\mathbf{b}_{-N+L} \cdots \mathbf{b}_{N-L}]$, we have,

$$\mathbf{B}_t = \sum_{k=-L}^{L} \mathbf{W}_{hk} \tilde{\mathbf{X}}_{t,k}$$

Now if we define

$$\tilde{\mathbf{B}}_{t,k} \equiv [\mathbf{b}_{k-N+2L} \cdots \mathbf{b}_{k+N-2L}]$$

and put $\mathbf{U}_t \equiv [\mathbf{u}_{-N+2L} \cdots \mathbf{u}_{N-2L}]$, then we have,

$$\mathbf{U}_t = \sum_{k=-L}^{L} \mathbf{W}_{hk}^T \tilde{\mathbf{B}}_{t,k}$$

Finally, if we put

$$\tilde{\mathbf{G}}_{t,k} \equiv [\mathbf{g}_{k-N+2L} \cdots \mathbf{g}_{k+N-2L}]$$

then we have for the natural gradient with respect to $\mathbf{W}_{hk}$,

$$\Delta \mathbf{W}_{hk} = \gamma_h^{l+1} \mathbf{W}_{hk} - \frac{1}{TN_2} \sum_{t=1}^{T} \tilde{\mathbf{G}}_{t,k} \mathbf{U}_t^T$$

where $N_2 = 2(N - 2L) + 1$.

We approximate the integral in (7.2) by a Riemann sum using the DFT, which samples the DTFT defined by the integral. If we make the definitions,

$$\begin{aligned} Q_{hijt\tau}^l &\equiv \alpha_{hij}^l \sqrt{\beta_{hij}^l}\, q_{hij}\left(y_{hijt\tau}^l\right) \\ D_h^l &\equiv \frac{N_1}{N_F} \sum_{n=1}^{N_F} \log \left| \det \sum_{k=-L}^{L} \mathbf{W}_{hk}^l \, e^{-i2\pi nk/N_F} \right| \\ P_{ht}^l &\equiv \gamma_h^l \exp(D_h^l) \prod_{\tau=-N+L}^{N-L} \prod_{i=1}^{n_h} \sum_{j=1}^{m_{hi}} Q_{hijt\tau}^l \end{aligned}$$

where N_F is the DFT length, then the $\hat{v}_{ht}$ and $\hat{z}_{hijt\tau}$ updates can be written,

$$\hat{v}_{ht}^l = \frac{P_{ht}^l}{\sum_{h'=1}^{M} P_{h't}^l}, \quad \hat{z}_{hijt\tau}^l = \frac{Q_{hijt\tau}^l}{\sum_{j'=1}^{m_{hi}} Q_{hij't\tau}^l} \tag{7.7}$$

Then,

$$\gamma_h^{l+1} = \frac{1}{T}\sum_{t=1}^{T} \hat{v}_{ht}^{l}\,, \quad \alpha_{hij}^{l+1} = \frac{1}{TN_1\gamma_h^{l+1}}\sum_{t=1}^{T} \hat{v}_{ht}^{l} \sum_{\tau=-N+L}^{N-L} \hat{z}_{hijt\tau}^{l}$$

The log likelihood of θ^l given $\{\mathbf{X}_t\}$, which we denote by $\bar{P}^l$, is calculated as,

$$\bar{P}^l \;=\; \sum_{t=1}^{T} \log\left(\sum_{h=1}^{M} P_{ht}^{l}\right) \tag{7.8}$$

$\bar{P}^l$ increases monotonically with iteration l.

7.5 Frequency Domain Formulation

Since the updates involve multichannel convolutions, it may be more efficient for longer filters to use fast Fourier transforms (FFTs) to perform convolution. Specifically, for the sources, $\mathbf{Y}$, we have,

$$\mathbf{B}_h(\omega) = \mathbf{W}_h(\omega)\mathbf{X}_h(\omega)$$

and,

$$\mathbf{U}_h(\omega) = \mathbf{W}_h^H(\omega)\mathbf{Y}_h(\omega)$$

where the superscript H denotes Hermitian, or conjugate transpose. We must revert to the time domain temporarily, however, to calculate the gradient function $\mathbf{G}$, forming $\mathrm{FFT}(g(\mathrm{IFFT}(\mathbf{Y}(\omega))))$. Then we have,

$$\Delta\mathbf{W}_h(\omega) = \gamma_h\mathbf{W}_h(\omega) - \frac{1}{TN}\,\mathbf{G}_h(\omega)\mathbf{U}_h^H(\omega)$$

The additional computation $\mathrm{FFT}(g(\mathrm{IFFT}(\mathbf{Y}(\omega))))$ distinguishes the time-domain based frequency formulation from algorithms which perform instantaneous ICA on individual frequency channels separately. The lack of any computations coupling the updates leads to the so called permutation problem, which results from the invariance of the ICA likelihood function with respect to permutations. Thus the sources determined by separate frequency channel ICA algorithms are not necessarily in the same order, and reconstruction of the time domain source requires matching the frequency sources across each channel. In the present formulation, we do not encounter the permutation problem, since the updates are kept consistent by the time domain joint optimization, and the frequency domain is employed only for efficiency of computation. This formulation is actually very similar to the "hybrid" frequency/time domain algorithm of Attias and Schreiner [5, §3.3], which employs a heuristic combination of time domain and frequency domain cost functions.

7.6 Multivariate Multichannel Blind Deconvolution

The determinant formula for Block Toeplitz matrices can be extended to multidimensional fields as well. The asymptotic formula for the determinant of a multidimensional multichannel convolution operator is,

$$\lim_{N\to\infty} \frac{1}{N} \log\det \overline{\mathbf{R}}_N = \frac{1}{(4\pi)^d} \int_{-\pi}^{\pi} \cdots \int_{-\pi}^{\pi} \log\det S_{\mathbf{W}}(\omega_1, \ldots, \omega_d)\, d\omega_1 \cdots d\omega_d$$

where $N_0 = 2N + 1$, and

$$S_{\mathbf{W}}(\omega_1, \ldots, \omega_d) = \sum_{\tau_1} \cdots \sum_{\tau_d} \mathbf{R}(\tau_1, \ldots, \tau_d) e^{-i(\omega_1\tau_1 + \cdots + \omega_d\tau_d)} =$$

$$\left(\sum_{\tau_1} \cdots \sum_{\tau_d} \mathbf{W}(\tau_1, \ldots, \tau_d) e^{-i(\omega_1\tau_1 + \cdots + \omega_d\tau_d)} \right) \left(\sum_{\tau_1} \cdots \sum_{\tau_d} \mathbf{W}^T(\tau_1, \ldots, \tau_d) e^{i(\omega_1\tau_1 + \cdots + \omega_d\tau_d)} \right)$$

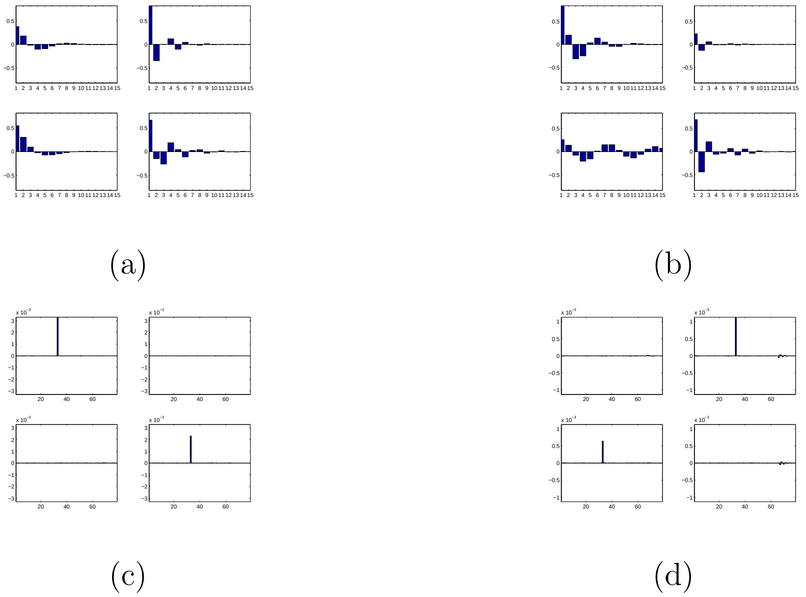
and we have,

$$\det S_{\mathbf{W}}(\omega) = \left| \det\left(\sum_{\tau_1} \cdots \sum_{\tau_d} \mathbf{W}(\tau_1, \ldots, \tau_d) e^{-i(\omega_1\tau_1 + \cdots + \omega_d\tau_d)} \right) \right|^2$$

The multivariate integration can again be approximated by Riemann integration, assuming computational feasibility of making a d-dimensional grid. An algorithm essentially identical to the one dimensional field case results, where one dimensional multichannel convolutions are replaced by d-dimensional multichannel convolutions with the d dimensional field of matrices making up the filter. The multidimensional multichannel deconvolution filters can again be inverted using the Fourier transform, to get the d-dimensional field of mixing matrices.

