Regularised Inference for Changepoint and Dependency Analysis in Non-Stationary Processes

Alexander James Gibberd

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
of
University College London

Department of Statistical Science
Department of Security and Crime Science

September 2017

I, Alexander James Gibberd confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

Multivariate correlated time series are found in many modern socio-scientific domains such as neurology, cyber-security, genetics and economics. The focus of this thesis is on efficiently modelling and inferring dependency structure both between data-streams and across points in time. In particular, it is considered that generating processes may vary over time, and are thus non-stationary. For example, patterns of brain activity are expected to change when performing different tasks or thought processes.

Models that can describe such behaviour must be adaptable over time. However, such adaptability creates challenges for model identification. In order to perform learning or estimation one must control how model complexity grows in relation to the volume of data. To this extent, one of the main themes of this work is to investigate both the implementation and effect of assumptions on *sparsity;* relating to model parsimony at an individual time-point, and *smoothness;* how quickly a model may change over time.

Throughout this thesis two basic classes of non-stationary model are studied. Firstly, a class of piecewise constant Gaussian Graphical models (GGM) is introduced that can encode graphical dependencies between data-streams. In particular, a group-fused regulariser is examined that allows for the estimation of changepoints across graphical models. The second part of the thesis focuses on extending a class of locally-stationary wavelet (LSW) models. Unlike the raw GGM this enables one to encode dependencies not only between data-streams, but also across time. A set of sparsity aware estimators are developed for estimation of the spectral parameters of such models which are then compared to previous works in the domain.

Note on Writing Style

This thesis predominantly follows a third person narrative, where throughout this thesis I use "we" to relate to the reader and myself. At some places I use the term "I" to relate to my own thoughts and beliefs.

List of Publications

This thesis is based in part on the following papers which are published in peer-reviewed journals/proceedings:

- A. Gibberd and J.D.B Nelson, 'Group-fused Graphical Lasso for Changepoint estimation in Multivariate Time-series', *IMA International Conference on Mathematics in Signal Processing*, 2014
- A. Gibberd and J.D.B Nelson, 'High Dimensional Changepoint Detection with a Dynamic Graphical Lasso', *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2014
- A. Gibberd and J.D.B Nelson, 'Sparsity in the Multivariate Wavelet Framework: A Comparative Study Using Epileptic Electroencephalography Data', *IET Intelligent Signal Processing*, 2015 (Chapter 6)
- A. Gibberd and J.D.B Nelson. 'Estimating Multi-Resolution Dependency Graphs within a Locally Stationary Wavelet Framework', *IEEE Global Conference on Signal & Information Processing*, 2015 (Chapter 6)
- A. Gibberd and J.D.B Nelson, 'Regularised Estimation of 2D-Locally Stationary Wavelet Processes', *IEEE Workshop on Statistical Signal Processing*, 2016 (Chapter 6)
- A. Gibberd and J.D.B Nelson, 'Estimating Dynamic Graphical Models from Multivariate Time-Series Data: Recent Methods and Results', *Lecture Notes on Artificial Intelligence*, 2016 (Chapters 2, 3)
- A. Gibberd and J.D.B. Nelson, 'Regularized Estimation of Piecewise Constant Gaussian Graphical Models: The Group-Fused Graphical Lasso', *Journal of Computational & Graphical Statistics*, 2017 (Chapter 3)
- A. Gibberd, M. Evangelou and J.D.B. Nelson, 'The Time-Varying Dependency Patterns of NetFlow Statistics', *IEEE International Conference on Data Mining*, 2016 (Chapter 3)

In memory of Dr James D.B. Nelson

Acknowledgements

First and foremost, I would like to dedicate this thesis to my late supervisor Dr James Nelson: Without James' unique mix of patience, creativity and passion for knowledge this thesis would never have been completed. I am, and always will be grateful for the time, freedom, and opportunities he afforded me throughout our time working together.

