Use and Development of Matrix Factorisation Techniques in the Field of Brain Imaging

Matthew Craig Pearce

Department of Pure Mathematics and Mathematical Statistics
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Churchill College June 2018
This thesis is dedicated to Kerry and to Alfred, who provided invaluable assistance in late-stage editing.
Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Matthew Craig Pearce
June 2018
Acknowledgements

The unfailing enthusiasm and patience of my supervisor, Simon White, must first be acknowledged. Advice from Paul Newcombe and Brian Tom pushed me in the right direction early on. Support from Karen Campbell and all the other members of the Cam-CAN project helped to bring the neuroimaging component of this thesis to fruition. Some of the work in chapter 5 was made available as Pearce and White (2016a) and part of Pearce and White (2016b) was used in chapter 6. Audience comments and questions in response to presentations and posters also proved useful, as has feedback from anonymous reviewers. Colleagues and peers at the MRC Biostatistics Unit have ensured that my time in Cambridge has been stimulating. The Medical Research Council funded this work, for which I am very grateful.
Abstract

Matrix factorisation treats observations as linear combinations of basis vectors together with, possibly, additive noise. Notable techniques in this family are Principal Components Analysis and Independent Components Analysis. Applied to brain images, matrix factorisation provides insight into the spatial and temporal structure of data.

We improve on current practice with methods that unify different stages of analysis simultaneously for all subjects in a dataset, including dimension estimation and reduction. This results in uncertainty information being carried coherently through the analysis.

A computationally efficient approach to correlated multivariate normal distributions is set out. This enables spatial smoothing during the inference of basis vectors, to a level determined by the data. Applied to neuroimaging, this reduces the need for blurring of the data during preprocessing. Orthogonality constraints on the basis are relaxed, allowing for overlapping ‘networks’ of activity.

We consider a nonparametric matrix factorisation model inferred using Markov Chain Monte Carlo (MCMC). This approach incorporates dimensionality estimation into the inference process. Novel parallelisation strategies for MCMC on repeated graphs are provided to expedite inference. In simulations, modelling correlation structure is seen to improve source separation where latent basis vectors are not orthogonal. The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) project obtained fMRI data while subjects watched a short film, on 30 of whose recordings we demonstrate the approach.

To conduct inference on larger datasets, we provide a fixed dimension Structured Matrix Factorisation (SMF) model, inferred through Variational Bayes (VB). By modelling the components as a mixture, more general distributions can be expressed. The VB approach scaled to 600 subjects from Cam-CAN, enabling a comparison to, and validation of, the main findings of an earlier analysis; notably that subjects’ responses to movie watching became less synchronised with age. We discuss differences in results obtained under the MCMC and VB inferred models.
# Table of contents

List of figures .............................. xvii

List of tables ............................. xxiii

1 Introduction ............................ 1

2 Matrix factorisation in fMRI ........ 5
   2.1 Matrix factorisation .................. 5
   2.2 A note on notation .................. 6
      2.2.1 A note on independence ........ 7
   2.3 Some factorisation models .......... 7
      2.3.1 Principal Components Analysis .. 7
      2.3.2 Factor Analysis ................ 8
      2.3.3 Independent Components Analysis 9
      2.3.4 Other notable linear methods ... 11
      2.3.5 Beyond linear models .......... 13
   2.4 Neuroimaging data .................. 14
      2.4.1 A brief introduction to fMRI .. 14
      2.4.2 Dimensions, content and format of fMRI data 16
      2.4.3 Preprocessing .................. 17
   2.5 Matrix factorisation of fMRI .... 18
   2.6 Dimension estimation for matrix factorisation 20
      2.6.1 Neuroimaging context .......... 20
      2.6.2 Minimum Description Length estimation 21
      2.6.3 Laplace approximation to Bayesian PPCA evidence 21
   2.7 Eigenvalue corrections ............ 23
   2.8 Analysis of group fMRI data ....... 24
      2.8.1 Group fMRI data ............... 24
# Table of contents

## 2.8 Reduction techniques
- 2.8.2 Reduction to a common space prior to concatenation ........................................................................ 25
- 2.8.3 Separate reduction prior to concatenation ..................................................................................... 25
- 2.8.4 Tensor factorisation ...................................................................................................................... 26

## 3 Failure modes for eigendecomposition methods
- 3.1 Introduction .................................................................................................................................. 29
- 3.2 Time-series or spatial processes? .................................................................................................... 29
- 3.3 Effect of model ‘transposition’ misspecification.
  - 3.3.1 Contrast with proposed vectors in spatial orientation ............................................................ 30
  - 3.3.2 Effect of low-sample size on estimation of spatial eigenvector ............................................. 32
  - 3.3.3 Related work on the high-dimension, low sample size regime .............................................. 33
  - 3.3.4 An objection to the high dimension, low sample size critique ............................................. 34
- 3.4 Dimension estimation in the high dimension, low sample size regime ........................................ 35
  - 3.4.1 High dimension low sample size simulation results ............................................................... 36
- 3.5 Impact of signal to noise ratio ........................................................................................................ 37
- 3.6 Point estimates of dimensionality .................................................................................................. 39
- 3.7 Two stage analysis: must the PCA solution contain the ICA solution?
  - 3.7.1 Perils of two-stage analysis ...................................................................................................... 41
- 3.8 Orthogonality and spatial ICA ......................................................................................................... 43
- 3.9 Summary ....................................................................................................................................... 46

## 4 Efficient models of structured bases
- 4.1 Introduction .................................................................................................................................... 49
  - 4.1.1 Structured Features .................................................................................................................. 49
  - 4.1.2 Computational considerations .................................................................................................. 50
  - 4.1.3 Graphs and Gaussian Markov Random Fields ......................................................................... 51
- 4.2 Covariance structures ...................................................................................................................... 52
  - 4.2.1 Definitions ................................................................................................................................. 52
  - 4.2.2 Groups of covariance structures ............................................................................................... 53
  - 4.2.3 Rules for construction of valid eigenvalues ............................................................................. 53
  - 4.2.4 Parameterisation of the Multivariate Normal distribution ..................................................... 54
  - 4.2.5 Dimension extensions of covariance structures ........................................................................ 56
- 4.3 Relationship to other parameterisations .......................................................................................... 57
  - 4.3.1 Multiplicative dimension extension via Kronecker product .................................................. 57
- 4.4 Example: smooth features .............................................................................................................. 58
  - 4.4.1 The neighbourhood model ....................................................................................................... 58
  - 4.4.2 Extension to additional dimensions .......................................................................................... 58
Table of contents

4.4.3 Illustration: Normalisation ........................................... 60
4.5 Discussion ................................................................. 63

5 Bayesian non-parametric matrix factorisation 65
5.1 Introduction ............................................................... 65
5.2 Model ...................................................................... 67
5.3 Gibbs sampler ........................................................... 68
  5.3.1 Conditional distributions of latent basis vectors, \( w_k \) ............. 69
  5.3.2 Activation of shared features within \( B \) ................................ 69
  5.3.3 Scaling variables for shared features, \( S_{k,t} \) .......................... 71
  5.3.4 Activating unique features ............................................ 71
  5.3.5 Drawing spatial feature vectors for unique features .................... 73
  5.3.6 Sampling the noise level \( \sigma^2 \) ........................................ 73
  5.3.7 Sampling covariance structure parameters ................................ 73
  5.3.8 Sampling the IBP strength \( \alpha \) ..................................... 75
  5.3.9 Sampling the IBP repulsion \( \beta \) ..................................... 75
  5.3.10 Sampling weight precisions \( \nu_k \) .................................... 76
  5.3.11 Sampling the weight means \( \tau_k \) .................................... 76
5.4 Simulated Example ....................................................... 76
5.5 Conclusion ................................................................. 81

6 Parallel Gibbs Sampling in Repeated Graphs 83
6.1 Motivation and approach ................................................. 83
6.2 Repeated graphs and their colouring .................................... 84
  6.2.1 Size of a repeated colouring ......................................... 85
6.3 The Palette Sampler ....................................................... 85
  6.3.1 Palettes .............................................................. 86
  6.3.2 Sampling from a palette is parallelisable ............................... 86
  6.3.3 Construction of palettes on repeated graphs .......................... 87
6.4 Examples ................................................................. 88
  6.4.1 Illustrative example with a random array .............................. 88
6.5 Application to nonparametric matrix factorisation ...................... 90
  6.5.1 The Indian Buffet Process .......................................... 90
  6.5.2 Previous sampling approaches ....................................... 91
  6.5.3 Improved parallelisation ............................................. 92
  6.5.4 Example results .................................................... 93
6.6 Discussion ................................................................. 94
# Table of contents

## 7 Bayesian nonparametric matrix factorisation for neuroimaging

7.1 Purpose and background ................................................. 97  
7.2 Data .............................................................................. 98  
  7.2.1 Size and preprocessing ........................................ 98  
  7.2.2 Concatenation ...................................................... 99  
7.3 Prior ............................................................................. 99  
7.4 Ranking of dimensions .................................................. 102  
7.5 Metrics .......................................................................... 102  
  7.5.1 Synchronisation ..................................................... 102  
  7.5.2 Smoothness .......................................................... 103  
  7.5.3 Magnitude ............................................................. 104  
7.6 Inference ....................................................................... 104  
  7.6.1 Initialisation .......................................................... 104  
  7.6.2 Results ................................................................. 104  
7.7 Conclusion ..................................................................... 111  
  7.7.1 Comments on results ............................................. 111  
  7.7.2 Further work ......................................................... 113  

## 8 Structured basis matrix factorisation via Variational Bayes

8.1 Motivation ..................................................................... 115  
8.2 Variational Bayes .......................................................... 116  
  8.2.1 The evidence lower bound .................................... 116  
  8.2.2 Co-ordinate updates .............................................. 117  
  8.2.3 Parametric families ................................................ 118  
  8.2.4 Stochastic variational inference ......................... 118  
  8.2.5 The re-parameterisation trick ............................. 119  
8.3 Structured Basis Matrix Factorisation ......................... 121  
  8.3.1 Aims ................................................................... 121  
  8.3.2 Model ................................................................. 122  
8.4 Inference ..................................................................... 124  
  8.4.1 Notation ............................................................... 124  
  8.4.2 Basis vectors ......................................................... 125  
  8.4.3 Mixture selection variables ................................. 126  
  8.4.4 Mixture selection parameters ......................... 126  
  8.4.5 Mixture scaling variables ................................. 127  
  8.4.6 Mixture scaling parameters ........................... 127  
  8.4.7 Spatial parameters .............................................. 127
# Table of contents

8.4.8 Noise parameters ................................................. 129
8.5 Software ............................................................ 130
8.6 Simulated example .................................................. 130

9 Application of Structured Matrix Factorisation to Neuroimaging Data 135

9.1 Introduction .......................................................... 135
9.2 Data and preprocessing .............................................. 136
9.3 Priors and optimisation constants ................................ 137
9.4 Model selection ....................................................... 138
  9.4.1 Magnitude ....................................................... 142
9.5 Results from selected model ....................................... 143
  9.5.1 Matching to Shirer maps. ................................. 146
  9.5.2 Matching to Harvard Oxford Atlas ..................... 149
  9.5.3 Regression against covariates ............................ 152
  9.5.4 Rendering of components ................................. 154
9.6 Commentary on results ............................................. 176

10 Discussion ............................................................ 179

10.1 Overview ........................................................... 179
10.2 Dimensionality of neuroimaging data .......................... 181
10.3 Comparison of MCMC and VB results .......................... 182
10.4 Further work ......................................................... 183

References .............................................................. 185
List of figures

2.1 (Belliveau et al., 1991) showed that it was possible to measure changes in
the activity of the visual cortex through MRI. ................................. 15
2.2 Top: a volume from the Cam-CAN dataset under 1mm Gaussian smoothing
FWHM spatial, and the same under 1mm followed by 10mm spatial smoothing. 18

3.1 In both images the left hand panel shows the true primary eigenvector and
the right hand panel the estimated eigenvector. Signal to noise ratio $\alpha = 100$. 33
3.2 Different patterns of signal decay, and their translation into empirical eigen-
values. ................................................................. 36
3.3 Behaviour of dimensionality estimation methods under transposition mis-
specification, with fixed $T = 1000$, $\alpha = 100$, $K = 5$. The figure on the left is
from a data generating process using a linear signal decay, while the figure on
the right uses a logistic signal decay. In both cases the estimated dimension
reduces to the smallest considered given a large enough observed dimension. 37
3.4 Behaviour of dimensionality estimation methods on simulated data when
$V/T = 2.5$, $T = 1000$ for varying levels of signal strength $\alpha$. In both cases
the true latent dimension is 10. Dotted lines show 25% and 75% percentile
estimates. ................................................................. 38
3.5 Estimated dimension under uniform signal decay and varying spatial auto-
correlation models. This suggests that the character of the signal decay curve
has more impact on estimated dimensionality than the nature of the ‘spatial’
autocorrelation. ................................................................. 39
3.6 Eigenvalue and Laplace approximation analysis of a subset of $N = 30$ partic-
ipants from Cam-CAN (Shafto et al., 2014). Note that the Laplace approxi-
mation score curve (red) is flat around its maximum. .......................... 40
3.7 Analysis of ICA performance in a similar experimental setup to (Beckmann et al., 2005). 3.7a) shows the two latent spatial maps, which overlap in 50% of their active area (squares offset from centres). 3.7b) the panels show the true latent time-courses with correlation 0.05, and the bottom two the inferred time courses. In 3.7d) the latent time courses were chosen to have correlation 0.2. Panels 3.7c) and 3.7e) show the corresponding inferred spatial maps for the time courses in 3.7b) and 3.7d) with correlations 0.05 and 0.2 respectively. We can see that in 3.7e) that ‘spatial’ ICA did not recover the true spatial maps when the correlation level was higher.

4.1 Visualisation of the eigenvectors for covariance structures on an 8 × 8 grid.

5.1 Different views of example images from the holdout dataset.

5.2 Analysis of the dimensionality of the dataset. Blue line shows the Laplace evidence approximation of the dimensionality obtained by Minka’s method (optimising $K^* = 33$). Black dashed line shows the first 100 eigenvalues of the data covariance matrix.

5.3 MCMC sampling traces for analysis of the same dataset under different priors.

5.4 Comparison of matched features. Matching done by cosine similarity after normalisation.

5.5 Distribution of inferred sources under iSSFA and FastICA. If one were interested in maximising the non-Gaussianity of inferred sources the sparsity of the iSSFA model produces a more optimal fit by the kurtosis criterion than does the FastICA methodology.

6.2 Illustration of propositions 6.2.1 and 6.3.2 on the example model.

6.3 Nonzero portion of an example IBP matrix, $K^+ = 20 \times T = 10$. Shared features rows are in the top half of the image, and unique feature rows in the bottom half.
6.4 Layout of resampling work for shared features in $B$. Here we are using 3 compute nodes (shown in different colours) and there are $K^{++} = 5$ shared features to resample. Each step is equivalent to parallel sampling of a palette. Hence 5 steps are required for a Gibbs pass through all shared features $B_{k,t}$ (shown here transposed).

7.1 Rapid increase in basis vectors under uninformative prior. While the trajectory suggests an asymptote and hence convergence may be found eventually, for neuroimaging analysis the computational requirements may be infeasible.

7.2 Trace plot of the number of features (solid blue) and noise level (dotted black) over MCMC iterations.

7.5 Synchronisation versus smoothness scores for each dimension in the final sample. More synchronised dimensions tend to be smoother.

7.3 Spatial maps and associated time-courses. Features selected for highest synchronisation score.

7.4 Spatial maps and associated time-courses. Features selected for lowest synchronisation score.

8.1 Histogram of the components of the simulated data.

8.2 Demonstration of model on synthetic data, using two different covariance structures.

9.1 a) ELBO curves over the optimisation process. b) Average likelihoods over the final 50 iterations for different dimensionalities under IID spatial structure. Dotted: upper and lower quartiles of the same.

9.2 Summaries of synchronisation levels and smoothness for different dimensionalities under IID spatial structure. Solid blue lines represent mean scores. Dotted green lines show upper and lower quartiles.

9.3 a) ELBO curves over the optimisation process, note that lower dimensional models converge more quickly than higher dimensional models. b) Solid: average likelihoods over the final 50 iterations for different dimensionalities under nDCT spatial structure. Dotted: upper and lower quartiles of the same. Note that the scale of b) is shared by Figure 9.1.

9.4 Summaries of synchronisation levels and smoothness for different dimensionalities under nDCT spatial structure. Solid blue lines represent mean scores. Dotted green lines show upper and lower quartiles.
9.8 Report for latent dimension 1. Smoothness: 2.165 (15). Synchronisation: 0.35 (16). Magnitude: 0.007 (19). HOA matches: -PG2, PT, POC, -STGad, -SC. ................................................................. 156
9.15 Report for latent dimension 8. Smoothness: 1.544 (2). Synchronisation: 0.466 (3). Magnitude: 0.016 (9). HOA matches: FP, JLC, LOCid, PG3, -LOCsd. ................................................................. 163


List of tables

2.1 Quantities for the MDL method .............................................. 21
2.2 Quantities for the LAP formula .............................................. 22
4.1 Collection of covariance structures ....................................... 63
5.1 Variables and their distributions. ........................................... 67
7.1 Prior parameters used in the analysis. Refer to section 5.2 for their interpretation. Spatial parameters have been given names $\theta_1 = \phi$ and $\theta_2 = \lambda$ for a spatial structure like that of section 4.4 with $\Delta = \phi I + \lambda L$ ........................................ 101
7.2 Latent dimensions scored .................................................... 110
8.1 Model specification for SMF .................................................. 122
8.2 Performance of structures on GMRF ..................................... 131
9.1 Settings used in VB run ....................................................... 137
9.2 Priors used in SMF analysis. ................................................ 138
9.3 Metrics for model selection under IID structure. ..................... 140
9.4 Metrics for model selection under nDCT structure. ................. 142
9.5 Posterior summary .............................................................. 143
9.6 Latent dimensions scored .................................................... 145
9.7 Shirer maps matched to latent maps. ..................................... 146
9.8 Latent maps matched to Shirer maps. ................................... 147
9.9 Key for Harvard Oxford Atlas regions. Laterally split regions merged. ......................................................... 150
9.10 Basis matched to Harvard Oxford atlas regions. Sign indicates whether loading is positive or negative in that region. .............................. 151
9.11 Latent dimensions scored by regression of synchronisation on covariates $(\text{sync} \sim c + b \ast \text{age})$ ...................................................... 152
Chapter 1

Introduction

This thesis is concerned with the use and development of matrix factorisation techniques in the field of brain imaging. This chapter provides a brief introduction to the motivation and framework for the topic. What, then, is matrix factorisation and how is it relevant to neuroimaging?

Matrix factorisation uses latent variables in a linear model to explain multivariate data of interest. The unifying idea is that given a $V \times T$ matrix of data we can summarise the important structure using around $K(V + T)$ latent variables. Thus, when $K$ is small, matrix factorisation methods effect dimension reduction or compression of data. In chapter 2 we provide technical background on different matrix factorisation techniques and their popular usage in the field of brain imaging.

Applied to brain imaging, the summaries produced by matrix factorisation will include spatial maps representing areas of the brain active (or suppressed) in a particular functional network together with temporal information representing how engaged those networks were for each subject over time. However, factorisation is applied to the data available rather than only to that portion of the data which would be of interest psychologically or clinically. Thus our summaries of neuroimaging data may capture events such as head movement or scanner artefacts. It falls then to analysts of neuroimaging data to develop models and methods capable of separating out the relevant content.

In chapter 3 we consider issues that have the potential to arise under popular matrix factorisation techniques for neuroimaging. We look at the circumstances that could lead to poor estimation of the latent dimensionality, $K$. We also consider the potential issues caused by orthogonality constraints; that is, whether our models should try to enforce zero correlation between latent spatial maps.

Currently popular methods for matrix factorisation treat each location in a spatial map as independent of its neighbours. From a Bayesian perspective this means that our prior belief
about the appearance of a spatial map representing a functional network in the brain is that it is as likely to look like TV static as it is to contain contiguous regions of activation. Data is currently smoothed before analysis in order to ensure contiguous regions of activation. This process averages out details across the brain. Chapter 4 sets out our approach to modelling correlation within latent spatial maps. We take a general approach to the problem, showing that it is possible to model spatial correlation efficiently by examining the algebraic properties of precision matrices with known eigenvectors.

In chapter 5 we extend a Bayesian nonparametric matrix factorisation model, including by incorporating an efficient model for spatially correlated basis vectors. Spatial correlation is modelled at the level of individual functional networks. In simulation we see that modelling spatial relationships increases the similarity of our recovered solution to the ground truth relative to the spatially independent prior available in previous work. This approach promises to relax the need to decide in advance of analysis the latent dimension \( K \) by exploring the distribution of possible latent dimensionalities.

Inference of the model in chapter 5 was achieved using the Markov Chain Monte Carlo (MCMC) method of Gibbs sampling. As is well known, MCMC methods are computationally intensive. However, by examining the conditional independence properties of the model’s graph, in chapter 6 we show that inference can be substantially parallelised. The result presented covers all graphical models that can be arranged into an array with dependencies only between nodes in the same column or row.

An application of our nonparametric model to neuroimaging data is made in chapter 7. We find that a loose prior results in too a high latent dimensionality from both computational and interpretive perspectives. We are thus motivated to use priors which limit the accuracy of the model in return for a high level of dimension reduction. Using metrics we are able to rank latent dimensions, and see that these metrics may help to separate artefactual and psychologically interpretable dimensions. Notwithstanding the deployment of the techniques described in chapter 6, the computational requirements of Markov Chain Monte Carlo for matrix factorisation in the high dimensional setting of neuroimaging are found to limit to the number of subjects whose data could reasonably be processed.

Chapter 8 addresses the issue of computational scalability by providing a Variational Bayes inference method for fixed dimension models. We call this Structured Matrix Factorisation (SMF) to reflect the modelling of correlation within the basis vectors. The work presented in this chapter covers a range of factorisation models where the basis vectors are themselves expected to be L2 sparse in a known basis. An extended example is provided in terms of a smoothness model suitable for neuroimaging. In chapter 9 we apply SMF to a neuroimaging dataset of 600 subjects obtained during a movie-watching task.
We conclude in chapter 10 with reflections on our main findings; in particular on the advantages and disadvantages of Markov Chain Monte Carlo and Variational Bayes for matrix factorisation in the high dimensional setting of neuroimaging.
Chapter 2

Matrix factorisation in fMRI

2.1 Matrix factorisation

We will be interested in models which express data vectors $y_i \in \mathbb{R}^V, i = 1, \ldots, T$ as lying close to a latent linear manifold. The latent manifold being spanned by $K$ basis vectors $w_k \in \mathbb{R}^V, k = 1, \ldots, K$, which can be arranged into a matrix, $W = [w_1, \ldots, w_K]$, which we will refer to as the basis. Each observation $i$ having a set of latent components $x_i \in \mathbb{R}^K$ which specify a location on the linear manifold and an ‘error’ term $\epsilon_i$ which captures the difference of the observed vector from the corresponding position on the latent manifold. This is to say, we will be interested in models which have the algebraic form:

$$y_i = Wx_i + \epsilon_i, \quad i = 1, \ldots, T$$  \hspace{1cm} (2.1)

where $K < V$ the method discussed will have effected dimensionality reduction. We add observation noise $\epsilon_i$ here for generality, although some of the more popular methods discussed in the following sections assume implicitly that $\epsilon_i = 0$. We can write this in matrix form as

$$Y = WX + E,$$  \hspace{1cm} (2.2)

in which $Y$ is a $V \times T$ matrix containing the observed vectors $y_i$ as columns, $X$ a $K \times T$ matrix containing the $x_i$ as columns, and $E$ is the corresponding matrix of observation noise. Methods which lead to equations of this form are sometimes referred to as matrix factorisation methods. We approve of this term because the matrix factorisation formulation allows for a range of models with dependencies and structure across both the $V$ and $T$ dimensions of the data, and treats $W$ and $X$ on a more equal footing, something which can be awkward to express in a vector format.
After setting the general algebraic form of a model a number of further details remain to be specified before it can be fit to data. These include: what constraints the variables obey; which variables are given probability distributions and, if so, what those distributions are; what is an appropriate value for $K$, the dimension of the latent basis; for optimisation based approaches what is the objective function? The possible resolutions of these questions lead to different models, and we touch now on some of the main examples.

### 2.2 A note on notation

In this thesis matrices will be denoted by upper case bold symbols, such as $M, \Sigma$ and so forth. Vectors will be denoted by bold lowercase symbols, such as $x, \xi$ and so on. We will generally not de-bold matrices or vectors when referring to specific indices.

We use ‘Matlab’ notation, by which $M_{i:}$ means the $i^{th}$ row of a matrix and $M_{:,j}$ is the $j^{th}$ column. Notation like $M_{:,J}$ would indicate the matrix formed by dropping the $j^{th}$ column of the matrix $M$. Or when $J$ is a set of indices $M_{:,J}$ is the matrix formed by dropping all the columns whose index is in $J$. Single rows picked from matrices will be treated as matrices with a single row, this means there is no inconsistency when one or more rows are thus picked out.

We may use a single superscript index on a matrix, such as $M^n$, to refer to the $n^{th}$ matrix from a collection, rather than a matrix power (generally the matrices referred to in this fashion are non-square data matrices and thus the intention should be clear from context).

The notation $D(v)$ indicates a diagonal matrix constructed with components of the vector $v$ comprising the main diagonal. The symbol $I$ represents the identity matrix, which may be subscripted with an integer where doing so would assist identification of the relevant dimensionality.

Some symbols that require definition are $\circ$ which signifies elementwise multiplication, $\otimes$ which is the Kronecker product, and $\oplus$ which is the Kronecker sum.

For ease of comprehension, we will try to be consistent throughout this thesis with the symbols chosen for key indices. Occassionally it may be necessary to break with this, for instance when reporting others’ work, but we will alert the reader whenever this is the case. An important source of notational confusion arises from the orientation of data matrices in various authors’ work. That is, whether the method analyses data $Y (V \times T)$ or $Y' (T \times V)$. Where reporting others’ work we will try to homogenise the orientation to $Y (V \times T)$ so that the number of columns represents the number of observations, and the number of rows represents the dimension of each observation.
2.2.1 A note on independence

The reader will recall that two random variables, \( z \) and \( y \) are independent if, and only if, their joint density function factorises as \( p(z,y) = p(z)p(y) \). We say that the entries or components of a random vector \( z \) are independent we mean that the density of the vector factorises as \( p(z) = \prod_i p(z_i) \).

The independence property is of interest for at least two reasons. First, the factorisation property of the joint density leads to mathematical tractability. For example, many integrals of substantial practical interest simplify when the random variables involved are independent; and in chapter 8 we make extensive use of this property.

Second, independence eases interpretation. For a collection of \( K \) independent random variables the distribution of the \( k^{th} \) variable is unaffected by the values the others happen to take. As an example, this means that unvisualised interactions between variables could not be present to mislead us when looking at \( K \) separate histograms. This can prove particularly helpful when \( K > 3 \) and we run out of spatial dimensions in which to visualise the data.

2.3 Some factorisation models

2.3.1 Principal Components Analysis

While we anticipate the reader will be familiar with Principal Components Analysis (PCA) we will recapitulate its mechanics and some of its history here. PCA is a classical method in statistics due to Pearson (1901). Those interested in a detailed history can consult Wold et al. (1987) or the introductory chapter of Jolliffe (2002).

A notable development came in the paper of Hotelling (1933), which appears to have introduced the canonical name for the method. Indeed, Hotelling anticipates the generality of latent variable modelling in the case of multivariate observed data, "It is natural to ask whether some more fundamental set of independent variables exists, perhaps fewer in number than the \( x \)'s, which determine the values the \( x \)'s will take." Here our \( y \) functions as Hotelling’s \( x \)'s. Principal components analysis emerged from a simplifying assumption that the latent variables had Gaussian distributions. Gaussian latent variables provide the simplification that zero correlation between two variables implies their independence, a fact which is not true in general for other distributions.

This is interesting as there is a popular view that PCA is a model-free technique, since the mathematical operations needed to obtain PCs are valid for component distributions other than the Gaussian. However, the motivation in Hotelling (1933) for the technique was the discovery of independent components. Another piece of folk knowledge is that PCA is
rotationally invariant. The method set out by Hotelling is not rotationally invariant since the components are ordered in terms of decreasing variance\(^1\). Of course, one can apply further procedures to the result of PCA, but that does not change what PCA itself is.

The essential formulation of the PCA model is that our observed data have a vector representation with respect to some latent basis plus noise; that is, the data obey equation 2.1. The components of the observations with respect to the basis are uncorrelated. Furthermore an ordering can be obtained by optimising the proportion of the variance in the data accounted for by each component. It turns out that the solution is obtained by eigendecomposition of the sample covariance matrix, with the ordering being determined by the size of the eigenvalues (descending). The basis \(W\) is found to consist of the first \(K\) eigenvectors of the sample covariance matrix. The principal components are easily obtained as \(X = W'Y\) due to the orthogonality of the basis. The interpretation of the components \(X_{:,t}\) for the \(t^{th}\) observation is as the coordinates of the closest point on the latent manifold to the observed vector \(Y_{:,t}\).

PCA is applied widely throughout the sciences, and so we cannot aim to recapitulate a thorough history here. However, the paper of Bishop (1999) is of particular note for this thesis. It provided a fully Bayesian analysis of the PCA model. This drew on a more developed probabilistic variant of PCA set out in Tipping and Bishop (1999). The main advantage of probabilistic or Bayesian PCA models is an explicit noise model and model for the components so that the fitting of the parameters (\(W\) and \(X\)) is regularised by the amount of noise estimated to be present. These works are noteworthy because they formed part of the path of development towards one of the more popular methods for both dimensionality selection and independent components analysis employed in the study of neuroimaging. We revisit this issue in section 2.6.

### 2.3.2 Factor Analysis

Factor Analysis derives from the work of Spearman (1904), with Thurstone (1931) providing a generalisation to multiple latent factors. Again, data are treated as a linear combination of a number of latent variables with additive noise, implying the same algebraic form as equation 2.1. In FA the components \(x\) are referred to as factors and the basis \(W\) as factor loadings.

Much has been written about Factor Analysis and we will not be able to do this vast literature justice here. It is challenging to obtain a canonical description of the method(s) in use, a fact also noted by Jolliffe (2002, p.150). For instance, in the statistical encyclopedia of Everitt and Howell (2005), neither confirmatory nor explanatory Factor Analysis were stated as a model with its underlying assumptions. While the seminal paper of Rubin and Thayer

\(^1\)Barring the unlikely event of repeated eigenvalues in finite sample covariance matrices
2.3 Some factorisation models

Some factorisation models stated at least three cases under different assumptions on the variables.

Caveats issued, let us consider as an example the maximum likelihood approach to FA of Lawley and Maxwell (1963). In terms of equation 2.1, the entries of the components $x$ and noise $\varepsilon$ are assumed to be vectors of independent Gaussians. The noise is modelled probabilistically as $\varepsilon \sim \mathcal{N}(0, \Psi)$ where $\Psi$ is a positive definite diagonal matrix. A restriction is made on $W$ such that $W^\prime \Psi W$ must be diagonal, meaning that the columns of $W$ will not usually be mutually orthogonal, unless $\Psi = cI$. This is a point of departure from the PCA method, where $W$ is assumed to be orthonormal. Another point of contrast between this method and PCA is that the directions of the basis vectors found by this method for Factor Analysis are invariant to rescalings of the input variables.

2.3.3 Independent Components Analysis

The Independent Components Analysis (ICA) literature of the 1980s and 1990s grew from the neural networks and signal processing literatures as a particular case of the general problem of ‘blind source separation’ (Jutten and Taleb, 2000). The idea of blind source separation is that the observed data are unknown functions (hence ‘blind’) of a number of latent variables (the ‘sources’). If we can recover the functions and the sources we will have learned something interesting about how the data were generated. A famous example from this class of problems is the ‘cocktail party’: given $V$ microphones at a cocktail party with $K$ guests, can we recover $K$ audio tracks, each comprising just a single speaker’s contribution to the conversation?

The seminal paper of (Comon, 1994) defined the method as, "The independent component analysis (ICA) of a random vector consists of searching for a linear transformation that minimizes the statistical dependence between its components." As such, ICA methods again lead to the algebraic form of equation 2.1, but under the goal that the latent components, $x$, should be as independent as possible. Thus ICA could be seen as picking up where Hotelling’s comment (see section 2.3.1) left off by seeking latent variables which maximise some measure of independence, but without the restriction to their following a Gaussian distribution.

It is difficult to show from a finite sample of data that the independence property holds. For example, the independence of two random variables $x,y$ implies zero interaction in centered moments of all orders (since under independence $\int (xy)^d p(x,y) dx dy = E[x^d]E[y^d]$). It is not possible in general, with finite samples, to be certain that all these requirements hold simultaneously. Thus the search for independent latent variables must use some proxy measure of independence in order to assess how good a particular solution is.
In two of the most widely used variants of the ICA model, InfoMax (Bell and Sejnowski, 1995) and FastICA (Hyvärinen and Oja, 2000), the basis, $W$, is taken to be an invertible matrix of dimension $V$ chosen to optimise some measure of the independence of the latent components $x$, while the noise $\epsilon$ is assumed to be zero.

Bell and Sejnowski (1995), framed the ICA problem as one of maximising the mutual information between observed data, $y$, and the latent variables, $z$, implied by an invertible non-linear transformation of the data. This involved the inference of the parameters $(W, b)$ of a transformation $z = g(W^{-1}y + b)$ by maximisation of the entropy of the $z$. The required calculations were provided for several non-linearities $g$, such as the logistic sigmoid. Note that this form of equation for $z$ is a single layer artificial neural network (ANN). Single layer because $z$ is the final ‘output’ and ANN because the concept of an affine transformation followed by an element-wise non-linear transformation is a very common model for the activity of artificial neurons. In terms of equation 2.1 the latent components of the basis are $x = g^{-1}(z) - b$.

In Hyvärinen and Oja (2000) an appeal was made to the central limit theorem, which, in broad brush strokes, says that sums of random variables are more Gaussian than the variables summed. So, if we observe $V$ different sums of $V$ variables we may be able to reverse the summation by constructing a further $V$ variables which are as non-Gaussian as possible. Non-Gaussianity can be measured in terms of negentropy (Hyvärinen et al., 2004). Negentropy is the difference in entropy of a distribution with density function $f$ and fixed covariance from a Gaussian density $g$ with the same covariance as $f$, that is $J[f] = H[g] - H[f]$ when $\mathbb{H}$ is the entropy functional. Negentropy is always nonnegative $J[f] \geq 0$ and can equal zero only when $f$ is itself a Gaussian density. It was shown in Hyvärinen and Oja (2000) that the maximisation of negentropy was the same, up to a constant, as the maximisation of mutual information approach to ICA derived in Bell and Sejnowski (1995).

In the most popular implementations of ICA, the concepts of dimension reduction and latent variables parted ways. The focus went towards the identification of independent latent variables. This is primarily because the family of methods including Bell and Sejnowski (1995) and Hyvärinen and Oja (2000) rely on the linear transformation of the observed data being invertible. In both cases special use is made of the Jacobian of a linear transformation, that is $|\text{det}W|$, meaning that $W$ must be square. The cumulant based JADE approach to ICA of Cardoso (1999), termed JADE, also seeks a square, invertible $W$. This implies that if we observe $V$ variables our ICA method will return $V$ latent variables which are as statistically independent as the implementing algorithm can make them. Whereas an ICA method that also implemented dimension reduction would require the inference of a transformation of the observed data to $K < V$ latent variables.
2.3 Some factorisation models

Other ICA methods exist which also implement dimension reduction; and the application of the adjective ‘non-square’ or ‘square’ when describing an ICA method is a shorthand for whether dimension reduction is implemented or not. For instance (Penny et al., 2001) built on corresponding work by (Minka, 2000) for probabilistic PCA to provide a framework for incorporating dimension reduction into an ICA model; and this model has been important in neuroimaging. An alternate approach was that of (Attias, 1999) which took the route of generalising FA rather than PCA.

2.3.4 Other notable linear methods

**Alternating Least Squares**

The work of Koren et al. (2009) detailed a matrix factorisation method which fit a model of the same algebraic form as equation 2.2 in which $Y$ was incompletely observed. The method has become somewhat famous in industrial applications by winning for the authors the top prize in a $1m competition sponsored by the online video service Netflix. In particular $Y_{i,j}$ represented the rating given by used $i$ to film $j$. Since most users had not rated most films $Y$ was very sparse, or alternately the matrix was mostly ‘missing data’. Their approach considered the rows of $W$ as $K$ vectors and the columns of $X$ as $K$ vectors. This facilitated consideration only of observed data and therefore a way of handling missing data, a task not possible with the methods discussed so far in this chapter. The approach is also noteworthy for our purposes as it forms an alternative interpretation of the basis matrix, and in chapter 3 we discuss the impact such decisions have on models for neuroimaging. The objective of the method was to minimise the sum of squared differences between the observations $Y_{i,j}$ and the model outputs $(WX)_{i,j}$. The fitting process iteratively held $W$ fixed and optimised with respect to $X$ then alternated by holding $X$ fixed and optimising $W$.

**Non-negative matrix factorisation**

Lee and Seung (2001) presented algorithms for non-negative matrix factorisation (NNMF). These imposed constraints that both the basis and components of the dataset should contain no negative entries. Inspired by the EM algorithm, they showed how to construct a sequence of estimates such that the objective function of the sequence was non-increasing. Two objective functions were considered, the Euclidean distance (similar to PCA and FA methods), and a divergence measure that reduced to Kullback-Leibler divergence ² in the case that the matrices obeyed a sum-to-one constraint. NNMF simplifies the interpretation of the latent

---

²The KL divergence from a distribution $p$ to a distribution $q$ is defined as $D_{KL}(q||p) = E_q \left[ \ln \left( \frac{q(x)}{p(x)} \right) \right]$
variables involved since there can be no ‘cancelling out’ of the contribution of one latent variable by another since all are non-negative.

**Exponential Family PCA**

The work of Collins et al. (2001) provided an extension of PCA for data, \( Y \), which are ill approximated by standard real valued PCA, such as count data, binary variables, and strictly positive variables. Instead of factorising the data matrix, we instead factorise the natural parameters \( \Theta = WX \) of the data when the entries \( Y_{i,j} \) are considered to be drawn from some exponential family distribution with inverse link function \( E[Y_{i,j}] = g(\Theta_{i,j}) \). This reduces to probabilistic PCA when the link function is the identity function, defining a Gaussian. The work of Mohamed et al. (2009) provided a Bayesian variant of this approach.

**Sparse PCA**

Sparse PCA (Zou et al., 2006) provides a model in which the components \( X \) of the data are expected to be sparse with respect to the basis \( W \). The components are obtained as \( X = B'Y \) through an \( V \times K \) matrix \( B \), so as to minimise the objective function

\[
||Y - WX||^2 + \lambda \sum_{k=1}^{K} ||B_{:,k}||^2,
\]

where \( \lambda \) is a regularisation parameter which encourages sparsity. The interpretation of the sparsity here is that only a few of the variables in each observation \( y_i \) are used in the calculation of the \( k^{th} \) component \( (X_{k,i} = B'_{:,k}Y_i) \) because most of the entries in \( B_{:,k} \) will be pushed close to zero. If the constraint \( W = B \) is introduced and the penalty term ignored then this model reduces to ordinary PCA.

**Sparse decomposition in a signal dictionary**

Sparsity may also effected by assuming the components of the data \( X \) are sparse with respect to the basis \( W \) under a transformation. For instance the method of Zibulevsky and Pearlmutter (2001) assumed that the data could be factorised as

\[
Y = WC\Phi
\]

Where \( C \) is a \( K \times M \) matrix and \( \Phi \) an \( M \times T \) matrix whose rows are fixed basis vectors (e.g. wavelet basis functions). So that the components of the data are \( X = C\Phi \). The goal of the technique was to infer an appropriate basis \( W \) for the space of observations, while
making the chosen pattern of activations $C$ as sparse as possible. The implication of this is the introduction of a dependence structure within the $T$ domain, and, potentially some regularisation through the sparsity constraint on $C$.

### 2.3.5 Beyond linear models

We have seen so far that the linear algebraic form of equation 2.1 has been at the heart of a range of models including PCA, FA, ICA and other variants. It is natural to ask where we can turn to when such a form is not appropriate. For instance, it would seem highly unlikely that a linear model would be the most suitable in a setting involving images of objects which may occlude one another, since their effects would not be additive.