7.7 Multichannel Deconvolution Experiment

We verified the algorithm with a simple 2 channel experiment using a mixture model with two multichannel mixing filters, shown in Figure 1. We generated 20000 iid Laplacian time points, and then generated the first half of the test data from the first model and the second half from the second model. We learned two multichannel unmixing filters of length 60. The unmixing filters were initialized to identity, and the model source densities were adapted with the Generalized Gaussian mixture family, and initialized with two mixture components with zero location, unit scale, and shape parameter 1.5. The learned filters were convolved with the known mixing filters to assess the convergence to the true inverse filter. As shown in Figure 1, the correct inverse filters were learned.



(a)

(b)

(c)

(d)

Figure 7.1: Toy experiment with two mixing multichannel filters (a) and (b). In (c) and (d) are plotted the multiple convolution with the learned deconvolving filters.

8

Application to Analysis of EEG

Electro-encephalograph (EEG) data is recorded from a sensor cap worn by the experimental subject. The sensors record electromagnetic wave voltages from source produced in the brain, as well as from the heart and muscles, and from ambient power line or other EM phenomena. An example of raw EEG data is shown in Figure 8.1.

The instantaneous ICA mixing model holds for EEG data since the electromagnetic waves that superpose to form the raw EEG recordings travel at the speed of light, and practical sampling rates cannot distinguish such short delays in the arrival time of the source signals at the sensors. The ICA sources themselves are not i.i.d., but standard ICA has been found to be successful in decomposing raw EEG recordings $\mathbf{x}(t) = [x_1(t) \cdots x_n(t)]^T$ into a superposition of source waveforms $\mathbf{s}(t) = [s_1(t) \cdots s_n(t)]^T$ [73, 74]. Figure 8.2 shows the separated source activations corresponding to the raw EEG in Figure 8.1. It is quite remarkable how well ICA is able to separate out eyeblinks (source 2) and heart beat (source 5) leaving other periodic waveforms clearly visible, whereas the raw EEG is dominated by the larger variance sources.

The columns $\mathbf{a}_1, \ldots, \mathbf{a}_n$ of $\mathbf{A}$ represent the instantaneous impulse response of the generating current on the sensors, which is the characteristic electromagenetic field distribution on the sensors associated with source currents in particular

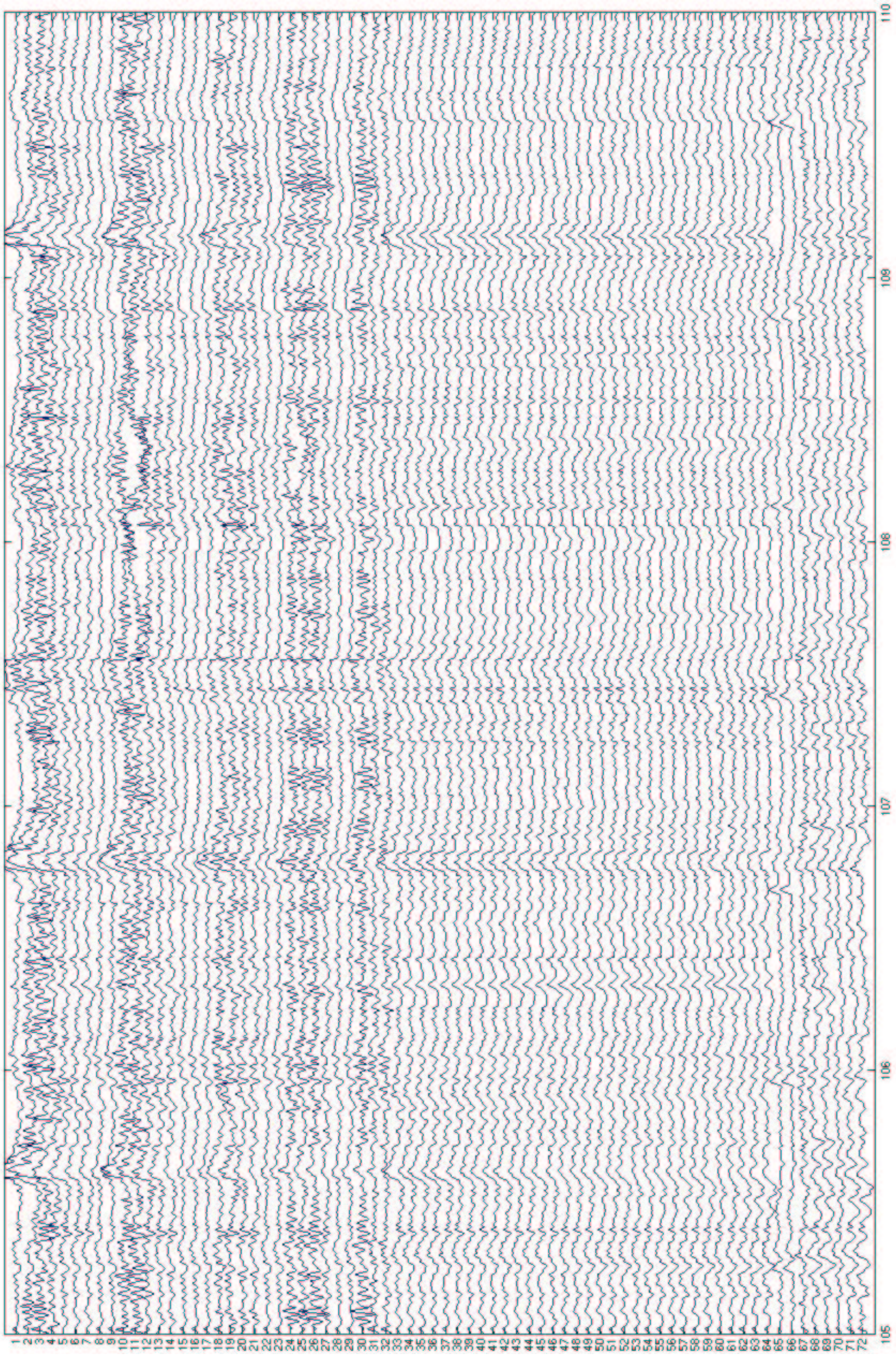


Figure 8.1: Plot of raw EEG data.