A very special gratitude goes to Defence Science Technology Laboratory (DSTL) for funding this work, and especially to Ralph for being so hospitable and the great discussions. I would also like to thank my second supervisor Ricardo Silva, collaborators; Matt Nunes, Sandipan Roy, Marina Evangelou, Niall Adams, and Marcela Mendoza. Thanks to Alex Immer for helping me navigate the horrors of entrepreneurship. Thanks to Robbie for helping with the proofreading effort.

Equally, while there are many who have helped me with my research, there are many more I am grateful for making my time as a PhD actually enjoyable! I'm eternally thankful to have had the Luncheon crew to provide a constant source of mid-day procastrination, coffee, and beer drinking. To all the guys in the stats office, I'm not sure how much work I actually did at my office (at that rate I should also thank the staff of Costa), but it was always a fun place to be. I'd also like to say thanks for JB and Hojjat for all the travels and chats, especially towards the end.

Last but not least, I am grateful for the continued and unconditional support of my family; June, Al, Ham, and Josie. Also a big shout out to Auntie Sue and Uncle Ray who helped me stave off starvation while writing this document. Finally, I really want to thank Anastasia for all the care, love and food she has provided throughout the process of writing this document. I'm very much looking forward to spending weekends together again.

Contents

Chapter 1

Introduction

High-dimensional correlated time-series are ubiquitous in many real world applications, from observations of how blood flows throughout the brain to understanding traffic flows across computer networks. The continuous development of sensing technologies places new requirements on the tools and methodologies that are used to gain understanding from data. Not only are new datasets acquired faster, and in greater resolutions, but they also measure more aspects of the world around us. In many applications, the number of features or variables that one may measure often outnumber the volume of points at which these may be sampled. Such *high-dimensional* situations pose serious challenges for statistical estimation due to the inherently large number of degrees of freedom associated with traditional models.

To avoid overfitting, in the high-dimensional setting one is required to make assumptions about the dynamics and dependency of variables. Typically, these will be encoded by a statistical model. However, even for very simple statistical models like linear regression, the number of model parameters may grow faster than data can be collected. To enable estimation of the model, it is often assumed that the data may be described by a smaller subset of the parameters required in the ambient dimension. While such assumptions help stabilise our model statistically; they give rise to challenges in computation associated with how we search the model space for the best set of parameters. These challenges are inextricably linked. For example, certain statistical models and assumptions allow for a simplified computational search, but potentially at the

expense of statistical performance. The assumptions that we make in order to identify a model often affect the level of insight one may obtain from the underlying data. It is therefore of paramount importance to consider the theoretical and empirical consequences of assumptions, both in a computational and statistical sense.

1.1 Motivational Applications

To give an idea of how and where methods developed in the thesis may be used, we here discuss two areas of critical importance to society; namely, neuroscience, and cyber-security. Both of these applications are examined to various depths further in the thesis, particularly, Sections 3.4 and 7.4.

1.1.1 Understanding Brain Activity

There are several neuroscientific objectives associated with the analysis of neurological data. For example, one may be interested in localising regions of the brain linked to certain tasks, determining how regions of the brain interact (functional mapping), or making predictions about psychological or disease status. In this thesis, the primary topic of interest is modelling dependency structure between variables; concerning neurological analysis this strongly relates to the objective of functional mapping within the brain. Specifically, if one represents the relationships between parts of the brain as a network, then robustly inferring this network structure is of paramount importance. Not only do the methods of this thesis aim to address the challenge of infering network strcutre, but also extend this to the setting where these dependencies can evolve over time.

Several forms of sensing technology may be used to measure brain activity: techniques such as *magnetoencephalography (MEG)* and *electroencephalography (EEG)* rely on sensing the magnetic and electrical activity within the brain, whereas *functional magnetic resonance imaging (fMRI)* data uses blood flow in the brain as a proxy for activity.