The neural networks and machine learning literatures have provided a continuing (cf. section 2.3.3) source of developments towards models where the observed data are treated as nonlinear transformations of latent variables. An important paper was that of Hinton et al. (2006) which established that it was possible to obtain useful results in image analysis through deep neural networks. In particular the method involved learning a neural network composed of multiple layers, one layer at time. At each layer an unsupervised model, the Restricted Boltzmann Machine (RBM), was inferred, except at the final layer, where the latent variables were used as predictors in a regression. In this respect it could be regarded as a non-linear alternative to a technique such as Principal Component Regression (PCR) in which the matrix of components $X$ produced by a PCA model is used as the design matrix for a regression. In another respect, the use of the RBM made the technique resemble a multi-layer ICA in which the variables in a layer were conditionally independent given the output of the previous layer, with a regression model on the topmost level linking to the dependent variable.

Although influential, the techniques in Hinton et al. (2006) were superseded by more conventional forms of backpropagation techniques based on stochastic gradient descent; for example see the seminal Krizhevsky et al. (2012). However, other work exists which provides a link between unsupervised probabilistic modelling of neural networks and use of backpropagation infrastructure. Variational Bayesian methods can be used in conjunction with gradient descent techniques by specification of an appropriate objective function (Graves, 2011; Kingma and Welling, 2014). Indeed, unsupervised deep neural networks have attracted sufficient attention recently to develop into the separate subfield of representation learning (Bengio et al., 2013). That literature concerns itself with learning the parameterisation of a nonlinear manifold close to which the data vectors live.

This thesis will focus on linear models for multivariate data. This is for both methodological and practical reasons. It is not easy to visualise the representation of data in terms of latent
variables through non-linear transformations. In particular the tangent space at different points on a nonlinear manifold may be visualisable, but may be different at each point on the manifold. In contrast, linear modelling implies that the tangent space is identical at all points on the latent manifold. A practical implication of this is that where linear methods can provide $K$ images that offer a global visualisation of the latent manifold nonlinear methods cannot. Although there may be intelligent ways of picking interesting locations on a nonlinear manifold to visualise locally. Other considerations are sample size and computation time. Neural networks methods typically require large sample sizes to train effectively.

We will later see in neuroimaging that our tangent space would be spanned by $K$ brain images, each of which was a visualisation of a three dimensional block, which is a sufficiently challenging quantity to visualise appropriately. Furthermore it is frequently exactly such representations of data structure that are sought where latent variable methods are used in neuroimaging.

### 2.4 Neuroimaging data

#### 2.4.1 A brief introduction to fMRI

The introduction of functional Magnetic Resonance Imaging (fMRI) is largely credited by Rosen and Savoy (2012) to seminal work by Belliveau et al. (1991) and Kwong et al. (1992). The work in Belliveau et al. (1991) provided early evidence that changes in cognition could be captured through Magnetic Resonance Imaging (MRI), but required administration of intravenous contrast agents. The work of Kwong et al. (1992) showed both the possibility of using endogeneous contrast agents to measure *in-vivo* changes in the brain’s functional activity through MRI and the acquisition of dynamic data, with images separated by seconds rather than minutes. The endogenous contrast agent being deoxyhemoglobin, which is found in human blood.

The potential of using blood oxygenation as a contrast agent was highlighted in Ogawa et al. (1990) which termed the signal Blood Oxygenation Level Dependent or BOLD. The underlying theory being that areas of the brain which do more work require more oxygen. The blood supply then adapts to these needs by increasing the availability of oxygenated blood in the region. Changes in oxygenation alter the local magnetic properties of the blood, which can then be detected through MRI. It was this link between the cognitive functioning of the brain and MRI signal which gave rise to the fMRI acronym.

The paper of Kwong et al. (1992) also set out methods for statistical analysis in its consideration of block experimental designs. The idea in these designs is to present one or
many stimuli or tasks to participants for a certain periods of time (blocks), usually with a control stimulus. The difference in averages between control and task blocks reveals the areas of the brain that are more or less active during the task. Block experimental design was a dominant method in fMRI neuroimaging throughout the early 1990s (Snyder and Raichle, 2012).

Resting state describes the acquisition of fMRI data where subjects are not instructed to undertake any particular task. A motivation for this is essentially that the brain consumes more energy than its weight would suggest, but that the changes in energy consumption due to tasks are very small (Raichle and Mintun, 2006; Zhang and Raichle, 2010). This suggests the study of intrinsic activity should be important to the understanding of brain function. Studies in this line of enquiry have found consistent evidence for networks of connectivity in resting state, termed the default mode network Buckner et al. (2008); Raichle (2015). In particular where lesion models (localised damage) fail to explain a disease, explanations might be sought in terms of wider disruption in the functional connectivity between brain regions, for instance in depression, schizophrenia or dementia (Broyd et al., 2009).

A related method to resting state fMRI is the acquisition of fMRI data under ‘naturalistic’ stimuli. Typically this implies the presentation of audio or video to subjects while being scanned. The aim here is to induce greater synchronisation between subjects (Jääskeläinen et al., 2008) under richer conditions which should recruit more complex functional machinery (Lahnakoski et al., 2012).

It may be no surprise that the biology of the brain, and in particular the BOLD signal, is more complex than this simple picture suggests. Issues include different neuronal densities and patterns of neuron usage across regions, and the inability to distinguish among different
types of neuronal activity, such as whether a connection is excitatory or inhibitory (Logothetis, 2008). Understanding of brain metabolism is imperfect, complicating the interpretation that should be given to changes in blood flow (Raichle and Mintun, 2006).

2.4.2 Dimensions, content and format of fMRI data

The purpose of this section is to ground the reader in the general dimensions and format of fMRI datasets. We believe that this sort of information is helpful context for the later discussion of analysis methods.

The number of subjects participating in a study, $N$, will often be on the order of tens, while a large study may have hundreds of subjects. The composition of the sample will depend on the scientific question under consideration. This can vary from populations as specific as professional jazz pianists (Limb and Braun, 2008) to populations as broad as cognitively healthy adults from the 18-87 age range (Shafto et al., 2014).

Each subject may have a number of sessions in the scanner, or runs, $R$. This occurs for several reasons. First, there may be different experimental conditions under investigation, such as a cognitive task and resting state (Shafto et al., 2014). Similarly, longitudinal measurements of subjects over time frames of weeks, months, or years may be required (Ward et al., 2003). Second, there are practical limits on the length; for instance in some experimental paradigms the risk of subjects falling asleep is higher on longer scans (Tagliazucchi and Laufs, 2014).

From the perspective of the data analyst, each scanning session will give rise to data comprising measurements at $H$ time-points of brain volumes. A volume is an array with three indices representing the different spatial dimensions. The elements within the array are termed voxels, for volumetric pixels. We will refer to the total number of voxels as $V$, and to the number of voxels on axis $i$ as $V_i$, so that $V = V_1V_2V_3$.

Alongside, and usually prior to, the functional MRI scan, an anatomical MRI scan will be performed. The difference between the two lying in what signal the scanner is tuned to track most closely. The most commonly used mode (T1 contrast) differentiates grey and white matter than the modes (T2, T2$^*$) used to track blood oxygenation for functional imaging. An anatomical scan provides a single volume for each subject, typically at a higher resolution than the functional data. It is usually these anatomical scans which will later be used to standardise the data across subjects by aligning them to a common space (Huettel et al., 2014, see chapter eight of).

The most popular storage format for fMRI data is Neuroimaging Informatics Technology Initiative (NIfTI). Files of this type contain not only the data, but also metadata. A single NIfTI file will typically contain the results for one run for one subject. A common piece
of metadata is the linear transformation needed to relate the voxel indices and a ‘standard’ space.

Alongside the functional and anatomical scans, a variety of other side information may be available. For instance, we may have covariates relating to the subject, particularly in terms of demographics and health status. We may have covariates related to the experimental setup, such as information regarding the stimulus presented to the subject at each time point in the scan. Alternate types of scans may also be available, for instance mapping vasculature within the subject’s brain. The availability, or not, of any of these data, together with the scientific question under consideration, will influence the choice of statistical analysis undertaken on the data.

2.4.3 Preprocessing

As set out by Monti (2011) a number of preprocessing steps are commonly applied to fMRI data prior to its statistical analysis: performing slice time correction, as 3D volumes are compilations of 2D slices taken close together in time; rigid transformations of the data, to account for subject movement; and nonlinear warping of data to a common template so that the spatial index should have the same interpretation across subjects.

The most common standard space for analysis is called MNI 152 (after the Montreal Neurological Institute, Grabner et al. 2006), which is an anatomical template based on the combined brains of 152 individuals. By aligning the data from particular studies with the template, researchers can be more specific about the location of findings in the brain and so compare findings from different studies more easily.

Band pass filtering involves the removal of high or low frequencies from the data. For fMRI data this means considering the time-series of data at each voxel. In particular, low frequency oscillations in brain activity, e.g. over the course of minutes, are commonly seen as artefactual and thus their removal can be beneficial for analysis, while the usefulness of attenuation of high frequencies is not clear (Skudlarski et al., 1999).

Spatial smoothing is a further pre-processing step commonly undertaken prior to analysis. The motivation is typically a combination of reducing the impact anatomical differences between individuals could have on the analysis, and of improving the signal-to-noise ratio by averaging out noise (Huettel et al., 2014, p. 315). Kernel smoothing is the method most commonly applied to achieve spatial smoothing of fMRI, which replaces each voxel value with a linear combination of its neighbours using a set of weights. The standard kernel used is Gaussian, so that the weights decay as a bell shaped curve with distance from the voxel being replaced (Brett, 1999). The rate of weight decay is reported in terms of Full Width at Half Maximum (FWHM) in units of millimetres. The FWHM is the length of an interval such
that the value of the function at the ends of the interval are half of the values at its maximum. FWHM can be converted to standard deviation, $\sigma$, via the relation $FWHM = 2\sqrt{2\log 2}\sigma$.

The use of a sequence of pre-processing steps, followed by later statistical analysis steps, is often referred to as a pipeline (Strother, 2006). For the purpose of this thesis we will take preprocessing, including alignment and temporal high-pass filtering, but not spatial smoothing as given. This will allow us to utilise data aligned to a common space via either linear or nonlinear methods.

The spatial smoothing step was identified by (Strother, 2006) as one of the most influential. Fig. 2.2 shows slices from the same volume within the Cam-CAN dataset under two different levels of smoothing. The value of the FWHM is typically fixed at an assumed value based on some combination of researcher experience and data quality. In chapter 4 we shall begin to see that spatial smoothing, as a linear method, can be integrated into statistical matrix factorisation models. This will have at least two advantages. First, it facilitates a data-driven approach to deciding on smoothing parameters. Second, by inferring smoothness from the data we do not have to globally blur the data prior to analysis.

### 2.5 Matrix factorisation of fMRI

We present here a sketch of the manner in which matrix factorisation is applied to single-subject fMRI data in an ICA analysis. Most fMRI analyses are of group data, and there are a number of ways to approach this, which we discuss in section 2.8. However, we believe the approach outlined here is a useful template for understanding the group approaches.

Consider again a $V \times T$ data matrix $Y$ where $V$ represents the number of voxels in the data, and $T$ the number of volumes obtained during a run. A common way for matrix factorisation to proceed here is to use PCA to reduce the dimension of each observation from $T$ to $K$, with $K$ chosen by a strategy such as those discussed in section 2.6. Using PCA we factorise our data...
2.5 Matrix factorisation of fMRI

into a $T \times K$ matrix $W^{PCA}$ and a $K \times V$ matrix of components $X^{PCA}$ as

$$Y' \approx W^{PCA} X^{PCA},$$

(2.3)

where the superscripts on the right hand side serve to differentiate this stage of factorisation from later ones. Note the transpose here (to maintain consistency with our previous discussion) the data are interpreted as a dataset comprising $V$ time series vectors of length $T$; we will return to this modelling choice in Chapter 3.

One of the usual square ICA methods (see section 2.3.3) is then used to further factorise the principal components into a $K \times K$ matrix $W^{ICA}$ and a $K \times V$ matrix of components $X^{ICA}$ as,

$$X^{PCA} = W^{ICA} X^{ICA}.$$  

(2.4)

Substituting from equation 2.4 back into the PCA decomposition of the data, equation 2.3, we see that

$$Y' \approx W^{PCA} W^{ICA} X^{ICA},$$

(2.5)

When the ICA basis matrix $W^{ICA}$ is orthonormal this second stage amounts to a rotation of the principal components, and otherwise we see that the independent components are in the span of the principal components.

ICA analyses of fMRI data which treat the data as $V$ observations (one for each voxel) of $T$ dimensional vectors (the timeseries through each voxel) are termed ‘spatial’ ICA, because each spatial index, $i = 1, \ldots, V,$ receives a set of latent components. The spatial approach was introduced by McKeown et al. (1998), and has now been incorporated, with modifications, into major software packages such as FSL (Beckmann and Smith, 2004) and SPM (Calhoun et al., 2001). The columns of $W^{PCA} W^{ICA}$ are $T$ dimensional vectors interpretable as time-courses. While rows of $X^{ICA}$ can be interpreted as vectors from the $V$ space inhabited by the original fMRI volumes.

The untransposed approach which instead starts by using PCA to factorise the $V \times T$ matrix $Y$ is known as ‘temporal’ ICA, because each volume receives latent components with a time-like index $i = 1, \ldots, T$. The spatial approach is more common than the temporal approach due to difficulties associated with the ratio of the dimension of each volume to the number of volumes observed $V / T$, a topic which we continue to prepare for next and discuss in further detail in chapter 3.
2.6 Dimension estimation for matrix factorisation

2.6.1 Neuroimaging context

We now consider some methods for estimating the appropriate dimensionality of matrix factorisation models that feature in widely used neuroimaging analysis pipelines. The order, $K$, of the decomposition will typically be determined by a method based on the eigenvalues of the $T \times T$ covariance matrix $\mathbf{Y}'\mathbf{Y}$, assuming the columns of $\mathbf{Y}$ are mean centred. We specify the $T \times T$ covariance matrix here because this is implied in spatial ICA methods. We discuss modifications for multi-subject data in Sec 2.8, and whether spatial or temporal ICA is most appropriate in chapter 3.

Available methods include a Laplace approximation to the model evidence of a Bayesian Probabilistic Principal Components Analysis model (LAP, Minka (2000)) or Minimum Description Length of linearly transformed multivariate gaussians (MDL, Wax and Kailath (1985)). Both of these methods were developed in the general case outside of neuroimaging and later applied to fMRI analysis.

Approaches based on the above methods have been influential due to their incorporation within widely distributed, used, and cited neuroimaging software. The formula based on MDL is used within the GIFT toolbox for the SPM software package Li et al. (2007); Rachakonda et al. (2007). The LAP formula is used within the FSL software package (Beckmann et al., 2005; Beckmann and Smith, 2004, 2005). In both cases additional modifications are made to the data or eigenvalues passed to the objective functions; discussed further in section 2.7. We present here the original formulae expressing the objective functions.

In both cases the formulae are structured as functions of the empirical eigenvalues and a proposed latent dimensionality, together with the dimensions of the data. We can construct a curve of scores for different settings of the latent dimensionality, $K = 0, \ldots, n - 1$ where $n$ is the observed dimensionality of the data vectors (number of eigenvalues of whatever covariance matrix is being used). We then choose the value of $K$ which optimises the score; minimise for MDL, maximise for LAP.

Both the MDL and LAP formulae are based on an assumption that the data vectors are independently, identically distributed. In neuroimaging this would correspond to an assumption that activity in the brain is not spatially dependent; this is a strong assumption to make when we have strong prior evidence that at least some local regions of the brain function as units. The method of Li et al. (2007) for estimating the appropriate dimension of fMRI data based on the LAP formulae attempts to address this by taking a spatial subsample of the data so that the effect of spatial dependence within the brain is mitigated.
We have still not resolved the issue of whether the ‘spatial’ or ‘temporal’ form of analysis is most appropriate. This resolves into the question of whether the dimension of the observed vectors and their quantity should be interchanged. We have already promised to discuss this in chapter 3. For now, we shall proceed on the basis of a ‘spatial’ interpretation with the number observed vectors being the number of voxels.

### 2.6.2 Minimum Description Length estimation

The MDL formula of Wax and Kailath (1985) involves the quantities set out in Table 2.1. It is assumed that we are dealing with data vectors that have a multivariate gaussian distribution and decompose in the linear form of equation 2.1.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>Dimension of observed vectors</td>
</tr>
<tr>
<td>V</td>
<td>Number of observed vectors</td>
</tr>
<tr>
<td>K</td>
<td>Proposed latent dimensionality</td>
</tr>
<tr>
<td>$\lambda_i$</td>
<td>$i^{th}$ empirical eigenvalue</td>
</tr>
</tbody>
</table>

Table 2.1 Quantities for the MDL method

The MDL formula itself is presented in equation 2.6 and involves the log likelihood of the data evaluated at its maximising value for a proposed latent dimensionality additively together with a penalty term depending on the number of parameters involved:

$$l_{MDL}((\lambda_i, K)) = -\log \left( \frac{\prod_{i=K+1}^{V} \lambda_i^{1/(T-K)}}{\frac{1}{p-k} \sum_{i=K+1}^{T} \lambda_i} \right)^{(T-K)V} + \frac{1}{2} K(2T - K) \log V \quad (2.6)$$

This quantity represents the length of a code required to transmit the model and a summary of the data under the model. Smaller is better, with a small $l_{MDL}$ indicating that the model effectively describes the variability in the data.

### 2.6.3 Laplace approximation to Bayesian PPCA evidence

The LAP formula is for a Bayesian probabilistic PCA model (Minka, 2000). This again means a model of the algebraic form of equation 2.1, with latent Gaussian components, an orthonormal basis and Gaussian observation noise. The notation used in the original
presentation is shown in Table 2.2. We again refrain from translating this notation into our own for the same reasons as above.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>expansion</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>Observed dimensionality</td>
<td>$\sum_{i=K+1}^{T} \lambda_j$</td>
</tr>
<tr>
<td>V</td>
<td>Number of observations</td>
<td>$T-K$</td>
</tr>
<tr>
<td>K</td>
<td>Latent dimensionality</td>
<td>$\lambda_i$</td>
</tr>
<tr>
<td>$\lambda_i$</td>
<td>$i^{th}$ empirical eigenvalue</td>
<td>$\lambda_i$ if $i \leq K$ else $\sigma$</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>noise variance</td>
<td>$\lambda_i$</td>
</tr>
<tr>
<td>$\hat{\lambda}_i$</td>
<td>$i^{th}$ modelled eigenvalue</td>
<td>$\lambda_i$</td>
</tr>
</tbody>
</table>

Table 2.2 Quantities for the LAP formula

Some requirements set out by Minka were that $V > T$ and $T > K$. Hence, if we model our neuroimaging data as $V$ observations of length $T$ time-series, these requirements will generally be met. However, if we suppose that what we really have are $T$ observations of $V$ dimensional spatial vectors we would usually violate these requirements. Which way round to look at the data is a modelling assumption, and we will return to some of its implications in chapter 3.

$$a = \prod_{i=1}^{K} \prod_{j=i+1}^{T} (\hat{\lambda}_j - \hat{\lambda}_i)(\lambda_i - \lambda_j)V$$

$$u = 2^{-K} \prod_{i=1}^{K} \Gamma((T - i + 1)/2)\pi^{-(T-i+1)/2}$$

$$p_{LAP}(\{\lambda_i\}, K) = u \left( \prod_{j=1}^{K} \lambda_j \right)^{-V/2} \sigma^{-V(T-K)/2} (2\pi)^{(m+K)/2} a^{-1/2} V^{-K/2}$$ (2.7)

It is worth noting that Minka (2000) also provides a simplified version of the LAP criterion which is a Bayesian Information Criterion (BIC). This is done by ignoring terms that do not grow with $V$. The BIC formula is nearly identical to the MDL criterion, and so we do not complicate our present task by examining it further in our simulations etc. This is also worth knowing, in that it establishes an expectation that the LAP should perform better in small samples, since it retains terms that become irrelevant in the limit of large sample sizes.
2.7 Eigenvalue corrections

In Beckmann and Smith (2004) a transformation is applied to the empirical eigenvalues prior to use of the LAP criterion for dimensionality selection. The intended transformation, as we understand it, is to calculate the eigenvalues of the \( T \times T \) covariance matrix and compare them to another set of \( T \) eigenvalues. A number of aspects of this approach are perplexing from a statistical and mathematical perspective. The other set being, we understand, the \( T \) quantiles of the ‘expected eigenspectrum of a random Gaussian matrix’. The authors here mean the eigenspectrum of the Wishart distributed sample covariance matrix.

First, the authors denote the inverse cumulative distribution of a (scalar) random variable \( \nu \) by \( G^{-1}(\nu) \). This is confusing since a statement is made equivalent to the density being supported on an interval whose upper endpoint is greater than unity, implying that arguments outside \([0,1]\) are being passed to an inverse CDF. In fact we believe (from looking at implementing code) that the intention was to describe acquisition of comparison ‘eigenvalues’ by taking \( i = 1, \ldots, T \) evenly spaced points \( \zeta(i) \) in \([0,1]\) and passing them through the inverse CDF to obtain \( \tilde{\lambda}(i) = G^{-1}_\nu(\zeta(i)) \) which function as a form of theoretical order statistics.

Second, ratios are then made of the empirical eigenvalues \( \lambda(i) \) over the theoretical eigenvalues \( \tilde{\lambda}(i) \) so that \( r(i) = \lambda(i)/\tilde{\lambda}(i) \). These \( r(i) \) are passed into the LAP criterion \( p_{LAP} \), which the reader will recall is a function of an empirical eigenspectrum, not a set of ratios. The paper does not provide a justification as to why this proposed correction is a valid procedure, and while it is possibly a failure of imagination we don’t see a clear reason why a justification should be available.

The reference from which the method appears to be inspired, Johnstone (2001), discussed a similar technique where the empirical and ‘theoretical’ eigenvalues were compared via a sort of QQ-plot. The author of that paper concluded of this visual aid that, "Without variability information on the null distribution, one cannot say with rigor whether this is real."

The effect of using the \( r(i) \) as ‘corrected’ eigenvalues might be interpreted as follows. Suppose we plot the descending empirical eigenvalues of the data, and regard these as a scree plot. The latent dimensionality \( K \) should be identified by a ‘kink’ in the plot such that the the last \( T - K \) eigenvalues are similar in value compared to the first \( k \) which are to be much larger. The LAP formula (equation 2.7) provides a way of scoring the extent to which this ideal is the case. Now, if the last \( T - K \) ordered eigenvalues are close to the expected order statistics, \( r(i) \) should be close to unity, and there should be a ‘kink’ at the \( K^{th} \) eigenvalue because afterwards there should be only a straight line . Since \( \text{eig}(\mathbf{W} \mathbf{W}^T + \sigma \mathbf{I}) = \text{eig}(\mathbf{W} \mathbf{W}^T) + \sigma \) and if \( K < T \) then \( \mathbf{W} \mathbf{W}^T \) has many zero eigenvalues so that if the signal (non zeros of \( \text{eig}(\mathbf{W} \mathbf{W}^T) \)) is sufficiently large, the smaller eigenvalues are most likely determined by the noise model. We
stress that this is only trying to provide an intuition for why an approach like analysis of the \( r(i) \) might have been inspired.

## 2.8 Analysis of group fMRI data

### 2.8.1 Group fMRI data

In section 2.5 we gave a sketch of matrix factorisation methods as applied to single subject fMRI data. The approach discussed operated in two stages with PCA factorisation followed by ICA factorisation. Similar approaches are commonly used for group datasets (and there may be further factorisation stages), and this raises the question of how to apply dimension reduction.

For clarity let us set out the dimensions of group data, and what we mean by concatenation of matrices. If we have \( N \) subjects, \( n = 1, \ldots, N \), in a dataset each of whom provide a \( V \times H \) matrix \( M^n \) then we have \( NH = T \) observations in total. By the horizontal concatenation of \( n = 1, \ldots, N \) matrices \( M^n \), which must all have the same number of rows, we mean the \( V \times T \) block matrix formed as,

\[
M = [M^1, M^2, \ldots, M^N]. \tag{2.8}
\]

Factorisation of a matrix formed by horizontal concatenation can also be viewed in block terms as

\[
M = AB = A[B^1, B^2, \ldots, B^N], \tag{2.9}
\]

where each of the blocks \( B^n \) has the same number of columns as the corresponding \( M^n \) and can be seen to represent latent components for the original data with respect to the space spanned by \( A \).

Whereas if \( M \) were the vertical concatenation of the matrices \( M^n \), we would mean an \( NV \times H \) block matrix:

\[
M = \begin{bmatrix} M^1 \\ M^2 \\ \vdots \\ M^N \end{bmatrix}, \tag{2.10}
\]

and this can be seen as having an analogous block factorisation to the case of horizontal concatenation above, but with the roles of rows and columns reversed.

Some options used for the first stage of factorisation in a group analysis are to: reduce each subject’s data using a separate model, then concatenate; to reduce each subject’s data
with a common model, then concatenate; or to concatenate the group’s data and reduce it collectively. We now discuss some notable methodologies which exemplify these approaches.

### 2.8.2 Reduction to a common space prior to concatenation

In Calhoun et al. (2001) the data are reduced using PCA with \( K_i = K \) (it appears) for all subjects. It is not clear how the authors choose the latent dimensionality for this stage: e.g. is \( K_i \) estimated for several subjects and the mode taken, or does the \( K_i \) come from a randomly chosen subject etc. The reduced data are concatenated in terms of rows. The reduced data is further reduced using the MDL criterion (or related AIC criterion). This is the method implemented in the GIFT software within the popular SPM package Rachakonda et al. (2007). A square ICA model is then fitted to the twice-reduced data.

A similar method to the above is called concat ICA in the MELODIC software within the popular FSL package. As far as we are aware the details of the approach are not published. Nevertheless we feel this is important to note as the software is one of the more popular packages for ICA of fMRI data; with a related paper, Beckmann and Smith (2004), having over 1500 citations.

### 2.8.3 Separate reduction prior to concatenation

In work by Shi and Guo (2014) and in Guo and Tang (2013) each subject’s data \( Y_i \) is reduced to its own rank \( K_i \) basis. The paper implies \( K_i = K \) for all subjects, but it is not clear which subject’s (or combination thereof) data is used to select \( K \), although it is clear that the method used is the LAP approximation discussed in section 2.6.3. The data are temporally concatenated and an ICA model fitted to the reduced data.

In the method of Varoquaux et al. (2010): \( K \) is chosen using the data from a single subject using the method of Mei et al. (2008); a separate rank \( K \) PCA is fitted for each subject; the reduced data are temporally concatenated (the resulting matrix is now of shape \( NK \times V \)). The results are further reduced using ‘Canonical Correlation Analysis’ (CCA) implemented as singular value decomposition of the concatenated matrix. It might be preferable to refer to this step as a further PCA, since it is not clear in what sense the concatenated matrix represents correlations to decompose. A novel approach to selection of the latent dimensionality is made at this stage using the bootstrap to estimate a threshold for singular values. ICA is then fitted to the result.

To make this concrete, suppose we had a dataset comprising two subjects Ann and Bob whose data are \( Y^1 \) and \( Y^2 \) respectively, then the first stage ‘spatial’ factorisation into \( T \times K \)
matrices $W^n$ and $K \times V$ matrices $X^n$ is:

$$Y_1' = W_1 X_1,$$

$$Y_2' = W_2 X_2,$$  \hspace{1cm} (2.11)

The component matrices $X^n$ are then vertically concatenated to form a $2K \times V$ matrix, $X$, which is then factorised into a $2K \times M$ matrix $A$ and an $M \times V$ matrix $B$

$$X = \begin{bmatrix} X_1^T \\ X_2^T \end{bmatrix} = \begin{bmatrix} A_1^T \\ A_2^T \end{bmatrix} B = AB,$$  \hspace{1cm} (2.12)

so that substitution of elements from equation 2.12 into equation 2.14 yields,

$$Y_1' = W_1 A_1 B,$$

$$Y_2' = W_2 A_2 B,$$  \hspace{1cm} (2.13)

The rows of $B$ are vectors in $\mathbb{R}^V$ space and so can be plotted as brain images and subjected to interpretation. Their interpretation is somewhat complicated by virtue of the vertical concatenation used in equation 2.12. The columns of the principal component matrices, $X^n$, used there are components of vectors in different rank $K$ vector subspaces of $\mathbb{R}^T$. The outcome is a set of common spatial patterns in the rows of $B$ and subject specific time-courses in the columns of the $W^n A^n$.

### 2.8.4 Tensor factorisation

The method of Beckmann and Smith (2005) is implemented as the Tensor ICA function of the FSL MELODIC software. In this approach the data are concatenated such that we have $NV$ observations of $H$ dimensional vectors. $K$ is chosen using the concatenated data using the LAP score, and the data reduced into $\mathbb{R}^K$. This means that the dimension estimation and reduction step is a traditional PCA as set out in section 2.3.1 rather than exhibiting some additional tensor property. The reduced data are modelled with a tensor form of ICA. Tensor ICA produces a factorisation involving a $N \times 1$ matrix $C$ of subject specific weights, a $H \times K$ matrix of shared time-courses $A$ and a matrix of spatial components $X$:

$$Y' = (C \otimes A)X,$$  \hspace{1cm} (2.14)
Where $\otimes$ is the Khatari-Rao product taking the columns of the matrices operated on to be blocks. Which is again of the same algebraic form as equation 2.2. Note that there are only $(N + H)K$ free parameters in the basis, which is generally far fewer than those available in the group methods described in sections 2.8.2 or 2.8.3 which have $NHK$. The structure of the basis suggests that it has the capacity to fit the data well when all of the subjects possess the same latent time courses up to a constant of proportionality, and may struggle when this is not the case.
Chapter 3

Failure modes for eigendecomposition methods

3.1 Introduction

The purpose of the present chapter is to explore some of the potential issues arising from use of conventional matrix factorisation pipelines in the neuroimaging setting. These include violations of assumptions, mis-estimation of latent dimensionality, conflicts caused by use of two stages of modelling and distortions introduced by model constraints. We conclude with some desiderata for later work.

3.2 Time-series or spatial processes?

Our concern here is to critically evaluate conventional ICA pipelines for fMRI data. Let us start from the beginning by reconsidering basic assumptions about the data matrix, \( Y \in \mathbb{R}^{V \times T} \). Should we interpret this as \( T \) observations of a \( V \) dimensional spatial process? Or \( V \) observations of a length \( T \) time-series? Or perhaps one observation of a \( V \times T \) spatio-temporal process, or possibly many such observations if we have multiple subjects? A further set of considerations arises in the context of group data, where each of \( N \) subjects brings \( H \) images. Do we concatenate temporally, that is should our data matrix be \( V \times NH \), or concatenate spatially for a \( H \times NV \) data matrix? Do we move from matrix to tensor methods? While not altering the data we have, the answers to these questions have an impact on the types of models it would be advisable to fit. This chapter focusses on the impact the modelling assumptions of conventional ICA pipelines could have, in particular with particular respect to dimensionality estimation.
Recall that for most fMRI studies we generally have more voxels per image than images $T \ll V$. Hence we cannot hope to estimate a fully parametrised $V \times V$ spatial covariance matrix $\Sigma_V$ with accuracy, as would be required for a ‘temporal’ ICA model. Even supposing we had a sufficient dataset for the task, given that in neuroimaging $V \gg 10^5$, the subsequent numerical eigendecomposition of $\Sigma_V$ would currently be infeasible due to memory restrictions and very slow to run as an $O(V^3)$ operation. This essentially rules out a conventional approach to temporal ICA. Although recent advances in computational methods (for instance, Smith et al. (2014)) could eventually lead to temporal ICA becoming more widely adopted. We therefore turn primarily to consideration of $\Sigma_T$ spatial ICA as discussed in sections 2.5 and 2.8.

### 3.3 Effect of model ‘transposition’ misspecification.

Suppose we make the spatial ICA modelling assumption and treat our data as $V$ observations of length $T$ time-series (or, perhaps, in a group analysis $NV$ observations of length $H$ time-series). What would we expect to happen if our assumption was wrong, and the data are actually $T$ observations of a $V$ dimensional spatial vector? That is suppose we have data:

$$Y = WX + E$$  \hfill (3.1)

Where we suppose that: $W$ is a $V \times K$ fixed orthonormal matrix that functions as a mixing matrix (or as the basis vectors of a principal components model, or latent features in the language of machine learning); $X$ is a $K \times T$ matrix of IID Gaussians representing our components or ‘sources’, $X_{i,j} \sim \mathcal{N}(0, \phi^2)$; and $E$ is a $T \times V$ matrix of IID Gaussians representing observation noise, $E_{i,j} \sim \mathcal{N}(0, \sigma^2)$.

We propose a simple analysis of this issue through consideration of the expectation of the $T \times T$ empirical covariance matrix, $\Sigma_T$,

$$\mathbb{E}[\Sigma_T] = \frac{1}{V-1} \mathbb{E}[(WX + E)^T(WX + E)]$$

$$= \frac{1}{V-1} \mathbb{E}[X^T W^T W X] + \frac{1}{V-1} \mathbb{E}[E^T E]$$

$$= \frac{K}{V-1} \phi^2 I_T + \frac{V}{V-1} \sigma^2 I_T$$  \hfill (3.2)
3.3 Effect of model ‘transposition’ misspecification.

This should be sufficient to point towards an issue. Namely that for $K < T \ll V$, and unless $\phi^2 \gg \sigma^2$, we could reasonably expect the signal in our empirical covariance matrix to be swamped by observation noise. In particular the proportion of the variance accounted for by our principal components does not scale well with increases in spatial resolution for a given level of observation noise.

For instance assume that the signal in one of the principal directions is 10 times stronger than the noise $\phi^2 = 10\sigma^2$, and we have $10^3$ times more spatial observations than temporal $V = 1000T$. Then the variance accounted for by the latent structure will be roughly on the order of 1% of total variance. To see this the contribution of the latent structure is $(K\phi^2)/(V - 1) \approx 10T\sigma^2/1000T = \sigma^2/100$ while the contribution of the observation noise is $V\sigma^2/(V - 1) \approx 1000T\sigma^2/1000T = \sigma^2$.

Looked at another way, if we think about the signal to noise ratio $SNR = \phi^2/\sigma^2$ then in order for the latent structure to account for about 50% of the variability in the covariance matrix we would have to assume, when $V = 1000T$, that $SNR \approx 1000$.

Directions of extension.

This way of looking at the issue is quite general as it enables a number of extensions. If component variances are made unequal but bounded, i.e. $\text{Var}(X_{i,j}) = \phi_j^2$ we simply replace $K\phi^2$ with $\sum_{j=1}^{K} \phi_j^2$ in the above, and this makes no real difference to the conclusions as we could swap the sum out for a bound with $K$ times the maximum $\phi_j$, or we could think about the average $\phi_j$.

If the components are made IID non-Gaussian with a fixed variance, the expectation still holds. Of course, this breaks the probabilistic-PCA model. This does not matter for classical PCA, but does matter when using probabilistic variants to do automatic dimensionality selection.

If orthonormality of columns of $W$ is relaxed then $W^TW \neq I_K$ and the analysis is more complicated. However, if $W$ is a stochastic matrix, independent of $X$, with independent columns consisting of zero mean random vectors of unit length, a similar result holds. To see this write $V = WX$, then,

$$\Sigma_{i,j} = \mathbb{E} [V^TV]_{i,j} = \sum_{k=1}^{K} \sum_{l=1}^{K} \mathbb{E} [X_{k,j}X_{l,j}] \mathbb{E} [W'_{k}W_{l}] .$$

(3.3)
Now if $i \neq j$ we see $E[\Sigma_{i,j}] = 0$ due to the zero mean and independence assumptions on the components $X_{r,s}$, whereas if $i = j$ we have:

$$E[\Sigma_{i,j}] = \sum_{k=1}^{K} \phi^2 ||W_{.,k}||^2 = K\phi^2,$$

(3.4)

at which point we are back to the situation faced in equation 3.2.

### 3.3.1 Contrast with proposed vectors in spatial orientation.

Further to this we can observe that this problem is to some extent avoidable. Consider examining the expected signal in the direction of a unit vector $v$ in the column space of a fixed, orthonormal basis $W$,

$$v^T \Sigma_v v = \frac{1}{T} E[v^T YY^Tv]$$

$$= \frac{1}{T} E[v^T WXX^TW^Tv] + \frac{1}{T} E[v^T EE^Tv]$$

$$= \phi^2 + \sigma^2$$

Hence, even if the signal-to-noise ratio, $\phi^2/\sigma^2$, is modest, if given a proposed basis vector in $v \in \mathbb{R}^V$ we should be able to tell whether it accounts for much signal. Indeed we can get the same value by taking the dot product with an individual observation:

$$E [(v^T y)^2] = E [(v^T Wx + v^T \epsilon)^2] = \phi^2 + \sigma^2$$

We note this here as a motivation for the later use and development of tools that do not require eigendecomposition of empirical covariance matrices, and instead operate mainly through taking dot products with individual images. Such procedures naturally fall out of both Gibbs sampling, see chapter 5, and Variational Bayes methods, see chapter 8, for matrix factorisation.

### 3.3.2 Effect of low-sample size on estimation of spatial eigenvector.

Figure 3.1 visualises the effect of low sample size on estimated spatial eigenvectors when the observed dimension exceeds the number of observations. In both images 10 latent features were chosen from the basis vectors of the Inverse Discrete Cosine Transform. An overall signal-to-noise ratio of around 100-to-1 was used to generate a dataset. The singular value decomposition was then used to estimate the population eigenvectors.
3.3 Effect of model ‘transposition’ misspecification.

![Leading spatial eigenvector when nobs=10000, dims=10000](image1)
![Leading spatial eigenvector when nobs=100, dims=10000](image2)

(a) Matched dimension and sample size.  
(b) High dimension, low sample size.

Fig. 3.1 In both images the left hand panel shows the true primary eigenvector and the right hand panel the estimated eigenvector. Signal to noise ratio $\alpha = 100$.

We can see in Figure 3.1 that when the number of observed dimensions matches the number of observations we can recover a reasonable estimate of the true principal eigenvector. However when the number of observed dimensions is an order of magnitude larger than the number of observations we fail to recover the principal eigenvector.

Despite this example being framed in terms of the ‘spatial’ eigenvectors, the results would impact any analysis of the smaller ‘temporal’ empirical covariance matrix due to the relationship $Y^T u = v$ between the respective eigenvectors. Where the conditions for this problem arise they would imply that PCA on the $T \times T$ covariance matrix was equivalent to projecting the data onto noise vectors.

The reader might, at this point, wonder why PCA still often leads to useful results in such regimes. Though beyond the scope of our discussion here, we might speculate that results from compressed sensing are relevant. In particular, it is often possible to obtain a good reconstruction of data from statistics obtained by projection of data on to random vectors (Candès and Wakin, 2008).

### 3.3.3 Related work on the high-dimension, low sample size regime

A notable work related to loss of eigenvector fidelity in small samples was that of Nadler (2008), which focused on the distribution of the first principal component in finite samples with high dimension. It was shown that when the dimension of the observations exceeds the number of observations the principal eigenvector of the empirical covariance matrix will no longer track the true principal component vector. This is particularly interesting for our purposes because similar undertakings have focused on obtaining asymptotic results; infinite dimension or infinite dimension and infinite sample size at a fixed ratio. Our concern is
Failure modes for eigendecomposition methods

foremost on appropriate methodology for applied data analysis, and therefore doubly finite, $T$ and $V$, sample results provide a better guide for consideration of whether the $V > T$ problem is likely to be an issue in particular applications of PCA, including neuroimaging.

While Nadler (2008) discussed PCA on the $V \times V$ covariance matrix, thus result has implications for PCA based on the $T \times T$ covariance matrix obtained by ‘transposing’ the observations. This is again because of the relationship $Yu = v$ between the eigenvectors of the two potential covariance matrices. The implication being that if with high probability $v$ fails to track the true spatial eigenvector, i.e. the estimated eigenvector is noise, then the related empirical temporal eigenvector will be noise also. For neuroimaging, this means that if the most appropriate model for brain images is a linear combination of spatial vectors we should be careful in attempting to use the eigenvalues of empirical covariance matrices.

An alternative approach to the high dimension, low sample size problem might potentially present itself through the work of Ledoit and Wolf (2004). In that paper the authors pursue a method for estimating the true covariance matrix using a linear combination of the empirical $V \times V$ covariance matrix with the identity matrix. In particular they provide results for choosing an optimal weighting of the two. The result is essentially a regularised empirical covariance matrix. However, even if this estimator is accurate, for an applied neuroimaging problem this leaves us with the task of solving a very large, dense numerical eigenvalue problem.

As commented on in section 2.8 it is common for PCA dimension to be estimated based on the images of a single subject from a dataset. This is again a procedure which further reduces the sample size used for PCA, and would therefore similarly exacerbate any problems caused by low sample size.

3.3.4 An objection to the high dimension, low sample size critique

It might be objected to the above critique, particularly that of section 3.3, that we made the assumption of increasing spatial resolution for a given level of observation noise $\sigma^2$. It could be suggested that in reality a higher resolution observation would have a lower amount of observation noise in each entry. For instance, and thinking about neuroimaging again, if observation noise is proportional to voxel volume, if we reduce voxel size then our observation noise will be smaller. However, such an objection would itself make a strong assumption about the scalability of the precision of MRI scanners.