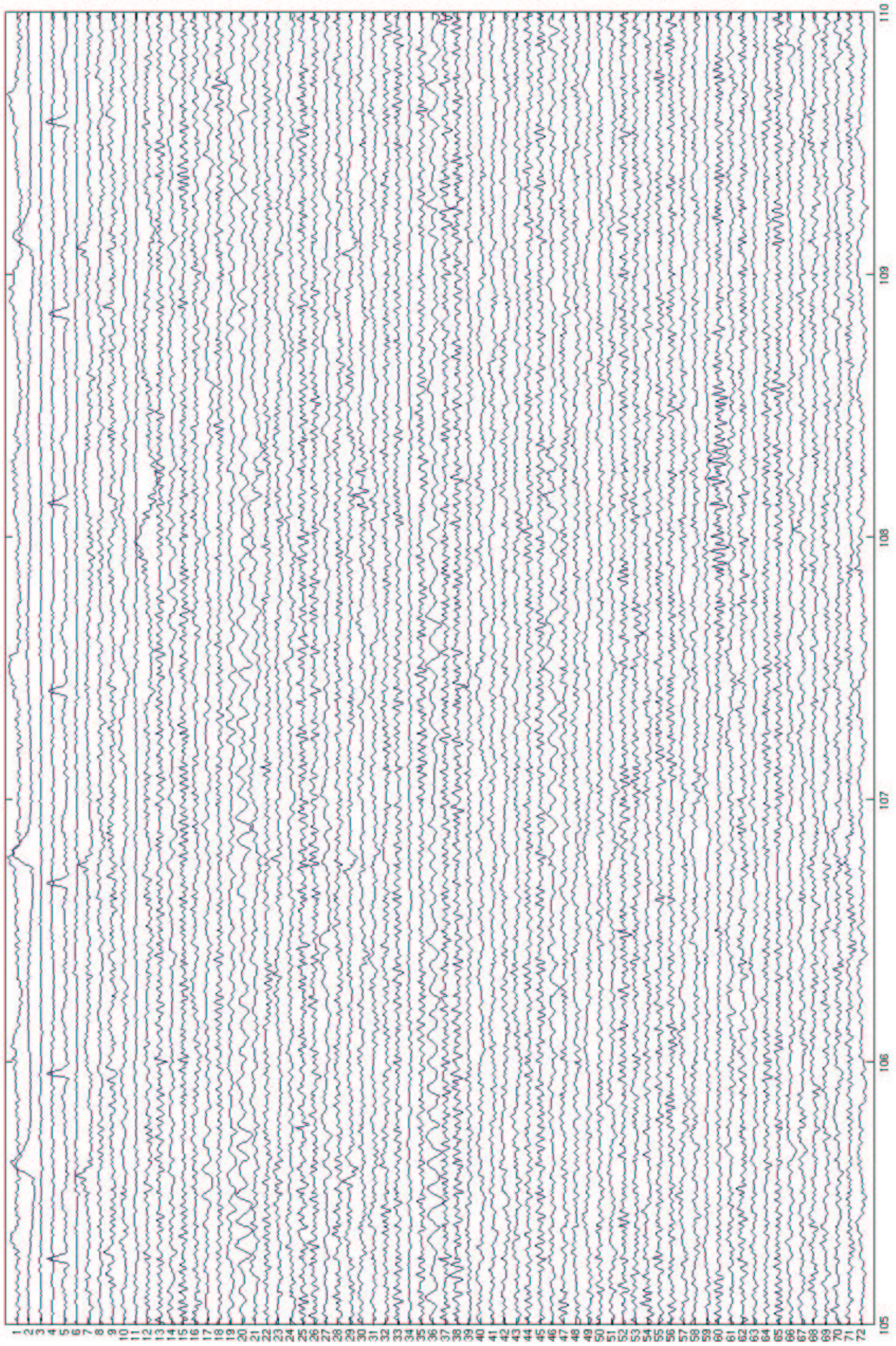


Figure 8.2: Plot of separated EEG source activations for same time frame as Figure 8.1.

locations in the brain, or with particular source current distributions. The inverse problem of determining the actual current distribution associated with a given sensor response is known since Helmholtz to be ill-posed, i.e. the generating current distribution associated with two dimensional sensor array response $\mathbf{a}_j$ is not unique. However, given reasonable certain prior assumptions on the localization or dipolarity of the sources, generating current sources can often be determined [73, 74].

The point, though, is that the characteristic responses are determined in a blind manner, assuming only the independence of the generating current sources. This decomposes the problem of identifying generating current sources for in recording from an experiment into two parts: (1) determination of responses associated with independent generating currents, and (2) determining the actual generating current distribution. Indeterminacy of the second problem does not have any affect of the determinacy of the first problem. In fact, it may be sufficient, as in Brain Computer Interfacing (BCI) for example, merely to determine the independent responses $\mathbf{a}_i$, $i = 1, \ldots, n$, and their activation signals $s_i(t)$, $i = 1, \ldots, n$, and detect and respond to "events" in the activation signal alone. Still, the gain for neurobiology is that, to the extent that the generating current sources can be determined, they are determined in a completely blind manner.

An alternative is to create a dictionary $\mathbf{A} \in m \times n$ of $n \gg m$ sensor responses $\mathbf{a}_j \in \mathbb{R}^m$, for all possible current source locations and orientations, and attempt to find sparse solutions to the underdetermined problem $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$. Drawbacks of this approach are that one must limit à priori, for combinatorial reasons, the possible generating sources with whose responses one populates the dictionary $\mathbf{A}$, for example to single dipole sources on a predetermined grid and with a predetermined set of possible orientations, and that for a large number of sensors m, and long observed sequences, one must perform an iteratively least squares algorithm or other nonlinear function optimization routine to find the sources $\mathbf{s}(t)$ for each observed time point $t = 1, \ldots, T$. In the case we consider, the number of sensors m is 254, and T is normally in the hundreds of thousands or longer. Processing such sequences becomes computationally difficult to do in a reasonable amount of time, even offline, but particularly for possible real time applications.

Assuming a complete basis $\mathbf{A} \in \mathbb{R}^{n \times n}$, on the other hand, and assuming low noise, i.e. low sensor noise or channel noise, which is reasonable with current technology, or combined with low pass filtering or shrinkage techniques, allows one to determine the sources $\mathbf{s}(t)$ simply through matrix multiplication $\mathbf{s}(t) = \mathbf{W}\mathbf{x}(t)$. The disadvantage of the standard complete basis ICA model, is that one is limited to decomposing the entire observation sequence of hundreds of thousands of time points of the usually non-stationary observed signal $\mathbf{x}(t) \in Rn$, into n sources, where n is on the order of hundreds.

It may actually be that, by limiting observation to particular times, for example before and after a particular stimulus in a repeated experiment, the number of sources may be on the order of hundreds, or fewer. The non-stationarity problem, however, will generally be an issue, both for the blind approach, and the sparse coding with known dictionary approach, unless one is able to populate the dictionary with sources transformed by various transformations associated with known non-stationary phenomena.

To address the problems of source number limitation in the complete case, as well as non-stationarity, we propose a linear process mixture model to model the generating sources, as well as characteristic non-stationary interference segments that modulate the sources. Such an approach can also model the temporal dependence of the sources using a linear process model for each source.

We performed experiments using the linear process mixture model and on EEG data collected from experiments by Julie Onton and Scott Makeig at the Swartz Center for Computational Neuroscience. We used three datasets described below.

1. The first dataset, referred to as “twoback”, is an experiment in which a

subject, wearing and EEG cap and seated at a computer, is shown a sequence of letters, and after each letter presentation, the subject responds with a left or right mouse click as follows: if the letter presented is the same as the one two letters ago (not the last one, the one before that) then left click, if not then right click. The computer then gives with auditory feedback as follows: If the subject's response is correct, a bell sounds indicating correct. If the subject's response is incorrect (the subject makes an mistake in memory exceeds time limit) then a buzzer sound plays indicating incorrect.

The data consists of 283,900 samples at $250Hz$ from 71 channels. The samples are made up of 668 concatenated segments of length 425, or 1.7 seconds. The segments are time locked to the computer's feedback after the subjects response, from 0.7 seconds before the feedback to 1.0 second after.

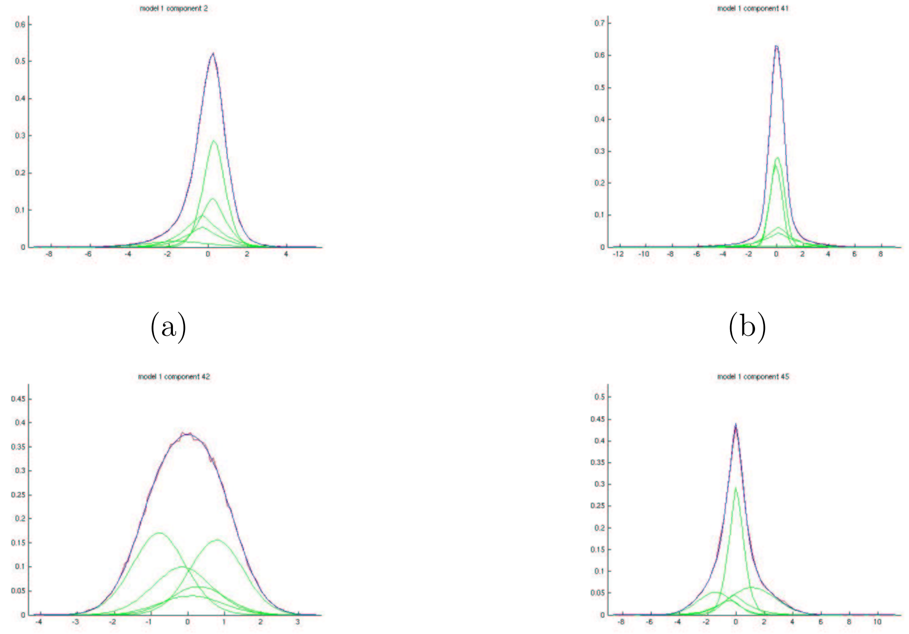
2. The second dataset, referred to as "wordfinger", consists of two tasks performed by the same subject. In the first task, the subject reads a sequence of words printed on a computer screen, and responds after each word with a right or left mouse click depending on whether the subject thinks the word is new or not. A feedback tone is given for correct, and a buzz when the subject is wrong. In the second task, the subject is given a sequence commands to individual fingers on the right or left hand. The finger movement is indicated associating adjacent sets of keys on the computer keyboard with the fingers on each hand. As in the previous task, the subject is given auditory feedback, tone for correct and buzz for wrong.