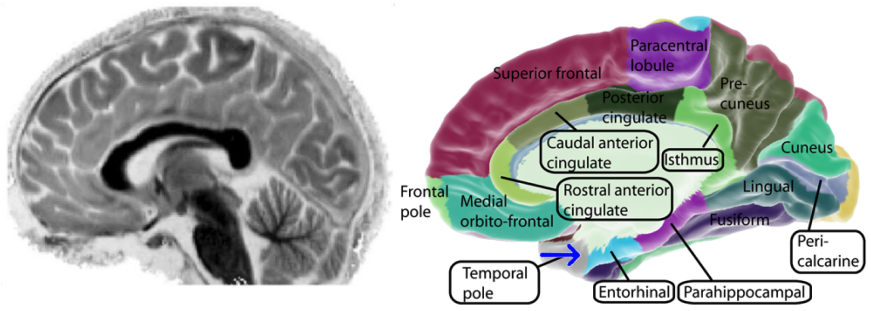


Figure 1.1.1 – Left: Integrated MRI image of my brain displaying high spatial resolution, image produced by 3-T Siemens Magnetom Trio MRI as part of the study by Zeki et al. (2014). Right: Example of anatomical regions of interest over which voxel wise data may be integrated. For each region, a time-series may be constructed (*Source: Wikipedia; Hagmann P, Cammoun L, Gigandet X, Meuli R, Honey CJ, et al.*).

FMRI Data

If we first consider fMRI data, it is possible to get very high spatial resolution, but low resolution in time[1]. For example, a typical fMRI analysis may capture around $p = 100,000$ voxels (the equivalent of a pixel in an image) over a period of several minutes, resulting in around $T = 200-2000$ time points. To maintain interpretability this activity is often aggregated into larger regions of interest, an example of possible aggregations is demonstrated in Figure 1.1.1. However, even in the aggregated setting, when one considers the estimation of correlation (a simple measure of variable dependency), the required matrix may have many parameters; a $p = 60$ dimensional correlation matrix requires the estimation of $d = p(p-1)/2 = 1770$ parameters. If one further considers that this matrix may change over time, then the problem is still clearly well in the high-dimensional regime.

Traditionally, the estimation of such networks assumes stationarity, i.e. a dependency network for the regions would be estimated assuming that it does not change over time. As such, these stationarity assumptions can affect the level of insight given by the analysis. Increasingly, the aim of studies is not only to find which regions interact with each other, but how this varies over time, or throughout various experimental situations such as performing different tasks.

[1]For a review on statistical analysis of fMRI data, see Lindquist (2008)

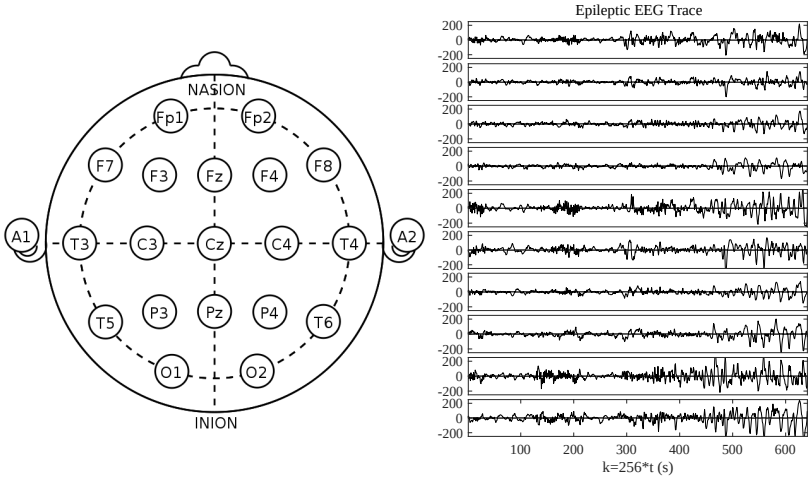


Figure 1.1.2 – Left: EEG electrode placement points for the international 10-20 standard. Right: EEG readings for 10 electrodes in the lead up to an epilepsy seizure, see Chapter 6 for more details.

To relax the assumption that the graph is constant over time requires extending the degrees of freedom within the chosen statistical model. Throughout this thesis, and especially in Chapter 3, several approaches to relaxing stationarity assumptions are discussed. In particular, a new type of dynamic graphical model is proposed which can operate in high-dimensions while also detecting changes (or *changepoints*) across many data-streams. While analysis of fMRI data is not explicitly discussed in this thesis, it is one of the areas where methods developed here may provide great value. Indeed, other researchers are investigating the application of algorithms presented in this thesis to fMRI data, and the similarly motivated work of Monti et al. (2014) and Xu et al. (2013) suggests increasing acceptance of these methodologies in the neuroscience community.