Such an objection would also neglect the role of structured noise such as small movements. For instance suppose a subject’s head shifts 0.01mm to the right, if we use 1mm edged voxels then in each voxel 1% of the material measured is swapped out, but with 0.1mm edged voxels then 10% of the material is swapped out. If it is not clear how this would generate variance,
imagine scanning a fake ‘brain’ composed of a fixed, chess-board style pattern of high and low ‘activation’. Now imagine the ‘brain’ oscillating left and right, from time point to time point this would affect the proportion of high and low ‘activation’ within each measured voxel - hence the time series through each voxel would show variance, despite the pattern of ‘brain’ activation being fixed. Due to the relative volumes involved, this effect would be more pronounced if the voxel sizes were reduced.

3.4 Dimension estimation in the high dimension, low sample size regime

The preceding sections have discussed issues connected to estimating eigenvalues from covariance matrices. While these observations are clearly relevant to PCA, they do not directly answer one of the key questions of this thesis: what is the appropriate dimension for matrix factorisation of neuroimaging data? We now consider the application of popular dimensionality estimation formulae to data generated under the conditions of interest.

In the following we simulate data $Y = WLZ + E$ where: $W$ is a $V \times K$ matrix with unit length columns which are orthogonal (orthogonal in expectation if random); $L$ is a $K \times K$ diagonal matrix representing signal strengths. In particular the sum of the main diagonal of $L$ will be normalised to $\alpha + K$ so that we can control the overall amount of signal by varying $\alpha$, while entries of $L$ are strictly greater than unity so that the signal in any structural direction is stronger than the noise (hence the $+K$ in normalisation); $Z$ is a $K \times T$ matrix such that $Z_{i,j} \sim N(0,1)$; and $E$ is a $V \times T$ matrix such that $E_{i,j} \sim N(0,1)$.

We can use different bases and signal strength profiles to see how popular dimensionality estimation formulae cope with different values of $V$ and $T$. The two formulae we will use were discussed in section 2.6 namely the Laplace approximation to the model evidence for a Bayesian PCA model of Minka (2000) which we will refer to this as LAP; and the Minimum Description Length based criterion for model order selection in what is essentially a PCA model of Wax and Kailath (1985) which we will refer to as MDL.

We call $l = \text{diag}(L)$ the signal decay curve, since $L^2$ represents the eigenvalues of the covariance matrix of the latent structural terms (i.e. of Cov($WLZ_{i,j}$)), at least when the basis is fixed. We call $l$ a decay curve because eigenvalues can always be ordered and therefore decay in some pattern. The eigenspectrum of the observed data $Y$ also contains noise from $E$. The interplay between the signal decay curve and the spectrum of the noise in the observed eigenspectrum will determine our dimensionality estimates.
3.4.1 High dimension low sample size simulation results

The following present results found when applying the MDL and LAP dimension estimation scoring formulae in the high dimension, low sample-size regime. This means potentially misapplying the methods when ‘transposing’ the roles of observed dimension and number of observations. We used the model defined by equation 3.1 to simulate data for varying dimensionalities $V$ when the true latent dimension was $K = 10$. For each setting of $V$ we performed 30 replications. Figure 3.3 shows the mean estimated latent dimension together at each setting, with the standard deviation in that estimate (which provides some notion of variability of the estimate but can provide values which are not interpretable).
3.5 Impact of signal to noise ratio

In the above analysis the quantity $\alpha$ represents the amount of signal attached to the latent basis and thus determining the signal-to-noise ratio (given standard normal noise). While Figure 3.2b showed the differences different signal decay patterns have on empirical eigenvalues, we now consider how those characteristically different eigenspectra impact on the estimated dimensionality under different settings of $\alpha$.

Furthermore, in probabilistic-PCA models such as those of Tipping and Bishop (1999) and Wax and Kailath (1985) the latent sources are assumed to be independent and identically distributed. As PCA underlies the popular ‘spatial’ ICA methods, we consider what would
happen with respect to dimensionality estimation if the voxel wise time series were spatially correlated. In this context the number of observations is $V$. If these are not independent but correlated, then $V$ is an over-estimate of the effective sample size. We would therefore expect this to interfere with the dimension estimation scoring formulae described above, since the score is a function of sample size.

Fig. 3.4 Behaviour of dimensionality estimation methods on simulated data when $V/T = 2.5$, $T = 1000$ for varying levels of signal strength $\alpha$. In both cases the true latent dimension is 10. Dotted lines show 25% and 75% percentile estimates.

Fig 3.4 shows the effect of varying the overall signal strength $\alpha$ under different assumptions about the latent feature type and the decay shape of the latent signals. In both cases shows we observe that the dimensionality estimation formulae begin to offer useful estimates when $\alpha > 25$ and offer better estimates by $\alpha = 100$. For $\alpha = 100$ the smallest $\phi_j$ for the linear decay curve is 2.82 and for our exponential decay curve is 7.12. This explains why, in this setting, the estimated dimension converges to the truth more quickly under the exponential than under linear decay.

In our observation from simulations, and when underlying ‘spatial’ basis vectors are orthogonal (at least in expectation), the shape of the signal decay curve matters more than the nature of the spatial auto-correlation. We show this in Figure 3.5 with the uniform signal decay curve and four different types of basis vectors, identity and IID bases are shown in a) and b) respectively and are nearly visually indistinguishable from spatially correlated bases drawn from a 2 dimensional autoregressive process or the inverse discrete cosine transform respectively.
3.6 Point estimates of dimensionality

So far we have covered how existing methods choose assumptions about the orientation of data, the construction of covariance matrices, the choice of eigenvalues and the selection of a latent dimension. We have not yet commented on the use that inferred latent dimensionality is put to. What should be clear at this stage is that there is some uncertainty associated with the latent dimensionality, caused variously by the signal-to-noise ratio of the data, the relative size of the number of observations and the observed dimension, and the shape of signal decay across latent dimensions. While not conclusive with respect to any particular application, the existence of these potential pitfalls at least introduces doubt that the $K$ obtained by optimising a criterion is inevitably the truth (or very close to it).

What happens then with our uncertainty about the latent dimensionality? For example, we may find that the dimensionality scoring curves produced by the LAP or MDL methods are flat around the optimum value. An illustration of this phenomenon is shown in Figure 3.6 using a subset of 30 participants from the Cam-CAN study (Shafto et al., 2014). We see that the Laplace approximation score curve is flat around its maximum.

Under current methods popular in neuroimaging (see section 2.8), uncertainty about the latent dimensionality is not propagated to later stages of the statistical analysis. An immediate consequence of this is felt during dimension reduction. In very broad terms: if our estimated value of $K$ is too small then downstream analyses will underfit the data, as we
Failure modes for eigendecomposition methods

will have given up too much information from the data; going the other way, if we estimated $K$ to be too large, we will overfit the data with downstream analyses fitting models to noise.

A more distant consequence occurs when, for example, statistical parametric maps of quantities associated with a particular ICA analysis are presented, those statistics are conditional on a fixed dimension analysis, when the dimensionality itself is uncertain.

The question then, is why we do not account for uncertainty in the selection of dimensionality in applied neuroimaging research? It seems the answer must be practicality. Consider two tactics that could be used to account for uncertainty in the latent dimensionality: take bootstrap samples of the data and run the whole analysis pipeline on each; or using the LAP approximation for model evidence to average over different settings of $K$.

Either strategy would involve running analyses many times. If our downstream analysis involved any computationally intensive tools, and for instance we tested $K = 1,...,100$, this would scale up the overall analysis time by a large multiple unless we had sufficient resources to run all analyses in parallel. As it is very common for fMRI software to require on the scale of days to complete a run, this would raise serious practical concerns. This is compounded when we consider the possibility in exploratory work of a variety of preprocessing settings; such as smoothing levels, high-pass filtering, and different varieties of motion correction.

Another issue that would be confronted is the presentation and combination of different models. For instance if we found bases $W^{(1)}$ and $W^{(2)}$ under dimensionality settings $K_1, K_2$ how do we match up a particular feature to show the effect that a change in the dimensionality has on it? Should we even expect the ‘same’ feature to be present in both models? This is a theoretically non-trivial question that would confront us even if raw computation were no longer an issue. Furthermore strategies that might work for, say, one dimensional time-series data, such as plotting matched functions on one graph would not work well for spatio-temporal fMRI data due to the challenges of visualising high dimensional quantities.
3.7 Two stage analysis: must the PCA solution contain the ICA solution?

In section 3.6 we discussed the potential impact that uncertainty around the appropriate setting of latent dimensionality might have on downstream analyses. Here we go into some of the issues that might come into play when Independent Components Analysis is used in a pipeline following Principal Components Analysis. We should therefore ask ourselves two questions: if we are interested in finding a rank $K$ basis for the data against which the components of the data are maximally independent why is it safe to assume that is coincident with the subspace spanned by the $K$ principal eigenvectors?

As mentioned in sections 2.3.3 and 2.5, a typical two-stage ICA pipeline first uses PCA to reduce the dimension of the data. This results in the ICA stage of the pipeline receiving the projection of the data onto the first $K$ basis vectors of the covariance matrix. The ICA model then finds a further transformation of the data from $\mathbb{R}^K$ to $\mathbb{R}^K$. In particular ICA provides a matrix factorisation of the data fed into it. So that the mixing matrix in the case of spatial ICA represents a rotation of the first $K$ eigenvectors of the $T \times T$ empirical covariance matrix of the data. We now set aside the idea that these eigenvectors could be estimated poorly in small samples or the dimension misspecified by autocorrelation and instead concentrate on the recovery of the ICA solution.

3.7.1 Perils of two-stage analysis

A counter example casts doubt on any hope that the ICA solution is guaranteed to be contained in the PCA subspace. Consider two random variables: a Gaussian, $x \sim \mathcal{N}(0, 1)$ and a Laplace $y \sim \mathcal{L}(0, 1/2)$ and define a random vector:

$$ z = \begin{bmatrix} x^2 \\ x \\ y \end{bmatrix} $$

(3.5)

Where we recognise that the $x^2$ entry will have a $\chi^2$ distribution, thus the independent components should be retrievable by some form of ICA since there is only one Gaussian entry. We see that $z_1$ is perfectly dependent on $z_2$, while $z_2$ contains more information that $z_1$ since the sign cannot be recovered. In any case, we would definitely want $z_3$ to be in the subspace in which we search for an ICA solution. However, the population covariance matrix...
Failure modes for eigendecomposition methods

is:

\[
\text{Cov}(z, z) = \begin{bmatrix}
2 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & \frac{1}{2}
\end{bmatrix}
\] (3.6)

As the eigenvalues are exposed on the main diagonal it is clear that if we performed PCA with \( K = 2 \) on the population covariance matrix we would entirely neglect \( z_3 \). So any downstream analysis would not be able to recover the true solution. We would have to hope, with a given finite sample, that the off diagonal entries of a data covariance matrix would be nonzero, that the subsequent numerical eigendecomposition would include a small loading on \( z_3 \) and that this would be enough for an ICA algorithm to work on. This is to say, we would have to hope that our PCA was ‘wrong’.

Of course, since the probability mass of any single real number under either the Laplace or Gaussian distribution is zero, empirical eigenvectors under this distribution should always have at least a small weight on \( z_3 \). Can ICA recover the solution when the eigenvectors chosen by PCA only have very small loadings on one of the true independent variables? To check this in our small example we simulated 1000 draws from the distribution above, and used a FastICA fitting algorithm (which is itself randomised) with \( K = 2 \). Recall this produces a factorisation \( Y \approx W^{\text{ICA}} X^{\text{ICA}} \) where \( W^{\text{ICA}} \) is \((3 \times 2)\) and \( X^{\text{ICA}} \) is \((2, 1000)\). Where \( M \) is the unmixing matrix produced by FastICA we have that \( W^{\text{ICA}} = W^{\text{PCA}} M^{-1} \) and \( X^{\text{ICA}} = M X^{\text{PCA}} \).

\[
\text{mean}( (W^{\text{ICA}})_{i,j} ) = \begin{bmatrix}
-0.27 & 0.956 \\
-0.374 & -0.028 \\
0.006 & 0.003
\end{bmatrix} \quad \text{var}( (W^{\text{ICA}})_{i,j} ) = \begin{bmatrix}
4.26 & 11.938 \\
0.847 & 0.138 \\
0.003 & 0.006
\end{bmatrix}
\] (3.7)

We can see from the information in equation 3.7 that in simulations the combined PCA/ICA procedure effectively drops loading on \( z_3 \), as we might expect from the theoretical analysis. The root of this problem is that PCA and ICA have different objective functions. PCA seeks the orthogonal rank \( K \) basis on which the components of the data have the greatest variance. ICA instead seeks to maximise the independence of the components, or, if this sounds too circular, the mutual information between the input data and the components Bell and Sejnowski (1995). However, ICA cannot recover solutions which are dropped from the search space at the PCA stage.

These objectives do partially share a common requirement. PCA requires lack of correlation between the components and this is achieved through the selection of orthogonal basis vectors. As independence implies lack of correlation, this fulfils a requirement of ICA that the components be independent. However the converse is not true, lack of correlation
does not imply independence. The covariance matrix in equation 3.6 shows an example of this in the relationship between \( x \) and \( x^2 \), which enjoy zero covariance (their covariance in fact being the skewness of a symmetric distribution) but where the second term is perfectly dependent on the first.

In terms of neuroimaging, this issue could lead to skipping of independent structural components which have a low variance; particularly where the selected dimension was too low.

### 3.8 Orthogonality and spatial ICA

Achieving independence between components is the optimisation criterion of the ICA model. In turn this implies that the inferred independent components, living in the rows of \( X^{ICA} \), should be uncorrelated in expectation. To see this consider the expectation of the dot product between the \( j^{th} \) and \( k^{th} \) components which when \( j \neq k \) we should have under an ICA model to be 
\[
E[\sum_{v=1}^{V} X_{j,v}^{ICA} X_{k,v}^{ICA}] = \sum_{v=1}^{V} E[X_{j,v}^{ICA} X_{k,v}^{ICA}] = 0 \text{ (assuming zero component means)}.
\]
This has a substantial implication for the interpretation of results in the field of neuroimaging. It implies that we should expect the spatial maps, often termed ‘networks’, produced by ICA to be pushed towards orthogonality.

Most applied work in neuroimaging uses such a ‘spatial’ ICA method. As noted by Beckmann et al. (2005), we might face a problem where spurious relationships were introduced in results in order to satisfy orthogonality constraints. We differ from the authors of that paper in believing this to be a potentially serious issue the interpretation of spatial maps, as shown in the following example.
Failure modes for eigendecomposition methods

Fig. 3.7 Analysis of ICA performance in a similar experimental setup to (Beckmann et al., 2005). 3.7a) shows the two latent spatial maps, which overlap in 50% of their active area (squares offset from centres). 3.7b) the panels show the true latent time-courses with correlation 0.05, and the bottom two the inferred time courses. In 3.7d) the latent time courses were chosen to have correlation 0.2. Panels 3.7c) and 3.7e) show the corresponding inferred spatial maps for the time courses in 3.7b) and 3.7d) with correlations 0.05 and 0.2 respectively. We can see that in 3.7e) that ‘spatial’ ICA did not recover the true spatial maps when the correlation level was higher.

We replicated an example from Beckmann et al. (2005) (their Figure 2) as far as possible from the information provided. The example aimed to demonstrate that the probabilistic orthogonality constraint would not lead to distortion of spatial signals in overlapping areas. As shown in Figure 3.7 using sine wave time courses similar to those used in that paper we
could replicate low distortion. However, this was a result of the sine waves used being nearly orthogonal with 0.05 correlation. In contrast, in Figure 3.7d the courses are again sine waves but with correlation 0.2. This level of correlation induced a distortion in the spatial maps, in contrast to Figure 3.7c which recovered two squares Figure 3.7e shows the inference of a rectangle with a stronger strip in the middle (bottom) and a contrast which could delete one of the weak strips and bolster the other.

![Fig. 3.8 Two further examples of correlated time courses which produce distorted results.](image)

We found that correlated Gaussians would often result in distortion when correlation was above 0.2 and almost always when correlation was above 0.3. But ICA assumes non-Gaussianity so its relevance as a test-case for ICA was questionable. However, we also exponentiated versions of Gaussian AR processes (e.g. correlation 0.23) and this produced
distortion (exponentiation renders the distribution non-Gaussian), as did correlated Bernoulli random variables (e.g. again with correlation 0.23).

Given this simple counter-example, the arguments put forward by Beckmann et al. (2005) cannot be relied upon in general when interpreting spatial maps from an ICA model. In general, where the true latent spatial maps are correlated ($E[X_i, X^T_j] \neq 0$), we should expect that a model pursuing an objective that requires orthogonality of the time-series basis and rewards stochastic orthogonality of spatial maps may well result in distortion of overlapping regions in the inferred spatial maps. The extent to which this is the case will depend on the level of correlation in the corresponding time series.

This example is also particularly simple in that only two latent spatial maps are present. In a real neuroimaging study, where a typical reported value of latent dimensionality might be $K = 50$, we may have to consider that a particular spatial map might overlap a small amount with each of the others while being only slightly correlated with each of the other time-courses. In which case the distortion may be subtle, for example zeroing out some regions in some spatial maps and splitting a single ‘network’ across several spatial maps elsewhere.

### 3.9 Summary

In this chapter we have covered several issues that may affect application of combined PCA/ICA methodologies to neuroimaging data. We considered the extent to which the empirical $T \times T$ covariance matrix could be trusted when the correct interpretation of the data was as spatial vectors and $T \ll V$. The impact of this on the estimation of dimensionality using eigenvalue based dimension selection criteria was considered. We saw that when the assumptions of the criteria were violated, the estimated dimension was not reliable. When the assumptions were valid we saw that the signal-to-noise ratio had a strong effect on the inferred dimensionality, with a low ratio corresponding to underestimates of dimensionality. The use of point estimates of dimensionality was discussed in terms of its downstream impact on reporting of statistics such as p-value maps, although the computational and reporting difficulties of undertaking multiple analyses was acknowledged.

Two-stage PCA/ICA methodologies and saw that these could drop independent components with low variability in favour of redundant, high variance components. The impact of stochastic orthogonality constraints was discussed. We demonstrated that in the case of spatial ICA these constraints could lead to distortion of true spatial maps in areas of overlapping activity, leading to a potential for misinterpretation.
These considerations lead us to seek methods which: do not require eigendecomposition of empirical covariance matrices; have a consistent objective function across stages of factorisation; can account for multiple possible settings of proposed dimensionality; and relax orthogonality constraints to avoid distortion.
Chapter 4

Efficient models of structured bases

4.1 Introduction

This chapter turns to the issue of modelling spatial relationships in high dimensional vectors. Specifically, we set out a framework for efficient handling of high dimensional Gaussians that is suited for incorporation into matrix factorisation models. This is accomplished by algebraic consideration of precision matrices with known eigenvectors. The approach allows the extension of models to multiple dimensions and the composition of models. The stage is set for inference of model parameters using gradient methods in later chapters. We present an extended example of a model for a smooth random vector designed, which in later chapters will be used to reduce the need for spatial smoothing of neuroimaging data during pre-processing. Generally the model is seen to enforce $L^2$ sparsity of the components of vector with respect to an alternate basis.

4.1.1 Structured Features

In neuroimaging we might wish to model a voxel in the brain as being likely to take on a value similar to its neighbours. This would correspond to the intuitive idea that if brains contain regions which behave as functional units and the grid of voxels is finer than the anatomical regions, we should see some positive correlation between voxel values. Moreover as the spatial resolution of our imaging increases we should expect spatial relationships to increase in importance. In addition to these \textit{a priori} considerations, recent works highlight the importance of spatial autocorrelation to the statistical analysis of fMRI data, whether in the regression setting where Eklund et al. (2016) identified the mishandling of spatial correlation as a cause for inflated false positive rates, or in Griffanti et al. (2017) which commented on the sensitivity of ICA results to pre-smoothing choices.
Matrix factorisation models typically do not capture correlation structure within latent basis vectors \( \mathbf{w} \), setting aside vector normalisation. For example in classical PCA (Mardia et al., 1979), we do not expect the \( i^{th} \) entry of a basis vector \( \mathbf{w}_i \) to have a similar value to the \( (i+1)^{th} \) entry. This is replicated in the Bayesian PCA of Tipping and Bishop (1999). Similarly, treatments such as the sparse PCA of Zou et al. (2006) modify the distributions of the latent variables to induce sparsity in one of the factors but without assuming that the nonzero coefficients follow a pattern (e.g. nonzeros come in blocks). Popular matrix factorisation techniques in recommender systems Koren et al. (2009) are similar. However for certain classes of data, e.g. locations in space, points in time, clusters of people, we may expect structure within our latent features, perhaps in the form of smooth landscapes, cyclic time-courses, or group-wise random effects.

### 4.1.2 Computational considerations

Modelling spatial correlation structure for latent vectors could be prohibitive for high dimensional data, as it implies handling at least one \( V \times V \) nondiagonal covariance matrix. In fMRI neuroimaging \( V \) can be on the order of \( 10^5 \) or \( 10^6 \). While, for instance, evaluating the density of a multivariate normal requires the determinant of a covariance or precision matrix. This is operation is \( O(V^3) \) under a general approach using the Cholesky decomposition. Approaches such as low rank approximations via eigendecomposition are likely to cause difficulties for accurate evaluation of density functions as these require the determinant, which depends on information excluded from the approximation.

If we knew the eigendecomposition of a positive definite precision matrix \( \mathbf{\Delta} = \mathbf{U}\mathbf{H}\mathbf{U}^T \), then we could evaluate its determinant in \( O(V) \) time as \( \ln |\mathbf{\Delta}| = \sum_{i=1}^V \ln H_i \). However, numerical difficulties would persist as the \( V \times V \) matrix \( \mathbf{U} \) may not fit into the memory of a computer, and, if it did, the required matrix-vector operations would still be \( O(V^2) \). A solution to this is the use of fast linear operators which encoded the actions of \( \mathbf{U} \) and \( \mathbf{U}^T \). So that if \( \mathbf{U}\mathbf{w} \) could be obtained in \( O(t(V)) \) time for some \( t(V) \ll V^2 \), we could perform useful tasks such as evaluating the density and sampling from the distribution in \( O(t(V)) \) time.

A somewhat similar approach arose in the form of circulant embedding (Dietrich and Newsam, 1997). This solved the problem of sampling from a Gaussian whose correlation matrix was circulant. Copies of this correlation matrix are embedded as blocks within an appropriately constructed circulant matrix. As a circulant matrix the eigendecomposition is known, with the Fourier Transform supplying the eigenvectors and the model the eigenvalues. This technique has the advantage of being adaptable to any appropriate circulant model. However this was an approach to sampling, while we also desire efficient evaluation of a density and its gradient. Furthermore the approach requires selection of the number of
blocks for the embedding, which was dependent on the length scale; as we aim to infer such parameters and require gradient information this is inconvenient.

The circulant embedding method was again taken up for inference of Gaussians in Rue and Held (2005). That work emphasised the interpretation of the precision matrix within the topic of Gaussian Markov Random Fields (GMRFs). The circulant method provided an approach to the special set of GMRFs with circulant correlation matrices. We next consider GMRFs more broadly before continuing on to develop a framework for tractable modelling of high dimensional Gaussians. We will later reunite the two topics with an extended example of an further interpretable GMRF with a special eigendecomposition.

4.1.3 Graphs and Gaussian Markov Random Fields

GMRFs provide a helpful framework for modelling of multivariate normal distributions (Rue and Held, 2005). A GMRF is a multivariate Gaussian some of whose entries are independent of each other, conditional on the rest of the vector. The zeros of the precision matrix encode conditional independence properties. This implies that if \( \mathbf{w} \sim \mathcal{N}(0, \Delta^{-1}) \) is a GMRF, then \( \Delta \) has some level of sparsity. When the model contains many conditionally independent variables, the precision matrix becomes sparse leading to some strategies for computational tractability. GMRFs have been used, for example, to model spatial distributions in epidemiology (Papageorgiou et al., 2014) and in climate science (Zammit-Mangion et al., 2015).

Suppose we wish to model an undirected graph with \( J \) vertices \( \mathcal{G} \) and edge set \( \mathcal{E} \). We say that vertices \( i \) and \( j \) are neighbours, written \( i \sim j \), if and only if \( (i, j) \in \mathcal{E} \) (where \( (i, j) \) is an unordered pair). We let \( \nu_i \) represent the number of neighbours of the \( i \)th node. This neighbourhood information allows the definition of a \( J \times J \) matrix known as the Laplacian, \( \mathbf{L} \), by setting:

\[
\mathbf{L}_{i,j} = \begin{cases} 
\nu_i & i = j \\
-1 & i \sim j \\
0 & \text{otherwise}
\end{cases}
\]

Due to the presence of one linear constraint, namely that \( L_{i,i} = -\sum_{j \neq i} L_{i,j} \), zero is always an eigenvalue of the graph’s Laplacian matrix. As such, the Laplacian is somewhat unsuitable for use as a precision matrix directly. Instead, some constant multiple of the identity matrix may be added in order to ensure invertibility. Let our precision matrix be given by \( \Delta = \theta_1 \mathbf{I} + \theta_2 \mathbf{L} \). This can be used to model draws from the graph as \( \mathbf{x} \sim \mathcal{N}(0, \Delta^{-1}) \).
Since $\Delta^{-1}$ is sparse, it is tempting to proceed with computation using sparse Cholesky decompositions. However, in high dimensions this approach does not always prove tractable or accurate. Rue and Held (2005) discuss how for circulant block matrices, analysis can proceed using an eigendecomposition that has the Discrete Fourier Transform as its eigenvectors. We find that by further formalising the modelling of multivariate Gaussians in terms of their eigendecomposition a range of models become possible.

4.2 Covariance structures

We now provide some formalism around covariance structures which all share the same known eigenvectors, $U$. The eigenvectors are important because if they are fixed in advance at certain values our computations can be made quicker.

For practical purposes our interest is in eigenvector matrices $U$ that be represented as fast linear operators. Where we would count as fast an operator able to perform matrix-vector multiplication with computational and storage complexities of less than $O(V^2)$. For instance the Fast Fourier Transform has a computational complexity of $O(V \ln V)$ and negligible storage complexity (Frigo and Johnson, 2005). However, the discussion that is to follow is agnostic with respect to how $U$ is represented or how fast we can execute its operation in practice.

Once the eigenvectors are fixed the free parameters of our covariance structures will be their eigenvalues. We would also like to provide various rules for the production of new covariance structures, and their extension to higher dimensions.

4.2.1 Definitions

Let the set of all positive definite matrices sharing orthonormal eigenvectors $U$ be called a covariance structure or precision structure and be denoted $G_U \subset \mathbb{R}^{N \times N}$. We can define a bijection between $G_U$ and $P_N$, the set of vectors in $\mathbb{R}^N$ containing all positive entries. Note this means that we will be ignoring degenerate covariance structures. To see this define the matrix function of a vector $g_U : P^N \rightarrow G_U$ such that

$$g_U(h) = UD(h)U^T,$$  \hfill (4.1)

which is diagonalised by $U$ and whose eigenvalues, $h$, are strictly positive by definition, hence $g_U(h) \in G_U$. The inverse function $g_U^{-1} : G_U \rightarrow P^N$ is given by

$$g_U^{-1}(Q) = \text{diag} \left( U^T QU \right),$$  \hfill (4.2)
4.2 Covariance structures

since $g_U^{-1}(g_U(h)) = \text{diag}(U^TUD(h)U^T) = \text{diag}(D(h)) = h$. Now because $Q \in G_U$ must have positive eigenvalues to be positive definite we see $g_U^{-1}(Q) \in P_N$.

4.2.2 Groups of covariance structures

The set $G_U$ forms an Abelian group with respect to the operation of matrix multiplication.

Identity element

Let $h = 1$ then $g_U(h) = UIU^T = I$.

Closure

Recall that we denote the elementwise product of two objects $a, b \in \mathbb{R}^M$ by $a \cdot b$.

Choose $Q = UD(h)U^T$, $R = UD(i)U^T \in G_U$. Then $QR = UD(h \cdot i)U^T = g_U(h \cdot i) \in G_U$.

Associativity

Choose $Q = UD(h)U^T$, $R = UD(i)U^T$, $S = UD(j)U^T \in G_U$. Then $[QR]S = UD(h \cdot i \cdot j)U^T = Q[RS]$.

Inverse

The inverse of $g_U(h)$ is $g_U(h^{-1})$ where, to be clear, $h^{-1}$ is the element-wise reciprocal. Then $g_U(h)g_U(h^{-1}) = UD(h \cdot h^{-1})U^T = UD(1)U^T = I$.

This property also means we can talk about ‘precision structures’ in the same way as ‘covariance structures’ since for a fixed set of eigenvectors they are members of the same set $G_U$.

Commutativity

Following from the closure property, choose $Q = g_U(h)$, $R = g_U(i) \in G_U$. Then $QR = UD(h \cdot i)U^T = RQ$.

4.2.3 Rules for construction of valid eigenvalues

It may be worthwhile making explicit some additional rules for constructing valid sequences of eigenvalues $h \in P_N$. While some of these may seem somewhat obvious, their enumeration here will allow us to simply use them later on.
Powers

As we have already seen, if \( h \in P_N \) then its elementwise reciprocal is also in \( P_N \). Similarly \( h^p \in P_N \) for some \( p \in \mathbb{R} \), although if \(|p|\) is very large then we will be inviting numerical issues.

Positive scalar multiplication

As we have already seen, if \( h \in P_N \) then for \( a \in \mathbb{R} \) we have that \( ah \in P_N \).

This, combined with the property that \( I \in G_U \) for all valid \( U \), leads usefully to the closure of the covariance structure under Bayesian updating with isotropic noise as we shall see in Chapters 5 and 8.

A side effect here is that unit normalised versions of covariance structures are valid, since if \( h \in P_N \) then \(|h|_2 > 0\) and so \( \frac{h}{||h||} \in P_N \).

Summation

If \( h, i \in P_N \) then \( h + i \in P_N \) also.

Parameterisation

Suppose we have a function \( h : \mathbb{R}^P \to P_N \) then we can parameterise the eigenvalues as \( h(\theta) \) for \( \theta \in \mathbb{R}^P \).

For clarity, when we use this facility we will often overload notation so that \( g_U(\theta) \) is understood to mean \( g_U(h(\theta)) \) for some valid function \( h \). Similarly, \( Q(\theta) \) is understood to resolve as \( Q = g_U(h(\theta)) \).

4.2.4 Parameterisation of the Multivariate Normal distribution

Let \( \Delta(\theta) \in G_U \) be a parameterised precision matrix given by,

\[
\Delta(\theta) = UD(h(\theta))U^T,
\]

where \( h \in P_V \) and is twice continuously differentiable. The first requirement ensures that \( \Delta \) is positive definite. The second that various gradient-based algorithms can operate with our model, in particular first derivatives are indispensable throughout this thesis, second derivatives are only required for the method of chapter 5. We will sometimes suppress the vector parameter \( \theta \) for notational convenience, i.e. \( \Delta = \Delta(\theta) \) and \( h = h(\theta) \).
4.2 Covariance structures

Density

The log density of the $N(\mathbf{0}, \mathbf{\Delta}^{-1})$ distribution can be evaluated as,

$$
\ln f(w|\theta) = c + \frac{1}{2} \sum_{v=1}^{V} \ln(h_i(\theta)) - \frac{1}{2} \sum_{i=1}^{V} h_i(\theta)(U^T w)^2,
$$

(4.3)

Since the determinant term resolves to a linear time function of the eigenvalues this is no longer the bottleneck for applied work. Instead the term for the exponent dominates the computation time through the computation of $U^T w$, which is $\mathcal{O}(V^2)$ in the worst case, hence our interest in $U$ that can be represented as fast operators.

Gradient

We also note that this formulation is differentiable with respect to the parameter $\theta$. We express the gradient of $h$ using the grad operator $\nabla$ whose operation on a $n-$vector valued function of $m$ parameters produces an $m \times n$ matrix whose $(i, j)$th entry is the partial derivative of the $j^{th}$ component of the vector with respect to the $i^{th}$ parameter (Magnus and Neudecker, 1988),

$$
\nabla_{\theta} \ln f(w|\theta) = \frac{1}{2} \sum_{i=1}^{V} \frac{1}{h_i(\theta)} \nabla_{\theta} h_i(\theta) - \frac{1}{2} \sum_{i=1}^{V} (U^T w)^2 \nabla_{\theta} h_i(\theta)
$$

$$
= \frac{1}{2} (\nabla_{\theta} h) (h^{-1} - (U^T w)^2)
$$

(4.4)

Which enables various optimisation and sampling procedures for its posterior distribution. Where in the second line we use the convention elementwise operations on vectors.

Hessian

The $(i, j)^{th}$ entry of the Hessian of the distribution with respect to the parameters of the precision matrix is given by,

$$
(\nabla_{\theta} \nabla_{\theta} \ln f(w|\theta))_{i,j} = \frac{1}{2} \sum_{i=1}^{V} \left( \frac{\partial h_i}{\partial \theta_i \theta_j} (h^{-1}_i - (U^T w)_i^2) - \frac{\partial h_i}{\partial \theta_i} \frac{\partial h_i}{\partial \theta_j} h^{-2}_i \right)
$$

(4.5)

Sampling

Parameterised in this fashion, the distribution may be sampled from as:

$$
w = UD(h(\theta))^{-1/2} z$$
When \( z \sim \mathcal{N}_V(0, I_V) \). Hence the time complexity of both density evaluation and sampling is determined by that of our linear operators. Again this will usually be dominated by the complexity of \( U \) (unless a peculiar choice is made for \( h(\theta) \)).

### 4.2.5 Dimension extensions of covariance structures

We may have a model for a one-dimensional spatial process and wish to extend it to two or more dimensions. The question arises how to go about this. There is an elegant way, which we call the graph extension; since for precision matrices related to graph Laplacians it is a restatement of a result in graph theory (Merris, 1994). We say related to because the addition of a matrix proportional to the identity is often required to ensure invertibility. However, even where the precision matrices have no (clear) interpretation as graph Laplacians we may still use the same operations to produce other valid precision matrices.

#### Graph extension via Kronecker sum

We can extend covariance matrices for two vector spaces to a covariance matrix on the tensor product as follows. Choose two sets of orthonormal eigenvectors \( U \in \mathbb{R}^{N \times N} \) and \( V \in \mathbb{R}^{M \times M} \) and define the operation, which we call the graph product, by \( \ast : (G_U, G_V) \to G_{U \otimes V} \) for \( Q = g_U(h) \) and \( R = g_V(i) \) by

\[
Q \ast R = (U \otimes V)(D(h) \otimes I_M + I_N \otimes D(i))(U \otimes V)^T = g_U \otimes V (D(h) \otimes I_M + I_N \otimes D(i)), \tag{4.6}
\]

We can see \( G_{U \otimes V} \) is a valid covariance structure since \( U \otimes V \) is orthonormal and all eigenvalues \( \text{diag}(D(h) \otimes I_M + I_N \otimes D(i)) \) are positive. This latter point serves to highlight a difference between our construction and the graph theoretic case, where a zero eigenvalue is ensured for the eigenvector 1.

The motivation for this definition and its graph theoretic origin is provided in the extended example of section 4.4.

#### Illustration: making second-order models

It may be worth illustrating an aspect of the graph extension that arises when trying to build richer models of spatial interaction. Suppose we wished to build a graphical-type model with relationships between second order neighbours.
First observe that \((Q \ast Q)^2 \neq Q^2 \ast Q^2\), even though both are members of \(G_{\otimes \otimes} U\) since for the former matrix:

\[
(Q \ast Q)^2 = (U \otimes V)(D(h) \otimes I_M + I_N \otimes D(i))(U \otimes V)^T(U \otimes V)(D(h) \otimes I_M + I_N \otimes D(i))(U \otimes V)^T
\]

\[
= (U \otimes V)(D(h) \otimes I_M + I_N \otimes D(i))(D(h) \otimes I_M + I_N \otimes D(i))(U \otimes V)^T
\]

\[
= g_{\otimes \otimes}(D(h^2) \otimes I_M + 2D(h) \otimes D(i) + I_N \otimes D(i^2))
\]

While for the latter matrix,

\[
Q^2 \ast Q^2 = (U \otimes V)(D(h^2) \otimes I_M + I_N \otimes D(i^2))(U \otimes V)^T
\]

\[
= g_{\otimes \otimes}(D(h^2) \otimes I_M + I_N \otimes D(i^2))
\]

So that the order of operations is important to modelling, since the additional term \(2D(h) \otimes D(i)\) in the eigenvalues of the former matrix will affect the strength of correlations between the spatial dimensions.

### 4.3 Relationship to other parameterisations

#### 4.3.1 Multiplicative dimension extension via Kronecker product

An alternative way to extend a model to multiple dimensions comes via the Kronecker product, as highlighted by Ren et al. (2013). In their formulation, if two vectors have correlation structures \(Q\) and \(R\) then the correlation structure of a separable model has correlation structure given by \(Q \otimes R\).

We can situate this approach within our formalism as follows. Begin by choosing members of known precision structures \(Q = g_U(h)\) and \(R = g_V(i)\). Now consider the eigendecomposition of the resulting precision structure,

\[
Q \otimes R = (UD(h)U^T) \otimes (VD(i)V^T)
\]

\[
= (U \otimes V)(D(h) \otimes D(i))(U \otimes V)^T.
\]

Which we recognise as a member of \(G_{U \otimes V}\). So that the eigenvalues are extended into the higher dimension in a multiplicative fashion rather than the additive fashion found in the graph extension.
4.4 Example: smooth features

4.4.1 The neighbourhood model

Consider a graph on one dimensional lattice such that each node $s_i$ is a neighbour of $s_{i+1}$ and $s_{i-1}$ wherever those indices are valid. That is, $s$ is a form of first order autoregressive process. Since $i \sim j$ if and only if $|i - j| = 1$ we have that $L$ is tridiagonal and thus sparse.

We can obtain an analytical expression for the eigendecomposition ($L = UDU^T$) of this Laplacian as the matrix encodes the recurrence requirements of the Discrete Cosine Transform (DCT). The analytical form of DCT-II is known (Strang (1999), Ahmed et al. (1974)) and therefore (with $j, k$ ranging over $0, \ldots, N-1$):

$$U_{jk} = \cos \left( (j + 1/2)k\pi/N \right) \quad (4.9)$$

$$\gamma_k = 2 - 2\cos(k\pi/N)$$

This can be integrated into our framework with $h_i(\theta) = \theta_1 + \theta_2 \gamma_i$ for fixed $\gamma_i, i = 1, \ldots, V$, giving:

$$Q(\theta) = \theta_1 I_V + \theta_2 U\text{diag} \{\gamma_i\} U^T$$

4.4.2 Extension to additional dimensions

While it is nice to have a model for smooth one dimensional features, our ultimate aim is to model neuroimages with three spatial dimensions. Here results from graph theory provide a solution for extension to higher dimensions.

A result stated in Merris (1994) is that the Laplacian matrix of the Cartesian product of two graphs is the Kronecker sum ($\oplus$) of those graphs’ Laplacian matrices. A useful implication for our present purposes is that if $L_1, L_2$ are two Laplacian matrices for models on the line of rank $N_1, N_2$ respectively with eigendecompositions $U_iD_iU_i^T$ then we can find the eigendecomposition of the Cartesian product graph Laplacian for a model of the plane as:

$$L_{(2D)} = L_1 \oplus L_2 \quad (4.10)$$

$$= (U_1 \otimes U_2)(D_1 \otimes I_{N_2} + I_{N_1} \otimes D_2)(U_1 \otimes U_2)^T$$

$$= U_{(2D)}D_{(2D)}U_{(2D)}^T$$

The Laplacian for a 3D model can then be produced by the Kronecker sum of a 2D model and a further 1D model, hence if $L_3$ functions for the extension to 3D as $L_2$ did for the extension.
to 2D we find that the eigendecomposition of the 3D case is:

$$L_{(3D)} = (L_1 \oplus L_2) \oplus L_3 = U_{(3D)}D_{(3D)}U_{(3D)}^T$$

Where

$$U_{(3D)} = U_1 \otimes U_2 \otimes U_3 \quad (4.11)$$
$$D_{(3D)} = D_1 \otimes I_{N_2} \otimes I_{N_3} + I_{N_1} \otimes D_2 \otimes I_{N_3} + I_{N_1} \otimes I_{N_2} \otimes D_3$$

This approach allows us to break a complex modelling problem into elements for which closed form solutions exist, or for which numerical methods are feasible. We next illustrate this in the case of an adjacency model on a regular $n$-dimensional grid. This is the motivation for our definition of the graph product in section 4.2.5.