3. The third dataset, referred to as "reaching" is a reaching experiment. A subject performed in two conditions, differing by the starting point of the movement. In each trial, a target LED was turned on and the subject was instructed to touch the target in one smooth movement. The LED on was marked with an event code for left, middle and right target. The LED stayed on until the target was touched or for 2500 ms in case of target missed. After

a random interval of 800-1200 ms, the next target appeared.

8.1 Generalized Gaussian Mixture source prior

The EEG sources have a variety of probability densities, exhibiting skew, multimodality and other features that cannot be captured with unimodal symmetric densities. Some examples of unusual histograms of separated sources are plotted in Figure 8.3. The green



(a) (b)

(c) (d)

Figure 8.3: Converged source distributions for some components using a five mixture Generalize Gaussian source model, showing (a) skewed, (b) heavy-tailed (c) sub-gaussian, and (d) sharply peaked densities. The mixture components of the model density are plotted in green, and the model density which is there sum is plotted in blue. The empirical histogram is plotted in red, before the model density, and is almost invisible as it is completely covered by the model density, showing exact agreement.

8.2 ICA Mixture Model with Adaptive Source Priors

We tested the instantaneous mixture model on the twoback and wordfinger datasets. The plots in Figures 8.6 and 8.8 show the data arranged such that each row is a a segment of EEG, and the segments are stacked on top of each other in increasing temporal order from the bottom. The time points within each trial are classified and assigned to one of the models, and colored the color corresponding to that model.

8.3 Linear Process Mixture Model Examples

We performed mixture deconvolution on a particular separated source with an alpha wave spectrum, i.e. a spectrum containing a spectral peak around 10Hz. The alpha signals are characterized by sustained alpha bursts, rather than persistent alpha activity. Thus the locally stationary model thus fits well, as the signal may be divided into segments in which alpha was active and segments in which alpha segments is non-active.

We ran a single channel linear mixture model decomposition with two generating filter models, each with a different adaptive i.i.d. driving process of two Generalized Gaussians. We decomposed the signal into segments by classifying time points according to which model was most likely to have generated the segment of which the time point is the center. The algorithm was able to successfully segment the signal into alpha and non-alpha segments. This is verified by showing that the spectrum of classified time points of one model contained an alpha peak, while the spectrum of the time points assigned to the other model has no peak at alpha.

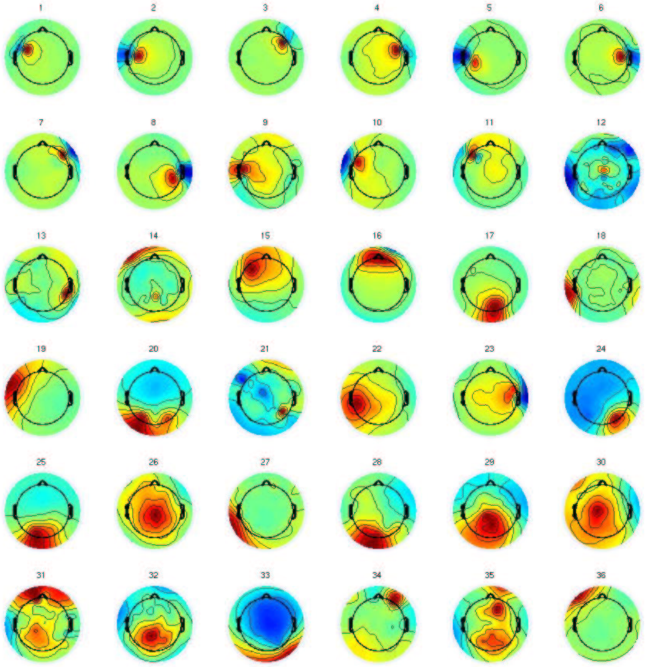


Figure 8.4: Scalp maps for the one of the two models learned from the twoback dataset. The components are arranged in order of maximum mutual information between the component activation y_{hik} and the probability signal. For two models, $h = 1, 2$, we have $v_{1k} = 1 - v_{2k}$, so there is really only one model probability signal. Apparently this model represents periods of muscle activity, while the other may represent task related components.

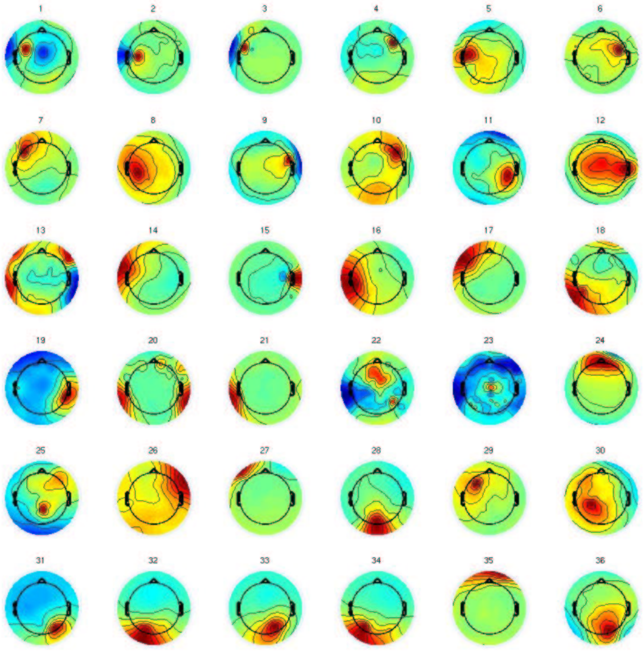


Figure 8.5: Scalp maps for the two models learned from the twoback dataset. The components are arranged in order of maximum mutual information between the component activation y_{hik} and the probability signal.

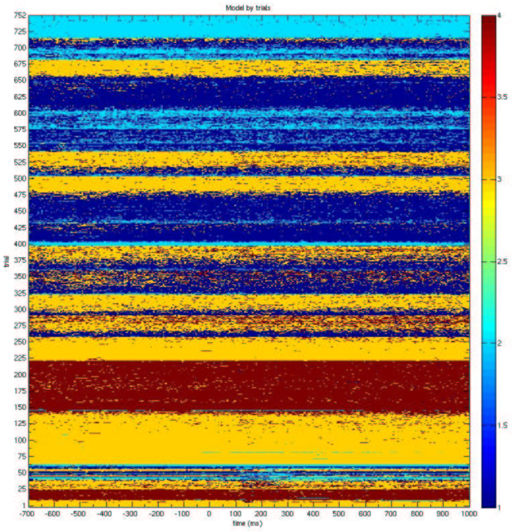
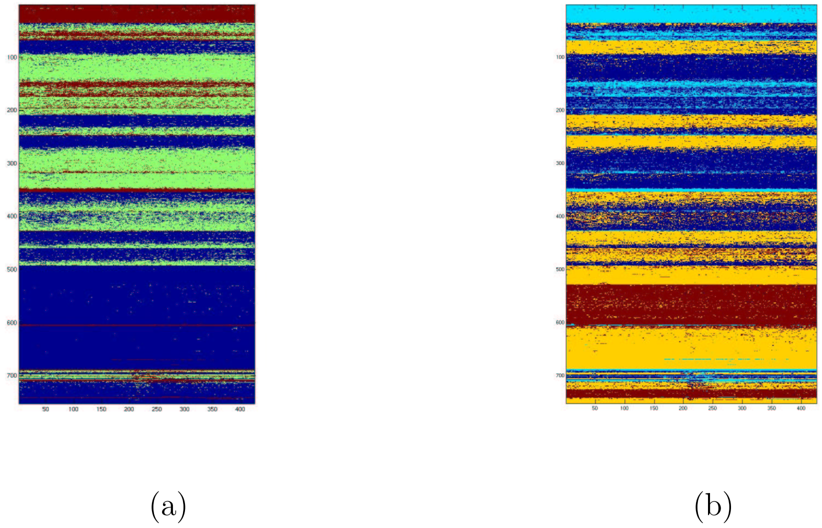