EEG Data

In contrast to fMRI data, EEG sensors provide very high resolution in time at the expense of spatial resolution (see Figure 1.1.2). Recording frequencies of 256Hz are common, which gives rise to an abundance of data, especially when monitoring can take place for prolonged periods of time. For example, epilepsy patients are routinely monitored for days or weeks prior to surgical

operations. In this data-rich environment, we may not expect to find ourselves in a high-dimensional setting. If we again consider the task of estimating a correlation matrix to describe cross-channel variation we can easily operate in the standard-dimensional setting where $T > p^2$. However, if we want to ask more questions of the data and describe not just its cross-channel variance, but also its auto-covariance structure, our models very quickly grow in size. For example, a p-dimensional lag h *vector auto-regressive (VAR)* model will have of order p^2h parameters. If we again consider that these models may have parameters which change over time, then the high-dimensional setting is soon reached.

In Chapter 7, a class of dynamic graphical model is developed that utilises wavelet basis functions. Such models can not only describe cross-covariance, but also how auto-covariance structures change over time. As an example application, it is demonstrated how these models may be used to characterise EEG activity in the lead up to, and throughout an epilepsy seizure. A particular novelty of the method is that it decomposes dependency structure as a network across a set of different length scales. Therefore, not only is the seizure activity clustered according to seizure, but one can readily see which electrodes appear dependent across a network. Potentially, this may give an indication of how epileptic activity spreads across the brain, which could be of particular value in understanding complex epielpsies where seizure activity is not well localised. If shown to be robust over multiple patients, features such as those proposed here may one day help clinicians improve epilepsy diagnosis and treatment.

1.1.2 Statistical Analysis of Network Traffic

Dynamic graphical models provide a valuable tool not just in the scientific domain, but also to help us understand complex systems in general. Large computer networks, such as the internet, are perhaps one of the most complex and data-rich systems available to study. As computing technology has progressed, it is not only possible to transfer more data across networks, but also monitor this activity itself. Again, in such a data-rich domain, one may question the need for high-dimensional statistics. For example, if one considers the popular network monitoring protocol NetFlow, it is not uncommon to collect

hundreds of gigabytes of data. However, when one considers that a large corporate network may have thousands of devices attached, it is again plausible that statistical models will have to operate in high-dimensional settings.

Given society's reliance on computer networks, such systems are increasingly being targeted by sophisticated cyber-criminals and other parties. Rather than cause immediate damage and expose themselves to defenders, attackers are increasingly choosing to infiltrate and remain active within a network for extended periods of time. These *Advanced Persistent Threats (APT)* are hard to detect due to the massive complexity and volume of activity within networks which can mask the subtle movements of an attacker (Friedberg et al. 2015). Traditional *Intrusion Detection Systems (IDS)* operate on a rule-based approach (Patcha et al. 2007), these can react very quickly to detect known threats as long as these correspond to previously modeled and coded patterns. Unfortunately, such hard-coded rules require frequent updating and given their high levels of specificity, such defences are being increasingly bypassed using so-called polymorphic attacks (Fogla et al. 2006)[2]. To counter these rule specification issues, a popular research direction in network anomaly detection is to adopt *machine-learning* based approaches. Generally speaking, these aim to model different classes of network activity, anomalous or normal, based on some algorithm which is trained on real network data. Two main strands of machine-learning methodologies are employed in the literature:

Discriminative methods: Act to classify network activity as normal or abnormal based on an explicit labelling of normality, i.e. one has access to some labelled data where the state of the network is known. Sometimes, this may be extended to consider specific types of anomaly, in a task which is known as anomaly identification (Iglesias et al. 2014).

Generative models: Aim to describe an underlying statistical distribution from which observed data might be generated. Anomalous activity can then defined with respect to the estimated distribution (Patcha et al. 2007). Many correlation based methods, for example *principle component analysis (PCA),* may be seen in this light (Ringberg et al. 2007).