We can extend this decomposition to 2D or 3D dimensional grids using either equation 4.10 or 4.11 respectively. These observations afford further insight into the random vector $s \sim \mathcal{N}(0, Q(\theta)^{-1})$ as

$$\ln f(s|\theta) \propto c + \theta_1 s^T s + \theta_2 \sum_{i=1}^{V} \gamma_i (U^T s)_i^2$$

from which we see that the distribution penalises by a flexible amount, $\theta_2$, those DCT-II coefficients, $(U^T s)_i^2$, corresponding to large eigenvalues, $\gamma_i$. Those indices $i$ for which the eigenvalues $\gamma_i$ is large are exactly those for which the corresponding eigenvectors represent high frequency activity, as illustrated in Fig 4.1a.
4.4.3 Illustration: Normalisation

An issue that must be addressed is the magnitude of the vectors drawn from distributions with correlation structure. If the distribution concerned is zero mean, then an increase in the correlation between entries will, unless we are careful, decrease the expected magnitude of the vector. This issue does not arise in classical matrix factorisations such as PCA since there the basis vectors are constrained to be of exactly unit length.

As an illustration, consider a model along the lines of Sec. 4.2.4, in particular $a \sim \mathcal{N}(0, \Delta^{-1}(\theta))$. Let the eigenvalues be single parameter linear functions $h_i(\theta) = 1 + \theta \gamma_i$, $\gamma_i \geq 0$. If the orthonormal eigenvectors $U$ are not the identity matrix then an increase in $\theta$ will increase the magnitude of the correlation between at least some entries of $a$:

$$
\text{Cor}(a_i, a_j) = \frac{\sum_{k=1}^{V} U_{i,k} U_{j,k} h_k(\theta)^{-1}}{\left(\sum_{k=1}^{V} U_{i,k} U_{i,k} h_k(\theta)^{-1}\right)^{1/2}\left(\sum_{k=1}^{V} U_{j,k} U_{j,k} h_k(\theta)^{-1}\right)^{1/2}}
$$

(4.12)

Without specifying the form of $U$ and $\gamma$ and choosing $i, j$ there is a limited amount we can do with Eqn. 4.12. However we can see that when $\theta = 0$ we find $h_i(\theta) = 1$, in which case the numerator becomes the dot product of two orthonormal vectors, so that for $i \neq j$, $\text{Cor}(a_i a_j) = 0$. If we take a particular structure, such as a DCT based precision matrix
using equation 4.9 on a vector $\mathbf{a} \in \mathbb{R}^{50}$ and consider $\text{Cor} (\mathbf{a}_{25}, \mathbf{a}_{26})$ we find the following relationship with $\theta$:

\[ \text{Corr}(\mathbf{a}_{25}, \mathbf{a}_{26}) \]

Fig. 4.2

However, this form for the $h_i(\theta)$ has side effects in terms of the norm of the vector which we can see by defining:

\[ m(\theta) = E \left[ \mathbf{a} \mathbf{a}^T \right] = E \left[ \mathbf{z}' \mathbf{D}(h(\theta))^{-1} \mathbf{z} \right] = \sum_{i=1}^{V} \frac{1}{h_i(\theta)} = \sum_{i=1}^{V} \frac{1}{1 + \theta \gamma_i} \]  

(4.13)

From which it can be seen that $m(\theta) \to 0$ as $\theta \to \infty$. This poses a problem as we would like to utilise distributions whose magnitude remains constant as their correlation level varies. However this suggests a solution to the problem if we use the reciprocal of $m(\theta)$ as a normalisation constant and define a new set of eigenvalue functions using some base eigenvalues $\gamma_i \geq 0$:

\[ h_i(\theta) = \left( \sum_{m=1}^{V} \frac{1}{g_m(\theta)} \right)^{-1} \quad g_i(\theta) = c(\theta) g_i(\theta) \]

So that under this parameterisation:

\[ m(\theta) = c(\theta) \sum_{i=1}^{V} \frac{1}{g_i(\theta)} = 1 \]  

(4.14)

It is useful at this point to note that the gradient of $c(\theta)$ is given by:
The gradient of $h$ is, then:

$$\nabla_\theta h_i(\theta) = \nabla_\theta \{ c(\theta)g_i(\theta) \} = g_i(\theta)\nabla_\theta c(\theta) + c(\theta)\nabla_\theta g_i(\theta) \quad (4.16)$$

One hundred samples from the normalised DCT structure under increasing values of $\theta$ (spaced evenly on the logarithmic scale from 1 to 10000) are shown in Fig. 4.3. We can see that the normalisation has the effect for large $\theta$ of generating vectors representing small fluctuations around a constant value. Whereas for low $\theta$ samples the sign fluctuates rapidly across space and a visually ‘grainier’ texture.

Fig. 4.3 Samples from the normalised DCT (nDCT) structure for $\theta \in (1, 10000)$
4.5 Discussion

These combinatorial rules for kronecker structure, powers, sums, and combination of spectral curves provide a powerful framework for working tractably with a range of nondiagonal latent covariance structures.

Rue and Held (2005) discuss the link between another graphical model and discrete linear transform. They showed that for circulant block precision matrices the relevant transform is the Discrete Fourier Transform (DFT). The difference between DFT and DCT-II being the relevant boundary conditions where DFT essentially considers graphs on a loop or torus, while DCT-II has free edges. While that work considered graphs on regular grids it did not make fully clear the transform interpretation or the implication in terms of penalisation.

Another popular linear transform that yields an interpretable graphical model is the Discrete Wavelet Transform (DWT). Suppose the inverse DWT provides the eigenvectors \( U \) of a precision matrix for a particular orthogonal wavelet basis. An example would be a model for a fully connected graph, or to make this more concrete, suppose that the exam results of children in a class were uniformly correlated with one another while each child was associated with an idiosyncratic random effect. Here the discrete Haar or D4 wavelet bases, combined with the eigenvalue 0 for the eigenvector 1, as usual, and the eigenvalue \( n \) for all other eigenvectors produce the required Laplacian matrix, \( nI - 11^T \). The main diagonal represents the idiosyncratic random effect and the uniform off-diagonals a group-wise random effect.

<table>
<thead>
<tr>
<th>( U )</th>
<th>complexity</th>
<th>graphical model</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCT</td>
<td>( \mathcal{O}(n \ln n) )</td>
<td>autocorrelation</td>
</tr>
<tr>
<td>DCT ( \otimes ) DCT</td>
<td>( \mathcal{O}(n' \ln n') ), ( n' = \max(n_1, n_2) )</td>
<td>2D spatial autocorrelation</td>
</tr>
<tr>
<td>DFT</td>
<td>( \mathcal{O}(n \ln n) )</td>
<td>circulant autocorrelation</td>
</tr>
<tr>
<td>DWT-Haar</td>
<td>( \mathcal{O}(n) )</td>
<td>fully connected groups</td>
</tr>
<tr>
<td>DWT ( \otimes ) DCT</td>
<td>( \mathcal{O}(\max{n_1, n_2 \log n_2 }) )</td>
<td>fully connected groups autocorrelated over time</td>
</tr>
<tr>
<td>Any</td>
<td>( \mathcal{O}(n^3) )</td>
<td>arbitrary precision matrix</td>
</tr>
<tr>
<td>Any ( \otimes ) DCT</td>
<td>( \mathcal{O}(\max{n_1^3, n_2 \log n_2 }) )</td>
<td>arbitrary precision matrix autocorrelated over time</td>
</tr>
</tbody>
</table>

The decompositions in (4.10) and (4.11) also help with Laplacians which are the product of characteristically different graphs. For instance suppose we were interested in modelling
daily fluctuations of temperature at different spatial locations. Midnight being both the end and start of a day, $L_1$ could represent a model where the daily timeseries was associated at its edges with a DFT based decomposition. While spatial dependency between nearby locations could be expressed with a 2D DCT based graph $L_2 \ast L_3$. These models are summarised in Table 2.1.
Chapter 5

Bayesian non-parametric matrix factorisation

5.1 Introduction

We again consider a $V \times T$ data matrix $Y$ comprising $T$ observations of $V$ dimensional vectors. As discussed in chapter 2 matrix factorisation methods decompose such multivariate observations into linear combinations of $K$ latent basis vectors, as per equation 2.2:

$$Y_{(V \times T)} = W_{(V \times K)} X_{(K \times T)} + E_{(V \times T)} \quad (2.2)$$

Dimension reduction is effected by approximating $Y$ with $WX$, thus using $K(V + T)$ which will generally be rather fewer than the $VT$ observed variables. Here $E$ is the difference between the low rank approximation and the data, whose sum of squares is the reconstruction error. We may refer to the entries of matrices playing the role of $X$ as components and rows of $W$ as features or basis vectors.

When applying matrix decomposition we generally face uncertainty about the number of latent features, $K$, required to explain the data. We start from the assumption that the reader wishes to learn the latent basis vectors, the components of the data in that basis and their number from the data in a single, coherent model for approximating the data.

As discussed in section 2.6 a number of criteria are available which assign a score to each choice of $K$. Dimensionality is estimated by selection of the $K$ whose score is optimal. Methods based on such tactics do not typically account for uncertainty in the choice of the optimal $K$. Methods based on the criteria discussed in section 2.6 also assume an orthonormal basis with no correlation structure within basis vectors beyond normalisation to unit length.
At the same time we seek decompositions with desirable characteristics, such as sparse activation of features, or constraints on the covariance structure of latent features. Due to the wide family of methods which conform to the algebraic structure in (2.2) methods which improve our accounting for dimensional uncertainty have a broad scope for application.

Viroli (2009) provided a Bayesian approach to matrix factorisation that accounted for uncertainty in $K$. In that scheme the dimension $K$ was sampled through reversible jump MCMC (RJMCMC) while the components $X$ were draws from Gaussian mixture models. Such a model implies that the representation of observations with respect to the latent features would be dense.

We will primarily be concerned with observations that are neuroimages. Work on natural images (Hyvärinen et al., 2009), together with successes in signal processing (Elad et al., 2010) and representation learning (Bengio et al., 2013), suggests that modelling representations of images as sparse with respect to a basis is a powerful and parsimonious strategy. In the formalism of equation 2.2 this means $X$ being sparse when $W$ was chosen appropriately. Hence the approach in Viroli (2009) would not be entirely satisfying as a model for neuroimaging data.

Sparsity is also a solution to certain forms of non-identifiability. For instance in PCA with Gaussian components the likelihood is unaffected by orthogonal rotations of the solution. Sparsity constraints can remove this indeterminacy, although it may be replaced with a reduced vulnerability in terms of permutation indeterminacy and rotations of subsets of features with identical sparsity patterns.

An alternate strategy to account for uncertainty in $K$ was proposed by Knowles and Ghahramani (2007) through the use of the Indian Buffet Process (IBP). A draw, $B$, from the $IBP(\alpha, \beta)$ distribution is a binary matrix and can be used in an elementwise product (denoted $\circ$) with a matrix of scaling coefficients $S$ to produce a sparse, random, weighting matrix, $X = S \circ B$. Here $S$ and $B$ are theoretically $(\infty \times T)$ but, for finite $T$, $B$ is guaranteed to have a finite number of nonzero rows, making computational use tractable. The decomposition in equation 2.2 becomes:

$$Y = W(S \circ B) + E \quad (5.1)$$

Knowles and Ghahramani (2007) provides models for infinite Independent Components Analysis (iICA) and infinite sparse Factor Analysis (isFA) by changing the elementwise distributional assumptions on entries of $S$ as following a Laplace or Gaussian distribution respectively. Note that when dealing with the IBP that the effective rank of the decomposition, $K$, is merely a statistic arising by counting the nonzero columns in $B$. This means $K$ is not a parameter to be sampled separately, and so specialist RJMCMC techniques are not required.
A limitation of the existing work on nonparametric matrix factorisation is the use of IID univariate distributions to model latent basis vectors which we reasonably expect to exhibit strong dependency structures. That is, existing models provide a route to sparsity with respect to a basis, but do not exploit prior beliefs about the structure of that basis. For instance a collection of time series that were sparse in the Fourier basis would lead to a smooth set of basis vectors, $W$. We prepared the ground for this problem in chapter 4 when we considered efficient models for covariance structures via parameterised eigendecompositions.

In section 5.2 we present the contribution of this chapter, an extended infinite Independent Structured Sparse Factorisation Analysis model. Section 5.3 provides a Gibbs sampling algorithm for our model. The utility of our method is demonstrated in section 5.4 where it is found to outperform PCA and the existing Bayesian nonparametric model on simulated data.

### 5.2 Model

In this section we propose an extension, infinite Sparse Structured Factor Analysis (iSSFA), to the work of Knowles and Ghahramani (2007). Our model provides nondiagonal covariance structure for the latent features. The full specification of our model is laid out in table 5.1.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Indices</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B \sim IBP(\alpha, \beta)$</td>
<td></td>
<td>component activations</td>
</tr>
<tr>
<td>$S_{kt} \sim \mathcal{N}(\tau_k, \nu_k^{-1})$</td>
<td>$k = 1, \ldots t = 1, \ldots, T$</td>
<td>component scalings</td>
</tr>
<tr>
<td>$w_k \sim \mathcal{N}(0, \Delta(\theta)^{-1})$</td>
<td>$k = 1,$</td>
<td>features</td>
</tr>
<tr>
<td>$E_{jt} \sim \mathcal{N}(0, \sigma^2 I)$</td>
<td>$t = 1, \ldots, T$</td>
<td>measurement error</td>
</tr>
<tr>
<td>$\sigma^2 \sim \text{InvG}(e_\sigma, f_\sigma)$</td>
<td></td>
<td>noise level</td>
</tr>
<tr>
<td>$\alpha \sim \mathcal{G}(e_\alpha, f_\alpha)$</td>
<td></td>
<td>IBP strength</td>
</tr>
<tr>
<td>$\beta \sim \mathcal{G}(e_\beta, f_\beta)$</td>
<td></td>
<td>IBP repulsion</td>
</tr>
<tr>
<td>$\nu_k \sim \mathcal{G}(e_\nu, f_\nu)$</td>
<td>$k = 1,$</td>
<td>scaling variable precisions</td>
</tr>
<tr>
<td>$\tau_k \sim \mathcal{N}(m_\tau, r_\tau^{-1})$</td>
<td>$k = 1,$</td>
<td>scaling variable means</td>
</tr>
<tr>
<td>$\ln \theta_p = \xi_p \sim \mathcal{N}(m_p, r_p^{-1})$, $p = 1, \ldots, P$</td>
<td></td>
<td>feature parameters</td>
</tr>
</tbody>
</table>

Table 5.1 Variables and their distributions.

Compared to Knowles and Ghahramani (2007) the enriched prior on $S$ allows different means and variances for each latent component. This adds expressivity to the model when sources are not identically distributed. Also, we observe that zero mean components in $S$ can lead more easily to degenerate situations with respect to sparsity in which $B = 1_k 1_T'$. We
Bayesian non-parametric matrix factorisation

will see in section 5.3.2 that non-zero means enable the relevant spatial vector \( w_k \) to explain residuals when deciding whether to activate a component for an observation \( B_{k,t} \).

Where the precision matrix of the latent basis vectors is modelled in the fashion developed in chapter 4 so that \( \Delta(\theta) \in G_U \) for some orthonormal basis \( U \) of \( \mathbb{R}^V \) and eigenvalues \( h(\theta) \). This imposition of a correlation structure on the latent basis vectors forms a point of differentiation both from both classical matrix factorisation techniques such as PCA (section 2.3.1) and the more recent Bayesian nonparametric approaches of Knowles and Ghahramani (2007) and Virol (2009).

Johnson et al. (2007) provided one approach to modelling spatial dependency in a matrix factorisation model. The strategy adopted there was to reduce the data in the spatial domain via a Discrete Wavelet Transform, retaining only the coefficients for the father wavelets (100/512% of the original data). Our model implies a different approach, where a spatial model is enforced on the basis vectors prior to mixing. To see the difference, taking the transform of a data vector after mixing, \( z = \Phi^T y \), results in coefficients that could be the result of many different bases. For instance if we take \( y = [1, 1]^T \):

\[
y = Wx = W^T x' = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{5} \\ 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (5.2)
\]

It should instead be clear that \( U^T W \) and \( U^T W' \) are not equal in general; for instance we could set \( U \) to be the relevant Discrete Wavelet Transform \( \Phi \). These quantities form a major part of the pdf of a high dimensional Gaussian (c.f. equation 4.3) in our model. As such, our model provides a way to distinguish between two such bases that the approach of Johnson et al. (2007) does not, by handling spatial correlation at the level of individual basis vectors rather than after mixing. In neuroimaging terms this means modelling spatial correlation within each functional network’s map separately, rather than globally in terms of the observed data.

### 5.3 Gibbs sampler

Building on the methods of Knowles and Ghahramani (2007) we can construct a Gibbs sampler for the model proposed in section 5.2. The sampler is approximate in that we use a Laplace approximation to the posterior distribution of the latent feature parameters.

Note that the notational conventions set out in section 2.2 continue to apply here.
5.3.1 Conditional distributions of latent basis vectors, $w_k$

Consider resampling the feature $w_k$. Define a residual:

$$e_t^k = y_t - W;_{-k}(B \circ S)_{-k,t}$$

The Gibbs conditional distribution for $w_k$ is, then, multivariate normal,

$$w_k|S, B, W;_{-k}, \sigma^2, Y \sim \mathcal{N}(\mu_{w_k}, \Sigma_{w_k})$$

with

$$\mu_{w_k} = \left(\frac{\sum_{t=1}^T S_{k,t}^2 B_k}{\sigma^2} B_k I_V + \Delta\right)^{-1} \left(\frac{1}{\sigma^2} \sum_{t=1}^T S_{k,t} B_{k,t} e_t^k\right)$$

$$\Sigma_{w_k} = \left(\frac{\sum_{t=1}^T S_{k,t}^2 B_k}{\sigma^2} I + \Delta\right)^{-1}$$

The feature vectors function as a basis for the latent data space. We can see from these conditional distributions that the Gibbs sampler bears a resemblance to the Gram-Schmidt orthogonalisation procedure. Each feature is treated as though it were the final vector in the procedure, the effects of the other features having been subtracted from the data to leave a residual vector which is then normalised by application of a covariance matrix.

We note also that $\Sigma_{w_k} \in G_U$ since, from the rules identity and positive scalar multiplication in sections 4.2.2 and 4.2.3, we know that for $Q \in G_U$ that $cI + Q \in G_U$ also. This means that we know the eigendecomposition of $\Sigma_{w_k}$ and can therefore resample $w_k$ quickly given a fast implementation of $U$ as an operator.

5.3.2 Activation of shared features within B

Sampling of $B$ occurs in two phases. We describe features as shared when $m_k = \sum_{t=1}^T B_{k,t} > 1$ and as unique when $m_k = 1$. A draw from the two parameter IBP is obtained by, for the $t^{th}$ observation, activating each shared feature with probability $\Sigma_{r=1}^{t-1} B_{k,r}/(\beta + t - 1)$ and then drawing a set of Poisson($\alpha \beta / (\beta + t - 1)$) unique features (Knowles and Ghahramani, 2007). The columns of the matrix are exchangeable so that conditional on the other observations, the $t^{th}$ observation may be treated as the $T^{th}$. When doing posterior inference, this structure leads naturally to two stages, in which the shared features are sampled as Bernoulli random variables, and a quantity of unique features sampled from a Poisson distribution. The following step deals with the resampling of shared features.
For \( \mathbf{B} \) we will approach sampling entry-wise, using the technique of Knowles and Ghahramani, 2007 in defining a ratio of conditionals \( r = r_1 r_p \). Where \( r_p \) is available from section 2.1 of that work, \( r_p = \frac{m_{k,t}}{\beta + T - 1 - m_{k,t}} \), with \( m_{k,t} \) being the number of times the \( k^{th} \) factor has been activated, excluding observation \( t \). In our case \( r_l \) is different with:

\[
    r_l = \frac{p(y_t|S_{-k,:}, S_{k,-t}, \mathbf{W}, \mathbf{B}_{-k,:}, \mathbf{B}_{k,-t}, \mathbf{B}_{k,t} = 1, \sigma^2)}{p(y_t|S_{-k,:}, S_{k,-t}, \mathbf{W}, \mathbf{B}_{-k,:}, \mathbf{B}_{k,-t}, \mathbf{B}_{k,t} = 0, \sigma^2)}
\]

The required marginal distribution of the numerator is again Gaussian:

\[
    y_t|S_{-k,:}, S_{k,-t}, \mathbf{W}, \mathbf{B}_{-k,:}, \mathbf{B}_{k,-t}, \mathbf{B}_{k,t} = 1, \sigma^2, \sim \mathcal{N}
    
    \left( \mathbf{W}_{-k}(\mathbf{S} \circ \mathbf{B})_{-k,t} + \mathbf{w}_k \tau_k, \sigma^2 \mathbf{I}_V + \frac{1}{V_k} \mathbf{w}_k \mathbf{w}_k^T \right)
\]

This presents an important modelling difference from the work in Knowles and Ghahramani (2007) where, due to the assumption of zero-mean components (\( \tau_k = 0 \)), the mean of the numerator distribution would not feature \( \mathbf{w}_k \). That fact would force the algorithm to explain the residual \( y_t - \mathbf{W}_{-k}(\mathbf{S} \circ \mathbf{B})_{-k,t} \) using only the covariance structure. Non-zero means, \( \tau_k \), allows a feature vector to help explain the residual when sampling a feature activation \( \mathbf{B}_{k,t} \).

Now set:

\[
    r_0 = y_t - \mathbf{W}_{-k}(\mathbf{S} \circ \mathbf{B})_{-k,t}
\]
\[
    r_1 = y_t - \mathbf{W}_{-k}(\mathbf{S} \circ \mathbf{B})_{-k,t} - \tau_k \mathbf{w}_k
\]

We can now evaluate \( \ln r_l \) as:

\[
    \ln r_l = \ln p(y_t|\mathbf{B}_{k,t} = 1, \cdot) - \ln p(y_t|\mathbf{B}_{k,t} = 0, \cdot)
\]
\[
= -\frac{1}{2} \ln \left( 1 + \frac{1}{\sigma^2 V_k} \mathbf{w}_k \mathbf{w}_k^T \right) - \frac{1}{2\sigma^2} \mathbf{r}_1^T \mathbf{r}_1
\]
\[
+ \frac{1}{2\sigma^2} \mathbf{r}_0^T \mathbf{r}_0
\]

Hence, despite both the precision and covariance matrices for the numerator being dense, the density may be evaluated in \( \mathcal{O}(V) \) time and memory as the main calculations are dot
products of dense vectors in $\mathbb{R}^V$. For the shared features, $k$, we can now sample $B_{k,t} \sim \operatorname{Bern}\left(\frac{r_l r_p}{r_l r_p + 1}\right) = \operatorname{Bern}\left((1 + \exp(-\ln(r_l r_p)))^{-1}\right)$.

We use the techniques described in Pearce and White (2016b) to efficiently parallelise these calculations across a compute cluster. This material is covered and generalised in chapter 6. The techniques involve particular choices for the order of Gibbs sampling to facilitate parallelisation, and thus are not needed here and in section 5.3.3 where the Gibbs distribution of single variables is concerned.

### 5.3.3 Scaling variables for shared features, $S_{k,t}$

Setting $\mu_t = W_{:, -k}(S_{:, k} \circ B_{:, k})$, the Gibbs conditional for $S_{k,t}$ is Gaussian with:

$$S_{k,t} | \cdot \sim \mathcal{N}(m_S, v_S)$$

with

$$m_S = \left( v_k + \frac{w_k w_k^T}{\sigma^2}\right)^{-1} \left( \frac{1}{\sigma^2} w_k^T (y_t - \mu_t) + v_k \tau_k \right)$$

$$v_S = \left( v_k + \frac{w_k w_k^T}{\sigma^2}\right)^{-1}$$

### 5.3.4 Activating unique features

In order to complete a draw from the Indian Buffet Process we must sample unique features for each observation, a task we will tackle by a Metropolis-Hastings step for each observation. The IBP is exchangeable (Griffiths and Ghahramani (2011), Knowles and Ghahramani (2007)) hence the $t^{th}$ observation may be treated as the $T^{th}$. Define $J_t$ to be the set of indices of $n$ new components for observation $t$. We set the number of such components and the parameters and weights associated with them as $\omega = (n, \{\tau_j\}, \{\nu_j\}, \{s_j\})$ where the $s_j$ are weights for features.

We make a proposal $\omega^*$ in three steps: sampling the number of new sources $n$, then their parameters $\{\tau_j\}, \{\nu_j\}|n$ and finally the weightings $s_j|\{\tau_j\}, \{\nu_j\}$. Note that at this point the feature vectors will be marginalised out. We then choose to accept or reject the proposal via an MH step. It will be convenient to write,

$$\mu = W_{:, -J_t}(S \circ B)_{-J_t}$$
since the values of the rows of $S, B$, and columns of $W$ corresponding to shared features enter in to our calculations only through the mean of $y_i$. Afterwards, conditional on MH acceptance, we will sample the new basis vectors $\{w_j\}_{j \in J_t}$ (see section 5.3.5). That is to say we block sample $p(\omega, \{w_j\}|\cdot)$ as $p(\{w_j\}|\omega, \cdot)p(\omega|\cdot)$ for each $t$.

To begin, we use the respective priors to generate proposals:

$$n \sim \text{Poi} \left( \frac{\beta \alpha}{\beta - T - 1} \right)$$

$$\tau_j | n \sim \mathcal{N}(m_n, r^{-1}_n), \ j \in J_t$$

$$v_j | n \sim \mathcal{G}(e_v, f_v), \ j \in J_t$$

$$s_j | n, \tau_j, v_j \sim \mathcal{N}(\tau_j, v_j^{-1}), \ j \in J_t$$

Then form an acceptance probability ratio where we make use of the proposal $q(\omega^*|\omega)$ being equal to the prior $p(\omega^*)$ so that:

$$u = \frac{p(\omega^*|y_t, \cdot)q(\omega|\omega^*)}{p(\omega|y_t, \cdot)q(\omega^*|\omega)} = \frac{p(y_t|\omega^*, \cdot)}{p(y_t|\omega, \cdot)}$$

Which forms the basis of the acceptance probability. We now focus on the density for $p(y_t|\omega)$, for which we need to marginalise out the features corresponding to the weights we sampled. In their prior, the features $\{w_j\}$ are, conditional on the hyperparameters, IID multivariate normal. Let us stack them into a single vector, and define a block matrix which we will use to manipulate them using bracketed $(i)$ notation to indicate the $i^{th}$ index in in $J_t$:

$$r = \begin{bmatrix} w_{(1)} \\ w_{(2)} \\ \vdots \\ w_{(n)} \end{bmatrix}, \ C = [S_{(1)}_d I_V, S_{(2)}_d I_V, \cdots, S_{(n)}_d I_V]$$

So that $r$ is $(nV \times 1)$ and $C$ is $(V \times nV)$. The conditional distribution of the observations is, then:

$$y_j | B_{d,t}, r, \cdot \sim \mathcal{N}_V (\mu + Cr, \sigma^2 I_V)$$

$$r \sim \mathcal{N}_{nV} (0, I_n \otimes \Delta^{-1})$$ (5.3)
5.3 Gibbs sampler

Hence, using results on the multivariate normal distribution (Bishop, 2006) the marginalised conditional we seek is given by,

\[ y_t | B_{J,t}, \cdot \sim \mathcal{N}(\mu, \sigma^2 I + \left( \sum_{j \in J_t} B_{j,t}^2 \right) \Delta^{-1}) \]  \hspace{1cm} (5.4)

and this density function provides the numerator and denominator of \( u \), enabling us to calculate the acceptance probability for the proposal. Note also that, again, due to the results of chapter 4 we know that the covariance matrix is in the same family \( G_U \) as the prior \( \Delta \) thus facilitating rapid evaluation of the density for efficient \( U \).

5.3.5 Drawing spatial feature vectors for unique features

This section continues with the notation used in section 5.3.4, as we are completing a block draw. For each observation \( t \) we need to draw the spatial features from the distribution \( S_{J,t} | \omega, \cdot \). Hence, reversing the conditioning in equation 5.3 we see that the posterior for \( r \) is again multivariate normal:

\[
\begin{align*}
\mathbf{r} | y_t, \cdot &\sim \mathcal{N}(\mu_r, \Sigma_r) \\
\mu_r &= \left( \frac{1}{\sigma^2} C^T C + I_n \otimes \Delta \right)^{-1} \left( \frac{1}{\sigma^2} C^T (y_t - \mu) \right) \\
\Sigma_r &= \left( \frac{1}{\sigma^2} C^T C + I_n \otimes \Delta \right)^{-1}
\end{align*}
\]

5.3.6 Sampling the noise level \( \sigma^2 \)

If we place an inverse gamma prior on \( \sigma^2 \sim \text{InvG}(e_\sigma, f_\sigma) \) then conditional on the other variables, the result is conjugate. Write \( \epsilon_t = y_t - W (S \circ B)_{:,t} \) and note that there are \( T \) such residuals. The conditional follows an inverse gamma distribution:

\[
\sigma^2 | Y, \cdot \sim \text{InvG} \left( \frac{VT}{2} + e_\sigma, f_\sigma + \frac{1}{2} \sum_{t=1}^{T} \epsilon_t^T \epsilon_t \right)
\]

5.3.7 Sampling covariance structure parameters

We will work with covariance structures that are parameterised by non-negative variables \( \theta \) so that the log normal distribution is an appropriate prior. On that basis let \( \xi_p = \ln \theta_p \) for \( p = 1, \ldots, P \) and let the prior comprise independent normal distributions so that \( f(\xi) = \prod_{p=1}^{P} \mathcal{N}(\xi_p; m_{\xi_p}, r_{\xi_p}) \).
We sample from the conditional posterior for $\xi$ using a Laplace approximation (Rue et al. (2009)) to the posterior distribution, that is to:

$$p(\xi|W,\cdot) \propto p(W|\xi,\cdot)p(\xi)$$

We find that this approach works well across scales of $\theta$ without the need for additional MCMC tuning parameters required by asymptotically exact methods, for instance slice sampling. To perform the Laplace approximation we must find the parameters which maximise the posterior distribution. Using the density for our parameterisation of the multivariate normal in equation 4.3 the log posterior is:

$$\ln f(\xi|W,\cdot) = c + \ln f(w|\theta(\xi)) + \ln f(\xi)$$

which has gradient,

$$\nabla_\xi \ln p(\xi|W) = \theta \circ \nabla_\theta \ln f(W|\theta) + \nabla_\xi \ln f(\xi)$$

where the gradient of the likelihood $\nabla_\theta \ln f(W|\theta)$ is available from equation 4.4. The Hessian is given by,

$$H = \nabla_\xi \nabla_\xi \ln f(\xi|W) = D(\theta \circ \nabla_\theta \ln f(W|\theta)) + \left(\theta \theta^T\right) \circ \nabla_\theta \nabla_\theta \ln f(W|\theta) + \nabla_\xi \nabla_\xi \ln f(\xi)$$

where the Hessian of the likelihood, $\nabla_\theta \ln f(W|\theta)$, can be calculated via equation 4.5 and the gradient in equation 5.6.

We then use numerical optimisation to find a MAP estimate $\hat{\xi}$, following which we sample from the approximate conditional posterior distribution:

$$\xi|W,\cdot \sim \mathcal{N}(\hat{\xi},H^{-1})$$
If required, this distribution can be used as a proposal for a corrected Metropolis-Hastings step with acceptance ratio:

\[
p(\xi' | \mathbf{W}, \cdot) N(\xi'; \hat{\xi}, \mathbf{H}^{-1})
\]

\[
p(\xi | \mathbf{W}, \cdot) N(\xi'; \hat{\xi}, \mathbf{H}^{-1})
\]

Which, if the approximation is good should have a high acceptance rate, and where all the quantities can be evaluated using previous computations. However, in the example of section 5.4 we sample from the Laplace approximation directly.

5.3.8 Sampling the IBP strength \( \alpha \)

We use a rate parameterised alternative to that in Knowles and Ghahramani (2007). Let \( H_T(\beta) = \sum_{i=1}^{T} \frac{\beta_i}{\sum_{i=1}^{T} \beta_i} \), then:

\[
P(\alpha | \mathbf{B}, \beta) \propto P(\mathbf{B} | \alpha, \beta) P(\alpha) \\
\propto \mathcal{G}(\alpha; K_+ + e_{\alpha}, f_{\alpha} + H_T(\beta))
\]

Where \( K_+ \) is the number of non-zero columns of \( \mathbf{B} \).

5.3.9 Sampling the IBP repulsion \( \beta \)

We will sample this via a metropolis step. We use the shape-rate parametrised Gamma prior \( \beta \sim \mathcal{G}(\epsilon_{\beta}, f_{\beta}) \). In order to ensure non-negativity we will work with parameter logarithms. We take a Normal distribution for the proposal \( q(\psi|\psi^*) = \mathcal{N}(\psi^*; \psi, \sigma^2) \). Write \( m_k = \sum_{t=1}^{T} B_{k,t} \), \( B(\cdots) \) is the beta function, \( H_T(\beta) \) is as above, and \( K_l \) is the number of columns whose entries correspond to the integer \( l \) expressed in binary (reading a column of \( \mathbf{B} \) downwards yields a binary number \( 0 \leq l \leq 2^T - 1 \)). The likelihood ratio is:

\[
L = \ln \frac{P(\mathbf{B} | \alpha, \psi^*)}{P(\mathbf{B} | \alpha, \psi)} = -\alpha H_T(\beta^*) + \alpha H_T(\beta) \\
= \sum_{k=1}^{K_+} [\ln \Gamma(T - m_k + \beta^*) - \ln \Gamma(T - m_k + \beta)] \\
+ K_+ [\ln \Gamma(T + \beta) - \ln \Gamma(T + \beta^*)] \\
+ \alpha (H_T(\beta) - H_T(\beta^*))
\]
Bayesian non-parametric matrix factorisation

So that the log Metropolis-Hastings ratio is found as:

\[
\ln a_{\beta \to \beta^*} = L + e_\beta \ln \beta^* - f_\beta \beta^* - e_\beta \ln \beta + f_\beta \beta
\]

\[
= L + e_\beta [\ln \beta^* - \ln \beta] + f_\beta [\beta - \beta^*]
\]

5.3.10 Sampling weight precisions \( \nu_k \)

The Gibbs distribution for \( \nu_k \) is a Gamma:

\[
v_k | \{ S_{k,t} \}, B_{k,t} = 1, \cdot \sim \mathcal{G}(\tilde{\epsilon}_v, \tilde{f}_v)
\]

with

\[
\tilde{\epsilon}_v = e_v + 1/2 \sum_{i=1}^{T} B_{k,i},
\]

\[
\tilde{f}_v = f_v + \frac{1}{2} \sum_{i=1}^{T} ((S_{k,t} - \tau)B_{k,t})^2
\]

5.3.11 Sampling the weight means \( \tau_k \)

The Gibbs distribution \( \tau_k | S, B, \cdot \) is normal with:

\[
\tau_k | S, B, \cdot \sim \mathcal{N}(\mu_\tau, \zeta_\tau)
\]

\[
\mu_\tau = \left( \nu_k \sum_{t=1}^{T} B_{t,k} + r \right)^{-1} \left( \nu_k \sum_{t:B_t,k=1} S_{t,k} + rm \right)
\]

\[
\zeta_\tau = \left( \nu_k \sum_{t=1}^{T} B_{t,k} + r \right)^{-1}
\]

5.4 Simulated Example

We will model \( \Delta(\theta) = g(\theta)I_V + h(\theta)L = \theta_1 I_V + \theta_2 L \). We will take \( L \) to be a multi-dimensional neighbourhood graph as discussed in section 4.4 and hence made efficient through the DCT. In our simulated example of section 5.4 this will be a 2D graph, while in neuroimaging example of chapter 7 it will be a 3D. However, the scheme provided below would work for other suitable \( Q \) as described in section 4.1.1 with little modification. Observe that the distribution of features requires that they be smooth to a level determined by the spatial hyperparameter \( \theta \), which we then learn from the data.
We generated $T = 3000$ observations on a $100 \times 100$ grid ($V = 10000$). We chose 2 dimensions as it offers a compromise between ease of visualisation and closeness to the neuroimaging case. Each observation was a linear combination of 50 features drawn as $w_k \sim \mathcal{N}(0, \mathbf{Q}(1, 100)^{-1})$ and then unit normalised. Scaling variables were drawn as $\mathbf{B}_{k,t} \sim \text{Bern}(1/20)$ with $\mathbf{S}_{k,t} \sim \mathcal{N}(1, \gamma_k)$, $50 < \gamma_k < 100$, $\sigma^2 \sim \mathcal{N}(0, 1)$. Additive noise was drawn IID $\mathbf{E}_{t,v} \sim \mathcal{N}(0, 1)$. A further holdout dataset of 500 images was drawn from the same distribution for performance testing. The first 10 observed images from the hold out data set are shown in Figure 5.1a and the corresponding latent images in Figure 5.1b.

(a) Observed images $\mathbf{Y}_{:,1:10}$ from holdout data

(b) Corresponding latent vectors $\mathbf{W}(\mathbf{S} \circ \mathbf{B})_{:,1:10}$

(c) Corresponding PCA reconstruction with $K^* = 33$

(d) Corresponding iSSFA reconstruction $\hat{\mathbf{E}}[\mathbf{W}(\mathbf{S} \circ \mathbf{B})_{:,1:10}]$. Note the similar length scale to the true basis vectors in b).

(e) Calculated as with d) but under the spatially IID prior. Note the shorter length scale, noisier appearance than the true latent basis vectors in b)

Fig. 5.1 Different views of example images from the holdout dataset.
Fig. 5.2 Analysis of the dimensionality of the dataset. Blue line shows the Laplace evidence approximation of the dimensionality obtained by Minka’s method (optimising $K^* = 33$). Black dashed line shows the first 100 eigenvalues of the data covariance matrix.

Six nodes of a high performance compute cluster were used to execute the analysis. Each node had twelve Intel Xeon E5-2620 2.00GHz CPUs and an Nvidia Tesla K20m GPU. The code was written in Julia (Bezanson et al. (2014)) in order to utilise the multiprocessing capabilities of the cluster. Features for the sampler were initialised using a K-means algorithm fit for 15 clusters.

Fig. 5.3 MCMC sampling traces for analysis of the same dataset under different priors.

Call the true latent data $X = W(S \circ B)$. Figure 5.1d shows the mean reconstruction of the latent data. That is the Monte Carlo approximation:

$$
\mathbb{E}[W(S \circ B)] \approx \hat{Y}_{iSSFA} = \frac{1}{N} \sum_{n=1}^{N} W^{(n)}(S^{(n)} \circ B^{(n)})
$$
where superscripts \((n)\) indicate samples from particular MCMC iterations. Figure 5.1e shows the same estimate under the spatially IID prior of Knowles and Ghahramani (2007). In both cases we used a thinned subsequence of the MCMC samples to obtain the estimate.

(a) Qualitative comparison of feature recovery. Top row: first ten true features. Second row: matched iSSFA features (final MCMC sample); third row matched FastICA features; bottom row: matched eigenvectors. The eigenvectors and the related ICA features appear to vary on a dissimilar scale to the true features. The iSSFA features are all of a similar scale to the true features.

(b) Top ten matched features under the spatially IID prior of Knowles and Ghahramani (2007) (final MCMC sample). Colour normalisation different to (a).

Fig. 5.4 Comparison of matched features. Matching done by cosine similarity after normalisation.

We can compare the reconstruction performance of iSSFA model to the reconstruction of the data obtained from PCA / SVD. We estimate the model order using Minka’s method resulting in a setting of \(K^* = 33\). Analysis of dataset dimensionality is shown in Figure 5.2. It is of note that Minka’s method still provides empirically better PCA performance than using the true \(K = 50\).

For the MCMC results the reconstruction error was calculated as \(\sum_{t,v} (\hat{Y}_{v,t} - \hat{Y}^{iSSFA}_{v,t})^2\). For PCA the solution is to reverse the matrix factorisation \(\hat{Y}^{EIG} = W^{EIG} X^{EIG}\) using the first \(K^*\) number of principal components. We found that on the holdout data the reconstruction error for PCA was 0.0152, for the IID model was 0.0119 and for the spatial model was 0.0091. Note that these values are different to the left hand y-axes of Figure 5.3 because they represent the evaluation of the reconstruction across samples rather than at a single sample.
This makes the ratio of the IID to spatial reconstruction error 1.3076. The reconstruction error using the first $K^* = 33$ principal eigenvectors is nearly 1.67 times greater than that of the spatial model.

We also compare the iSSFA results to another standard matrix decomposition method ICA, implemented through the FastICA algorithm (Hyvärinen and Oja (2000)). This methodology builds on the PCA solution (as set out in section 2.3.3) by further factorising the sources $W^{EIG} = W^{ICA}X^{ICA}$. Substituting this into the PCA decomposition of the data $Y = W^{EIG}W^{ICA}X^{ICA} = W^{ICA}X^{ICA}$. A comparison of these different sets of features with the true features is shown in Figure 5.4 where it is visible that our nonparametric method finds features that are more similar to the true features.