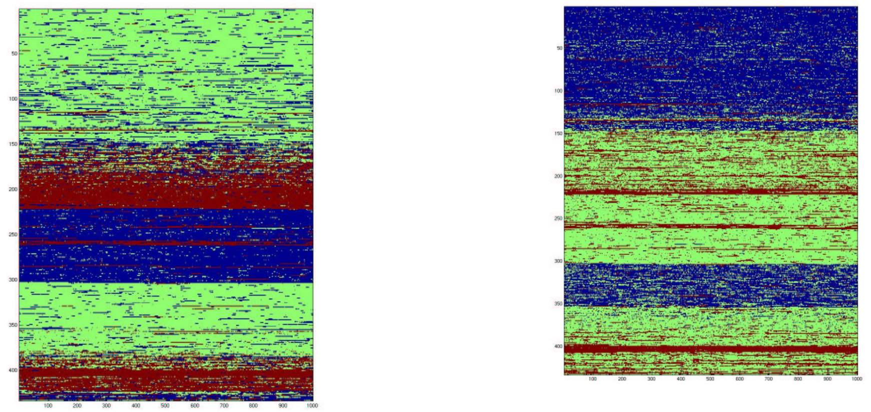
Figure 8.6: Plot of classified twoback data using four models. Each row is one trial lasting 1.7 seconds. Each time point is colored one of four colors corresponding to the model under which it has the highest likelihood. The trials are in temporal order, and the model regions are clearly extended in time. Two of the models seem to represent different periods of specific types of muscle activity.



(a)

(b)

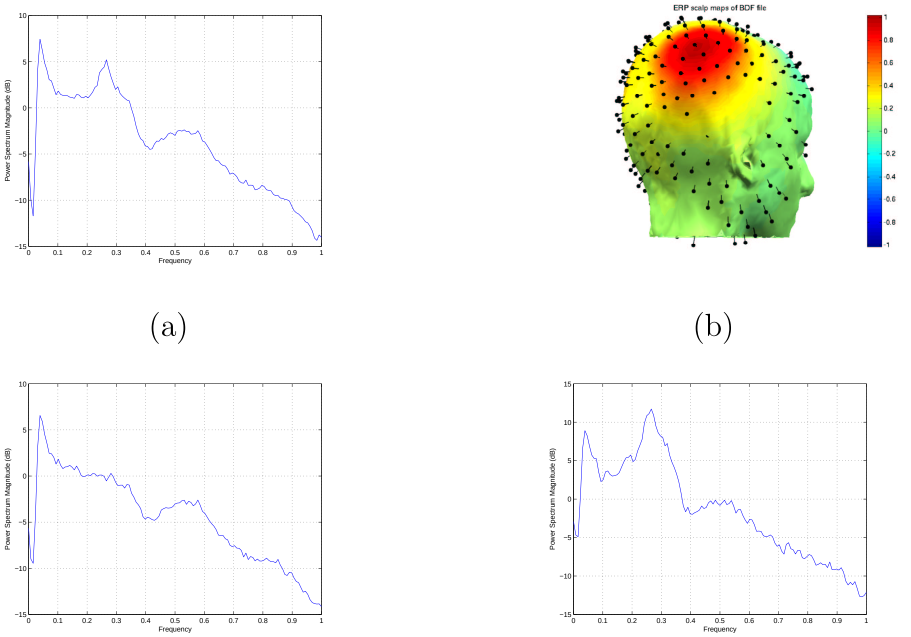
Figure 8.7: Classification of the twoback data with three models, and with four. This plot shows that there is consistency in the model classification over number of models, as the model change points for the three model seem to be mostly a subset of the change points in the four model. This has been found in other datasets as well.



(a)

(b)

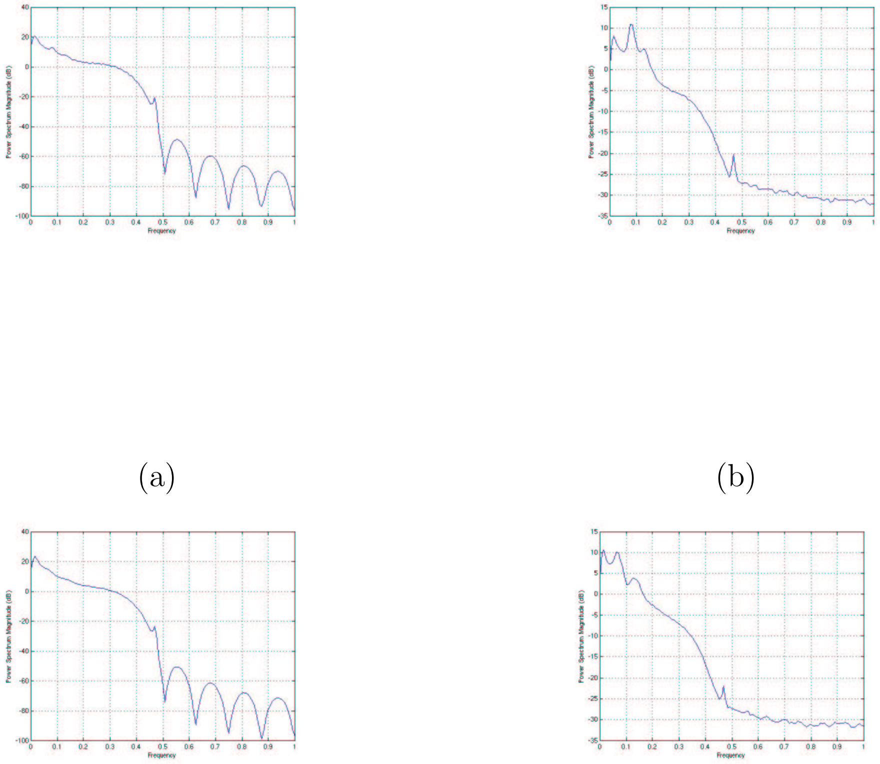
Figure 8.8: Classification of wordfinger dataset with three models using the (a) Generalized Gaussian mixture model and (b) Logistic mixture source model. The sharp switch from green to blue in (a) marks the change from the word task to the finger task, though the green model is used again later for the finger data. The Logistic model in (b) also has a sharp model switch at the task change. The red model in (b) is not well localized, and the Generalized Gaussian model seems to have specialized more than the Logistic model, perhaps due to the greater flexibility in the Generalized Gaussian mixture model (with adaptive shape parameter.)



(a) (b)

(c) (d)

Figure 8.9: Experiment with EEG data. (a) shows the original spectral density of the brain component with dipole estimate shown in (b). A mixture of deconvolving filters is applied to this source. The resulting psd's in (c) and (d) clearly show a division into alpha and non-alpha segments.