[2]A polymorphic attack is one which is capable of automatically (or easily) adjusting the way it appears to network monitoring systems, i.e. they do not have a fixed signature.

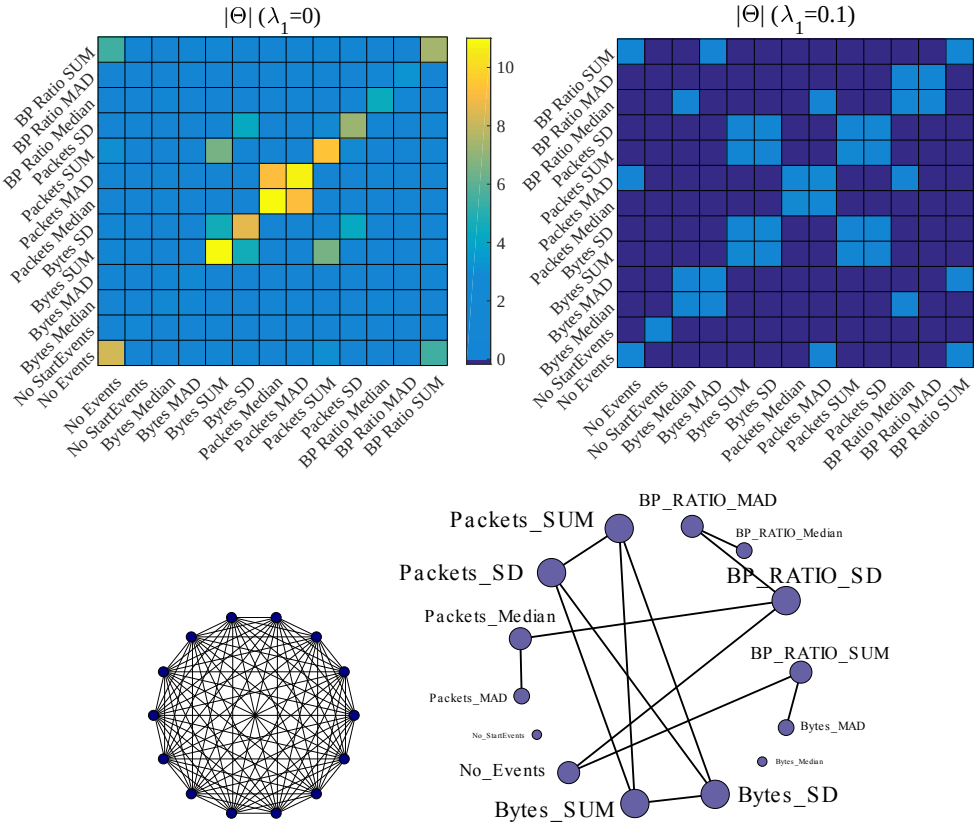


Figure 1.1.3 – Dense (left) and sparse (right) estimates of precision (inverse covariance) matrices obtained from network traffic features. A graphical modelling approach is represented by the sparse model whereby many of the entries in the matrix are zero. The graphical models corresponding to these matrices are given below. The assumption of sparsity clearly provides a more interpretable model and gives increased insight into the data-generation process. The size of the nodes in the estimated graph represent the relative degree (number of edges) associated with each node. A more complete analysis of this data can be found in Section 3.4.2.

To train a discriminative model, we need observations that relate to both network features X, which describe traffic flows, and the network state variable Y, which labels whether traffic is anomalous or not. In many situations, we simply don't know whether the network is in an abnormal state, i.e. we cannot measure Y and do not necessarily know whether the network is under attack or not. In the case where we cannot observe the network state directly, a generative approach can prove useful for defining anomalies. Rather than model the conditional distribution $P(Y|X)$, a generative model aims to describe the *joint*