FastICA attempts to decompose a signal into independent components by maximising the non-Gaussianity of inferred sources $X^{ICA}$, using a metric based on an approximation to negentropy. As can be seen in Fig 5.5 iSSFA produces more highly non-Gaussian estimates of the source distribution than does the FastICA algorithm applied to the same dataset. This is due to the sparsity induced by the Indian Buffet Process, which turns the iSSFA source distribution into a spike-and-slab model. Using sample excess kurtosis as a measure of non-Gaussianity (higher is more super-Gaussian), in the last MCMC iteration iSSFA scored 73.8 while the corresponding FastICA score was 1.1. The shapes of the histograms indicate that this ranking would hold under other reasonable metrics.

Another metric for the quality of the blind source separation is defined in Knowles and Ghahramani (2011) for a $V \times K$ matrix $M$ and a $V \times L$ matrix $N$ as $E_r(M, N) = \sum_{k=1}^{K} \min_{j \in 1, \ldots, L} \| M_{:,k} - N_{:,j} \|^2$. Here we found the ratio of errors to be $E_r(W, \hat{W}^{ICA}) / E_r(W, \hat{W}^{iSSFA}) = 1.41$. Whereas under the spatially IID nonparametric prior the same ratio was 0.87, i.e. worse than FastICA. This suggests that, where appropriate, modelling spatial continuity will help to provide a more accurate blind source separation; and also that the number of extra features in the iSSFA model is unlikely to be a sufficient explanation for its better score than ICA.

Figure 5.4a shows four rows of features. The top row visualises the first true ten latent features. The following three rows show other types of features matched by cosine similarity: iSSFA features from the final MCMC iteration; FastICA features; and finally PCA features. Figure 5.4 shows matched features obtained under the iSSFA prior of Knowles and Ghahramani (2007). We can see that modelling latent covariance structure has allowed the model to find a solution which more closely matches the ground truth than FastICA, which produces noisy features that vary on the wrong spatial scale. We also see that even within nonparametric models, accounting for latent covariance has improved feature recovery relative to the IID prior.
Fig. 5.5 Distribution of inferred sources under iSSFA and FastICA. If one were interested in maximising the non-Gaussianity of inferred sources the sparsity of the iSSFA model produces a more optimal fit by the kurtosis criterion than does the FastICA methodology.

Hence, at least when considering sparsely activated data, nonparametric sparse factor analysis can provide a more accurate reconstruction of the original data than can the eigendecomposition. Furthermore relaxation of orthogonality constraints can lead to features which are, individually, more similar to latent features.

5.5 Conclusion

In this chapter we have extended the nonparametric approach to matrix factorisation of Knowles and Ghahramani (2007). Specifically, we provided a more flexible prior on the components and a fully ‘spatial’ prior on the basis vectors. In simulations we saw that the extended model was superior for separation of correlated latent basis vectors, both in terms of separation metrics and in terms of MCMC convergence. We also saw that our spatial nonparametric approach performed better than PCA-ICA methods in simulation, which we suppose is a pre-requisite for improvements in neuroimaging data. In the next chapter we lay out the theory behind a method relied upon in our software to increase the speed of MCMC inference of factorisation models through parallelisation.
Chapter 6

Parallel Gibbs Sampling in Repeated Graphs

6.1 Motivation and approach

Gibbs sampling (Geman and Geman, 1984) is a Markov Chain Monte Carlo procedure (Gilks et al., 1995) for a model containing $M$ latent variables $Z_1, \ldots, Z_M$ and data $Y$. An iteration of Gibbs sampling requires some permutation $\pi$ of the $M$ variables which are then sampled from their distributions, conditional on the most recently sampled values of the other latent variables and the data, $p(Z_{\pi(m)}|Z_{\pi(1)}^{t}, \ldots, Z_{\pi(m-1)}^{t}, Z_{\pi(M)}^{t-1}, \ldots, Z_{\pi(M)}^{t-1}, Y)$ for $m = 1, \ldots, M$ and where superscripts indicate values drawn at particular iterations.

In order to compute the conditional distribution of $Z_{\pi(m)}$ we must, in the general case, wait until the preceding variables have been sampled from their own conditional distributions. This limits the scope for parallelisation of Gibbs samplers since it results in processors sitting idle while waiting to be informed of the results of calculations performed elsewhere. In highly parallel environments there are further data transfer overheads that can form a substantial computational bottleneck (Singh and Reddy, 2015). We call the sending of these updates regarding recently sampled values synchronisation steps.

Let us denote an undirected graph with vertices $\mathcal{V}$ and edges $\mathcal{E}$ by $G$. When $x \in \mathcal{V}, y \in \mathcal{V}$ we say that $x$ and $y$ are neighbours, denoted $x \sim y$, when $(x, y) \in \mathcal{E}$. Observing that the ordering is irrelevant for $\sim$ since the graph is undirected. In the following vertices will represent random variables, and edges conditional dependencies.

Conditional independence between groups of latent variables facilitates parallelisation. Recall that if a set of random variables $A$ is conditionally independent of another $B$ given a third $C$ then $p(A|B, C) = p(A|C)$. For Gibbs sampling this means that in order to sample
variables in set $A$ we need only know the values of variables in set $C$ and do not need to know the values of variables in set $B$. In turn this implies that $x \in A$ and $y \in B$ may be sampled in parallel given $C$ without a synchronisation step.

The question now is how to identify enough conditional independencies in a model to permit a useful level of parallelisation. The Chromatic Gibbs Sampler of Gonzalez et al. (2011) develops one strategy for doing so. Colouring a graph is equivalent to partitioning the vertices into sets (colours), such that no two members of the same set are neighbours. The chromatic sampler is based on the observation that if the undirected dependency graph of a probability model is coloured the variables belonging to each colour are conditionally independent given the remaining variables, and so may safely be sampled in parallel. Furthermore, after all variables of a particular colour have been sampled processors may exchange messages with each other synchronously, rather than serially.

This chapter provides a method that is superior to the Chromatic Gibbs Sampler in a class of graphs for which it is possible to match colours with one another in order to create multiple queues of work that may be distributed. Synchronisation is only required at the end of a work queue, hence reducing the number of synchronisation steps required significantly. Section 6.3 demonstrates that Gibbs samplers based on this strategy respect conditional independence properties. Repeated graphs are defined in section 6.2, with an illustrative example using a random array provided in section 6.4. The random array model forms the template for the application of our strategy to parallel sampling of the Indian Buffet Process based matrix factorisation model of chapter 5.

### 6.2 Repeated graphs and their colouring

**Definition** Let a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be split into multiple subgraphs $\{\mathcal{G}_s = (\mathcal{V}_s, \mathcal{E}_s)\}_{s=1}^{S}$ such that: $\{\mathcal{V}_s\}$ is a partition of $\mathcal{V}$; and each subgraph $\mathcal{G}_s$ is isomorphic to a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Let us index the vertices as $Z_{s,j} \in \mathcal{V}_s$ where the second index specifies the vertex $Z_j \in \mathcal{V}$ to which $Z_{s,j}$ corresponds (if there is more than one possible bijection, pick one and stick to it). If $Z_{s,j} \sim Z_{s,l}$ only if either $j = l$ or $s = t$ then we call $\mathcal{G}$ a repeated graph comprising $S$ repetitions of protograph $\hat{\mathcal{G}}$. (For clarity the case $s = t, j = l$ would correspond to a loop, which is not permitted since the conditional dependence of a random variable on itself makes little sense.)

Note that for a given model we may find that the model as a whole does not represent a repeated graph, but that some subset of the model does. In such a case the results of this chapter can be used for accelerating sampling of the repeated graph portion alone.
We could describe such models with plate diagrams in which vertices within a plate were perhaps connected to the corresponding vertices at different plate indices. For an example of a repeated graph with diagrams the reader should skip briefly to section 6.4.

### 6.2.1 Size of a repeated colouring

**Proposition 6.2.1.** Suppose the repeated graph $\mathcal{G}$ comprises $S$ repetitions $\mathcal{G}_1, \ldots, \mathcal{G}_S$ of the protograph $\mathcal{G}$. Let $\hat{c}$ be a $k$-colouring of the vertices of $\mathcal{G}$ such that $\hat{c} : \mathcal{V} \rightarrow \{1, \ldots, k\}$. We can extend the $k$-colouring $\hat{c}$ of $\mathcal{G}$ to a $k\lceil S/k \rceil$-colouring of $\mathcal{G}$.

**Proof.** Let $\pi_s$ be the cyclic permutation of $1, \ldots, k$ by $s$ steps to the left. Let the colouring of $\mathcal{G}$ be given by $c(Z_{t,j}) = k \cdot (t/k) + \pi_{t \% k}(\hat{c}(Z_j))$ where, to be clear, $t/k$ is the quotient of integer division and $t \% k$ is the remainder. For any, $t, j, s, l$ such that $Z_{t,j} \sim Z_{s,l}$ either $t = s$ or $j = l$ since $\mathcal{G}$ was a repeated graph. Now if $t = s$ we know that the graph $\mathcal{G}_t$ is isomorphic to $\mathcal{G}_s$ and since $c$ is just a permutation and increment of $\hat{c}$ it is a valid colouring of $\mathcal{G}_t$. If instead $j = l$ then wlog supposing $t < s$ we have that $c(Z_{t,j}) \neq c(Z_{s,j})$ (whether or not $Z_{t,j} \sim Z_{s,j}$) since if $t/k < s/k$ then $c(Z_{t,j}) = k(t/k) + l < k(s/k) + m = c(Z_{s,j})$ and $|m - l| < k$ while if $t/k = s/k$ then $c(Z_{t,j}) - t/k = \pi_t(\hat{c}(Z_j)) \neq \pi_s(\hat{c}(Z_j)) = c(Z_{s,j}) - t/k$ since $0 < s - t < k$ and $\pi$ was a cyclic permutation of $1, \ldots, k$. \[\square\]

Proposition 6.2.1 also provides a way to upper bound the chromatic number of a repeated graph $\mathcal{G}$ by solving the smaller problem of colouring its protograph $\mathcal{G}$, the sparser the connections between repetitions, the looser the bound will be. Having an optimal colouring is not necessary. However, the smaller the colouring the greater the parallelisation opportunity.

An implication of this proposition is that the Chromatic Gibbs sampler of Gonzalez et al. (2011) would require $k\lceil S/k \rceil$ synchronisation steps in any sampling scheme on a repeated graph. We will show next that by constructing a set of palettes on a repeated graph that it is possible to achieve a scheme with just $k$ synchronisation steps.

### 6.3 The Palette Sampler

A strategy for parallel Gibbs sampling requires the identification of appropriate groups of variables. We first define a criterion for such groups, which we call *palettes*. It is then demonstrated that palettes preserve important conditional independence properties through use of colourings, thus enabling parallelisation.

Finding an optimal vertex colouring is, in general, an NP-complete problem (Karp, 1972). Hence, at first glance, the conditions that must be met to construct a palette could render
proporation 6.3.1 impractical. However, for repeated graphs, which include many time-series models, spatial models and longitudinal models it is possible to straightforwardly construct a useful palette.

6.3.1 Palettes

Let \( G = (V, E) \) be a graph. If \( P = \{P_i\} \) is a set such that: i) \( P_i \subset V \); ii) if \( i \neq j \) then \( P_i \cap P_j = \emptyset \); and iii) if \( x \in P_i \) and \( y \in P_j \) then \( x \approx y \); we call \( P \) a palette and its elements, \( P_i \), queues.

Note that the definition does not require that the palette represents all vertices, i.e. that \( \bigcup_i P_i = V \). (We will later find a set of palettes which does.)

6.3.2 Sampling from a palette is parallelisable

To sample a palette in parallel is to distribute the queues \( P_i, i = 1, \ldots, N \) among the processors. Each processor can then serially samples the random variables within a queue. When all sampling is complete, processors synchronise by exchanging messages consisting of the random variables just sampled. Note that sampling a single palette is not a full Gibbs iteration, unless \( \bigcup_{d \in s} d = V \), and we will have to split the full vertex set into a number of palettes.

The following proposition is that sampling a palette in parallel corresponds to sampling random variables from the correct conditional distributions.

**Proposition 6.3.1.** Let \( P \) be a palette with \( M \) queues on a graph \( G = (V, E) \) and let \( O = V - \bigcup_i P_i \), that is the set of vertices not contained within the palette. The following holds, \( p(P|O) = \prod_i p(P_i|O) \).

**Proof.** For a palette \( P \) Consider the joint conditional probability of the palette \( p(P|O) \).

\[
p(P|O) = p(P_1, \ldots, P_n|O) \\
= p(P_i|O, P_1, \ldots, P_{i-1}, P_{i+1}, \ldots, P_N) p(P_1, \ldots, P_{i-1}, P_{i+1}, \ldots, P_N|O) \quad (6.1)
\]
Since $\mathcal{P}$ was a palette for any $x \in \mathcal{P}_i$ then for any $y \in \bigcup_{j \neq i} \mathcal{P}_j$ we have that $x \sim y$. Hence the Markov blanket (Pearl et al., 1989) of $x$ is contained in $(\mathcal{P}_i - x) \cup O$ and so,

$$p(x|O, \mathcal{P}_1, ..., \mathcal{P}_{i-1}, \mathcal{P}_{i+1}, ..., \mathcal{P}_N, \mathcal{P}_i - x) = p(x|O, \mathcal{P}_i - x)$$

After expanding the conditional for $\mathcal{P}_i$ in terms of its members we see this means that:

$$p(\mathcal{P}_i|O, \mathcal{P}_1, ..., \mathcal{P}_{i-1}, \mathcal{P}_{i+1}, ..., \mathcal{P}_N) = p(\mathcal{P}_i|O)$$

As $i$ was selected arbitrarily, by repeated use of the same argument (6.1) becomes:

$$p(\mathcal{P}|O) = \prod_{i}^N p(\mathcal{P}_i|O) \tag{6.2}$$

We can use this result to propose a palette sampler by forming palettes, distributing the queues from the palette to different processors and then, in parallel, sampling the random variables within each queue serially.

### 6.3.3 Construction of palettes on repeated graphs

The following proposition establishes that we can split the vertices of a repeated graph into a set of palettes.

**Proposition 6.3.2.** Given the repeated graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of $S$ repetitions of a protograph $\hat{\mathcal{G}} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$ for which we have a $k$-colouring, $\hat{c}$. There exists a set of palettes $Q = \{\mathcal{P}^s\}$ such that $\bigcup \mathcal{P}^s$ is a partition of $\mathcal{V}$.

**Proof.** For $j = 1, \ldots, |\hat{\mathcal{V}}|$ and $s = 1, \ldots, k$, with $\pi_r$ a permutation of $1, \ldots, |\hat{\mathcal{V}}|$ by $r$ steps right set

$$\mathcal{P}^s_j = \{Z_{t, \pi_r(j)} | t = k(s-1) + 1, \ldots, \min(k(s-1) + \lfloor S/k \rfloor, S)\}.$$ 

For fixed $j$ we see that $\mathcal{P}^s_j \subset \mathcal{V}$ for all $s$. While for fixed $j$, $q \neq r$ that for $Z_{a,b} \in \mathcal{P}^j_q$ and $Z_{c,d} \in \mathcal{P}^j_r$ because $k \leq |\hat{\mathcal{V}}|$ we see that $b = \pi_q(j) \neq \pi_r(j) = d$. Whereas if $a = c$ then $q = r$, by the definition of $\mathcal{P}^s_j$, and this is false by assumption. Since $\mathcal{G}$ was a repeated graph and $a \neq c$ and $b \neq d$ we have that $Z_{a,b} \sim Z_{c,d}$, and also that $\mathcal{P}^j_q \cap \mathcal{P}^j_r = \emptyset$. Hence $\mathcal{P}^j$ is a palette.
For given $Z_{t,j}$ we can find $s,l$ such that $Z_{t,j} \in \mathcal{P}_s^l$ by identifying the $s$ such that $k(s-1) + 1 \leq t \leq \min(k(s-1) + \lfloor S/k \rfloor, S)$ and taking $l = \pi_{-s}(j)$ hence $\cup_j \mathcal{P}_s^l$ is a partition of $\mathcal{V}$.

For the palette constructed in proposition 6.3.2 the number of synchronisation steps required is $|\hat{\mathcal{V}}|$ per iteration. Since by proposition 6.3.1 we know that we can sample the palette $\mathcal{P}_s^l$ in parallel conditional on the other palettes $\mathcal{O}^l = \cup_{l \neq j} \mathcal{P}_s^l$. There are $|\hat{\mathcal{V}}|$ such palettes and we must synchronise once after each palette.

We have shown, at least for repeated graphs, that the Palette Sampler requires $|\hat{\mathcal{V}}|$ synchronisation steps as opposed to $k \lceil S/k \rceil$ for the Chromatic Sampler on a repeated graph coloured as per proposition 6.2.1 (which if $k = \chi(\mathcal{G})$ and the repetitions are fully connected will be the best case for $\chi(\mathcal{G})$). The classic Gibbs Sampler requires $|\mathcal{V}|$, and as such would not be worth parallelising in this fashion.

### 6.4 Examples

#### 6.4.1 Illustrative example with a random array

Suppose that we have a random $5 \times 3$ array, $Z$ in which each row is jointly dependent, that is $(Z_{t,1}, Z_{t,2}, Z_{t,3}), t = 1, \ldots, 5$ are dependent upon one another, and each column, $(Z_{1,j}, Z_{2,j}, Z_{3,j}, Z_{4,j}, Z_{5,j}), j = 1, \ldots, 3$, are jointly dependent. Also, conditional on the other variables in the same row and column, a cell is independent of the rest of the array, e.g.

$$p(Z_{t,j}|Z - Z_{t,j}) = p(Z_{t,j}|\{Z_{t,l}\}_{l=1}^3 \cup \{Z_{t,j}\}_{t=1}^5 - Z_{t,j})$$

For example if the rows represent time and the columns individuals, this would yield a model for cohorts varying over time. If the rows represent people and the columns products this yields a model where responses within individuals are dependent and responses within products are dependent. One possible Bayesian network giving rise to such a dependency structure for 3 locations over 5 time points is depicted as a plate diagram in Figure 6.1.
Fig. 6.2 Illustration of propositions 6.2.1 and 6.3.2 on the example model.

After moralisation (Lauritzen and Spiegelhalter, 1988) to convert the directed links we can colour the corresponding array as shown in Figure 6.2a, where our colour labels correspond to \((b\ r\ g\ o\ a\ t)\).

The construction of the palettes as per proposition 6.3.2 leads to the following work queues,

\[
\mathcal{P}_1^1 = \{X_{1,1}, X_{2,1}\}, \quad \mathcal{P}_1^2 = \{X_{1,2}, X_{4,2}\}, \quad \mathcal{P}_1^3 = \{X_{5,3}\} \\
\mathcal{P}_2^1 = \{X_{1,2}, X_{2,2}\}, \quad \mathcal{P}_2^2 = \{X_{3,3}, X_{4,3}\}, \quad \mathcal{P}_2^3 = \{X_{5,1}\} \\
\mathcal{P}_3^1 = \{X_{1,3}, X_{2,3}\}, \quad \mathcal{P}_3^2 = \{X_{3,1}, X_{4,1}\}, \quad \mathcal{P}_3^3 = \{X_{5,2}\}
\]

and these are depicted in Figure 6.2b.
6.5 Application to nonparametric matrix factorisation

The limitations inherent to current factorisation methods for neuroimaging invite a Bayesian nonparametric approach capable of integrating dimensionality estimation, dimension reduction and model fitting steps. The model in chapter 5 we built on Knowles and Ghahramani, 2007 in utilising the Indian Buffet Process (IBP, Ghahramani and Griffiths, 2005) as a prior on a binary feature activation matrix $B_{\infty \times T}$. The IBP distribution ensures that for finite $T$ the number of nonzero rows is finite and random, allowing us to account for uncertainty in the intrinsic dimensionality. The matrix factorisation becomes (with $\circ$ denoting the Hadamard product)

$$Y = W(B \circ S) + E$$ (6.3)

Here $S$ plays a role determining how active features are while $B$ determines if they are active. Discussion of modelling assumptions are beyond the scope of this current chapter, which instead considers computational aspects of the most challenging step, inference on $B$.

6.5.1 The Indian Buffet Process

The generative story for IBP matrices was set out in Ghahramani and Griffiths, 2005. Consider the vector $m = B1$. We call feature $k$ shared if $m_k > 1$ and unique if $m_k = 1$. Let $K^+$ be the number of nonzero entries in $m$, and $K^{++}$ be the number of shared features. To draw from the two-parameter IBP distribution (Knowles and Ghahramani, 2007) we run through observations $1, \ldots, T$. The $t^{th}$ observation first activates features which have previously been activated with probability $m_k / (\beta + t - 1)$. The $t^{th}$ observation then activates a random number $j \sim \text{Poi}(\alpha \beta / \beta + t - 1)$ of new features.

Note that the $T$ observations are exchangeable. Also, in the matrix factorisation setting the ordering of the rows is irrelevant to the likelihood and thus indeterminate. Gibbs sampling from the posterior distribution of the IBP matrix $B$ involves two phases: sampling the shared and the unique feature activations for each observation. Sampling shared feature activations is by far the more computationally challenging element.
Fig. 6.3 Nonzero portion of an example IBP matrix, $K^+ = 20 \times T = 10$. Shared features rows are in the top half of the image, and unique feature rows in the bottom half

### 6.5.2 Previous sampling approaches

Knowles and Ghahramani (2007) considered sequential cycling through all $K^{++} \times T$ entries $B_{k,t}$ in order to resample feature activations in a manner equivalent to:

$$\mathbf{B}_{k,t} \sim \text{Bern}\left(\left(1 + \exp\left(-\ln\left(l_{k,t} p_{k,t}\right)\right)\right)^{-1}\right)$$  \hspace{1cm} (6.4)

Where $l_{k,t}$ is a likelihood ratio expressing the conditional probability of observing image $Y_{:,t}$ if feature $k$ is activated versus deactivated. That is $l_{k,t} = g_k(r_{k,t})$, which involves calculating the residual $r_{k,t} = Y_{:,t} - W(S \circ B)_{-k,t}$, where the $-k$ notation means all columns except the $k^{th}$. Thus $r_{k,t}$ is expensive to compute for large $V$. Furthermore note that calculation of $r_{k,t}$ is dependent on the other feature activations for image $t$.

On the other hand the contribution to the activation probability from the prior $p_{k,t}$ is dependent on the number of other images which have activated this feature, by virtue of
the IBP’s strong-become-stronger attribute. In particular Knowles and Ghahramani (2007) showed that \[ r_p = \frac{m_{k,t}}{b + 1 - m_{k,t}} \] where \( m_{k,t} = m - B_{:,t} \), that is ignoring the \( t^{th} \) column of \( B \).

Together these attributes imply that \( B_{k,t} \) is conditionally dependent on all entries in the same column \( B_{k,s} \) and the same row \( B_{l,t} \). This is irrelevant for a traditional serial Gibbs sampler but of considerable importance if we wish to parallelise our sampling algorithm.

Doshi-Velez et al. (2009) proposed two parallelisation approaches. The first approach ignored the conditional dependency structured and sampled all entries \( B_{k,t} \) in parallel (up to the number of available cores). The second approach used the first to generate the proposal for an Metropolis-Hastings step, ensuring a valid MCMC transition kernel. While the first approach may be useful, it is not suitable for scientific neuroimaging work as the precision of the method could be questioned in a discipline subject to increasing scrutiny of probabilistic results. The second approach would be too slow for our application due to high MH rejection rates. Hence we sought an approach capable of fully utilising computational resources while providing valid MCMC results.

6.5.3 Improved parallelisation

The first step in parallelisation comes from observing that if we take two entries \( B_{k,t} \) and \( B_{l,s} \) then if \( t \neq s \) and \( k \neq l \) then they are conditionally independent. This, combined with the exchangability of rows of \( B \) means that we can employ proposition 6.3.2 for constructing a parallelisation scheme. To do so we divide our data into \( H \) chunks such that the \( c^{th} \) chunk is the temporal (column-wise) concatenation of a number of our individual datasets. Let \( I_c \) be a set containing the indices belonging to the \( c^{th} \) chunk. We then distribute each chunk to separate computational nodes. If each node then operates serially on a separate feature, \( k \), we have applied proposition 6.3.2 and no conditional dependencies are violated.

The second parallelisation for shared features comes from observing that for fixed \( k \) and \( a \leq s \leq b \) then the residual vectors \( r_{k,s} \) are unaffected by activations sampled when \( t \neq s \). Thus we calculate batches of residuals in parallel \( R_{k,s} = Y_{:,c} - W(S \circ B)_{:,k} \). This linear algebraic problem is well suited to GPU hardware, as is the subsequent evaluation of likelihood contributions \( g_k(R_{:,s}) \). Afterwards we sequentially sample the \( B_{s,k} \) using our precomputed likelihood ratios for \( a \leq s \leq b \) which is a series of \( b - a \) simple calculations of \( p_{s,k} \).

Combining these two steps we split the data into chunks whose size is determined by the capacity of our GPUs. Each node is assigned a different feature \( k \) to work on. We then use the second observation to parallelise away computational bottlenecks in the computation of residuals for the images contained in each chunk. After each node has completed its task, messages in the form of a vector \( B_{k,l} \) are exchanged between nodes to ensure synchronisation.
6.5 Application to nonparametric matrix factorisation

Fig. 6.4 Layout of resampling work for shared features in \( \mathbf{B} \). Here we are using 3 compute nodes (shown in different colours) and there are \( K^{++} = 5 \) shared features to resample. Each step is equivalent to parallel sampling of a palette. Hence 5 steps are required for a Gibbs pass through all shared features \( \mathbf{B}_{k,t} \) (shown here transposed).

Each node then sets \( k \leftarrow k + 1 \) and begins work on the next feature unless \( k = K^{++} \) in which case \( k \leftarrow 1 \), until each node has cycled through all \( K \) features. This is illustrated in Fig. 2. As this scheme works for arbitrary \( K^{++} \) the process for resampling shared features in parallel adapts as the matrix factorisation dimension changes across MCMC iterations. If we have fewer nodes than chunks, then after \( \mathbf{B}_{k,t} \) has been resampled for all \( t \) in memory our nodes set new chunks into their respective GPUs and proceed as before until all sampling has been completed.

### 6.5.4 Example results

It is difficult to provide a perfect comparison between parallelised and non-parallel cluster-scale codes due to the effects of choice of dataset size and hardware capabilities. However, it is easy to show that our GPU accelerated parallel code performs much more efficiently than is possible using a serial sampler.

We consider application to a dataset consisting of \( N = 30 \) individuals each providing \( T_C = 193 \) fMRI images of dimension \( 61 \times 73 \times 61 \). This corresponds to the dimensions of realistic fMRI datasets.

In order to determine best-case performance for a serial sampler we defined a function equivalent to \( g_k(\mathbf{r}_{k,t}) \) for fixed \( k, t \) and \( \mathbf{r}_{k,t} \), which benefits our estimate of serial speed by not requiring any array slice memory access. The kernel was written in Julia and compiled prior to speed testing. We evaluated the kernel on a 12 core compute node, allowing implicit
Parallel Gibbs Sampling in Repeated Graphs

BLAS multi-threading as any serial sampler would undoubtedly take advantage of similar capabilities. The kernel took on average 0.029s to execute over 1000 repetitions.

A single Gibbs iteration for the shared $B_{k,t}$ requires $TK^{++}$ evaluations of $g_k(r_{k,t})$. Hence for our neuroimaging scale dataset $T = 30 \cdot 193$ and $K^{++} = 50$. As such an optimistic estimate of the serial execution time for this step would be around 8430s.

We then ran our parallel code, also written in Julia and CUDA C, using six GPU equipped compute nodes. This kernel did more work by resampling all shared $B_{k,t}$ and the corresponding $S_{k,t}$ rather than simply calculating $g_k(r_{k,t})$. The kernel took on average 35.49s over 10 repetitions. This time also includes message passing between nodes for synchronisation. Hence we conservatively estimate the speedup on this scale problem to be 237 fold. This shows the parallelisation to be efficient as the ratio of GPU/CPU cores involved is 256, a near linear scaling. Similar computational advantages persist under rank-one updates of residuals as these are also amenable to GPU-based linear algebra.

6.6 Discussion

In this chapter we have considered improved ways of taking advantage of conditional independence properties in graphical models. These results provide a way to speed up inference algorithms through better orderings of variables. They do not require further heuristics or the introduction of Metropolis steps.

It was shown that repeated graphs represent a substantial class of models. Notably we provided a template example of a random array model supported row and column dependencies. In a concrete example identified as a bottleneck elsewhere in the literature, the method provided a two-order of magnitude speed up. The efficiency gain in our example is a function of the number of colours in the example at hand. Generally the more repetitions $S$ of the protograph in a random array model, the greater the benefits.

On a theoretical level, the conditional independence properties identified here may go some way to explaining the good performance of asynchronous Gibbs sampling discussed in Doshi-Velez et al. (2009). Asynchronous Gibbs being the sampling of all variables in parallel without regard to dependence properties. Under the analysis presented here if the number of rows and the number of columns are both much larger than the number of cores available then there is a substantial chance under random selection of indices that the variables being sampled in parallel are conditionally independent of one another.

The ability to make parallelisation gains in the fashion of this paper may be of particular use when fitting models for scientific purposes. That is, where stronger guarantees on the asymptotic sampling distribution are required.
We primarily considered a strategy for parallelising two-dimensional array models. However the idea extends to higher dimensional tensor models where variables sharing the same index are dependent and variables not sharing an index are conditionally independent of one another.
Chapter 7

Bayesian nonparametric matrix factorisation for neuroimaging

7.1 Purpose and background

We now apply the nonparametric matrix factorisation model of chapter 5 to fMRI neuroimaging data. Our primary aim is to demonstrate its ability to produce sensible results at scale; and it was the scale of neuroimaging data that warranted the development of an efficient approach to spatial inference in chapter 4 and to parallelisation in chapter 6.

The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) study Shafto et al. (2014) is concerned with the traits associated with healthy ageing. Stage 2 data include fMRI, MEG, and demographic data acquired from nearly 700 individuals ranging from 18-87 years of age Taylor et al. (2017). The study captured fMRI data under three different conditions including resting state, film watching and sensorimotor tasks.

We focus in this chapter on the film watching task in which subjects watched an 8 minute edit of the Hitchcock drama *Bang! You’re Dead* (1954). We focus on this experimental condition in order to replicate the analysis of Campbell et al. (2015), which provides a benchmark ICA analysis of data from the Cam-CAN study. The primary question of that paper was the nature of the relationship between age and synchronisation of brain activity across subjects; in particular by analysis of data from $N^* = 218$ Cam-CAN participants. The free-form nature of the stimulus necessitates a relatively unconstrained analysis in terms of the time courses and brain regions involved. This is in contrast to, say, an experimental block design where the time-course(s) is assumed within the design matrix of multiple linear regression. Hence matrix factorisation models are well suited to the requirements of the analysis.
Spatial modelling of fMRI data is well motivated by the fact that the spatial resolution of fMRI is finer than typical parcellations of the brain into functional regions. Hence we should expect neighbouring voxels to often share similar values. Incorporating this knowledge into our model should reduce the need for spatial smoothing of fMRI data as a preprocessing step. We saw in chapter 5 that in a case where the ground truth was known modelling auto-correlation produced a superior identification of the latent basis vectors.

Furthermore current approaches to ICA in neuroimaging rely on eigendecomposition of covariance matrices for dimensionality estimation (see section 2.5). This may be problematic for a number of reasons outlined in chapter 3 including high-dimensionality with low sample size or low signal-to-noise ratio. The application of these methods to produce point estimates of dimensionality does not carry over uncertainty from PCA to ICA stages of analysis. The model used here provides an alternate route to factorisation unifying what would otherwise be the PCA and ICA stages of a model.

7.2 Data

7.2.1 Size and preprocessing

This chapter presents only an analysis of \( N = 30 \) subjects from the larger Cam-CAN dataset. Each subject contributed \( H = 193 \) images for a total of \( T = HN = 193 \times 30 = 5790 \), which after preprocessing had a spatial dimension of \( V = 61 \times 73 \times 61 = 271633 \). The data were preprocessed as in (Campbell et al., 2015) apart from high-pass filtering, which we implemented separately, and spatial smoothing where we used available data that had been smoothed with a 1mm FWHM kernel rather than 8mm. We discuss the size limitation of our data subset in section 7.7.1. A different analysis of a larger dataset is undertaken in chapter 9.

Further pre-processing steps undertaken in the work presented here were: subject-specific centring of data vectors (subtracting each subject’s ‘mean brain’ from their data); unit normalisation of the data (considered as spatial vectors); high-pass filtering of the data by attenuating voxel-wise time-series of less than 0.004Hz (roughly, this attenuated four frequencies which would not have been able to complete two cycles during the movie); zeroing of voxels more than two-neighbours away from a voxel in one of the cortical or sub-cortical areas of the Harvard-Oxford atlas (we found this step to be useful in removing some artefacts occurring outside the brain).
7.3 Prior

Table 7.1 states the parameters of the prior used in the main analysis of this chapter which is reported in section 7.6.2. For the most part we used relatively vague prior parameters.

An exception was made with respect to the aim of maintaining a tractable latent dimensionality. This was because under a vague prior on the observation noise ($\sigma^2$, prior (1,1)) and IBP parameters ($\alpha$, $\beta$ each with (1,1) priors), we found that a very large number of basis vectors was found under the nonparametric model. This judgement of ‘very large’ is made relative to the usual estimates, e.g. in Campbell et al. (2015) $\hat{K} = 56$ components were identified in a larger dataset. With respect to the noise level, this occurs because when the observation noise is very low, deviations from the model tend to be better explained by the presence of an additional latent basis vector than by observation noise (c.f. section 5.3.4).

As an example of this effect, figure 7.1 shows the trajectory of $K$ through the first few thousand iterations under a vague prior on the noise and IBP parameters. We see that $K$ increases roughly logarithmically with MCMC iterations. This behaviour is not in itself unreasonable as the observed dimension $V$ of the brain images is still much larger than $K \approx 180$. However ‘large’ latent dimensionality poses a number of problems. Computation becomes intractable meaning the chain could not be run to convergence. That is, a single
Fig. 7.1 Rapid increase in basis vectors under uninformative prior. While the trajectory suggests an asymptote and hence convergence may be found eventually, for neuroimaging analysis the computational requirements may be infeasible.

run could take months with no certainty of convergence. Even if we cared to do this as a proof of concept we would not recommend such a prior for use in further analyses. The results become hard to interpret, since large $K$ can lead to many subject-specific components. Furthermore the combinations of many non-artefactual components become complex to consider simultaneously when attempting to understand the implications of the results for brain function.
Table 7.1 Prior parameters used in the analysis. Refer to section 5.2 for their interpretation. Spatial parameters have been given names \( \theta_1 = \phi \) and \( \theta_2 = \lambda \) for a spatial structure like that of section 4.4 with \( \Delta = \phi I + \lambda L \).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_\alpha )</td>
<td>1.0</td>
</tr>
<tr>
<td>( e_\alpha )</td>
<td>1.0</td>
</tr>
<tr>
<td>( f_\beta )</td>
<td>25.0</td>
</tr>
<tr>
<td>( e_\beta )</td>
<td>25.0</td>
</tr>
<tr>
<td>( f_\nu )</td>
<td>0.1</td>
</tr>
<tr>
<td>( e_\nu )</td>
<td>0.1</td>
</tr>
<tr>
<td>( m_\phi )</td>
<td>0.0</td>
</tr>
<tr>
<td>( n_\phi )</td>
<td>0.01</td>
</tr>
<tr>
<td>( m_\lambda )</td>
<td>0.0</td>
</tr>
<tr>
<td>( n_\lambda )</td>
<td>0.01</td>
</tr>
<tr>
<td>( f_\sigma )</td>
<td>7863.7754</td>
</tr>
<tr>
<td>( e_\sigma )</td>
<td>1.0</td>
</tr>
<tr>
<td>( m_\tau )</td>
<td>1.0</td>
</tr>
<tr>
<td>( n_\tau )</td>
<td>10.0</td>
</tr>
</tbody>
</table>

When we placed a informative prior on the noise so as to form a soft floor on the noise level to which the MCMC inference was able to converge (at least as far as one is able to tell with a single chain). This prior worked to limit the accuracy of the model, so that deviations from the model could be well-explained by noise rather than by the existence of further latent basis vectors. This is visible from the density used in the acceptance ratio for proposals of unique features (equation 5.4) implying that as the noise level becomes large, the likelihood term tends towards unity while the proposal distribution will generally favour more compact models.

This possibility arises because the full conditional distribution on the noise level in the Gibbs sampler (see 5.3.6) is an Inverse Gamma distribution with mean \( (f_0 + SSR)/(e_0 + VT/2 - 1) \) where \( SSR \) is the sum of squared residuals in the model with prior rate \( f_\sigma \) and prior shape \( e_\sigma \). Hence a large prior rate has the same effect on the full conditional of the noise as does a large sum of squared residuals (i.e. a poorly fitting model).

In particular we can set \( e_\sigma = 1, f_\sigma = c/2 \) and suppose that \( SSR \) tends to zero then the mean of the full conditional looks like \( c/TV \). We can choose \( c \) by considering the mean sum
of squares of the data,

$$MSS = \frac{1}{VT} \sum_{n=1}^{T} \sum_{v=1}^{V} y_{n,v}^2$$  \hspace{1cm} (7.1)$$

By further setting \( f_\sigma = dVT \cdot MSS/2 \) the conditional mean becomes \( E[\sigma^2|\cdot] \approx d \cdot MSS \). As a result, \( d \) can be interpreted as a proportion of the mean sum of squares to be accounted for by the noise model. This is reminiscent of the ‘proportion of variance explained’ in discussions of Principal Components Analysis. Since the variance of the noise full conditional is (roughly) \( d \cdot MSS/(V \cdot T) \) with both \( d \) and \( MSS \) less than unity and \( TV \) large, the variance around the mean will be very small. We speak of this as a floor, since if the sum of squared errors is much bigger than the prior rate \( SSE \gg f_0 \) then the prior rate will not have much influence on the mean of the full conditional. In the analysis presented in this chapter we used \( d = 0.05 \) so that the modelled noise should represent at least 5% of the variance in the data.

### 7.4 Ranking of dimensions

In order to summarise the results of inference, we need a method to select the dimensions of interest for visualisation and analysis. A common measure used in conventional ICA models is the percentage of variance explained by each dimension (for instance this is implemented in FSL). This is derived from PCA, since the ICA mixing matrix is an orthogonal rotation of the principal component matrix factorisation. The reader will recall that the variance of each of the principal components divided by their sum represents the fraction of variance in the original data accounted for by that component.

In the matrix factorisation pursued here neither the temporal nor the spatial factors are constrained to be orthogonal. Hence an attempt to use the same measure for this method would result in double counting when calculating the percentage of variance explained using latent dimensions in isolation. For this reason we provide three methods for ranking of the components.

### 7.5 Metrics

#### 7.5.1 Synchronisation

One way to identify biologically plausible components in an fMRI dataset containing some external stimulus would be to rank components by how closely the time-course of activation
was replicated across subjects. The intuition being that if certain brain regions are reliably recruited in response to stimulus then the pattern of activation should be similar across subjects. Clearly this metric will be of less use for the identification of components which do not vary much in response to stimuli.

Let the expectation of the time course of components for dimension \( k \) belonging to the \( n^{th} \) subject in the dataset be defined as \( \bar{x}_{k,n} = \mathbb{E}_q [X^{(n)}_{k,:}] \) (a row vector). Define the group mean \( \bar{x}_k = \frac{1}{N} \sum_{n=1}^{N} x_{k,n} \). We define the inter-subject synchronisation for the \( k^{th} \) feature as:

\[
c_k = \frac{\sum_{n=1}^{N} \bar{x}_{k,n} \bar{x}'_k}{||\bar{x}_{k,n}||_2 ||\bar{x}_k||_2}
\]

(7.2)

So that the synchronisation score is the cosine similarity with the group mean time-course for the \( k^{th} \) dimension.

This metric is of interest for at least two reasons: synchronisation was the subject of previous study (Campbell et al., 2015) and holds scientific relevance through the generalisation of patterns of activation across many subjects; secondly, matrix factorisation methods are known to pick up components related to events not of scientific interest such as head movement, and such components should \textit{a priori} have a low level of inter-subject synchronisation.

### 7.5.2 Smoothness

The intuition for this metric is that spatial maps with larger ‘clusters’ of activation will typically be smoother than spatial maps with small ‘clusters’ of activation. This is because smoothness implies only small differences in value between nearby locations. Larger regions of activation are typically seen to relate to meaningful structures in the brain, while many small regions of activation are often indicative of artefactual components.