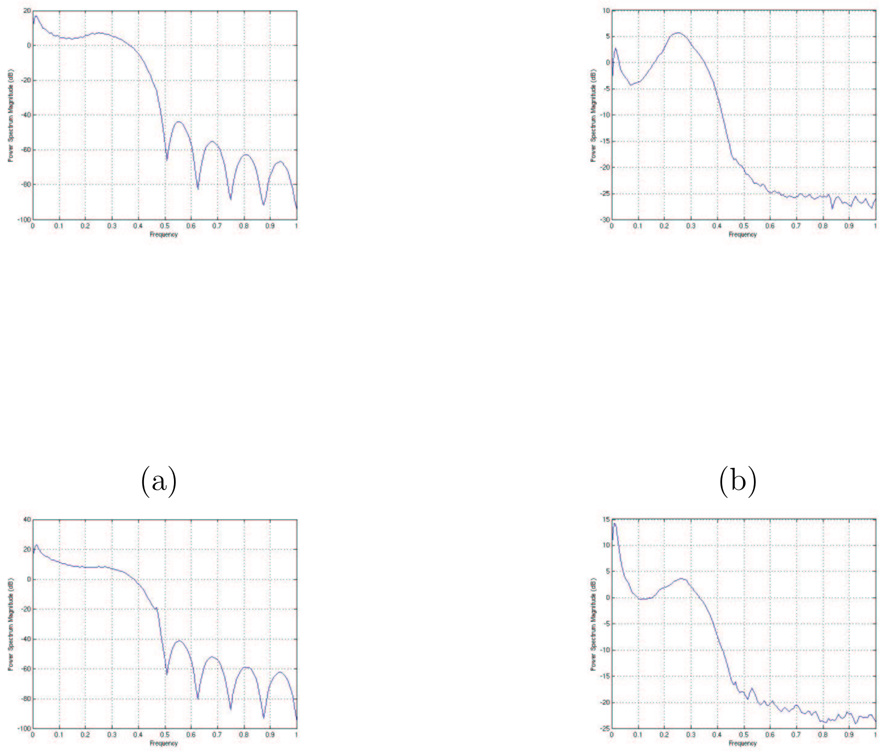
(a)

(b)

(c)

(d)

Figure 8.10: Examples of spectral density enhancement by using only classified data segments from a chosen model to compute the spectrum, rather than the entire time series. The left plots (a) and (c) are spectral densities of the entire time series for particular separated sources. The right plots (c) and (d) are the corresponding spectra of the model time points only. Using only the model time points clearly has a huge effect on the resolution of the spectrum. Also, the classified segments are not limited to particular chosen frequency bands, as would be the case with bandpass filtering, but rather adapt to the frequency response of independent signals, allowing complicated multimodal spectra.



(a)

(b)

(c)

(d)

Figure 8.11: More examples of spectral density enhancement by using only classified data segments from a chosen model to compute the spectrum, rather than the entire time series. The left plots (a) and (c) are spectral densities of the entire time series for particular separated sources. The right plots (c) and (d) are the corresponding spectra of the model time points only. Using only the model time points clearly has a huge effect on the resolution of the spectrum. Also, the classified segments are not limited to particular chosen frequency bands, as would be the case with bandpass filtering, but rather adapt to the frequency response of independent signals, allowing complicated multimodal spectra.

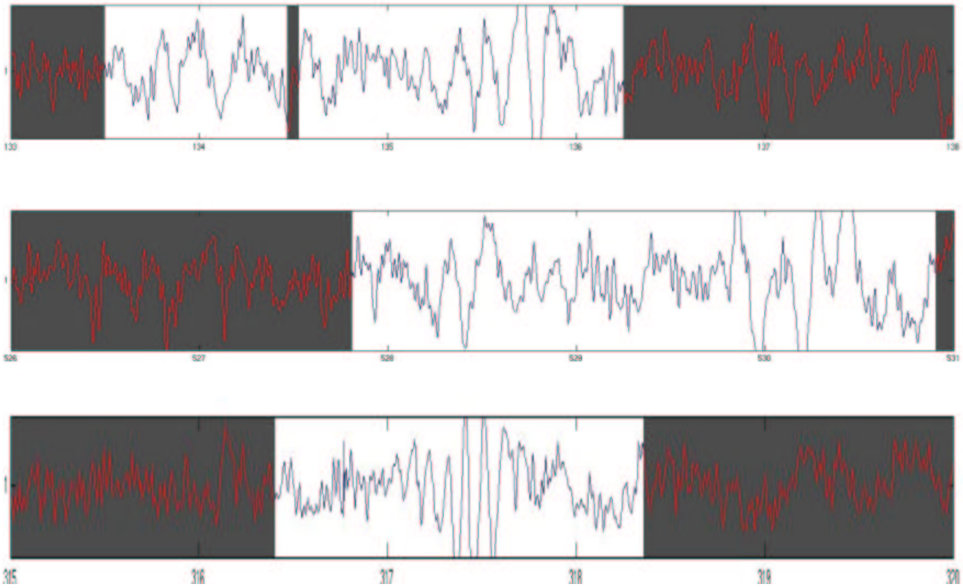


Figure 8.12: Example of segments claimed by the "good" model used to obtain the spectral estimate on the right in Figures 8.10 and 8.11.

Appendix A

Scale mixtures and the Mellin transform

Given that a density is a Gaussian scale mixture, one might wonder whether there is a general way of determining the scale mixing density given the density itself. This purpose is served by the Mellin transform which is to scale convolutions as the Fourier transform is to ordinary convolutions. The Mellin transform does not seem to be in wide use in the statistics and engineering communities, though a knowledge of its properties greatly simplifies the analysis of scale convolutions, as well as many important differential equations with power coefficients.

The Mellin transform [99, 97, 30], is defined by,

$$\mathcal{M}\left[f(x)\,;s\right] \;=\; \tilde{f}(s) \;=\; \int_0^\infty x^{s-1} f(x)\,dx \tag{A.1}$$

for $s \in \mathbb{C}$ such that the integral is convergent.

A.0.1 Basic properties

If $x^k g(x)$ and $x^k h(x) \in L(0,\infty)$ for some $k \in \mathbb{R}$, then [99, Thm. 44],

$$f(x) = \int_0^\infty \frac{1}{\xi}\, g\!\left(\frac{x}{\xi}\right) h(\xi)\,d\xi \quad\Rightarrow\quad \tilde{f}(s) = \tilde{g}(s)\tilde{h}(s) \tag{A.2}$$

and $x^k f(x) \in L(0,\infty)$. Thus under appropriate conditions, we can solve for $h(x)$ by inverting the transform. If $x^k f(x) \in L(0,\infty)$, then (A.1) can be inverted almost everywhere using the formula [99, Thm. 28],

$$\mathcal{M}^{-1}\left[\tilde{f}(s)\,;x\right] \;\equiv\; \frac{1}{2\pi i}\int\limits_{k-i\infty}^{k+i\infty} x^{-s}\tilde{f}(s)\,ds \;=\; \frac{f(x+)+f(x-)}{2} \tag{A.3}$$

where $f(x+)$ and $f(x-)$ denote the right and left hand limits of f at x. Thus, for example, when h in (A.2) is continuous on $(0,\infty)$, we have,

$$h(x) = \mathcal{M}^{-1}\left[\frac{\tilde{f}(s)}{\tilde{g}(s)}\,;x\right], \quad x \in (0,\infty)$$

We can similarly solve integral equations of the form,

$$f(x) = \int_0^\infty g(\xi x)h(\xi)\,d\xi$$

Using the following two properties, which follow from the definition (A.1),

$$\mathcal{M}\left[x^a f(x)\,;s\right] = \frac{1}{a}\tilde{f}\left(\frac{s}{a}\right), \quad \mathcal{M}\left[f(x^a)\,;s\right] = \tilde{f}(s+a) \tag{A.4}$$

we have $\mathcal{M}\left[t^{-1}h(t^{-1})\,;s\right] = \tilde{h}(1-s)$, and it follows that,

$$f(x) = \int_0^\infty g(\xi x)h(\xi)\,d\xi \quad \Rightarrow \quad \tilde{f}(s) = \tilde{g}(s)\tilde{h}(1-s) \tag{A.5}$$

Like the Laplace transform, the Mellin transform can be used to convert integro-differential equations into algebraic equations. We use in particular the following relation. Let D denote the differential operator. The transform of the operator $(-x)^n D^n$ is given by,

$$\mathcal{M}\left[(-x)^n D^n f(x)\,;s\right] \;=\; \frac{\Gamma(s+n)}{\Gamma(s)}\tilde{f}(s) \;=\; s(s+1)\cdots(s+n-1)\tilde{f}(s) \tag{A.6}$$

There are two basic integrals that are used, $\int_0^x f(t)dt$ and $\int_x^\infty f(t)dt$, and there are two corresponding definitions of fractional integrals. The *Riemann-Liouville fractional integral* [30, p. 113], for $\alpha > 0$ and non-integral, is defined by,

$$D^{-\alpha}f(x) = \frac{1}{\Gamma(\alpha)}\int_0^x (t-x)^{\alpha-1}f(t)\,dt$$

The *Riemann-Liouville fractional derivative*, D^{β}, $\beta > 0$, is defined by the same formula, with α replaced by $-\beta$.