distribution of the network features $P(X_1, X_2, \ldots, X_p)$. Whether the network is in a normal or anomalous state can be defined relative to the estimated $P(X_1, X_2, \ldots, X_p)$. However, defining and learning a full joint distribution, as opposed to conditional discriminative, or marginal models is hard due to the inherent complexity of such models. At this stage, a graphical modelling approach can add significant value as they possess the flexibility to model many distributions, but are robust due to their parsimonious construction. As demonstrated in Figure 1.1.3, graphical models enable enhanced interpretation of a dataset by highlighting dependencies between variables. In this case, the graphical structure is derived in a static manner from features derived from computer network traffic. When the graphical model is sparse, that is, there are only a few edges selected, the degrees of freedom in the model are reduced as fewer parameters are needed to specify the distribution. In the application to computer network modelling, the added robustness of graphical models may contribute to reducing the false positive rate of detecting anomalies. The application of graphical models to modelling computer network data is examined further in Chapter 3.

1.2 Thesis overview

In the rest of this introductory chapter I will give a brief overview of the thesis structure and notation. There are two principle literature reviews contained within this thesis. The first, "Regularised Learning" (Chapter 2) provides a relatively mathematical introduction to high-dimensional estimation and convex optimisation; the second, *"Locally Stationary Wavelet Processes"* is contained in Chapter 5 and relates to more traditional time-series literature. The remaining chapters contain what should be considered as the main contributions of this work.

It is worth remarking at this point that the traditional boundaries between the domains of *computer science*, *statistics*, and *signal processing* are being eroded. Indeed, the developing field of *machine-learning* may be seen as a result of the inter-disciplinary requirements for modern statistical modelling. The work presented throughout this thesis is very much in-line with this convergence of disciplines. For example, the first half focusses on Gaussian graphical models, which may be traditionally seen from a statistical and

optimisation perspective; the second half is very much born out of statistical signal-processing and the mathematical development of wavelet systems.

A brief summary of chapters is given below:

Chapter 2: Regularised estimation is introduced in the context of M-estimation. The requirement for regularisation is examined in the context of linear regression in high-dimensions. Different ℓ_p regularisation schemes alongside optimisation machinery are introduced. Gaussian graphical models (GGM) are defined in the i.i.d. setting and different model-selection schemes discussed. Finally, a theoretical framework for analysing M-estimators in high-dimensions is discussed.

Chapter 3: Dynamic GGM are introduced, recent literature is discussed, and several ways of relaxing stationarity assumptions are compared. Two novel estimators for dynamic graphical models; the *group-fused graphical lasso (GFGL)*, and *independently fused graphical lasso (IFGL)* are compared. Two algorithms are proposed for estimation of such models. Synthetic experiments are used to highlight the respective properties of the estimators. The chapter concludes by considering the applications of dynamic graphical models in genetics and cyber-security.

Chapter 4: A theoretical analysis of the changepoint error with the GFGL estimator is constructed. In standard dimensional settings (p fixed) asymptotic changepoint consistency is demonstrated. In high-dimensions the GFGL estimator is discussed in the context of the M-estimation framework introduced in Chapter 2.

Chapter 5: Until this point in the thesis, all models assume that data is independently, but not identically drawn. A class of *locally-stationary wavelet (LSW)* and Fourier based models are introduced that enable the modelling of auto-covariance structures. Some statistical properties and previously proposed methods for spectral estimation in these models are discussed.

Chapter 6: The potential high-dimensional nature of spectral estimation motivates regularisation of the empirical spectrum. We discuss and implement fused smoothers for the spectrum of LSW processes. Such fused estimation is extended to 2-D fields, giving rise to applications in image processing.

Chapter 7: This chapter translates ideas from Chapter 3 to the estimation of the multivariate LSW spectrum. Synthetic experiments suggest that regularisation can improve estimation performance and recover graphical spectral structure. An application to modelling EEG data throughout epilepsy seizures demonstrates the benefit of regularisation in terms of separating seizure pathways and enabling enhanced interpretation of EEG data.

Chapter 8: The thesis concludes with a discussion of how these methods can be further extended. In particular, it is discussed how one may extend the M-estimation framework of Chapters 2 and 4 to the locally-stationary wavelet setting.