In order to quantify smoothness, let \( L \) be the Laplacian matrix for the DCT based covariance structure as discussed in chapter 4. With the \( k^{th} \) basis vector as \( w_k \in \mathbb{R}^V \) we define the smoothness score for the \( k^{th} \) latent dimension as:

\[
g_k = w_k'Lw_k
\]

(7.3)

In terms of fMRI data, this represents a weighted sum of the squared differences between a voxel and its neighbours. Another way to think of \( g_k \) is as \(-2\) times the exponent in the prior likelihood of \( w_k \), which is the only element of that distribution to change for different \( k \) and hence forms a suitable quantity for comparison.
7.5.3 Magnitude

It may be informative to have a measure of how much each latent dimension contributes to the magnitude of the reconstructed data. Neither the $w_k$ or $X_k$ is strongly normalised (e.g. unit normalised), but whose magnitude may vary subject to a penalty imposed by the model. Therefore we must include a value from each to perform such a calculation. We define the magnitude score as the product, for the $k^{th}$ component of the $L^2$ norm of the basis vector with the $L^2$ norm of the components, normalised by the mean of the same across all latent dimensions.

$$m_k = \frac{W_k' w_k \cdot X_k' X_k}{\frac{1}{K} \sum_{i=1}^{K} W_i' w_i \cdot X_i' X_i}$$

(7.4)

A higher value of $m_k$ indicates that the component makes a larger (in the $L^2$ sense) contribution to the magnitude of the reconstruction. It is not possible to provide a percentage of variance explained calculation for each latent dimension, as is done in methods derived from PCA, because the basis vectors are not constrained to be orthogonal. We normalise $m_k$ here as otherwise the magnitude of the components can be very large or small and it is easier to read values close to unity.

7.6 Inference

7.6.1 Initialisation

We initialise the model by use of the K-means algorithm, regarding the data as a collection of spatial vectors to cluster. For reasons linked to the discussion in section 7.3, particularly the potential for a large number basis vectors, we initialise the model with a modest number, $K = 20$, of basis vectors. The initial component weights were estimated using the LASSO.

7.6.2 Results

Computation

We ran a Markov Chain Monte Carlo inference procedure with the above data and model for 15,000 iterations. The inference took on average 40.5 seconds per iteration, using 6 compute nodes equipped with GPUs; totalling a week long run. Due to the length of the inference process and its resource intensity we did not run simultaneous chains for comparison under the same prior.
Latent dimension

Figure 7.2 shows a trace plot of the number of latent dimensions in the matrix factorisation is plotted against the iteration in solid blue. We can see that after iteration 4,000 the chain settles on $K = 43$ features. We can also see several small ‘hairs’ on the trace plot, before iteration 5000, indicating iterations at which a new feature was activated (just for one subject) and then subsequently discarded. The lack of such ‘hairs’ after iteration 5000 is linked to the informative prior we imposed, which put a soft limit on the number of basis vectors.

Noise level

The dotted black line in figure 7.2 shows the inferred noise level. The corresponding observation noise level is stable at around $1.244 \times 10^{-5}$. This noise level is unreasonably large because it implies that the average norm of a brain image vector is then greater than unity, after the data vectors have been unit normalised. This could raise a concern that the reconstruction of the data $\hat{Y} \approx WX$ might be very poor. However, the mean squared error $1/(TV) \sum_{m,n} (\hat{Y} - WX)^2_{m,n}$ at the final iteration was $2.44 \times 10^{-6}$ implying that the reconstruction (as a whole) accounted for about 66% of the variance in the data.

Factors

We selected the latent components with the two highest and two lowest inter-subject synchronisation scores for visualisation in figures 7.3 and 7.4. The brains are displayed in axial slices, starting from the bottom of the brain (in the bottom left-hand corner) and proceeding vertically through the brain (in the columns of each image) to reach the top most slice (in the top right right-hand corner). Each slice is shown in the ‘radiological’ convention with the
right hand side of the brain on the left hand side of the image. The bar at the bottom of the image has a dual purpose, showing ± the largest absolute value of the image at the ends of the bar with the centre showing the colour of a zero value, while the maximum intensity of the bar (red) points to the right hand side of the brain. The numerical values of the extrema differ from component to component, but the visualisation process normalises these.

We do not threshold the images. This is for two reasons. First, we are interested in the characteristics of the model’s output. Excessive post processing could obscure this; while details of the relevant neuroanatomy are quite clear in the images. Where the anatomy is not clear, this is a helpful signal regarding artefacts. Second, as pointed out by Friston et al. (2002) in the context of regression models for fMRI, the notion of using the mechanics of z-statistic calculation on objects representing information about a Bayesian posterior to single out ‘significant’ activation would not be conceptually coherent.

In the case of the highly synchronised dimensions shown in figure 7.3 we see both a high degree of spatial auto-correlation with contiguous clusters of activation. We also note the fine degree of spatial resolution, for instance in figure 7.3c we can see a high resolution ‘fleur de lys’ in the precuneus. In contrast the dimensions with lower inter-subject synchronisation in figure 7.4 have spatial maps have a grainer character. This negative relationship is graphed in figure 7.5.

![Fig. 7.5 Synchronisation versus smoothness scores for each dimension in the final sample. More synchronised dimensions tend to be smoother.](image)

This qualitative judgement is quantitatively supported by Table 7.2 which shows the scores of the latent dimensions under the metrics described in section 7.5 alongside the ranks
7.6 Inference

(a) Feature 34 under axial slicing
(b) Subject specific time-series for dimension 34
(c) Feature 41 under axial slicing
(d) Subject specific time-series for dimension 41

Fig. 7.3 Spatial maps and associated time-courses. Features selected for highest synchronisation score.
Bayesian nonparametric matrix factorisation for neuroimaging

(a) Feature 24 under axial slicing

(b) Subject specific time-series for dimension 24

(c) Feature 15 under axial slicing

(d) Subject specific time-series for dimension 15

Fig. 7.4 Spatial maps and associated time-courses. Features selected for lowest synchronisation score.
of the components under the various metrics. In terms of synchronisation we see that the two most synchronised dimensions have scores on this metric of, 0.541 and 0.456 while the two least synchronised components have scores of and 0.181 and 0.178. While we will not delve deeply into psychological interpretation, a little discussion may be warranted. We can see that the more synchronised components load heavily and symmetrically on regions of cortex rather than on white matter or cerebrospinal fluid making them candidates for some interpretation. Whereas the less synchronised components show subcortical structure and a ‘halo’ effect suggestive of head motion in the case of dimension 15, and an artefact heavily influenced by a few individuals in the case of dimension 24. Note the different x-axis limits in the subject specific timeseries, with the less synchronised dimensions displaying a wider range of values.
Table 7.2 Latent dimensions scored

<table>
<thead>
<tr>
<th>K</th>
<th>Smooth</th>
<th>Smooth Rank</th>
<th>Mag</th>
<th>Mag Rank</th>
<th>Sync</th>
<th>Sync Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.827</td>
<td>23</td>
<td>0.821</td>
<td>16</td>
<td>0.203</td>
<td>33</td>
</tr>
<tr>
<td>2</td>
<td>3.251</td>
<td>39</td>
<td>0.806</td>
<td>17</td>
<td>0.212</td>
<td>22</td>
</tr>
<tr>
<td>3</td>
<td>3.13</td>
<td>34</td>
<td>0.851</td>
<td>14</td>
<td>0.24</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>2.301</td>
<td>11</td>
<td>0.899</td>
<td>12</td>
<td>0.361</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>1.521</td>
<td>1</td>
<td>2.8</td>
<td>3</td>
<td>0.321</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>2.037</td>
<td>5</td>
<td>2.772</td>
<td>4</td>
<td>0.21</td>
<td>26</td>
</tr>
<tr>
<td>7</td>
<td>2.968</td>
<td>29</td>
<td>0.935</td>
<td>11</td>
<td>0.233</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>2.51</td>
<td>15</td>
<td>1.085</td>
<td>9</td>
<td>0.239</td>
<td>13</td>
</tr>
<tr>
<td>9</td>
<td>2.025</td>
<td>4</td>
<td>3.176</td>
<td>2</td>
<td>0.21</td>
<td>25</td>
</tr>
<tr>
<td>10</td>
<td>1.861</td>
<td>3</td>
<td>2.28</td>
<td>6</td>
<td>0.255</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>2.461</td>
<td>13</td>
<td>1.27</td>
<td>8</td>
<td>0.195</td>
<td>39</td>
</tr>
<tr>
<td>12</td>
<td>2.167</td>
<td>7</td>
<td>3.735</td>
<td>1</td>
<td>0.19</td>
<td>41</td>
</tr>
<tr>
<td>13</td>
<td>2.375</td>
<td>12</td>
<td>0.786</td>
<td>19</td>
<td>0.379</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>2.851</td>
<td>25</td>
<td>0.804</td>
<td>18</td>
<td>0.208</td>
<td>28</td>
</tr>
<tr>
<td>15</td>
<td>2.273</td>
<td>10</td>
<td>2.511</td>
<td>5</td>
<td>0.181</td>
<td>42</td>
</tr>
<tr>
<td>16</td>
<td>2.648</td>
<td>18</td>
<td>0.709</td>
<td>22</td>
<td>0.204</td>
<td>31</td>
</tr>
<tr>
<td>17</td>
<td>2.7</td>
<td>20</td>
<td>1.395</td>
<td>7</td>
<td>0.209</td>
<td>27</td>
</tr>
<tr>
<td>18</td>
<td>2.942</td>
<td>28</td>
<td>0.969</td>
<td>10</td>
<td>0.205</td>
<td>29</td>
</tr>
<tr>
<td>19</td>
<td>2.709</td>
<td>21</td>
<td>0.837</td>
<td>15</td>
<td>0.226</td>
<td>17</td>
</tr>
<tr>
<td>20</td>
<td>2.719</td>
<td>22</td>
<td>0.857</td>
<td>13</td>
<td>0.211</td>
<td>24</td>
</tr>
<tr>
<td>21</td>
<td>2.698</td>
<td>19</td>
<td>0.758</td>
<td>20</td>
<td>0.215</td>
<td>20</td>
</tr>
<tr>
<td>22</td>
<td>3.296</td>
<td>41</td>
<td>0.624</td>
<td>25</td>
<td>0.199</td>
<td>37</td>
</tr>
<tr>
<td>23</td>
<td>2.561</td>
<td>16</td>
<td>0.667</td>
<td>24</td>
<td>0.218</td>
<td>19</td>
</tr>
<tr>
<td>24</td>
<td>3.161</td>
<td>36</td>
<td>0.565</td>
<td>29</td>
<td>0.178</td>
<td>43</td>
</tr>
<tr>
<td>25</td>
<td>2.847</td>
<td>24</td>
<td>0.581</td>
<td>28</td>
<td>0.201</td>
<td>35</td>
</tr>
<tr>
<td>26</td>
<td>3.03</td>
<td>32</td>
<td>0.602</td>
<td>26</td>
<td>0.193</td>
<td>40</td>
</tr>
<tr>
<td>27</td>
<td>2.48</td>
<td>14</td>
<td>0.592</td>
<td>27</td>
<td>0.262</td>
<td>8</td>
</tr>
<tr>
<td>28</td>
<td>2.875</td>
<td>26</td>
<td>0.557</td>
<td>30</td>
<td>0.202</td>
<td>34</td>
</tr>
<tr>
<td>29</td>
<td>2.581</td>
<td>17</td>
<td>0.542</td>
<td>31</td>
<td>0.222</td>
<td>18</td>
</tr>
<tr>
<td>30</td>
<td>3.601</td>
<td>43</td>
<td>0.498</td>
<td>36</td>
<td>0.252</td>
<td>11</td>
</tr>
<tr>
<td>31</td>
<td>2.936</td>
<td>27</td>
<td>0.531</td>
<td>32</td>
<td>0.255</td>
<td>9</td>
</tr>
<tr>
<td>32</td>
<td>1.73</td>
<td>2</td>
<td>0.727</td>
<td>21</td>
<td>0.311</td>
<td>7</td>
</tr>
<tr>
<td>33</td>
<td>2.271</td>
<td>9</td>
<td>0.677</td>
<td>23</td>
<td>0.442</td>
<td>3</td>
</tr>
<tr>
<td>34</td>
<td>2.054</td>
<td>6</td>
<td>0.443</td>
<td>43</td>
<td>0.541</td>
<td>1</td>
</tr>
<tr>
<td>35</td>
<td>3.207</td>
<td>37</td>
<td>0.491</td>
<td>37</td>
<td>0.204</td>
<td>32</td>
</tr>
<tr>
<td>36</td>
<td>2.972</td>
<td>30</td>
<td>0.498</td>
<td>35</td>
<td>0.196</td>
<td>38</td>
</tr>
<tr>
<td>37</td>
<td>3.262</td>
<td>40</td>
<td>0.529</td>
<td>33</td>
<td>0.213</td>
<td>21</td>
</tr>
<tr>
<td>38</td>
<td>3.155</td>
<td>35</td>
<td>0.461</td>
<td>39</td>
<td>0.233</td>
<td>14</td>
</tr>
<tr>
<td>39</td>
<td>3.009</td>
<td>31</td>
<td>0.487</td>
<td>38</td>
<td>0.229</td>
<td>16</td>
</tr>
<tr>
<td>40</td>
<td>3.215</td>
<td>38</td>
<td>0.453</td>
<td>41</td>
<td>0.205</td>
<td>30</td>
</tr>
<tr>
<td>41</td>
<td>2.169</td>
<td>8</td>
<td>0.513</td>
<td>34</td>
<td>0.456</td>
<td>2</td>
</tr>
<tr>
<td>42</td>
<td>3.321</td>
<td>42</td>
<td>0.444</td>
<td>42</td>
<td>0.212</td>
<td>23</td>
</tr>
<tr>
<td>43</td>
<td>3.12</td>
<td>33</td>
<td>0.461</td>
<td>40</td>
<td>0.2</td>
<td>36</td>
</tr>
</tbody>
</table>
7.7 Conclusion

It is helpful to have quantitative confirmation of this differences between textures as in the displays of axial slices in Figures 7.4 and 7.3 each spatial map has a slightly different colour normalisation, and fine-grained visual comparison is not reliable. The top five vectors ranked by smoothness, as shown in Table 7.2 had an average score of 1.835, whereas the bottom ranked five had an average score of 3.346.

Scores

Further to the above, there is a correlation of 0.35 between the rank of a component under the synchronisation and smoothness metrics, indicating that more synchronised components are expected to be somewhat smoother than the average component.

With respect to the magnitude score we note that the highest values occur in the block of components present at initialisation \( k \leq 20 \). This suggests that the initial settings are somewhat sticky. A further comment on the magnitude scores is that they are anti-correlated with synchronisation, the coefficient being \(-0.364\). This suggests that psychologically more interesting components are weak signals in terms of their mean square magnitude.

7.7 Conclusion

7.7.1 Comments on results

When applied to neuroimaging data the methods developed above have a number of advantages and also some limitations. We have demonstrated that it is possible to scale a Bayesian non-parametric method to high-dimensional neuroimaging data. Whereas conventional methods for matrix factorisation are typically reliant on the eigendecomposition (see chapter 3), the method used here is not. Hence it offers a free-standing alternative to existing approaches, and might be used to validate the results obtained by other methods.

We saw that inferred Bayesian matrix factorisation model was able to produce a solution in which the latent time-courses were more synchronised across subjects than under a conventional ICA model. Examining synchronisation between subjects is a question of some scientific interest in psychology. Hence the methods presented here may prove to be of assistance in work where inter-subject synchronisation is a key issue.

More generally, the synchronisation metric we proposed for ranking may be useful through its suggestion that a latent dimension is task-related and interpretable. Indeed we saw in section 7.6 that those factorisation dimensions whose component time-series were most synchronised across individuals had spatial maps which loaded on cortical areas more strongly than the least synchronised individuals.
This outcome may have been helped by the removal of orthogonality constraints from component-wise time-courses. As discussed in section 3.8, a high-level view of the underlying biology provides no justification for the imposition of orthogonality constraints. We say ‘may have helped’ because, of course, our method differs from conventional ICA, as practised in neuroimaging in at least three other ways: spike-and-slab components, variable factorisation dimension $K$, and dependency within basis vectors. It is challenging to attribute the differing outcome in terms of synchronisation to the lack of orthogonality constraints on time-series alone due to the combination of factors. Some of these factors are impossible to unbundle within an IBP based factorisation model inferred with MCMC. However, orthogonality constraints on time-series would certainly confound any attempt to recover genuinely correlated activity, while the other considerations mentioned would not.

While the empirical analysis of this chapter has provided some interesting insights into matrix factorisation of neuroimaging data under a non-standard approach, there are limitations to the approach.

First and foremost, MCMC proved to be insurmountably slow for this model. Even with a GPU equipped high performance compute cluster, and after implementation of the novel parallelisation strategy set out in chapter 6 and efficient modelling from chapter 4 it takes several weeks to obtain useful results. This has negative effects: it means that running multiple chains is too computationally expensive to be practical; and the time required to investigate pre-processing choices and plausible prior parameters limits our ability to test the sensitivity of the results to changes in assumptions. We consider the latter point is more important than the former because it makes it difficult to compare and contrast between different choices of approach. While important, the issue of MCMC convergence is here, somewhat moot due to the need for a soft cap on the number of basis vectors implemented through the prior on the noise level.

Of lesser computational importance, but worth recording here, are the disk-space requirements for MCMC on such a model. The procedure outputs $O(K(V + T))$ bytes of data per iteration, hence over a long MCMC run we may require several multiples of the original disk space. The chain discussed in section 7.6 occupies 0.65TB of space on disk, while the entire original dataset of over six hundred subjects occupies around 0.6TB, so that the disk used for the chain was around 20 times greater than the original data for 30 subjects. Disk space requirements scale linearly with both the latent dimension and the number of samples drawn. Compression on-the-fly could slow down the sampler, while thinning the chain would risk loss of useful information such as the ‘hairs’ on trace plot which indicate exploration of the dimensionality. Hence we felt there was little practical to be done about the issue without a complex approach to storage. Together with the run-time requirements, this limits the
scope for adoption in the applied setting to researchers with access to specialist computing facilities.

Furthermore, as MCMC is a sampling based method, there is a question of how to summarise and present the obtained information about the posterior distribution. When the model parameters are fixed in number and may effectively be considered as a collection of univariate distributions, summarisation of results is relatively straightforward. For non-parametric matrix factorisation (and other high-dimensional) models it is not: for instance basis vectors may be activated and discarded over the course of a run so that we cannot simply take the approach of ‘averaging’ over samples (this can be seen easily when we consider that the matrices we store may be of different sizes). This is compounded by the nature of neuroimaging data, where a run from each subject provides a realisation of a 4D spatio-temporal process and the basis vectors contained in $W$ are properly viewed as samples of 3D spatial processes. While in models with univariate parameters we can use the two-dimensional nature of the page to visualise a histogram, or a heatmap for bivariate parameters, such approaches are not available for high-dimensional objects. Hence we adopted the expedient approach of examining the results from the final MCMC iteration. This was computationally wasteful and failed to take full advantage of information about uncertainty contained in the samples.

The model we used was a non-parametric linear model, with dimensionality selection through the Indian Buffet Process. For neuroimaging we found that this had a substantial drawback under a non-informative prior on the observation noise. The difficulties encountered were suggestive of brain-images lying very close (i.e. low observation noise) to a non-linear manifold so that the linear approximation could be constantly and profitably improved by the addition of further basis vectors (i.e. by explaining the data better than noise).

### 7.7.2 Further work

The discussion in section 7.7.1 suggests a number of possible directions for development of this work. One possibility follows from the biologically plausible fit we obtained in terms of inter-subject synchronisation and interpretable spatial maps. Namely, the investigation of inference methods which could obtain qualitatively similar results more quickly. We begin to take this strand of investigation up in chapter 8. This line of investigation would also seem to be a pre-requisite for an analysis of differences between groups of subjects or the effects of covariates such as demographics, as such analyses would seem to require a larger (sub-)sample of the dataset than that which we were able to use in section 7.6.

The second thread for development arises from the ‘large’ number of latent basis vectors required under an uninformative prior. In particular, while not reported above due to lack
of space (e.g. we would need 90+ pages to fully visualise the chain reported in figure 7.1), solutions with large $K$ frequently contained basis vectors serving just a single individual. This effect was largely curtailed in the results reported in section 7.6 by means of an informative prior. This suggests nonparametric matrix factorisation might be useful for the purpose of removing artefacts and homogenising data by performing factorisations on small groups of subjects and retaining only the components that were generally shared between subjects.

The large $K$ issue raises a further possibility: that a nonlinear latent manifold would provide a better model than the linear manifold implied by the matrix factorisation model. Since a linear approximation to a non-linear manifold can usually be improved by the addition of further basis vectors (e.g. consider a piecewise linear approximation of $y = x^2$). If this were correct a cause of the large $K$ phenomenon would be the nonparametric model adding progressively more basis vectors to better approximate a nonlinear manifold. However, alternative nonlinear models such as neural networks are generally less interpretable than linear models. For example the linear approximation visualised in section 7.6 had a finite number of basis vectors which span a regression hyperplane. For nonlinear models we would have to produce different visualisations for different points in the space, perhaps either by sampling from a generative model, or by visualising the tangent space of the surface at that point. This could lead to an interesting set of tradeoffs between increased approximation accuracy and model interpretability.
Chapter 8

Structured basis matrix factorisation via Variational Bayes

8.1 Motivation

In the last chapter we utilised Markov Chain Monte Carlo to fit a non-parametric linear model to neuroimaging data. This helped to shed light on the intrinsic dimensionality of fMRI neuroimaging under quite different assumptions to those based on eigendecompositions.

The Indian Buffet Process distribution we applied, whose properties were developed in the machine learning community (Ghahramani and Griffiths, 2005; Knowles and Ghahramani, 2007, 2011; Teh et al., 2007), was the engine for this dimensionality estimation. However, we found this approach had its own limitations. As discussed in chapter 7, because our model was non-parametric we found that in order to stabilise inference an informative prior on the additive noise level was needed. This imposed an effective cap on the number of components. Nevertheless, the cap enabled a switch from assuming a particular latent dimensionality (as in eigendecomposition based approaches), to assuming a minimum acceptable noise level and finding the dimensionality which matched. As computation is a major challenge in these models, and non-parametric algorithms are inevitably both more complex to code and require more memory allocation (in the absence of truncation), this begged the question of whether fixed dimension methods would have a greater value.

Further to our developing insights into the dimensionality of fMRI data, the MCMC method involves generating a large number of samples. We found this created two challenges for neuroimaging data. First, is the issue of how to summarise the generated information about the posterior distribution. For example, a large part of the interest in the ICA method in fMRI neuroimaging is the production of $K$ maps of the brain. MCMC on our non-parametric
model produced \( g = 1, \ldots, r_2 \) samples where \( K^g \) varied over iterations. Furthermore, ‘label-switching’ is known to be a phenomenon, where the meaning of, say, the \( g^{th} \) basis vector changes over the course of the MCMC chain (for the analogue in mixture models see Stephens, 2000). This makes it difficult to present an average or summary view of an individual basis vector, that is, spatial map of the brain. Hence the model may fulfil the scientific purpose of dimensionality estimation under noise level constraints, but be limited in terms of the interpretability of the resulting basis. This necessitates visualisations of single samples from the model, which both somewhat defeats the purpose of obtaining many samples describing the posterior, and introduces unnecessary noise into the resulting visualisations.

Secondly, an MCMC chain with \( r_2 \) samples requires on the order of \( r_2 K (V + T) \) disk space on a computer. In practice this can be many terabytes of storage, and several multiples of the size of the original dataset. If the goal of the dimensionality reduction is compression rather than insight into the latent structure of the data, this is clearly not an attractive scenario.

### 8.2 Variational Bayes

In this section we recapitulate the variational approach to Bayesian inference. Variational Bayes (VB) is an optimisation based approach to inference, and is therefore a potential alternative to Markov Chain Monte Carlo.

As a number of Variational Bayes techniques have been published only recently, we sketch their motivation and derivation in order that the reader can appreciate how they fit together in our later applied work. In particular, attention is paid to techniques which will be of use in developing scalable software for matrix factorisation.

Particularly attractive features of the VB include: stochastic optimisation, using mini-batches of data for faster inference; closed form updates for conjugate exponential families in a similar fashion to Gibbs sampling; and stochastic gradient updates for non-conjugate distributions. Furthermore, the solution consists of the parameters of an optimal approximation to the posterior. This makes the posterior means of the various quantities a clear choice for visualisation, particularly relative to MCMC which entailed dealing with choice of a particular sample, or else averaging over samples while dealing with label switching issues. The solution takes a fraction of the disk space compared to MCMC sampling.

#### 8.2.1 The evidence lower bound

Motivated by the limitations of our MCMC approach to inference in a non-parameteric matrix factorisation, we developed a fixed dimension model inferred using variational Bayesian
methods. Variational Bayes (VB) is an optimisation based method, which targets a model evidence lower bound (ELBO). As it helps in the understanding and interpretation of the results, we re-derive this bound below, in which $y$ is the observed data, $\theta$ our model parameters, with $p$ the true probability distribution and $q$ an approximate distribution over $\theta$.

\[
p(y) = \int p(y, \theta) d\theta = \int \frac{q(\theta)}{q(\theta)} p(y, \theta) d\theta = \mathbb{E}_q \left[ \frac{p(y, \theta)}{q(\theta)} \right]
\]

\[
\ln p(y) \geq \mathbb{E}_q [\ln p(y, \theta) - \ln q(\theta)] = \mathbb{E}_q [\ln p(y, \theta)] + \mathbb{H}[q(\theta)] = \mathcal{L}
\]

(ELBO)

Where the second line follows by Jensen’s inequality, $\mathbb{H}$ is the entropy functional $\mathbb{H}[g] = \mathbb{E}_q [\ln g(\theta)]$, and we have labelled the right hand side of ELBO as $\mathcal{L}$ since this will be an objective function later.

The goal of the VB method is to find a distribution $q$ which maximises this lower bound (Blei et al., 2017; Fox and Roberts, 2012). In its most general incarnation, an implementation of VB may directly try to find an optimal $q$ via the Euler-Lagrange equations (assuming the required regularity conditions) since the ELBO is a functional of $q$. This approach is somewhat uncommon, however, in that it requires a great deal of specialisation to a model in solving the required differential equations.

A negative point for VB is the lack of the posterior convergence guarantee associated with MCMC. However, VB does provide a principled lower bound on the evidence provided through the ELBO. Another downside to VB is that it will not provide the information on multimodal posteriors available through MCMC.

### 8.2.2 Co-ordinate updates

A common way to make the maximisation of ELBO more tractable is through a ‘mean-field’ approximation, which, for our purposes, is the choice of an approximation which factorises conveniently as $q(\theta) = \prod g q(\theta_g)$ where the $\theta_g$ may be groups of variables. This leads to iterative schemes in which the ELBO is maximised with respect to the factor $q(\theta_g)$ alone, holding the other factors fixed. For a group of co-ordinates $\theta_g$, and their complement $\bar{\theta}_g = \theta \setminus \theta_g$, this results in a target:

\[
\ln p(y) \geq \mathbb{E}_{q(\theta_g)} [\mathbb{E}_{q(\bar{\theta}_g)} [\ln p(y, \theta)]] + \mathbb{H}[q(\theta_g)] + c
\]

(8.1)
Fox and Roberts (2012) show that the difficulty of solving the Euler-Lagrange equations can be avoided by reformulating the bound as

\[
\ln p(y) \geq -KL(q(\theta_g)||q^*(\theta_g)) - \mathbb{H}[q(\tilde{\theta}_g)] + \ln Z
\]  

(8.2)

with \(Z\) being the appropriate normalisation constant. Since KL is nonnegative and \(KL(q(\theta)||q^*(\theta)) = 0\) exactly when \(q(\theta) = q^*(\theta)\) the optimal variational distribution for \(\theta_g\), holding the rest fixed, is

\[
q^*(\theta_g) = Z^{-1} \exp(\mathbb{E}_{q(\bar{\theta}_g)}[\ln p(y, \theta)])
\]

which leads naturally to an iterative updating scheme by repeatedly cycling through the various \(\theta_g\).

### 8.2.3 Parametric families

An assumption that makes maximisation of the ELBO tractable is that the \(q(\theta_g)\) come from parametric families of distributions \(q(\theta_g|\phi)\). This allows us to maximise ELBO through choice of \(\phi\), which is usually a finite dimensional vector.

In the somewhat special case of distributions hailing from the exponential family and in particular where conjugacy relationships hold within the hierarchy of a model, handling the optimal \(q^*(\theta_g)\) becomes particularly simple (Bishop et al., 2002; Bishop and Winn, 2003). It is possible to write the exponent in \(q^*(\theta_g)\) as a dot product involving sufficient statistics \(t(\theta_g)\) and natural parameters \(\eta(d, \bar{\theta}_g)\)

\[
\mathbb{E}_{q(\bar{\theta}_g)}[\ln p(y, \theta)] = E_{q(\bar{\theta}_g)}[\eta(y, \bar{\theta}_g)]'t(\theta_g)
\]  

(8.3)

Which leads to the identification of \(q^*(\theta_g)\) as being from the same family as the prior on \(\theta_g\) and with natural parameter \(\phi = \mathbb{E}_{q(\bar{\theta}_g)}[\eta(y, \bar{\theta}_g)]\).

### 8.2.4 Stochastic variational inference

So far we have treated all the latent variables \(\theta_g\) on an equal basis, however it is useful to draw a distinction between ‘local’ and ‘global’ variables. The essential idea here is that global variables influence the likelihood of many observations, while local variables influence the likelihood of only a few observations. Or, cast as directed acyclic graphs, global variables have many children, while local variables have few. For example, in a matrix factorisation,
the components of an observed vector with respect to a basis appear only in the likelihood of that observation, whereas the basis vectors appear in all the observations.

Indeed, it was a similar observation about the local effects of certain variables which enabled our insight into the appropriate parallelisation structure for the Gibbs sampler for non-parametric matrix factorisation in chapter 6. As discussed in section 8.1 above a primary goal of dimension reduction in brain imaging is the identification of interpretable basis vectors. A limitation of Gibbs sampling is that for every update to the global basis vectors, we must also update the local components for each observation. But consider updating the local variables for a few of observations: we can already imagine that these revised components imply a change in the distribution of our global basis vectors.

It is natural to look for inference methods which allow us to extract information about the global variables of interest more quickly. The stochastic variational inference (SVI) method of (Hoffman et al., 2013) utilises the relationship in section 8.3. Consider an index \( g \) for a global variable \( \theta_g \) by replacing the expectation \( h(g) = E_q(\theta_g)[\eta(y, \bar{\theta}_g)] \) with an unbiased estimate of \( h(g) \) by randomly selecting an \( S \) size subset of the data and/or variables \( \theta_i \) whose likelihoods depend on \( \theta_g \), we can calculate an equivalent \( \tilde{\eta}(y, \bar{\theta}_g) \) whose downstream contributions are scaled by \( N/S \) so that \( E[\tilde{\eta}(y, \bar{\theta}_g)] = \eta(y, \bar{\theta}_g) \).

Consider a brief example. Suppose \( y_i \sim \text{Exp}(\lambda) \), \( i = 1, \ldots, N \), with a conjugate prior on \( \lambda \) we can calculate the subset mean \( \bar{y}^{(S)} \) and calculate a subset statistic \( \tilde{\eta}(y, \bar{\theta}_g) = N\bar{y}^{(S)} \) so that the expectation \( E[\tilde{\eta}(y, \bar{\theta}_g)] = N/\lambda = E[\eta(y, \bar{\theta}_g)] \).

Now such unbiased estimators of the natural parameters entail an addition of variance to our estimates. The work of Robbins and Monro (1951) established conditions for iterative methods based on such estimates to converge to a minimum when \( \phi^{(t)} = (1 - \rho^{(t)})\phi_{t-1} + \rho^{(t)}\phi^{(t)} \) and \( \phi^{(t)} = E_q(\theta)[E[\tilde{\eta}(y, \bar{\theta}_g)]] \). These require sequences of weights chosen so that \( \sum_{t=1}^{\infty} \rho_t = \infty \) and \( \sum_{t=1}^{\infty} \rho_t^2 < \infty \).

### 8.2.5 The re-parameterisation trick

So far we have focussed on conjugate exponential family relationships within a hierarchical model. These have the benefit of yielding closed form updates (corresponding to being able to analytically solve a gradient ascent problem). However, we may need to include within our model relationships which are not conjugate. An approach to dealing with this situation is the re-parameterisation trick (Kingma and Welling, 2014; Kucukelbir et al., 2015; Rezende et al., 2014). Consider a version of equation 8.1 in which the variational distribution of our
co-ordinates of interest are given a parametric distribution \( q(\theta_g; \phi_g) \):

\[
\mathcal{L} = \mathbb{E}_{q(\theta_g|\phi_g)}[\mathbb{E}_{q(\tilde{\theta}_g)}[\ln p(y, \theta)]] + \mathbb{H}[q(\theta_g; \phi)] + c \tag{8.4}
\]

We would like to do gradient ascent on this objective function with respect to the co-ordinate \( c \) on \( \theta \).\( z \) similar interchange on the rightmost term.\n
The idea of the re-parameterisation trick is to perform a change of co-ordinates, so that the stochasticity is expressed by a standard probability distribution which does not depend on \( \phi \). Since \( q(z) = b(\theta_g, \phi) \) (we use \( z \) and \( \Phi \) as this is not necessarily, but very frequently is, a normal distribution).\n
The gradient is,

\[
\nabla_{\phi} \mathcal{L} = \nabla_{\phi} \left( \mathbb{E}_{q(\theta_g|\phi_g)}[\mathbb{E}_{q(\tilde{\theta}_g)}[\ln p(y, \theta)]] + \mathbb{H}[q(\theta_g; \phi)] \right)
\]

\[
= \mathbb{E}_{\Phi(z)} \left[ \nabla_{\phi} \mathbb{E}_{q(\tilde{\theta}_g)} \left[ \ln \left( p(y, \tilde{\theta}_g(z)) \left| \frac{\partial \theta_g}{\partial z} \right. \right) \right] \right] + \nabla_{\phi} \mathbb{H}[q(\theta_g; \phi)] \tag{8.5}
\]

Since \( q(\theta_g; \phi) \) is typically from a standard parametric family, a closed form is usually known for its entropy \( \mathbb{H} \) in terms of the parameters \( \phi \) and so it is not necessary to perform a similar interchange on the rightmost term.

For example, if we consider \( q(\theta_g; \phi) = \mathcal{N}(\mu, \sigma^2) \) with \( \phi = (\mu, \sigma^2) \) we could reparameterise this \( z = b(\theta_g, \phi) = (\theta_g - \mu) / \sigma \) with \( q(z) = \mathcal{N}(0, 1) \). With a standard normal prior on \( \theta_g \) and a Gaussian likelihood \( p(y|\theta_g) = \mathcal{N}(\theta_g, 1) \) then \( \ln p(y|\theta_g) = c - (1/2)(y - \theta_g)^2 = c - (1/2)(y^2 - 2y\theta_g + \theta_g^2) \) and the gradient of our objective would be:

\[
\nabla_{\phi} \mathcal{L} = \mathbb{E}_{\Phi(z)} \left[ \nabla_{\phi} \ln p(y, \theta_g) \left| \frac{\partial \theta_g}{\partial z} \right. \right] + \nabla_{\phi} \mathbb{H}[q(\theta_g; \phi)]
\]

\[
= \mathbb{E}_{\Phi(z)} \left[ \frac{\partial}{\partial \phi} \left( \ln p(y, \theta_g) + \ln \left| \frac{\partial \theta_g}{\partial z} \right| \right) \right] + \nabla_{\phi} \frac{1}{2} \ln(2\pi e \sigma^2)
\]

\[
= \mathbb{E}_{\Phi(z)} \left[ \frac{N\bar{y} - (N+1)(\mu + \sigma z)}{N\bar{y} - (N+1)(\mu + \sigma z) + \frac{1}{2} z} \right] \left[ \begin{array}{c} 0 \\ 1/\sigma \end{array} \right]
\]

\[
= \left[ N\bar{y} - (N+1)\mu \right] \left[ 1/\sigma - (N+1)\sigma \right] \tag{8.6}
\]
In the current, simple example, we were able to analytically integrate out $z$ for the last line. Equating the gradient to zero and solving for $\mu$ and $\sigma$ would show that the variational distribution under the reparameterisation trick in this case replicates the exact posterior distribution. In more complex scenarios we would ascend the stochastic expectation of the gradient instead. In such cases, further transformations may be necessary to ensure that the variance remains positive when doing numerical gradient ascent, such as by defining a new parameter of interest $\omega$ and the transformation $\sigma = e^\omega$, followed by further application of the chain rule. The penultimate line is in a form such that we could evaluate the expectation using Monte Carlo samples from the fixed distribution $z \sim \Phi$. We would then use this estimated gradient for a gradient ascent procedure.

### 8.3 Structured Basis Matrix Factorisation

#### 8.3.1 Aims

We move now to an application of the VB approach to a matrix factorisation problem. The model of chapter 5 took a spike and slab approach to the modelling of components. The Indian Buffet Process was used to achieve a non-parametric approach to linear modelling. VB inference for IBP factorisation models was previously considered by Doshi et al. (2009). However, as described in chapter 7, an infeasibly large number of basis vectors was found under an non-informative prior on the observation noise. Instead we used an informative prior to cap the number of basis vectors through a floor on the noise level. We anticipate that VB inference on the same model would provide a faster route to an unwieldy result. In order to simplify the model, and assist its scalability, we dispense with the Indian Buffet Process and utilise a fixed dimension binary matrix for modelling of the spike element of the components.

The matrix factorisation approaches to neuroimaging data discussed in chapter 2 involved a stage of Independent Components Analysis. ICA is nominally a ‘distribution free’ method (although typically constrained by choice of approximations). The model of chapter 5 placed spike and gaussian slab distributions on the components. In order to make possible investigation into the benefits of a more flexible prior we incorporate a mixture model into the slab element of the components. Mixture models provide a general route to modelling unknown densities. For example, in the limiting case of Dirichlet Process mixtures of normals (infinite mixtures), then under some conditions the posterior distribution is consistent with the true distribution (Ghosal et al., 1999).
8.3.2 Model

We begin with a $V \times T$ data matrix $Y$ comprising $T$ observations of $V$ dimensional vectors. We factorise the data into an $V \times K$ latent basis, $W$, and a $K \times T$ matrix of components $X$. The components $X_{k,n}$ are given a flexible distribution through a spike-and-slab model in which the slab is a Gaussian mixture with $J$ components, $s_{k,n} \in \mathbb{R}^J$, which we shall henceforth call scaling variables in order to avoid confusion with the vector components. We account for observation noise $E_{m,n}$ arranged into a matrix $E$:

$$Y = WX + E$$

$$X_{k,n} = b'_{k,n}(s_{k,n,1}, \ldots, s_{k,n,J}, 0) = b'_{k,n}q_{k,n}$$

The model, with its variables and their distributions is specified in Table 8.1. All variables not provided with a distribution are to be supplied with the prior, namely $\alpha, a, b, e, f, m,$ and $v$.

Table 8.1 Model specification for SMF

<table>
<thead>
<tr>
<th>Variable</th>
<th>Distribution</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Y_{n}$</td>
<td>$\mathcal{N}(WX_{n}, \psi^{-1}I_{M})$</td>
<td>observations</td>
</tr>
<tr>
<td>$W_{k}$</td>
<td>$\mathcal{N}(\phi_k, \Delta^{-1})$</td>
<td>basis vectors</td>
</tr>
<tr>
<td>$s_{k,n}$</td>
<td>$\mathcal{N}(\tau_k, D(\sigma_k)^{-1})$</td>
<td>scaling variables</td>
</tr>
<tr>
<td>$(\sigma, \tau)_{k,j}$</td>
<td>$\mathcal{N}(m_j, v_j^{-1})\Gamma(e_j, f_j)$</td>
<td>scaling prior</td>
</tr>
<tr>
<td>$b_{k,n}$</td>
<td>$\text{Cat}_{J+1}(\rho_k)$</td>
<td>categorical mixture selectors</td>
</tr>
<tr>
<td>$\rho_k$</td>
<td>$\text{Dir}_{J+1}(\alpha)$</td>
<td>mixture prior</td>
</tr>
<tr>
<td>$\psi$</td>
<td>$\Gamma(e_{\psi}, f_{\psi})$</td>
<td>isotropic noise level</td>
</tr>
<tr>
<td>$\theta_p$</td>
<td>$\ln \mathcal{N}(a, b^{-1})$</td>
<td>dependence parameters</td>
</tr>
</tbody>
</table>

One of our goals is to provide an interesting, tractable model for the precision matrix of the basis vectors. We follow the approach of chapter 4 and structure this via an eigendecomposition involving an orthonormal matrix $U$ and $P \leq V$ parameters which parametrise a positive definite diagonal matrix $H(\theta)$ having differentiable components such that $H(\theta)_{mm} = h_m(\theta)$ (some calculations will be simplified when the $h_m$ are linear functions):

$$\Delta(\theta) = UH(\theta)U'$$

We will generally suppress the dependence of the precision matrix on $\theta$ for notational convenience. As an example, when $U'$ represents the Discrete Cosine Transform, we can
find a function $D(\theta)$ which ensures the resulting precision matrix is sparse and encodes an autoregressive process. A few observations serve to highlight the utility of this approach. The computational implication of this is that we can use fast linear operators to tractably evaluate the likelihood even on high dimensional ($V > 10^5$) data.