The *Weyl fractional integral*, for $\alpha > 0$ and non-integral, is defined by,

$$W^{-\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \int_x^{\infty} (t - x)^{\alpha - 1} f(t)\, dt$$

For the *Weyl fractional derivative*, W^{β}, $\beta > 0$, let n be the smallest integer greater than β. Then the Weyl fractional derivative is defined by,

$$W^{\beta} f(x) = (-D)^n \, W^{n-\beta} f(x)$$

The Mellin transform of the Weyl fractional integral or derivative (α positive or negative), is given by,

$$\mathcal{M}\left[W^{-\alpha} f(x)\,; s\right] = \frac{\Gamma(s)}{\Gamma(s+\alpha)}\, \tilde{f}(s+\alpha) \tag{A.7}$$

Bibliography

[1] S.-I. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

[2] S.-I. Amari, T.-P. Chen, and A. Cichocki. Stability analysis of learning algorithms for blind source separation. *Neural Networks*, 10(8):1345–1351, 1997.

[3] D. F. Andrews and C. L. Mallows. Scale mixtures of normal distributions. *J. Roy. Statist. Soc. Ser. B*, 36:99–102, 1974.

[4] H. Attias. A variational Bayesian framework for graphical models. In *Advances in Neural Information Processing Systems 12*. MIT Press, 2000.

[5] H. Attias and C. E. Schreiner. Blind source separation and deconvolution: The dynamic component analysis algorithm. *Neural Computation*, 10:1373–1424, 1998.

[6] H. Barlow. What is the computational goal of the neocortex? In *Large-scale neuronal theories of the brain.* MIT Press: Cambridge, MA, 1994.

[7] O. Barndorff-Nielsen, J. Kent, and M. Sorensen. Normal variance-mean mixtures and z distributions. *International Statistical Review*, 50:145–159, 1982.

[8] M. J. Beal and Z. Ghahrarmani. The variational Bayesian EM algorithm for incomplete data: with application to scoring graphical model structures. In *Bayesian Statistics 7*, pages 453–464. University of Oxford Press, 2002.

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

[10] A. Benveniste, M. Goursat, and G. Ruget. Robust identification of a nonminimum phase system. *IEEE Transactions on Automatic Control*, 25(3):385–399, 1980.

[11] A. Benveniste, M. Métivier, and P. Priouret. *Adaptive algorithms and stochastic approximations.* Springer-Verlag, 1990.

[12] T. Berger. *Rate Distortion Theory: a mathematical basis for data compression.* Prentice Hall, Englewood Cliffs, NJ, 1971.

[13] C. M. Bishop and M. E. Tipping. Variational relevance vector machines. In C. Boutilier and M. Goldszmidt, editors, *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pages 46–53. Morgan Kaufmann, 2000.

[14] S. Bochner. *Harmonic analysis and the theory of probability.* University of California Press, Berkeley and Los Angeles, 1960.

[15] J. M. Borwein and A. S. Lewis. *Convex Analysis and Nonlinear Optimization.* Sringer, 2000.

[16] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

[17] P. J. Brockwell and R. A. Davis. *Time Series: theory and methods.* Springer Series in Statistics. Springer, 1991.

[18] J.-F. Cardoso. Iterative techniques for blind source separation using only fourth order cumulants. In *Proc. EUSIPCO*, pages 739–742, 1992.

[19] J.-F. Cardoso and B. H. Laheld. Equivariant adaptive source separation. *IEEE. Trans. Sig. Proc.*, 44(12):3017–3030, 1996.

[20] K. Chan, T.-W. Lee, and T. J. Sejnowski. Variational learning of clusters of undercomplete nonsymmetric independent components. *Journal of Machine Learning Research*, 3:99–114, 2002.

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

[22] Q. Cheng. On the unique representation of non-gaussian linear processes. *The Annals of Statistics*, 20:1143–1145, 1992.

[23] R. A. Choudrey and S. J. Roberts. Variational mixture of Bayesian independent component analysers. *Neural Computation*, 15(1):213–252, 2002.

[24] A. Cichocki and S. Amari. *Adaptive Blind Signal and Image Processing.* John Wiley & Sons, Ltd., West Sussex, England, 2002.

[25] R. R. Coifman and M. V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Trans. Information Theory*, 38(2):713–718, 1992.

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

[27] S. F. Cotter, J. Adler, B. D. Rao, and K. Kreutz-Delgado. Forward sequential algorithms for best basis selection. In *Proceedings Vision, Image, and Signal Processing*, pages 235–244. IEE, 1999.

[28] S. F. Cotter, K. Kreutz-Delgado, and B. D. Rao. Backward sequential elimination for sparse vector subset selection. *Signal Processing*, 81:1849–1864, 2001.

[29] T. Cover and J. Thomas. *Elements of Information Theory.* John Wiley and Sons, Inc., 1991.

[30] L. Debnath. *Integral transforms and their applications.* CRC Press, 1995.

[31] 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*, 39:1–38, 1977.

[32] A. P. Dempster, N. M. Laird, and D. B. Rubin. Iteratively reweighted least squares for linear regression when errors are Normal/Independent distributed. In P. R. Krishnaiah, editor, *Multivariate Analysis V*, pages 35–57. North Holland Publishing Company, 1980.

[33] D. L. Donoho. On miminum entropy deconvolution. In D. F. Findlay, editor, *Applied Time Series II*, New York, 1981. Academic Press.

[34] S.C. Douglas, A. Cichocki, and S. Amari. Multichannel blind separation and deconvolution of sources with arbitrary distributions. In *Proc. IEEE Workshop on Neural Networks for Signal Processing, Almelia Island Plantation, FL*, pages 436–445, 1997.

[35] T. Eltoft, T. Kim, , and T.-W. Lee. Multivariate scale mixture of Gaussians modeling. In J. Rosca et al., editor, *Proceedings of the 6th International Conference on Independent Component Analysis*, pages 799–806. Slpringer-Verlag, 2006.

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

[37] M. Figueiredo. Adaptive sparseness using Jeffreys prior. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

[38] H. M. Finucan. A note on kurtosis. *Journal of the Royal Statistical Society. Series B (Methodological)*, 26(1):111–112, 1964.

[39] J. Fischer. An algrorithm for discrete linear L_p approximation. *Numerische Mathematik*, 38(1):129–139, 1981.

[40] H. Gazzah, P. A. Regalia, and J.-P. Delmas. Asymptotic eigenvalue distribution of block Toeplitz matrices and application to blind SIMO channel identification. *IEEE Trans. Information Theory*, 47(3):1243–1251, 2001.

[41] D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(3):367–383, 1992.

[42] Z. Ghahramani and M. J. Beal. Variational inference for Bayesian mixtures of factor analysers. In *Advances in Neural Information Processing Systems 12*. MIT Press, 2000.

[43] M. Girolami. A variational method for learning sparse and overcomplete representations. *Neural Computation*, 13:2517–2532, 2001.

[44] T. Gneiting. Normal scale mixtures and dual probability densities. *J. Statist. Comput. Simul.*, 59:375–384, 1997.

[45] G. H. Golub and C. F. Van Loan. *Matrix computations*. The Johns Hopkins University Press, 1996.

[46] Robert M. Gray. Information rates of autoregressive processes. *IEEE Transactions on Information Theory*, 16(4):412–421, 1970.

[47] E. J. Hannan. *Mutltiple Time Series*. John Wiley & Sons, Inc., New York, 1970.

[48] D. H. Hubel and T. N. Wiesel. Receptive fields of single neurones in the cat's striate cortex. *J. Physiol.*, 148:574–591, 1959.

[49] D. H. Hubel and T. N. Wiesel. Functional architecture of macaque monkey visual cortex. *Proc. R. Soc. Lond. B*, 198:1–59, 1977.