1.3 Notation

Specific notation, i.e. what individual characters mean, may change throughout the thesis, their meaning should thus be considered in the local context. In general, vectors are denoted in bold font and lower case and matrices are bold and upper case. For example:

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_p)^\top \in \mathbb{R}^p$$

$$\boldsymbol{A} = \begin{pmatrix} A_{1,1} & \cdots & A_{1,q} \\ \vdots & \ddots & \vdots \\ A_{p,1} & \cdots & A_{p,q} \end{pmatrix} \in \mathbb{R}^{p \times q} .$$

Table 1 provides a summary of notation for matrices and vectors. Asymptotic notation is fairly standard. For positive sequences $\{a_n\}$, $\{b_n\}$; $a_n = \mathcal{O}(b_n)$ means there exists a constant c_1 such that $a_n \leq c_1 b_n$. Similarly, $a_n = \Omega(b_n)$ means there is a constant c_2 such that $a_n \geq c_2 b_n$. Analogously, for functions we denote $f(x) = \mathcal{O}(g(x))$ to mean $|f(x)| \leq c_3 |g(x)|$ for constant c_3. Capitilised non-bold letters are used to describe random variables, if the variable is multivariate then this is indicated by an arrow. For example; $\vec{X} := (X_1, X_2, \ldots, X_p)^\top \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes p variables drawn from a multivariate normal distribution.

It is worth noting that some probabilistic statements in this thesis may slightly abuse the notation. For example, I may write $P[\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}, \sigma]$ which

actually means what is the probability of obtaining samples $\boldsymbol{y}$ from some random variable $\vec{Y}$ and it's associated distribution (in this case, this should be parameterised in terms of $\boldsymbol{X}, \boldsymbol{\theta}, \sigma$).

In much of the thesis the notation can become quite complicated simply due to the number of indexes required. For example, we will deal with quantities which may vary as a function of; time, space, scale, or direction, all of which require indexing. Generally, time-indexed quantities will have the notation $\{x^{(t)}\}_{t=1}^{T} = \{x^{(1)}, \ldots, x^{(t)}, \ldots, x^{(T)}\}$. Super-scripted indices are encased in brackets to differentiate them from the exponentiation operators. For example, the notation $(\boldsymbol{X}^{(t)})^{-1}$ refers to the inversion of the matrix $\boldsymbol{X}^{(t)}$ indexed by t.

Table 1 – Notation for properties and operations on matrices $\boldsymbol{A} \in \mathbb{R}^{p \times p}$ and vectors $\boldsymbol{x} \in \mathbb{R}^{p}$.