The model penalises the basis vector according to its magnitude and a weighted combination of its components in the transformed space implied by the choice of $U$

$$x'Ax = \theta_1||x||_2^2 + ||H(\theta)^{1/2}U'x||_2^2$$

So, to continue our example, by taking $U'$ to be the Discrete Cosine Transform and $H$ to be related eigenvalues we can penalise non-smooth basis vectors.

**Previous work**

Considerations similar to ours regarding two-stage factorisation methodologies helped to motivate the work of Harrison et al. (2015). That paper proposed a hierarchical generative model for fMRI images with inference via Variational Bayes. They proposed an IID spike-and-slab model placed on the spatial distribution in order to emulate ICA-like spatial sparsity, with an autoregressive model placed on temporal activity. Our modelling approach differs in a number of ways.

First we place a correlated, rather than IID spatial prior on the spatial distribution. As we saw in chapter 5, modelling spatial correlation, where appropriate, has the potential to assist deconvolution in a manner not achieved by an IID prior. The authors of that paper considered it impractical to model spatial smoothness, "by choosing not to impose a spatial smoothness constraint the computational complexity of our algorithm is reduced, thereby allowing the analysis of hundreds of subjects simultaneously." We shall demonstrate in chapter 9 that this limitation does not hold in practice.

Second, our model is not hierarchical. This causes it to be slightly less computationally flexible, through fewer opportunities for parallelisation. However our model is perhaps more easily interpretable, as we do not need to consider how much individual departures from the global basis might affect the qualitative characteristics of the basis from subject to subject.

Third, we impose a spike-and-slab model in the temporal dimension to effect sparse activation of basis vectors.

Lastly, by employment and combination of more advanced VB techniques, our inference algorithm scales better with the number of subjects in a dataset. For example, the minibatch technique leads to an $O(1)$ rather than to an $O(n)$ algorithm in theory (in practice the need for smaller step sizes will mean that this is not exactly the case).
8.4 Inference

To fit the model we employ the variational Bayes approach. As the model contains both conjugate and non-conjugate distributions, a mix of co-ordinate ascent (Bishop et al. (2002)) and reparameterised gradient ascent (Kingma and Welling, 2014; Kucukelbir et al., 2015) tactics are used to take advantage of computational efficiencies where possible. Note that we cannot yet use fully automated software such as that described by Kucukelbir et al. (2015) due to our requirement to scale the inference using a GPU cluster and the need for linear operators not yet implemented in standard software. It should be noted that co-ordinate ascent is closely related to gradient ascent using the natural gradient, which under conjugacy yields closed form solutions for optimal updates Hoffman et al. (2013). Hence there is a strong a priori reason to use conjugacy relationships.

In the following subsections a variable marked with a hat denotes the expectation of the relevant random variable under its variational distribution, so that \( \hat{\tau}_k \) denotes the expectation of \( \tau_k \) under its variational posterior distribution. Due to mean field assumptions leading to variational distributions from known families these expectations will generally have closed form solutions, and so we will comment on them further only where necessary.

8.4.1 Notation

We will use a form of Matlab notation, so that \( Y_{i,:} \) refers to the \( i^{th} \) row vector of a matrix, \( Y_{:,j} \) the \( j^{th} \) column vector, \( Y_{:,-j} \) an \( V \times T - 1 \) matrix formed of all columns except the \( j^{th} \), and so forth. We will define the function \( D : \mathbb{R}^B \to \mathbb{R}^{B\times B} \) to be the diagonal matrix formed of the argument’s components, in order. The symbol \( \circ \) denotes an elementwise product. Wherever a function is used on a vector which does not have a usual vector interpretation we mean the element-wise vector analogue of the function, this occurs in particular for \( \ln x \) and \( x^2 \).

As it will be used several times, the log likelihood of the data can be redefined in terms of a residual at the \( n^{th} \) observation and \( k^{th} \) component. We define this residual variable as:

\[
r_{k,n} = Y_{:,n} - W_{:,k}X_{-k,n}
\]  

(8.7)

The log likelihood of the \( n^{th} \) observation given all the latent variables can be written as:

\[
\ln p(Y_{:,n}|\cdot) = \text{const} + \frac{V}{2} \ln \psi - \frac{1}{2} (r_{k,n} - W_{:,k}X_{k,n})' \Psi (r_{k,n} - W_{:,k}X_{k,n})
\]

\[
= \text{const} + \frac{V}{2} \ln \psi - \psi \frac{r'_{k,n} r_{k,n}}{2} - \frac{\psi}{2} (W_{:,k}X_{k,n})'(W_{:,k}X_{k,n}) + \psi (W_{:,k}X_{k,n})'r_{k,n}
\]

(8.8)
8.4 Inference

8.4.2 Basis vectors

We can write the log likelihood and prior

\[
\ln p(Y|W_{:-k}, \cdot) = c - \frac{1}{2} \sum_{n=1}^{N} (r_{n,k} - X_{k,n} W_{:-k})' \Psi (r_{n,k} - X_{k,n} W_{:-k})
\]

\[
\ln p(W_{:-k}) = c - \frac{1}{2} (W_{:-k} - \phi_k)' \Delta (W_{:-k} - \phi_k)
\]

In the case of uniform noise covariance \( \Psi = \psi I_M \) we have a new set of eigenvalue functions:

\[
(\hat{H}_k)_{nm} = \mathbb{E}_q[\psi] \sum_{n=1}^{T} \mathbb{E}_q[X^2_{k,n}] + \mathbb{E}_q[h_m(\theta)]
\]

So that a closed form for the variational distribution is,

\[
\hat{W}_{:-k} = \hat{\Lambda}_k^{-1} \left( \hat{\Delta} \phi_k + \hat{\Psi} \sum_{n=1}^{T} \hat{X}_{k,n} \hat{r}_{n,k} \right)
\]

\[
q(W_{:-k}) = \mathcal{N}(\hat{W}_{:-k}, \hat{\Delta}_k^{-1})
\]

It should be noted that \( \mathbb{E}_q[X^2_{k,n}] = \sum_{j=1}^{T} \mathbb{E}_q[s^2_{k,n,j}] \mathbb{E}_q[b_{k,n,j}] \), again due to the properties of the categorical distribution, the mean field approximation and the zero scalar corresponding to \( b_{k,n,J+1} \). Also \( \mathbb{E}_q[X_{k,n}] = \sum_{j=1}^{T} \mathbb{E}_q[s_{k,n,j}] \mathbb{E}_q[b_{k,n,j}] \).

These expressions can be calculated efficiently in the case of uniform noise variance \( \Psi = \psi I_M \). In the case of diagonal \( \Psi \) an efficient solution to \( \hat{\Delta}_k \) would imply the ability to quickly decompose the sum of any two positive definite matrices given the solution of one (here \( \Delta \)) and would probably constitute a major result in linear algebra. In certain cases with sparse \( \Delta \) and diagonal \( \Psi \) we may have relatively efficient numerical solutions. Hence ‘traditional’ factor analysis models with diagonal \( \Psi \) would appear to be efficient only for small \( V \) or in relatively special cases (e.g. sparse \( H \) or when we can be sure that \( ||\Delta|| \gg ||\Psi|| \)).
8.4.3 Mixture selection variables

As a categorical distribution, the log prior on the mixture selection variables $b_{k,n}$ can be written as

$$\ln p(b_{k,n}) = b'_{k,n} \ln \rho_k$$

Taking the expectation of the likelihood over all variables except $b_{k,n}$ we have

$$E_q[\ln f(Y_{:,n}|b_{kn},\cdot)] = \text{const} - E_q[\frac{\psi}{2}X^2_{k,n}W'_{:,k}W_{:,k}] + E_q[\psi X_{k,n}W'_{:,k}r_{k,n}]$$

Again writing $q_{k,n} = (s_{k,n,1}, \ldots, s_{k,n,J}, 0)$, we see that

$$E_q[X^2_{k,n}] = \sum_{j=1}^{J+1} E_q[q^2_{k,n,j}] b_{k,n,j}$$

due to the properties of the categorical distribution, the mean field approximation and the zero scalar corresponding to $b_{k,n,J+1}$. This allows is to identify that the optimal variational categorical distribution, where soft($r_i$) = exp($r_i$)/$\sum_j \exp(r_j)$ is the softmax function:

$$q(b_{k,n}) = \text{Cat}(b_{k,n}^*)$$

$$b_{k,n} = \text{soft} \left( E_q[\ln \rho_k] + E_q[\psi] \left( E_q[W'_{:,k}] E_q[r_{k,n}] E_q[q_{k,n}] - 1/2 E_q[W'_{:,k}W_{:,k}] E_q[q^2_{k,n}] \right) \right)$$

All expectations in the last line having closed form solutions expressed in terms of the relevant variational parameters.

8.4.4 Mixture selection parameters

The model here is a conjugate pair of Dirichlet and multinomial distributions, thus the variational optimum is found at:

$$q(\rho_k) = \text{Dir}(\hat{\alpha}_k)$$

$$\hat{\alpha}_k = \alpha + \sum_{n=1}^{T} E [b_{k,n}]$$

Recalling that the $b_{k,n}$ vectors are of length $J + 1$, taking into account the spike component.
8.4.5 Mixture scaling variables

The mixture scaling variables have a multivariate normal posterior which we can identify by careful examination of the log likelihood:

\[
q(s_{k,n}) = \mathcal{N}(\tilde{s}_{k,n}, \bar{s}_{k,n})
\]

\[
\tilde{s}_{k,n} = (E_q[W_{:,k}]) E_q[\psi] D(E_q[b_{k,n,1:J}]) + D(E_q[\sigma_{k}])^{-1}
\]

\[
\bar{s}_{k,n} = \tilde{s}_{k,n} \left( D(E_q[\sigma_{k}]) E_q[\tau_{k}] + E_q[\psi] E_q[W_{:,k}] E_q[r_{k,n}] E_q[b_{k,n,1:J}] \right)
\]

8.4.6 Mixture scaling parameters

The optimal variational parameters for the slab mixture distributions on the \(k^{th}\) component can be found via conjugacy.

\[
q(\tau_{k,j}, \sigma_{k,j}) = \mathcal{N}(\hat{m}_{k,j}, \hat{v}_{k,j}, \hat{e}_{k,j}, \hat{f}_{k,j})
\]

\[
\hat{m}_{k,j} = \frac{m_{j}v_{j} + T\bar{s}_{k,:,j}}{v_{j} + T}
\]

\[
\hat{v}_{k,j} = v_{j} + T
\]

\[
\hat{e}_{k,j} = e_{j} + \frac{T}{2}
\]

\[
\hat{f}_{k,j} = f_{j} + \frac{1}{2} \sum_{n=1}^{N} (s_{k,n,j} - \bar{s}_{k,:,j})^2 + \frac{v_{j}T}{v_{j} + T}(\bar{s}_{k,:,j} - \tau_{k,j})^2
\]

8.4.7 Spatial parameters

To infer the spatial parameters \(\theta\) we use the reparameterisation tactic (Kingma and Welling (2014), Kucukelbir et al. (2015)). We use an elementwise transformation to the log-scale followed by an elliptical standardisation involving parameters \(\rho = (\mu_1, \ldots, \mu_p, \omega_1, \ldots, \omega_p)\):

\[
\mu_i + \xi_i e^{\omega_i} = \zeta_i = \ln \theta_i
\]

We use a standard normal variational distribution, \(\xi_i \sim \Phi\). The likelihood of the basis vectors is:

\[
\ln f(W|\theta, \cdot) = c + \frac{K}{2} \ln |\Delta| - \frac{1}{2} \sum_{k=1}^{K} (W_{:,k} - \phi_k)^T \Delta (W_{:,k} - \phi_k)
\]
\[ = c + \frac{K}{2} \sum_{m=1}^{V} \ln h_m(\theta) - \frac{1}{2} \sum_{k=1}^{K} (W;_k - \phi_k)^T \Delta(W;_k - \phi_k) \]

Which has derivatives with respect to the spatial parameters \( \theta \):

\[
\frac{\partial \ln f(Y|\theta, \cdot)}{\partial \theta_i} = \frac{K}{2} \sum_{m=1}^{V} \frac{\partial h_m(\theta)}{\partial \theta_i} h_m(\theta) - 1 - \frac{1}{2} \sum_{k=1}^{K} \sum_{m=1}^{V} \frac{\partial h_m(\theta)}{\partial \theta_i} \left( U'(W;_k - \phi_k) \right)_m^2
\]

While the prior on \( \theta \) is:

\[
\ln f(\theta_i) = c - \ln(\theta_i) + \ln(b) - \frac{b}{2} \ln(\theta_i)^2 - \frac{1}{2b} a^2 + b \ln(\theta_i) a
\]

With derivatives:

\[
\frac{\partial \ln f(\theta_i)}{\partial \theta_i} = -b \ln(\theta_i) + (ab - 1) \frac{1}{\theta_i}
\]

Now the Jacobian of the log transformation is:

\[
\ln |J(\zeta \rightarrow \theta)| = \sum \zeta_i
\]

The ELBO objective function can be expressed as

\[
\mathcal{L}(\rho) = \mathbb{E} [\ln f(Y, \zeta) + \ln |J|] + \sum_{p=1}^{P} \omega_p
\]

Which has derivatives with respect to the variational parameters \( \rho \):
\[
\frac{\partial \mathcal{L}(\rho)}{\partial \mu_i} = E_q \left[ \left( \frac{\partial \ln p(Y, \zeta)}{\partial \theta_i} + \frac{\partial \ln |J|}{\partial \zeta_i} \right) \frac{\partial \zeta_i}{\partial \mu_i} \right] \\
\frac{\partial \mathcal{L}(\rho)}{\partial \omega_i} = E_q \left[ \left( \frac{\partial \ln f(Y|\theta)}{\partial \theta_i} + \frac{\partial \ln |J|}{\partial \omega_i} \right) \xi_i e^{\omega_i} \right] + 1
\]

8.4.8 Noise parameters

We can write the likelihood of the data using an appropriate residual \( r_n = Y_{:,n} - WX_{:,n} \) as:

\[
\ln f(Y|\Psi, W, X, \cdot) = c + \frac{T}{2} \ln |\Psi| - \frac{1}{2} \sum_{n=1}^{N} r_n^\top \Psi r_n
\]

With \( \Psi \) being proportional to the identity matrix \( \Psi = \psi I_M \), the conjugate prior distribution is a Gamma distribution. For the uniform noise case the likelihood reduces to:

\[
\ln f(Y|\Psi, W, X, \cdot) = c + \frac{TV}{2} \ln \psi - \frac{\psi}{2} \sum_{n=1}^{N} r_n^\top r_n
\]

Which under a Gamma, \( \Gamma(\hat{e}, \hat{f}) \), prior yields a variational posterior \( q(\psi) = \Gamma(\psi; \hat{e}, \hat{f}) \) where

\[
\hat{e} = e + \frac{TV}{2} \\
\hat{f} = f + \frac{1}{2} \sum_{n=1}^{N} Y_{:,n}^\top Y_{:,n} - \sum_{n=1}^{N} Y_{:,n}^\top WX_{:,n} + \frac{1}{2} \text{tr} \left( E[W^\top W] \sum_{n=1}^{N} E[X_{:,n} X_{:,n}^\top] \right)
\]

It is helpful to observe that (with \( \delta_s^s \) the Kronecker delta):

\[
E[W^\top W]_{k,j} = \hat{W}_{:,k}^\top \hat{W}_{:,j} + \delta_k^j \sum_{m=1}^{M} (\hat{H}_k)^{-1}_{mm}
\]

\[
E[X_{:,n} X_{:,n}^\top]_{j,k} = \delta_k^j E[s_{k,n}^2] + (1 - \delta_k^j) E[b_{k,n,1:L}] E[s_{k,n}] E[b_{k,n,1:L}] E[s_{l,n}] E[b_{l,n,1:L}]
\]
8.5 Software

We have produced software to fit the model above. The package is written in the Julia language, and will (eventually) be made available as SMF.jl. The package has routines for CPU and CUDA compatible GPU. The availability of GPU brings a substantial speedup for high dimensional (large $V$) data. The package natively supports multi-node processing, which, at present we use solely for distributed memory capabilities; although we plan to utilise greater parallelisation for model fitting. The implemented covariance structures for the basis ($\Delta$) are:

- DCT - for smooth features without associated edges. With and without unit normalisation.
- DFT - for smooth features with associated edges.
- DWT - for features with continuity or sparsity constraints. The supported wavelets are those provided by Wavelets.jl.
- IID - independence structure with common variance.

The model is fit using a minibatch strategy specified by the user. Simple loading of data can be done by passing an array. Functionality is also provided for distributed construction of batches by passing filenames and pre-processing functions.

8.6 Simulated example

Here we consider the performance of the above model on a simulated dataset. We generated 1200 samples of $32 \times 32$ pixel images from the linear model Eqn. 2.2. We fixed the observation noise at unity. The individual basis vectors were drawn from a DCT based covariance structure as outlined in Chapter 4 with $h_i = 1 + 100\gamma_i$ with $\gamma_i$ as in Eqn. 4.9. The vector components were distributed, with $p = 0.3, b = 10$ as:

\[
X_{i,j} = m_{i,j,1}z_{i,j,1} - m_{i,j,2}z_{i,j}
\]

\[
m_{i,j,k} \sim Bern(p)
\]

\[
z_{i,j,1} \sim \mathcal{G}(2b, 2)
\]

\[
z_{i,j,2} \sim \mathcal{G}(b, 1/2)
\]
The reason for this slightly exotic distribution was to also test whether use of a mixture model for the components brought practical benefits or not. A histogram of the resulting components is shown in Fig. 8.1.

![Histogram of the components of the simulated data.](image)

Fig. 8.1 Histogram of the components of the simulated data.

We fit models to the resulting data using four different covariance structures. Optimisation was run for 500 steps, at which point the procedure had converged in each case. As the dataset was relatively small it was not necessary to use minibatches. A summary of the performance of the models is given in Table 8.2, which provides the final ELBO of the model, the likelihood at final iteration, the mean square error, the source separation and the total training time. Source separation is defined as the mean maximum absolute correlation between the fitted basis vectors and the true basis vectors; this means the separations score takes a value between zero and unity, with unity representing a perfect reconstruction of the true basis vectors. We can see that the normalised DCT model performs best in terms of mean square error and source separation.

<table>
<thead>
<tr>
<th>Structure</th>
<th>ELBO</th>
<th>Log Likelihood</th>
<th>MSE</th>
<th>Separation</th>
<th>Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCT</td>
<td>-2669582.486</td>
<td>-2593124.500</td>
<td>3.968</td>
<td>0.443</td>
<td>64.826</td>
</tr>
<tr>
<td>nDCT</td>
<td>-44146121.963</td>
<td>-1837556.625</td>
<td>1.159</td>
<td>0.898</td>
<td>64.364</td>
</tr>
<tr>
<td>IID</td>
<td>-2799899.761</td>
<td>-2650840.500</td>
<td>4.375</td>
<td>0.154</td>
<td>62.527</td>
</tr>
<tr>
<td>WLT-db2</td>
<td>-3374906.397</td>
<td>-2450559.250</td>
<td>3.142</td>
<td>0.445</td>
<td>68.668</td>
</tr>
</tbody>
</table>
It would seem attractive to use the ELBO as a criterion for model selection. However, the bound depends on the choice of prior and on the choice of variational posterior. As such use of the ELBO for model comparison has to assume that the bound is tight for both models. While this sometimes holds in practice it is not guaranteed by theory (Blei et al., 2017). It would seem from Table 8.2 that our case falls into the latter bracket, with the best performing model having the least favourable ELBO. This is because the strong regularisation imposed by the prior substantially reduces the variational expectation of the log prior for the basis vectors. This phenomenon would also affect choice of latent dimensionality within a precision structure, since increasing $K$ would decrease the ELBO considerably.

Unfortunately, the MSE and Separation metrics are only available for simulated data. For real data it would appear that the likelihood presents the best basis for model selection. An alternative that we explore further in chapter 9 is the use of other metrics of interest, tailored to the particular application.

The reconstructions of the latent data under the normalised-DCT and IID covariance structures are shown in Figures 8.2a and 8.2b respectively. The difference in mean square error performance is visible in the IID model, which misses both weak and strong signals. This is was because the model shrank the usage of all but one component to zero, meaning that many of the received signals could not be well expressed. Similarly the DCT and WLT-db2 models used only two latent dimensions, with components of the others shrunk to zero, with commensurate penalties on our performance measures.

It is worth noting that these performance discrepancies are a function of sample size. When $V/T$ is large, all the models perform well. However, as $V/T$ reduces the performance of the IID and then other structures suffers most. This is particularly pertinent for the consideration of neuroimaging applications where $V/T$ is typically much greater than one, and is around one only for the largest datasets. Thus if we believe some form of spatial autocorrelation is appropriate, we may be able to achieve superior source separation by accounting for it in our models.
8.6 Simulated example

Fig. 8.2 Demonstration of model on synthetic data, using two different covariance structures.
Chapter 9

Application of Structured Matrix Factorisation to Neuroimaging Data

9.1 Introduction

Continuing the work of chapter 8, we model neuroimaging data using Structured Matrix Factorisation (SMF) via variational Bayes. A key goal is to demonstrate that the approach scales to large neuroimaging datasets. In doing so, we aim to overcome one of the main limitations of MCMC based inference: lack of scalability.

The SMF approach promises to provide a useful counterpoint in debates on how to factorise large neuroimaging datasets. In Calhoun et al. (2015) the PCA stage of factorisation in a PCA-ICA model formed the key issue, in particular whether researchers should model subject specific scaling of spatial eigenmaps. In contrast, with SMF the model has one stage rather than two, and so the question simply does not arise in this form. Instead, the inferred basis must explain variation across all subjects. The ability to handle larger datasets under the variational Bayes approach (than the MCMC work of chapter 7) will help to ensure that the contribution of subject-specific variations is dwarfed by that of common patterns.

The SMF approach also offers desirable modelling attributes for neuroimaging, such as sparse feature activation and spatial constraints on basis vectors. The first implies that brain networks can be activated and deactivated, and in the case of smaller datasets could indicate whether certain basis vectors are subject-specific. Spatial constraints on basis vectors lessen the need for preprocessing step leading to the retention of higher resolution data, while enabling the inference of appropriate smoothing parameters from the data.

\footnote{That is, with a singular value decomposition for the $n^{th}$ subject $Y = UDV^T$ whether to carry through $DV^T$ or just $V^T$ to the next stage of concatenation and factorisation}
We return to the Cambridge Centre for Ageing and Neuroscience data, a smaller subset of which was analysed in chapter 7. A primary finding of Campbell et al. (2015) was that older subjects’ response to movie watching, as captured by fMRI, is less synchronised than that of younger subjects. We revisit this research question using a larger sample of data from a later stage of the project and investigate under our alternate statistical assumptions whether this conclusion can be validated.

9.2 Data and preprocessing

We used imaging from 600 subjects in the CamCAN dataset. The data were preprocessed in the same fashion as in section 7.2.1. This is a larger, later stage dataset than that used in Campbell et al. (2015) (\(N^* = 218\)).

In this analysis we clipped the images to remove empty space at the sides of, behind, and in front of the brain. Slices at the bottom of the images containing the brainstem were also removed. This reduced data dimension \(V\) by about 38% without sacrificing much content of interest. The result reduced memory requirements and improved computational efficiency. Cropped data also improved visualisation, since less of each image is empty space.

Settings for the analysis, including the number of observations, dimensionality, optimisation parameters and so forth are set out in Table 9.1. Our minibatch size of five subjects was linked to the capacity of the GPUs available. We found that with the relatively small minibatch size compared to the large overall number of subjects that optimisation of the ELBO was better facilitated by use of small step sizes.
Table 9.1 Settings used in VB run

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Interpretation</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V$</td>
<td>Observed dimension</td>
<td>170448</td>
</tr>
<tr>
<td>$V_1, V_2, V_3$</td>
<td>Data shape</td>
<td>53, 67, 48</td>
</tr>
<tr>
<td>$T$</td>
<td>Number of observations</td>
<td>115800</td>
</tr>
<tr>
<td>$K$</td>
<td>Latent dimension</td>
<td>20</td>
</tr>
<tr>
<td>$J$</td>
<td>Mixture dimensions</td>
<td>1</td>
</tr>
<tr>
<td>$N$</td>
<td>Number of subjects</td>
<td>600</td>
</tr>
<tr>
<td>$B$</td>
<td>Number of batches</td>
<td>120</td>
</tr>
<tr>
<td>$F$</td>
<td>Files per batch</td>
<td>5</td>
</tr>
<tr>
<td>$r_1$</td>
<td>Spatial step size</td>
<td>1.0f-7</td>
</tr>
<tr>
<td>$r_2$</td>
<td>Spatial step sims</td>
<td>25</td>
</tr>
<tr>
<td>$r_3$</td>
<td>Closed form step size</td>
<td>0.1</td>
</tr>
<tr>
<td>$r_4$</td>
<td>Decay rate</td>
<td>0.5</td>
</tr>
<tr>
<td>$r_5$</td>
<td>Burn ticks</td>
<td>10</td>
</tr>
</tbody>
</table>

Despite the work of chapter 8 on mixture models for components, we found in setting up the analysis that $J > 1$ was not helpful in analysis of this neuroimaging data. In particular, more complex distributions substantially affected computation time, while failing to improve the metrics of interest. Our hypothesis as to why this is the case is that a single normal scaling variable with low precision forms a sufficiently flexible model and that it is challenging for the model to identify latent variables nested more deeply within the hierarchy.

### 9.3 Priors and optimisation constants

The prior parameters used in the analysis to follow are listed in Table 9.2. As we have over a hundred thousand images it is anticipated that the priors will be noninformative relative to the mass of data in the likelihood. To see how the data and prior compete, refer to the derivation in chapter 8. We would be less sanguine were the spatial model on the basis complex, however, the two models examined in this chapter are both univariate.
The initialisation of the algorithm used in this chapter was random; rather than being based on k-means and the LASSO as in chapter 7. After the previous work which lay behind the simulations of chapter 8, we felt that more advanced initialisation procedures did not reliably add much to the solution, while potentially complicating the attribution of a success in a run.

### 9.4 Model selection

In order to facilitate performance evaluation we ran models for dimensionality settings $K = 5, 10, 20, 30, 50, 75, 100$ for both of the IID and normalised DCT based covariance structures for the basis (discussed previously in chapters 4 and 8). We also took the view that this would facilitate comparison of the outputs produced under the different covariance structures.

The first issue we must address is model selection, in terms of both dimensionality and covariance structure. We first present information relating to the IID spatial structure. Figure 9.1 shows the trajectory of the ELBO for different dimensionalities together with the average likelihood of the final 50 iterations while 9.2 shows the level of between-subject synchronisation and smoothness of the basis vectors under different dimensionalities. The relevant numerical values are shown in Table 9.3, where the ELBO column instead shows the average ELBO at the final 5 checkpoints (one checkpoint every 10 iterations). The relevant numerical values relating to these plots are shown in Table 9.3.
9.4 Model selection

Fig. 9.1 a) ELBO curves over the optimisation process. b) Average likelihoods over the final 50 iterations for different dimensionalities under IID spatial structure. Dotted: upper and lower quartiles of the same.

Fig. 9.2 Summaries of synchronisation levels and smoothness for different dimensionalities under IID spatial structure. Solid blue lines represent mean scores. Dotted green lines show upper and lower quartiles.
Table 9.3 Metrics for model selection under IID structure.

<table>
<thead>
<tr>
<th>K</th>
<th>ELBO</th>
<th>Like</th>
<th>SyncMean</th>
<th>SyncMax</th>
<th>SmoothMean</th>
<th>SmoothMin</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>$9.154 \times 10^{10}$</td>
<td>$9.161 \times 10^{10}$</td>
<td>0.482</td>
<td>0.559</td>
<td>1.331</td>
<td>0.903</td>
</tr>
<tr>
<td>10</td>
<td>$9.131 \times 10^{10}$</td>
<td>$9.158 \times 10^{10}$</td>
<td>0.433</td>
<td>0.536</td>
<td>2.014</td>
<td>1.148</td>
</tr>
<tr>
<td>20</td>
<td>$9.191 \times 10^{10}$</td>
<td>$9.153 \times 10^{10}$</td>
<td>0.425</td>
<td>0.548</td>
<td>2.428</td>
<td>1.318</td>
</tr>
<tr>
<td>30</td>
<td>$9.130 \times 10^{10}$</td>
<td>$9.141 \times 10^{10}$</td>
<td>0.375</td>
<td>0.521</td>
<td>2.666</td>
<td>1.674</td>
</tr>
<tr>
<td>50</td>
<td>$9.073 \times 10^{10}$</td>
<td>$9.152 \times 10^{10}$</td>
<td>0.408</td>
<td>0.561</td>
<td>3.389</td>
<td>2.424</td>
</tr>
<tr>
<td>75</td>
<td>$8.840 \times 10^{10}$</td>
<td>$9.150 \times 10^{10}$</td>
<td>0.389</td>
<td>0.601</td>
<td>3.411</td>
<td>2.653</td>
</tr>
<tr>
<td>100</td>
<td>$8.906 \times 10^{10}$</td>
<td>$9.140 \times 10^{10}$</td>
<td>0.366</td>
<td>0.444</td>
<td>3.249</td>
<td>2.269</td>
</tr>
</tbody>
</table>

We see from Table 9.3 that the model with the highest ELBO has $K = 20$. However, we can see from Figure 9.1a that for the IID model that there is a good deal of fluctuation in the ELBO. This is due to a combination of the stochastic minibatch optimisation process and because, for the IID model, the ELBO tracks the likelihood very closely. We also see that performance of the higher dimensional models degrades through the optimisation process for the higher dimensional models under the IID structure. Our interpretation is that this degradation of performance arose from lack of a sufficient penalty model on choice of the basis; although it could also be a symptom of inadequate optimisation parameters for large $K$, or an interaction of these two causes. This would be consistent with closeness of the ELBO and the likelihood. Figure 9.2b shows that as the latent dimension is increased there is a trend towards increasing roughness of the basis vectors under an IID covariance structure.

When we turn to the normalised DCT covariance structure, we see in Figure 9.3a a smooth rise in the ELBO for all models. This is a result of the probabilistic normalisation penalty imposed on the basis by the prior. This effect drowned out the noise induced in the ELBO by the stochastic optimisation that we saw in the IID case. Table 9.4 shows that imposing structure on the spatial covariance resulted in smoother basis vectors at all dimensions except $K = 5$, and does not appear to lead to an increasing trend towards roughness with greater dimensionality. In terms of inter-subject synchronisation, neither covariance structure seems to dominate the other.
Fig. 9.3 a) ELBO curves over the optimisation process, note that lower dimensional models converge more quickly than higher dimensional models. b) Solid: average likelihoods over the final 50 iterations for different dimensionalities under nDCT spatial structure. Dotted: upper and lower quartiles of the same. Note that the scale of b) is shared by Figure 9.1.

Fig. 9.4 Summaries of synchronisation levels and smoothness for different dimensionalities under nDCT spatial structure. Solid blue lines represent mean scores. Dotted green lines show upper and lower quartiles.
Table 9.4 Metrics for model selection under nDCT structure.

<table>
<thead>
<tr>
<th>K</th>
<th>ELBO</th>
<th>Like</th>
<th>SyncMean</th>
<th>SyncMax</th>
<th>SmoothMean</th>
<th>SmoothMin</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>-1.66×10^9</td>
<td>9.149×10^{10}</td>
<td>0.493</td>
<td>0.614</td>
<td>1.563</td>
<td>1.066</td>
</tr>
<tr>
<td>10</td>
<td>-9.71×10^{10}</td>
<td>9.150×10^{10}</td>
<td>0.426</td>
<td>0.554</td>
<td>1.617</td>
<td>1.032</td>
</tr>
<tr>
<td>20</td>
<td>-2.90×10^{11}</td>
<td>9.149×10^{10}</td>
<td>0.406</td>
<td>0.577</td>
<td>1.904</td>
<td>1.285</td>
</tr>
<tr>
<td>30</td>
<td>-5.18×10^{11}</td>
<td>9.130×10^{10}</td>
<td>0.424</td>
<td>0.535</td>
<td>2.024</td>
<td>1.551</td>
</tr>
<tr>
<td>50</td>
<td>-1.00×10^{12}</td>
<td>9.141×10^{10}</td>
<td>0.373</td>
<td>0.471</td>
<td>2.002</td>
<td>1.242</td>
</tr>
<tr>
<td>75</td>
<td>-1.53×10^{12}</td>
<td>9.129×10^{10}</td>
<td>0.34</td>
<td>0.513</td>
<td>1.702</td>
<td>1.106</td>
</tr>
<tr>
<td>100</td>
<td>-2.23×10^{12}</td>
<td>9.151×10^{10}</td>
<td>0.379</td>
<td>0.422</td>
<td>1.824</td>
<td>1.291</td>
</tr>
</tbody>
</table>

For the nDCT model the ELBO is monotonically decreasing with K, indicating a better lower bound on the marginal likelihood of the data for lower dimensional models. On the other hand the highest average likelihood is seen for the highest dimensional model. This seems to be a side-effect of the introduction of a strong regularisation penalty. In particular the expected unit normalisation of the basis vectors. This reasoning is supported by reference to Table 8.2, where the ground truth was known, we can see that the ELBO of such normalised models is far lower for an normalised DCT model than an IID or unnormalised DCT model but the blind source separation was superior. Our suggestion therefore is to examine further the dimensionality $K = 20$ suggested by the IID model, but under the nDCT covariance structure which has preferable smoothness properties.

9.4.1 Magnitude

It may be informative to have a measure of how much each latent dimension contributes to the magnitude of the reconstructed data. Since neither the $w_k$ or $X_{k,:}$ is strongly normalised, we must include a value from each to perform such a calculation. We define the magnitude score as the product, for the $k^{th}$ component of the expected squared L2 norm of the basis vector with the mean expected value of the components squared.

$$m_k = \frac{1}{N} \mathbb{E}_{q} [w_k'w_k] \mathbb{E}_{q} [X_{k,:}X_{k,:}']$$

A higher value of $m_k$ indicates that the component makes a larger (in the L2 sense) contribution to the magnitude of the reconstruction. It is not possible to provide a % of variance calculation for each latent dimension, as is done in methods derived from PCA, because the basis vectors are not constrained to be orthogonal. Note this calculation is somewhat different than the analogue in chapter 7.
9.5 Results from selected model

We present in this section the fitted results for the $K = 20$ dimensional model under the nDCT spatial covariance model identified in section 9.4. Commentary on the results are largely reserved until section 9.6.

Table 9.5 provides a summary of some of the main facts about the variational approximation to the posterior, including the final minibatch estimate of the ELBO and the likelihood (average values over the final 50 iterations are available in Table 9.4). We can see that the penalty attached to the model is very substantial by comparing the likelihood and the ELBO. Recall from chapter 8 that in simulated data the likelihood was not much higher than the ELBO. Also shown are the means and 95% credible intervals for the noise level, the spatial smoothing parameter, and the average means and credible intervals for the spike activation probabilities.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Interpretation</th>
<th>2.5% Quantile</th>
<th>Mean</th>
<th>97.5% Quantile</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELBO</td>
<td>-</td>
<td>-</td>
<td>$-2.898 \times 10^{11}$</td>
<td>-</td>
</tr>
<tr>
<td>likelihood</td>
<td>-</td>
<td>-</td>
<td>$9.15 \times 10^{10}$</td>
<td>-</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Noise</td>
<td>181758.246</td>
<td>181761.855</td>
<td>181765.465</td>
</tr>
<tr>
<td>$\theta_1$</td>
<td>Autocorrelation level</td>
<td>16.1</td>
<td>1677.136</td>
<td>10957.628</td>
</tr>
<tr>
<td>$b_{1:20}$</td>
<td>Selection variables</td>
<td>0.348</td>
<td>0.351</td>
<td>0.354</td>
</tr>
</tbody>
</table>

To interpret the noise $\psi$, recall that this is the precision of the observation errors and that we have $V = 170448$ observed dimensions. Each image was unit normalised and so when $\frac{V}{\psi} \approx 0.938$ we see that the $K = 20$ dimensional model may account for only around 6% of the variance in the data; the percentage of variance type calculation is possible here as we are discussing the model as a whole, rather than attributing explanatory power to individual latent dimensions.

The spatial parameter mean of $\mathbb{E}_{q}[\theta] \approx 1677$ (median 420) indicates a fairly high level of smoothing. In terms of the Full Width at Half Maximum type calculation this value is found at the neighbouring voxel so around 4mm, the quarter maximum voxel is the $4^{th}$ neighbour so around 16mm away. Figure 9.5a shows the covariance between the 50,000$^{th}$ voxel (picked arbitrarily based for being on the interior of the image) and other voxels in an image using the posterior expectation of the smoothing parameter.
(a) Shown is the covariance between the 50,000th voxel and other voxels.

(b) Histogram of non-zero components.

The posterior for the spike activation probabilities shows that in a sampling scenario, we would expect a randomly chosen component to be non-zero around 35% of the time. Figure 9.5b shows a histogram of the non-zero components in the data, since although the VB model provides a multiplication by the posterior activation probability rather than a sample from a Bernoulli distribution, the posterior probability may be so low as to effectively zero out many components. This means filtering out of components within a small multiple of machine precision is necessary to be able to see interesting features of the non-zero components.

Table 9.6 shows the scores and ranks of the latent variables under the metrics defined in section 7.5. We can see that the top-ranked dimension in terms of smoothness and synchronisation is number 14, which is also highly ranked by the magnitude metric (redefined for this chapter in section 9.4.1). As shall be seen in section 9.5.1 this corresponds to an auditory component. A rendering of the full spatial map and subject specific timeseries can be found in Figure 9.21.
| K | Smooth Rank | Mag Rank | Mag Rank | Smooth Rank | Mag Rank | Mag Rank | Smooth Rank | Mag Rank | Smooth Rank | Mag Rank | Smooth Rank | Mag Rank | Smooth Rank | Mag Rank | Smooth Rank | Mag Rank |
|---|-------------|---------|---------|-------------|---------|---------|-------------|---------|-------------|---------|-------------|---------|-------------|---------|-------------|---------|-------------|
| 1 | 2.165 | 15 | 0.001 | 15 | 0.35 | 16 | 0.021 | 0.012 | 19 |
| 2 | 2.287 | 18 | 0.001 | 19 | 0.446 | 4 | 0.014 | -0.162 | 3 |
| 3 | 1.812 | 9 | 0.002 | 9 | 0.388 | 12 | 0.015 | 0.056 | 15 |
| 4 | 2.181 | 16 | 0.002 | 7 | 0.42 | 9 | 0.058 | 0.09 | 11 |
| 5 | 1.834 | 10 | 0.001 | 18 | 0.412 | 11 | 0.024 | 0.07 | 13 |
| 6 | 1.676 | 5 | 0.001 | 13 | 0.378 | 14 | 0.028 | 0.07 | 12 |
| 7 | 2.127 | 13 | 0.003 | 3 | 0.337 | 18 | 0.024 | -0.03 | 17 |
| 8 | 1.544 | 2 | 0.002 | 8 | 0.466 | 3 | 0.016 | 0.221 | 2 |
| 9 | 2.132 | 14 | 0.004 | 1 | 0.422 | 7 | 0.048 | -0.018 | 18 |
| 10 | 1.619 | 4 | 0.001 | 16 | 0.336 | 19 | 0.019 | 0.06 | 14 |
| 11 | 1.719 | 7 | 0.001 | 10 | 0.42 | 8 | 0.016 | -0.139 | 7 |
| 12 | 2.335 | 19 | 0.002 | 5 | 0.431 | 5 | 0.033 | 0.153 | 5 |
| 13 | 1.898 | 12 | 0.001 | 17 | 0.344 | 17 | 0.015 | -0.04 | 16 |
| 14 | 1.285 | 1 | 0.003 | 2 | 0.576 | 1 | 0.015 | -0.249 | 1 |
| 15 | 1.842 | 11 | 0.003 | 4 | 0.471 | 2 | 0.035 | -0.143 | 6 |
| 16 | 2.222 | 17 | 0.001 | 14 | 0.386 | 13 | 0.045 | -0.114 | 8 |
| 17 | 1.677 | 6 | 0.001 | 12 | 0.422 | 6 | 0.011 | 0.156 | 4 |
| 18 | 1.558 | 3 | 0.001 | 20 | 0.362 | 15 | 0.015 | 0.112 | 9 |
| 19 | 2.371 | 20 | 0.001 | 11 | 0.327 | 20 | -0.018 | 0.008 | 20 |
| 20 | 1.796 | 8 | 0.002 | 6 | 0.413 | 10 | 0.013 | -0.097 | 10 |
9.5.1 Matching to Shirer maps.