[50] A. Hyvärinen. Independent component analysis in the presence of Gaussian noise by maximizing joint likelihood. *Neurocomputing*, 22:49–67, 1998.

[51] A. Hyvärinen, P. O. Hoyer, and M. Inki. Topographic independent component analysis. *Neural Computation*, 13(7):1527–1558, 2001.

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

[53] T. S. Jaakkola. *Variational Methods for Inference and Estimation in Graphical Models*. PhD thesis, Massachusetts Institute of Technology, 1997.

[54] T. S. Jaakkola and D. Haussler. Probabilistic kernel regression models. In D. Heckerman and J. Whittaker, editors, *Proceedings of the 7th Intl. Work. on AI and Statistics*. Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1999.

[55] T. S. Jaakkola and M. I. Jordan. A variational approach to Bayesian logistic regression models and their extensions. In *Proceedings of the 1997 Conference on Artificial Intelligence and Statistics*, 1997.

[56] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. In M. I. Jordan, editor, *Learning in Graphical Models*. Kluwer Academic Publishers, 1998.

[57] S. Karlin and A. Novikoff. Generalized convex inequalities. *Pacific J. Math.*, 13:1251–1279, 1963.

[58] S. Karlin and W. J. Studden. *Tchebycheff Systems: with applications in analysis and statistics*. Interscience, New York, 1966.

[59] J. Keilson and F. W. Steutel. Mixtures of distributions, moment inequalities, and measures of exponentiality and Normality. *The Annals of Probability*, 2:112–130, 1974.

[60] T. Kim, T. Eltoft, and T.-W. Lee. Independent vector analysis: An extension of ICA to multivariate components. In J. Rosca et al., editor, *Proceedings of the 6th International Conference on Independent Component Analysis*, pages 165–172. Slpringer-Verlag, 2006.

[61] K. Kreutz-Delgado. A scaled gradient projection method for concave function optimization. Technical report, ECE Department, University of California San Diego, 1999.

[62] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. Sejnowski. Dictionary learning algorithms for sparse representation. *Neural Computation*, 15(2):349–396, 2003.

[63] K. Kreutz-Delgado and B. D. Rao. Focuss-based dictionary learning algorithms. In *Wavelet Applications in Signal and Image Processing VII: Proc. of SPIE*, volume 4119. SPIE, 2000.

[64] H. Lappalainen. Ensemble learning for independent component analysis. In *Proceedings of the First International Workshop on Independent Component Analysis*, 1999.

[65] T. Lee. *Independent Component Analysis*. Kluwer Academic Publishers, 1998.

[66] T.-W. Lee, M. S. Lewicki, and T. J. Sejnowski. ICA mixture models for unsupervised classification of non-gaussian classes and automatic context switching in blind signal separation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(10):1078–1089, 2000.

[67] M. S. Lewicki and B. A. Olshausen. Probabilistic framework for the adaptation and comparison of image codes. *J. Opt. Soc. Am. A*, 16(7):1587–1601, 1999.

[68] M. S. Lewicki and T. J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12:337–365, 2000.

[69] H. Lu, Y. Fainman, and R. Hecht-Nielsen. Image manifolds. In *Applications of Artificial Neural Networks in Image Processing III: Proc. of SPIE*, volume 3307. SPIE, 1998.

[70] D. G. Luenberger. *Linear and Nonlinear Programming.* Addison-Wesley, second edition, 1984.

[71] D. J. C. MacKay. Ensemble learning and evidence maximization. Unpublished manuscript, 1995.

[72] D. J. C. Mackay. Comparison of approximate methods for handling hyperparameters. *Neural Computation*, 11(5):1035–1068, 1999.

[73] S. Makeig, A.J. Bell, T-P. Jung, and T.J. Sejnowski. Advances in neural information processing systems. In D. Touretzky, M. Mozer, and M. Hasselmo, editors, *Independent component analysis of electroencephalographic data*, volume 8, pages 145–151. MIT Press, Cambridge, MA, 1996.

[74] S. Makeig, T-P. Jung, D. Ghahremani, A.J. Bell, and T.J. Sejnowski. Blind separation of event-related brain responses into independent components. *Proc. Natl. Acad. Sci. USA*, 94:10979–10984, 1997.

[75] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Processing*, 41(12):3397–3415, 1993.

[76] A. Mansour and C. Jutten. What should we say about the kurtosis? *IEEE Signal Processing Letters*, 6(12):321–322, 1999.

[77] M. Miranda and P. Tilli. Asymptotic spectra of Hermitian block Toeplitz matrices and preconditioning results. *SIAM Journal of Matrix Analysis and Applications*, 21(3):867–881, 2000.

[78] R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 355–368. Kluwer, 1998.

[79] B. A. Olshausen and D. J. Field. Natural image statistics and efficient coding. *Network: Computation in Neural Systems*, 7:333–339, 1996.

[80] J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables.* Academic Press, 1970.

[81] J. A. Palmer and K. Kreutz-Delgado. A general framework for component estimation. In *Proceedings of the 4th International Symposium on Independent Component Analysis*, 2003.

[82] J. A. Palmer, K. Kreutz-Delgado, and S. Makeig. Super-Gaussian mixture source model for ICA. In *Proceedings of the 6th International Symposium on Independent Component Analysis and Blind Source Separation*, Lecture Notes in Computer Science. Springer, 2006.

[83] J. A. Palmer, K. Kreutz-Delgado, D. P. Wipf, and B. D. Rao. Variational EM algorithms for non-gaussian latent variable models. In *Advances in Neural Information Processing Systems*. MIT Press, 2005.

[84] H.-J. Park and T.-W. Lee. Modeling nonlinear dependencies in natural images using mixture of Laplacian distribution. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2004. MIT Press.

[85] D. T. Pham. Mutual information approach to blind separation of stationary sources. *IEEE Trans. Information Theory*, 48(7):1935–1946, 2002.

[86] D. T. Pham and Ph. Garat. Blind separation of instantaneous mixture of sources via an independent component analysis. *IEEE Trans. Signal Processing*, 44(11):2768–2779, 1996.

[87] D. T. Pham, Ph. Garat, and C. Jutten. Separation of mixture of independent sources through a maximum likelihood approach. In J. Vandewalle, R. Boite, M. Moonen, and A. Oosterlinck, editors, *Signal Processing VI: Proceedings of EUSIPCO '92, Bruxelles, Belgium*, pages 771–774. Elsevier, Amsterdam, 1992.

[88] B. D. Rao, K. Engan, S. F. Cotter, J. Palmer, and K. Kreutz-Delgado. Subset selection in noise based on diversity measure minimization. *IEEE Trans. Signal Processing*, 51(3), 2003.

[89] B. D. Rao and I. F. Gorodnitsky. Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm. *IEEE Trans. Signal Processing*, 45:600–616, 1997.

[90] B. D. Rao and K. Kreutz-Delgado. An affine scaling methodology for best basis selection. *IEEE Trans. Signal Processing*, 47:187–200, 1999.

[91] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods.* Springer, 1999.

[92] R. T. Rockafellar. *Convex Analysis.* Princeton, 1970.

[93] M. Rosenblatt. *Gaussian and Non-Gaussian Linear Time Series and Random Fields.* Springer, 2000.

[94] G. Samorodnitsky and M. S. Taqqu. *Stable non-gaussian random processes: Stochastic models with infinite variance*. Chapman and Hall, New York, 1994.

[95] L. K. Saul, T. S. Jaakkola, and M. I. Jordan. Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research*, 4:61–76, 1996.

[96] C. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.

[97] I. N. Sneddon. *The use of integral transforms*. McGraw-Hill, 1972.

[98] M. E. Tipping. Sparse Bayesian learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[99] E. C. Titchmarsh. *Introduction to the theory of Fourier integrals*. Oxford: Clarendon Press, second edition, 1948.

[100] V. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, Inc., 1998.

[101] D. V. Widder. *The Laplace Transform*. Princeton University Press, 1946.

[102] D. Wipf, J. Palmer, and B. Rao. Perspectives on sparse bayesian learning. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, Cambridge, MA, 2003. MIT Press.

[103] C. F. Jeff Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

[104] W. I. Zangwill. *Nonlinear Programming: A Unified Approach*. Prentice-Hall, 1969.