As in Campbell et al. (2015) we compared our inferred latent maps, in this case the mean for the variational approximation to the posterior distribution of each basis vector, to those found in Shirer et al. (2012). The motivation for the matching being for validation against previously established resting state networks. Although a further effect is to help link the inferred maps to regions whose interpretation is already established (at least to some extent). The results of the comparison are presented in Table 9.8 in which the latent maps inferred in our SMF analysis are matched with Shirer maps. The latent map with the highest absolute correlation with the Shirer map is shown. The resulting matches are ordered in terms of the absolute correlations. Table 9.7 performs the same comparison but reversing the roles of the two sets of maps, so that each Shirer map is matched with a latent map.

Table 9.7 Shirer maps matched to latent maps.

<table>
<thead>
<tr>
<th>ShirerName</th>
<th>ShirerMap</th>
<th>Corr</th>
<th>Rank</th>
<th>ModelMap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auditory</td>
<td>2</td>
<td>0.24598597</td>
<td>1</td>
<td>14</td>
</tr>
<tr>
<td>Precuneus</td>
<td>9</td>
<td>0.20821625</td>
<td>2</td>
<td>19</td>
</tr>
<tr>
<td>Language</td>
<td>6</td>
<td>0.19898617</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Visuospatial</td>
<td>14</td>
<td>0.19560085</td>
<td>4</td>
<td>13</td>
</tr>
<tr>
<td>VDMN</td>
<td>13</td>
<td>0.19162165</td>
<td>5</td>
<td>20</td>
</tr>
<tr>
<td>HighVisual</td>
<td>5</td>
<td>0.18689552</td>
<td>6</td>
<td>17</td>
</tr>
<tr>
<td>PostSalience</td>
<td>8</td>
<td>0.1736484</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>DDMN</td>
<td>4</td>
<td>0.15679751</td>
<td>8</td>
<td>18</td>
</tr>
<tr>
<td>RECN</td>
<td>11</td>
<td>0.13089931</td>
<td>9</td>
<td>19</td>
</tr>
<tr>
<td>PrimVisual</td>
<td>10</td>
<td>0.12746644</td>
<td>10</td>
<td>12</td>
</tr>
<tr>
<td>AnteriorSalience</td>
<td>1</td>
<td>0.10408882</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>LECN</td>
<td>7</td>
<td>0.0883586</td>
<td>12</td>
<td>19</td>
</tr>
<tr>
<td>Sensorimotor</td>
<td>12</td>
<td>0.06800441</td>
<td>13</td>
<td>5</td>
</tr>
<tr>
<td>BasalGanglia</td>
<td>3</td>
<td>0.048556305</td>
<td>14</td>
<td>9</td>
</tr>
</tbody>
</table>
Table 9.8 Latent maps matched to Shirer maps.

<table>
<thead>
<tr>
<th>ModelMap</th>
<th>Corr</th>
<th>Rank</th>
<th>ShirerMap</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>0.24598597</td>
<td>1</td>
<td>2</td>
<td>Auditory</td>
</tr>
<tr>
<td>19</td>
<td>0.20821625</td>
<td>2</td>
<td>9</td>
<td>Precuneus</td>
</tr>
<tr>
<td>8</td>
<td>0.19898617</td>
<td>3</td>
<td>6</td>
<td>Language</td>
</tr>
<tr>
<td>13</td>
<td>0.19560085</td>
<td>4</td>
<td>14</td>
<td>Visuospatial</td>
</tr>
<tr>
<td>20</td>
<td>0.19162165</td>
<td>5</td>
<td>13</td>
<td>VDMN</td>
</tr>
<tr>
<td>18</td>
<td>0.19127625</td>
<td>6</td>
<td>2</td>
<td>Auditory</td>
</tr>
<tr>
<td>17</td>
<td>0.18689552</td>
<td>7</td>
<td>5</td>
<td>HighVisual</td>
</tr>
<tr>
<td>16</td>
<td>0.18167242</td>
<td>8</td>
<td>9</td>
<td>Precuneus</td>
</tr>
<tr>
<td>6</td>
<td>0.1811087</td>
<td>9</td>
<td>6</td>
<td>Language</td>
</tr>
<tr>
<td>10</td>
<td>0.17541996</td>
<td>10</td>
<td>2</td>
<td>Auditory</td>
</tr>
<tr>
<td>11</td>
<td>0.17380337</td>
<td>11</td>
<td>14</td>
<td>Visuospatial</td>
</tr>
<tr>
<td>5</td>
<td>0.1736484</td>
<td>12</td>
<td>8</td>
<td>PostSalience</td>
</tr>
<tr>
<td>2</td>
<td>0.1702254</td>
<td>13</td>
<td>2</td>
<td>Auditory</td>
</tr>
<tr>
<td>4</td>
<td>0.16214238</td>
<td>14</td>
<td>14</td>
<td>Visuospatial</td>
</tr>
<tr>
<td>9</td>
<td>0.14813633</td>
<td>15</td>
<td>6</td>
<td>Language</td>
</tr>
<tr>
<td>15</td>
<td>0.13992758</td>
<td>16</td>
<td>14</td>
<td>Visuospatial</td>
</tr>
<tr>
<td>3</td>
<td>0.13984765</td>
<td>17</td>
<td>5</td>
<td>HighVisual</td>
</tr>
<tr>
<td>7</td>
<td>0.1321702</td>
<td>18</td>
<td>8</td>
<td>PostSalience</td>
</tr>
<tr>
<td>12</td>
<td>0.12746644</td>
<td>19</td>
<td>10</td>
<td>PrimVisual</td>
</tr>
<tr>
<td>1</td>
<td>0.12072123</td>
<td>20</td>
<td>8</td>
<td>PostSalience</td>
</tr>
</tbody>
</table>

The Figure 9.6a shows a summary of latent maps which match most strongly with Shirer maps by correlation with the absolute value of the latent map.
(a) The figure shows the top ten basis vectors matched to the Shirer maps. Ranking runs from top-left down. A summary of each basis vector consisting of the axial, coronal and sagittal slices with the highest L2 norm is shown. This means the colour scales are comparable across features here.
9.5.2 Matching to Harvard Oxford Atlas

We also matched the latent spatial maps to the anatomical regions of the Harvard Oxford Atlases (HOA) supplied with the FSL software (Desikan et al., 2006; Frazier et al., 2005; Goldstein et al., 2007; Jenkinson et al., 2012; Makris et al., 2006). The regions covered are listed in Table 9.9, where we have introduced shortened names for the regions in order to save space elsewhere. These shortened names are unique and consistent within this document, but do not necessarily correspond to identifiers used elsewhere in the literature.
### Table 9.9 Key for Harvard Oxford Atlas regions. Laterally split regions merged.

<table>
<thead>
<tr>
<th>Code1</th>
<th>FullName1</th>
<th>Code2</th>
<th>FullName2</th>
</tr>
</thead>
<tbody>
<tr>
<td>AG</td>
<td>Angular Gyrus</td>
<td>PGad</td>
<td>Parahippocampal Gyrus, anterior division</td>
</tr>
<tr>
<td>B</td>
<td>Background</td>
<td>PGpd</td>
<td>Parahippocampal Gyrus, posterior division</td>
</tr>
<tr>
<td>COC</td>
<td>Central Opercular Cortex</td>
<td>POC</td>
<td>Parietal Operculum Cortex</td>
</tr>
<tr>
<td>CGad</td>
<td>Cingulate Gyrus, anterior division</td>
<td>PP</td>
<td>Planum Polare</td>
</tr>
<tr>
<td>CGpd</td>
<td>Cingulate Gyrus, posterior division</td>
<td>PT</td>
<td>Planum Temporale</td>
</tr>
<tr>
<td>CC</td>
<td>Cuneal Cortex</td>
<td>PG2</td>
<td>Postcentral Gyrus</td>
</tr>
<tr>
<td>FMC</td>
<td>Frontal Medial Cortex</td>
<td>PG3</td>
<td>Precentral Gyrus</td>
</tr>
<tr>
<td>FOC</td>
<td>Frontal Operculum Cortex</td>
<td>PC</td>
<td>Precuneous Cortex</td>
</tr>
<tr>
<td>FOC2</td>
<td>Frontal Orbital Cortex</td>
<td>SC</td>
<td>Subcallosal Cortex</td>
</tr>
<tr>
<td>FP</td>
<td>Frontal Pole</td>
<td>SFG</td>
<td>Superior Frontal Gyrus</td>
</tr>
<tr>
<td>HG</td>
<td>Heschl’s Gyrus</td>
<td>SPL</td>
<td>Superior Parietal Lobule</td>
</tr>
<tr>
<td>IFGpo</td>
<td>Inferior Frontal Gyrus, pars opercularis</td>
<td>STGad</td>
<td>Superior Temporal Gyrus, anterior division</td>
</tr>
<tr>
<td>IFGpt</td>
<td>Inferior Frontal Gyrus, pars triangularis</td>
<td>STGpd</td>
<td>Superior Temporal Gyrus, posterior division</td>
</tr>
<tr>
<td>ITGad</td>
<td>Inferior Temporal Gyrus, anterior division</td>
<td>SC2</td>
<td>Supracalcarine Cortex</td>
</tr>
<tr>
<td>ITGpd</td>
<td>Inferior Temporal Gyrus, posterior division</td>
<td>SGad</td>
<td>Supramarginal Gyrus, anterior division</td>
</tr>
<tr>
<td>ITGtp</td>
<td>Inferior Temporal Gyrus, temporoooccipital part</td>
<td>SGpd</td>
<td>Supramarginal Gyrus, posterior division</td>
</tr>
<tr>
<td>IC</td>
<td>Insular Cortex</td>
<td>TFCad</td>
<td>Temporal Fusiform Cortex, anterior division</td>
</tr>
<tr>
<td>IC2</td>
<td>Intracalcarine Cortex</td>
<td>TFCpd</td>
<td>Temporal Fusiform Cortex, posterior division</td>
</tr>
<tr>
<td>JLC</td>
<td>Juxtapositional Lobule Cortex</td>
<td>TOFC</td>
<td>Temporal Occipital Fusiform Cortex</td>
</tr>
<tr>
<td>LOCid</td>
<td>Lateral Occipital Cortex, inferior division</td>
<td>TP</td>
<td>Temporal Pole</td>
</tr>
<tr>
<td>LOCsd</td>
<td>Lateral Occipital Cortex, superior division</td>
<td>A</td>
<td>Accumbens</td>
</tr>
<tr>
<td>LG</td>
<td>Lingual Gyrus</td>
<td>A2</td>
<td>Amygdala</td>
</tr>
<tr>
<td>MFG</td>
<td>Middle Frontal Gyrus</td>
<td>C</td>
<td>Caudate</td>
</tr>
<tr>
<td>MTGad</td>
<td>Middle Temporal Gyrus, anterior division</td>
<td>CWM</td>
<td>Cerebral White Matter</td>
</tr>
<tr>
<td>MTGpd</td>
<td>Middle Temporal Gyrus, posterior division</td>
<td>H</td>
<td>Hippocampus</td>
</tr>
<tr>
<td>MTGtp</td>
<td>Middle Temporal Gyrus, temporoooccipital part</td>
<td>LV</td>
<td>Lateral Ventricular</td>
</tr>
<tr>
<td>OFG</td>
<td>Occipital Fusiform Gyrus</td>
<td>P</td>
<td>Pallidum</td>
</tr>
<tr>
<td>OP</td>
<td>Occipital Pole</td>
<td>P2</td>
<td>Putamen</td>
</tr>
<tr>
<td>PG</td>
<td>Paracingulate Gyrus</td>
<td>T</td>
<td>Thalamus</td>
</tr>
</tbody>
</table>
Matches were made by the cosine similarity, meaning the dot product of the vectors under L2 normalisation. Rankings were made in terms of the absolute value cosine similarity. Signs were reintroduced in reporting in order to indicate the direction of correlation between different regions within a component, with matched signs indicating positive correlation and opposing signs indicating negative correlation. The top five matches for each spatial map are listed in Table 9.10. This method has a substantial drawback in that larger HOA regions have a greater potential for containing sub-regions of positive and negative loading which would result in a low score; for instance the ‘Lateral Occipital Cortex, superior division’ is rather large. We nevertheless feel the labelling has some utility for enabling ‘reading’ of the spatial maps, but we should bear in mind that the top five scores may not always contain the regions that stand out visually but instead regions in which there is substantial, consistent activity.

Table 9.10 Basis matched to Harvard Oxford atlas regions. Sign indicates whether loading is positive or negative in that region.

<table>
<thead>
<tr>
<th>Basis vector</th>
<th>Match1</th>
<th>Match2</th>
<th>Match3</th>
<th>Match4</th>
<th>Match5</th>
</tr>
</thead>
<tbody>
<tr>
<td>w1</td>
<td>-PG2</td>
<td>PT</td>
<td>POC</td>
<td>-STGad</td>
<td>-SC</td>
</tr>
<tr>
<td>w2</td>
<td>-T</td>
<td>LOCsd</td>
<td>FP</td>
<td>-JLC</td>
<td>-HG</td>
</tr>
<tr>
<td>w3</td>
<td>-IFGpt</td>
<td>LOCid</td>
<td>-TFCad</td>
<td>OFG</td>
<td>POC</td>
</tr>
<tr>
<td>w4</td>
<td>-PT</td>
<td>FOC</td>
<td>STGpd</td>
<td>TOFC</td>
<td>-SPL</td>
</tr>
<tr>
<td>w5</td>
<td>H</td>
<td>MTGad</td>
<td>FP</td>
<td>-OFG</td>
<td>CGpd</td>
</tr>
<tr>
<td>w6</td>
<td>HG</td>
<td>PP</td>
<td>-FP</td>
<td>JLC</td>
<td>-OFG</td>
</tr>
<tr>
<td>w7</td>
<td>LG</td>
<td>H</td>
<td>PG</td>
<td>-MTGpd</td>
<td>JLC</td>
</tr>
<tr>
<td>w8</td>
<td>FP</td>
<td>JLC</td>
<td>LOCid</td>
<td>PG3</td>
<td>-LOCsd</td>
</tr>
<tr>
<td>w9</td>
<td>-IC2</td>
<td>-CC</td>
<td>-SGad</td>
<td>MTGpd</td>
<td>-LG</td>
</tr>
<tr>
<td>w10</td>
<td>LOCid</td>
<td>-IFGpo</td>
<td>HG</td>
<td>PC</td>
<td>PP</td>
</tr>
<tr>
<td>w11</td>
<td>-IFGpo</td>
<td>-ITGpd</td>
<td>P</td>
<td>LOCid</td>
<td>-SGpd</td>
</tr>
<tr>
<td>w12</td>
<td>-MTGpd</td>
<td>CC</td>
<td>SPL</td>
<td>TP</td>
<td>SC</td>
</tr>
<tr>
<td>w13</td>
<td>LOCsd</td>
<td>MTGad</td>
<td>PGpd</td>
<td>SC2</td>
<td>-HG</td>
</tr>
<tr>
<td>w14</td>
<td>ITGtp</td>
<td>-MTGpd</td>
<td>-FOC2</td>
<td>-A2</td>
<td>-TFCpd</td>
</tr>
<tr>
<td>w15</td>
<td>-MTGpd</td>
<td>IFGpo</td>
<td>-TP</td>
<td>ITGtp</td>
<td>-SC</td>
</tr>
<tr>
<td>w16</td>
<td>-P</td>
<td>-IFGpo</td>
<td>PG</td>
<td>PG2</td>
<td>-MTGtp</td>
</tr>
<tr>
<td>w17</td>
<td>-ITGpd</td>
<td>IC2</td>
<td>-PG2</td>
<td>-OP</td>
<td>-MTGad</td>
</tr>
<tr>
<td>w18</td>
<td>-PC</td>
<td>-PGpd</td>
<td>-IFGpt</td>
<td>-CWM</td>
<td>-OFG</td>
</tr>
<tr>
<td>w19</td>
<td>COC</td>
<td>-A2</td>
<td>PG3</td>
<td>-TOFC</td>
<td>IFGpt</td>
</tr>
<tr>
<td>w20</td>
<td>A</td>
<td>LV</td>
<td>PGad</td>
<td>PG</td>
<td>MTGtp</td>
</tr>
</tbody>
</table>
### 9.5.3 Regression against covariates

We regressed the synchronisation scores for each latent dimension against three covariates separately: age, gender and Cattell score (an intelligence measure). The results are presented in Table 9.11. Our investigation of whether the results of Campbell et al. (2015) could be replicated led to a focus on the age covariate. Plots of the regressions of synchronisation on age are provided in Figure 9.7a, which also include the values of the slope coefficients.

Table 9.11. Latent dimensions scored by regression of synchronisation on covariates (sync $\sim c + b \times \text{age}$).

<table>
<thead>
<tr>
<th>K</th>
<th>p-value</th>
<th>Slope</th>
<th>$R^2$</th>
<th>Rank (by p-value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0012</td>
<td>0.041</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0017</td>
<td>0.0717</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>0.0332</td>
<td>-0.0005</td>
<td>0.0076</td>
<td>19</td>
</tr>
<tr>
<td>4</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0013</td>
<td>0.0608</td>
<td>13</td>
</tr>
<tr>
<td>5</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0015</td>
<td>0.0799</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.001</td>
<td>0.0655</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>0.2809</td>
<td>-0.0003</td>
<td>0.0019</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0021</td>
<td>0.1475</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0011</td>
<td>0.0325</td>
<td>16</td>
</tr>
<tr>
<td>10</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0014</td>
<td>0.0658</td>
<td>11</td>
</tr>
<tr>
<td>11</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.002</td>
<td>0.1696</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>0.0062</td>
<td>-0.0006</td>
<td>0.0125</td>
<td>18</td>
</tr>
<tr>
<td>13</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0014</td>
<td>0.0675</td>
<td>10</td>
</tr>
<tr>
<td>14</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0019</td>
<td>0.1059</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.002</td>
<td>0.1037</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.001</td>
<td>0.03</td>
<td>17</td>
</tr>
<tr>
<td>17</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0019</td>
<td>0.0928</td>
<td>5</td>
</tr>
<tr>
<td>18</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0014</td>
<td>0.0688</td>
<td>8</td>
</tr>
<tr>
<td>19</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0008</td>
<td>0.0341</td>
<td>15</td>
</tr>
<tr>
<td>20</td>
<td>$&lt; 10^{-4}$</td>
<td>-0.0016</td>
<td>0.0681</td>
<td>9</td>
</tr>
</tbody>
</table>

It can be seen from Figure 9.7a that the slope of the regression is negative across all components. In this respect we would suggest that one of the findings of the earlier work Campbell et al. (2015) can be reproduced under different modelling assumptions: that inter-subject synchronisation decreases with age. Our use of the term replication here is in the
sense of the same data leading to the similar findings under a substantially altered analysis methodology. Also note that the synchronisation scores reported here were to the overall group mean of a larger sample of subjects than the earlier work ($N = 600$ vs $N = 218$) and so it is to be expected that overall synchronisation to a group mean would be lower.

To comment on the size of these effects, the negative relationship between synchronisation and age here is a slope of around $-0.002$. Between the minimum age, 18, and the mean age, 54.4, the expected decline in synchronisation would be $-0.07$. The mean synchronisation was around 0.405 so representing variation of about 17%. We feel this makes the effect large enough to warrant discussion. However, we can also see from Figure 9.7a that the variation between individuals at each age level is comparable to or greater than the difference between the end points of the trend line. This implies that in terms of predicting how synchronised a particular individual is to the group, the effect will be of limited use.

A further insight to be gained from Figure 9.7a is that there are some individuals who appear to be completely unsynchronised with the group. These are usually few enough
in number that further study of the synchronisation statistics could provide an avenue for automatic identification of anomalies such as caused by the scanner or by head motion, or individuals who were not focussed on the task. This idea could perhaps be validated in future work by additionally obtaining a rating from each participant on leaving the scanner regarding whether they had followed the task or not and investigating the predictive capacity of synchronisation scores.

9.5.4 Rendering of components

This section provides plots of the spatial maps and associated subject specific time-courses for all components, together with a summary of many of the statistics presented in the tables above.

The spatial maps are visualised via axial slicing. These maps represent the mean of the variational approximation to the posterior distribution to $w_k$. As in our MCMC work of section 7.6.2 we do not present thresholded images since this would distort the qualitative characteristics of the maps and because a $z$-statistic interpretation would be inappropriate.

Note that the coloured bar at the bottom of each spatial map serves both to center the colour scale to be white, while the red side of the colour bar points to the right hand side of the brain so that the images are presented in ‘radiological’ format.

The subject specific time-courses are presented for all $N = 600$ subjects for each component, rendered with a thin line so that the mass of activity is visible. Each is a complete series of 193 values, with each tick representing a 2.47 second gap. The thick coloured line represents the sequence of points where speech was present in the movie provided as stimulus to the participants. The sequence of speech impulses having been convolved with the model haemodynamic response function as used in Campbell et al. (2015). For ease of interpretation, the signs of the subject specific time-series have been flipped so that the mean correlation with the talking time-series is non-negative. The underlying correlation is reported in Table 9.6. The signs of the spatial maps have not been altered. We present the talking time series in these graphs in order to alleviate the need for repetition of very similar graphs at a later point. However we stress that it is not necessary to interpret the ‘talking’ time-series for every dimension.

The statistics and summary information presented include the synchronisation score for each feature and, in brackets, its rank, and similar for spatial smoothness and magnitude scores. Being rank 1 is ‘best’ under all metrics. Also presented are the top five matches by cosine similarity to anatomical features in the Harvard Oxford Atlas, together with the direction of the match.
We would again caution that the signs may be flipped through multiplication by a negative component; as is the case with all other commonly used matrix factorisations in neuroimaging. Therefore the sign attached to both spatial maps and matches with HOA regions should be read as follows: within a dimension activity in regions with the same sign are positively correlated, activity in regions with opposing signs are negatively correlated. The average direction of the components at different points in time can be read from the graph of the subject specific time-series by an interested reader.
Fig. 9.8 Report for latent dimension 1. Smoothness: 2.165 (15). Synchronisation: 0.35 (16). Magnitude: 0.007 (19). HOA matches: -PG2, PT, POC, -STGad, -SC.
Fig. 9.9 Report for latent dimension 2. Smoothness: 2.287 (18). Synchronisation: 0.446 (4). Magnitude: 0.015 (11). HOA matches: -T, LOCsd, FP, -JLC, -HG.
Fig. 9.10 Report for latent dimension 3. Smoothness: 1.812 (9). Synchronisation: 0.388 (12). Magnitude: 0.013 (13). HOA matches: -IFGpt, LOCid, -TFCad, OFG, POC.
Fig. 9.11 Report for latent dimension 4. Smoothness: 2.181 (16). Synchronisation: 0.42 (9). Magnitude: 0.021 (7). HOA matches: -PT, FOC, STGpd, TOFC, -SPL.
Fig. 9.12 Report for latent dimension 5. Smoothness: 1.834 (10). Synchronisation: 0.412 (11). Magnitude: 0.007 (17). HOA matches: H, MTGad, FP, -OFG, CGpd.
Fig. 9.13 Report for latent dimension 6. Smoothness: 1.676 (5). Synchronisation: 0.378 (14). Magnitude: 0.007 (18). HOA matches: HG, H, -FP, -JLC, -OFG.
Fig. 9.14 Report for latent dimension 7. Smoothness: 2.127 (13). Synchronisation: 0.337 (18). Magnitude: 0.039 (2). HOA matches: LG, PP, PG, -MTGpd, JLC.
Fig. 9.15 Report for latent dimension 8. Smoothness: 1.544 (2). Synchronisation: 0.466 (3). Magnitude: 0.016 (9). HOA matches: FP, JLC, LOCi, PG3, -LOCsd.
Fig. 9.16 Report for latent dimension 9. Smoothness: 2.132 (14). Synchronisation: 0.422 (7). Magnitude: 0.072 (1). HOA matches: -IC2, -CC, -SGad, -MTGpd, -LG.
Fig. 9.17 Report for latent dimension 10. Smoothness: 1.619 (4). Synchronisation: 0.336 (19). Magnitude: 0.016 (10). HOA matches: LOCid, -IFGpo, HG, PC, PP.
Fig. 9.18 Report for latent dimension 11. Smoothness: 1.719 (7). Synchronisation: 0.42 (8). Magnitude: 0.014 (12). HOA matches: -IFGpo, -ITGpd, P, LOCid, -SGpd.
Fig. 9.19 Report for latent dimension 12. Smoothness: 2.335 (19). Synchronisation: 0.431 (5). Magnitude: 0.026 (4). HOA matches: -MTGpd, CC, SPL, TP, SC.
Fig. 9.20 Report for latent dimension 13. Smoothness: 1.898 (12). Synchronisation: 0.344 (17). Magnitude: 0.007 (16). HOA matches: LOCsd, MTGad, PGpd, SC2, -HG.
Fig. 9.21 Report for latent dimension 14. Smoothness: 1.285 (1). Synchronisation: 0.576 (1). Magnitude: 0.022 (6). HOA matches: ITGtp, -MTGpd, -FOC2, -A2, -TFCpd.
Fig. 9.22 Report for latent dimension 15. Smoothness: 1.842 (11). Synchronisation: 0.471 (2). Magnitude: 0.024 (5). HOA matches: -MTGpd, IFGpo, -TP, ITGtp, -SC.
Fig. 9.23 Report for latent dimension 16. Smoothness: 2.222 (17). Synchronisation: 0.386 (13). Magnitude: 0.01 (15). HOA matches: -P, -IFGpo, PG, PG2, -MTGtp.
Fig. 9.24 Report for latent dimension 17. Smoothness: 1.677 (6). Synchronisation: 0.422 (6). Magnitude: 0.012 (14). HOA matches: -ITGpd, IC2, -PG2, -OP, MTGad.
Fig. 9.25 Report for latent dimension 18. Smoothness: 1.558 (3). Synchronisation: 0.362 (15). Magnitude: 0.006 (20). HOA matches: -PC, -PGpd, -IFGpt, -CWM, -OFG.
Fig. 9.27 Report for latent dimension 20. Smoothness: 1.796 (8). Synchronisation: 0.413 (10). Magnitude: 0.019 (8). HOA matches: A, LV, PGad, PG, MTGtp.
9.6 Commentary on results

It is not our aim here to venture detailed psychological interpretations of the components. Instead we focus on the extent to which our methodology picks out reasonable anatomical features that might support further neuroscientific interpretation.

We saw in section 9.5.4 that the basis vectors yield detailed images of neuroanatomy. A number of features stand out on a macro level: the basis vectors are visually smooth with contiguous regions of activation, as we might anticipate under the model; the basis vectors are symmetric; areas of (de-)activation are generally in regions dominated grey matter, while white matter and cerebrospinal fluid are relatively muted; motion artefacts such as halos are not present. All these facets are suggestive of functional networks capable of bearing reasonable psychological interpretations.

We speculate that this is the case for several reasons. The large number of subjects involved means that movement or other artefacts present in a single subject’s data contribute relatively little to the overall likelihood of the data, so that inference will push basis vectors towards explaining variation across many different subjects. The minibatch method of variational inference provides a reinforcing explanation. Suppose one of the inferred basis vectors represented an artefact of movement specific to one subject. The minibatch update for any batch not containing that subject would not emphasise that vector. Thus the artefact basis vector would be increasingly overwritten during the minibatch optimisation process; even if the artefact vector introduced a large mean square error.

Component 14 was ranked first both in terms of synchronisation and smoothness. With respect to synchronisation, we can see from Figure 9.21 that the subject wise time series closely track speech in the film (shown in bold red). The related spatial map shows concentrated activity in the temporal lobe, which is associated with hearing, language and vision. Component number 15 was the second most synchronised component with time series, shown in Figure 9.22, that do not track speech closely but consistently shown subjects to be either above or below zero together for sustained periods of time. The associated spatial map is more complex in this case, with negative loadings on the temporal pole and on a pattern similar to the default mode network (Buckner et al., 2008) consisting of precuneus, lateral parietal areas and the frontal pole and positive weightings.

Components 14 and 15 also illustrate the benefits of the lack of pre-smoothing of data since we can see certain structures in much detail. In the case of component 14 we can see individual gyri separated by white matter suggesting that when there is sufficient signal closely neighbouring areas will not ‘bleed’ into one another. In the case of component 15 we can see a detailed impression of the precuneous region, though it extends into contiguous regions.
outside the HOA delineated precuneous (negative area of activation with the appearance of a fleur-de-lys or club in the middle of the back of the brain).
Chapter 10

Discussion

10.1 Overview

In this thesis we have considered analyses of multivariate datasets via matrix factorisation. These methods reveal temporal and spatial structure when applied to neuroimaging data. In chapter 2 we saw that multiple stages of factorisation were often involved, most commonly using Principal Components Analysis and Independent Components Analysis. Chapter 3 considered limitations of current methods. Two stage matrix factorisation methods for group data generally optimised different objectives at each stage. These objectives were not necessarily consistent with one another, leading us to search for methods able to provide a unified analysis. We also considered the implications of orthogonality constraints on the output of factorisation models, and the potential for distortion of true underlying functional networks.

We identified opportunities for improving the handling of uncertainty around the number of latent basis vectors. An assumption of popular methods was a voxel-wise time series interpretation of fMRI data. In particular, if it is more appropriate to model fMRI data as observations of spatial vectors then dimension estimation based on a time series interpretation may fail as the empirical eigenspectrum could be dominated by noise. Furthermore, if the time series interpretation were appropriate but spatial dependence present, the decay of the eigenspectrum has the potential to lead to incorrect estimates of latent dimensionality. The computational complexity of the empirical eigendecomposition also posed a challenge to single stage matrix factorisation for larger datasets. These potential hazards motivated a search for methods which did not rely on empirical eigenspectra.

We did not find the voxel-wise time-series approach to fMRI conducive to the handling of spatial dependence across the brain. A common practice resulting from this limitation was reliance on strong spatial smoothing of data prior to analysis, using assumed values for
parameters. Visual contiguity of activity clusters is assured by averaging the values of nearby voxels; however this comes at the cost of blurring of the data. As brain images are high dimensional, explicit modelling of spatial correlation structure is challenging. In chapter 4 we provided a computationally efficient approach to handling multivariate normal random vectors. This utilised fast linear operators as the eigenvectors of nondiagonal covariance matrices, while graph theoretic results enabled the extension of models to spatial data. In the case of a spatially autoregressive model for volumetric images we saw that the model had an interpretation as a Gaussian Markov Random Field in which each voxel was connected to its six neighbours. While our focus was on neuroimaging, the machinery will have utility for sparse factorisation of general data under spatio-temporal constraints on the basis.

We developed a Bayesian nonparametric model for matrix factorisation, incorporating an efficient treatment of spatial correlation in chapter 5. This saw the level of spatial smoothness inferred from the data, rather than being assumed by the researcher. It further enabled smoothing to be applied at the level of individual basis vectors, rather than globally as a preprocessing step. While the extension in the use of the Indian Buffet Process from Knowles and Ghahramani (2007) provided an alternate approach to latent dimension estimation. In order to scale the Markov Chain Monte Carlo inference approach to neuroimaging data, we developed in chapter 6 a strategy for parallel Gibbs sampling of certain classes of graphical models.

Chapter 7 saw our first application to neuroimaging data, involving $N = 30$ subjects from the Cam-CAN dataset Shafto et al. (2014). Useful results were obtained under an informative prior to limit the number of basis vectors. This approach, however did not scale computationally to the degree required, and we discuss the issues at greater length below. Metrics to assist with the ranking and identification of components of interest were provided. These included: the smoothness of the spatial basis vectors, which favours maps with contiguous regions of activation; inter-subject synchronisation, which identifies dimensions whose time series of activation are consistent across subjects; and magnitude, which measures the scale of the contribution a component makes to the data and was intended to replace ‘percentage of variance explained’ for non-orthonormal bases.

In order to undertake larger scale analyses, we proposed a Structured Matrix Factorisation (SMF) model inferred via variational Bayes in chapter 8. This combined a number of VB techniques to handle both large datasets and non-conjugate spatial models. This led to a larger neuroimaging analysis in chapter 9 using $N = 600$ subjects from the Cam-CAN dataset; this scale is a limitation of the dataset rather than the SMF method. We found the smoothness and synchronisation metrics to be useful for selecting interpretable components. The synchronisation measure relies on subjects having a similar response to a stimulus, and
so has utility in analyses of data generated under naturalistic stimuli or block designs, and we anticipate would have less utility for resting state data. We did not see a metric based on component magnitude selecting interpretable components reliably, as physiological or scanner artefacts can have a large magnitude.

10.2 Dimensionality of neuroimaging data

Non-parametric modelling provided insight into the quality of linear approximations to real neuroimaging data. We saw in the applied setting that an uninformative prior on the observation noise level led to a rapidly expanding set of basis vectors. This caused model size and computation times to increase to the point where we were unable to run MCMC chains to convergence under the uninformative prior.

A partial solution was to use an informative prior to set a floor on the reconstruction accuracy as a fraction of the data sum of squares. We were able to run MCMC chains to convergence under such priors. The MCMC algorithm then determined an appropriate dimensionality for the resulting approximation to achieve that level of accuracy. This provides a similar dimensionality estimation criterion to that used in PCA, percentage of variance explained, but without the employment of eigenvalues. However, the computational limitations of MCMC were still sufficient to motivate work on an alternative approach.

On the other hand, the linear approximations of the data in these bases produced very small residuals. There are two possible causes for this. Consider brain images as points in an \( V \) dimensional space. The first is that the data truly lie very near to a \( K \) dimensional linear subspace of the ambient space, and that only computational constraints prevented us from obtaining MCMC results under that prior. A second interpretation is that the data lie very near to a nonlinear manifold, a linear approximation of which accounts for most of the variance in the data. The first interpretation seems too hopeful, that reality fits perfectly our assumptions. The second would seem more realistic, and could provide an explanation for an increasing number of basis vectors, namely that the nonparametric model has the facility to keep improving the fit of a linear manifold to a nonlinear one by the addition of further basis vectors.

The limitations of MCMC motivated the choice in chapter 8 to adopt a fixed dimensionality inferred with Variational Bayes. In simulations we saw that the ELBO was helpful in identifying the correct latent dimensionality. However, for higher dimensional data under a normalised penalty on the basis vectors, we found that the utility of the ELBO for model selection declined. As discussed in sections 8.6 and 9.4 we found that the ELBO was not necessarily comparable between different models, a phenomenon stemming from the depen-
dence of the bound on the prior and variational approximation, as well as its nature as bound rather than an estimate. In the end we alighted on a strategy of selecting dimensionality under an IID model using the ELBO as a criterion, and presenting results of a spatially correlated model with that dimensionality due to the superior properties in terms of spatial smoothness.

10.3 Comparison of MCMC and VB results

It is constructive to compare the results from the Variational Bayes approach to inference of a matrix factorisation model the results using Markov Chain Monte Carlo based inference. The most notable difference is the vast advantage in terms of computational efficiency of Variational Bayes over Markov Chain Monte Carlo. The analyses of neuroimaging data in sections 9.4 and 9.5 used 600 subjects and required around a day to fit models of all the orders reported on. Whereas our largest MCMC run used data from 30 subjects and required on the order of several weeks of computation time. Hence VB fulfils a role which MCMC cannot match, even with the tailored parallelisation scheme of chapter 6.

The primary mechanism for the computational efficiency of Variational Bayes here derives from the minibatch approach of Hoffman et al. (2013). In Gibbs sampling every component in $X$ must be resampled in order to complete an iteration before resampling of $W$. This means much time is spent sampling elements of $X$ conditional on a value of $W$ which we already know may be due to change radically given previously sampled elements of $X$. In fact we are permitted to resample $W$ multiple times within an iteration, but doing so counts only as a single iteration for MCMC purposes. Under VB only a fraction of the variational parameters for $X$ need be updated before we are able to update the parameters for $W$.

The tradeoff for increased speed is a need to set optimisation parameters which are not present in the Gibbs sampler, in order to compensate for differences between the optimal parameters implied by different minibatches. In our experience this is not too onerous, and we should recall that non-Gibbs MCMC methods such as Metropolis-Hastings also require parameters for transition proposals.

An important difference in the results from the models, to our mind, is the lower % of variance explained under the VB approach. Our MCMC results achieved accurate reconstruction performance. However, the IBP based matrix factorisation inferred by MCMC was allowed to have a relatively larger number of basis vectors per participant. We explained in chapter 7 the necessity of a strong prior on the noise level to prevent the introduction of large numbers of basis vectors.

In broad terms the largest model fitted in section 9.4 had $K = 100$ so that with $N = 600$ the number of basis vectors per subject was $1/6$, and for the model reported in detail in section
9.5 with $K = 20$ there were $1/30$ basis vectors for each subject. Whereas in the MCMC based model on a smaller sample of $N = 30$ we obtained convergent chains with roughly $K = 40,\ldots,90$ giving between $4/3$ and $3$ basis vectors per subject. This would suggest at least some of the reconstruction performance of the MCMC method can be explained by simply having a larger model relative to sample size. Fitting a similarly complex model under SMF would require between $K = 800, K = 2400$ basis vectors, if scaled linearly. Or if an IBP type model were appropriate, and the number of basis vectors required increases logarithmically with sample size, then $K$ in the low hundreds might suffice.

A further observation regarding the SMF models presented in chapter 9, and perhaps related to the lower ratio of basis vectors per subject, is the greater utilisation of basis vectors between subjects. This means that the vectors must represent patterns common between many subjects. Indeed the visualised spatial maps showed loading on grey matter regions and a high level of symmetry. We were satisfied with the visually crisp highlighting of neuroanatomical features such as the precuneus and auditory regions. While the subject specific time-series showed a good deal of synchronised activity; with very few extreme levels of activation specific to single subjects. Both characteristics would seem to indicate the biological plausibility of the extracted components and spatial maps.

### 10.4 Further work

The approaches set out in this thesis could support a number of developments. The model of chapter 5 can handle subject specific variation such as movement and other artefacts by the addition of subject specific components. This manifested itself in the MCMC neuroimaging results of chapter 7, where a large number of basis vectors were fit. We saw in that analysis, and in others not reported, that large numbers of basis vectors often translated into certain basis vectors representing subject specific variation. This suggests utility as a denoising method where the nonparametric approach is used on groups of subjects. The data would then be reconstructed while discarding subject-specific basis vectors. Some soft constraint on the number of basis vectors might be required in order to ensure that at least some components were shared amongst subjects. This approach would have the advantage of providing a clear numerical criterion on which latent dimensions were artefacts, rather than requiring expert judgement.

The VB results on neuroimaging set out in chapter 9 used a time course of speech that had been derived by convolving instances of speech in the film with a model hemodynamic response function. This quantified one aspect of an otherwise free-form, ‘naturalistic’ stimulus. The most synchronised component clearly tracked the speech time series. This
suggests that deriving further ways of quantifying naturalistic stimuli and using these for ranking of the components would be a viable metric. Our usage was post-inference, but it would be possible to incorporate such inputs as the mean vectors in a mixture of multivariate normals in order to assess how well the vectors matched the inputs. This might also serve to stabilise positive and negative signs in the inferred basis.

One of our aims at the beginning of this work was to involve covariates such as demographics within the factorisation model. This goal might be realised through the use of covariates in the spike element of the spike-and-slab component model. That is, we could let the spike element for the $k^{th}$ component, $b_k$, depend on covariates, $z$, such as age by forming a linear predictor $\eta_k = \beta_k^T z$ and modelling $p(b_k = 1) = \expit(\eta_k)$. This technique would be an adaptation to the factorisation setting of a similar technique used in mixture modelling (Papageorgiou et al., 2014).

With data obtained in a movie watching task, we observed that the VB results were most useful for identifying regions related to audio and visual processing. An idea arises from block experimental designs, which provide contrast between the brain regions used in ‘treatment’ and ‘control’ settings. Translating this to the paradigm of naturalistic stimuli, we would suggest that including data from subjects under a variety of different stimuli could provide a more fully rounded picture of functional networks. By variety we mean stimuli and tasks which would be thought to involve quite different functional networks. To illustrate, if it is fair to assume that movie watching does not require much executive control, then use of a task which required substantial executive control might lead to superior separation of functional networks involved in both tasks; since executive control networks only slightly stimulated by the movie would be better separated out from audiovisual networks. This has potential utility whether or not responses to the contrast task were themselves of scientific interest.

Selection of model dimensionality is a persistent challenge. A path around the challenge might involve averaging of SMF models of different orders and covariance structures. For instance the difference between two groups in the most synchronised, the smoothest, or the dimension most correlated with a pre-specified seed could be measured in each of many models and the results aggregated. This would require additional work specifying the outcomes or groups of interest, but reduce the burden of model selection and the challenge of visualising many high dimensional objects. It would further require a metric by which to average models, and we saw that the natural choice for SMF, the ELBO, behaved very differently under different covariance structures. This could suggest the use of an alternative metric, such as averaging by use of smoothness or synchronisation scores (a scheme reminiscent of Thompson sampling).
References


References


References


References