Notation	Description	Example
$\boldsymbol{A}_{\cdot,i}$	Vector constructed from entries in column i of matrix $\boldsymbol{A}$	
$\boldsymbol{A}_{\setminus ii}$	The matrix $\boldsymbol{A}$ with elements $A_{ii} = 0$	
$\boldsymbol{A}^{\top}$, $\boldsymbol{x}^{\top}$	Transpose of matrix or vector	
$\boldsymbol{A}^{-1}$	Inverse of matrix $\boldsymbol{A}$	
$\boldsymbol{A} \succ 0$	Matrix $\boldsymbol{A}$ is positive definite	$\boldsymbol{x}^{\top}\boldsymbol{A}\boldsymbol{x} > 0$ for all $\boldsymbol{x} \neq \boldsymbol{0}$
$\boldsymbol{A} \succeq 0$	Matrix $\boldsymbol{A}$ is positive semi-definite	$\boldsymbol{x}^{\top}\boldsymbol{A}\boldsymbol{x} \geq 0$ for all $\boldsymbol{x} \neq \boldsymbol{0}$
$\langle \boldsymbol{x}, \boldsymbol{y} \rangle$	Inner product of vectors	$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^{\top}\boldsymbol{y}$
$\langle\!\langle \boldsymbol{A}, \boldsymbol{B} \rangle\!\rangle$	Inner product of matrices	$\langle\!\langle \boldsymbol{A}, \boldsymbol{B} \rangle\!\rangle = \mathrm{tr}(\boldsymbol{A}^{\top}\boldsymbol{B})$
$\|\boldsymbol{x}\|_p$	p-norm for vector $\boldsymbol{x}$	$\|\boldsymbol{x}\|_p = (\sum_{i=1}^{p} \|x\|_i^p)^{1/p}$
$\|\boldsymbol{x}\|_0$	ℓ_0 pseudo-norm (number of non-zero elements in $\boldsymbol{x}$)	$\|\boldsymbol{x}\|_0 = \|\{x_i \neq 0 \, , i = 1, \ldots, p\}\|$
$\|\boldsymbol{x}\|_\infty$	Largest value in $\boldsymbol{x}$	$\max_{1 \leq i} \|x_i\|$
$\|\boldsymbol{A}\|_F$	Frobenius norm of matrix	$\sum_{i,j=1}^{p} \|A_{i,j}\|^2$
$\mathrm{tr}(\boldsymbol{A})$	Trace of $\boldsymbol{A}$	$\mathrm{tr}(\boldsymbol{A}) = \sum_{i=1}^{p} A_{i,i}$
$\|\|\|\boldsymbol{A}\|\|\|_2 \equiv \|\boldsymbol{A}\|_2$	Induced ℓ_2, ℓ_2 norm, or spectral norm of matrix	$\max_{\|\boldsymbol{x}\|_2 = 1} \|\boldsymbol{A}\boldsymbol{x}\|_2$
$\|\boldsymbol{A}\|_\infty$	Largest element in matrix	$\max_{1 \leq i,j \leq p} \|A_{i,j}\|$
$\|\boldsymbol{A}\|_{2,1}$	Group-mixed ℓ_2/ℓ_1 norm (row/column)	$\sum_{j=1}^{p} \|A_{j,\cdot}\|_2$

In Chapters 5,6, 7 it is convenient to use some notation from the signal-processing literature. Primarily, these relate to operations acting on discretely indexed functions $f[t]$, where the square brackets imply that the argument of

the function is integer valued. Notation for some operations on such functions is summarised in Table 2.

Table 2 – Notation for operations on a discrete function f.

Notation	Description	Example
$f[t]$	A function supported on $t \in S \subseteq \mathbb{Z}$	
$f \uparrow l[k]$	Up-sampling of f by l and then taking the kth element	$f \uparrow l[k] = \begin{cases} f[k/l] & k = nl\ n \in \mathbb{Z} \\ 0 & \text{otherwise} \end{cases}$
$f \downarrow l[k]$	Downsampling by l and then taking the kth element	$f \downarrow l[k] := f[lk]$
$(f * g)[t]$	Convolution of signal f with filter g	$(f * g)[t] := \sum_{k=-\infty}^{\infty} f[k]g[t-k]$

Chapter 2

Regularised Estimation

> ... for theories (of equal scope) rendering equally probable our observational data (which, for brevity I shall call equally good at "predicting"), fitting equally well with background knowledge, the simplest is most probably true – Swinburne 1997

From a statistical estimation viewpoint, the significance of a model component or parameter can be viewed in terms of a model selection problem. One may construct a loss function which tells us how well the model fits some data, a lower value of this function implies the model more adequately describes the data. Formally, let us construct this function as $L(M, \boldsymbol{\theta}, \boldsymbol{X})$, where $M \in \mathcal{M}$ indexes a model with parameters $\boldsymbol{\theta} \in \mathcal{P}(M)$, and the matrix $\boldsymbol{X}$ relates to some observed data. Additionally, to account for differences in perceived model complexity, one should penalise this with a function $R(M, \boldsymbol{\theta})$. A more *complex* model should have a larger value of $R(\cdot)$. An optimal identification of model and parameters may then be found through balancing the two terms, such that

$$(\hat{M}, \hat{\boldsymbol{\theta}}) = \underset{M \in \mathcal{M}, \boldsymbol{\theta} \in \mathcal{P}(M)}{\arg\min} \left[L(M, \boldsymbol{\theta}, \boldsymbol{X}) + R(M, \boldsymbol{\theta}) \right] . \tag{2.0.1}$$

In statistics such a formulation is referred to as an M-estimator; however, such frameworks are popular across all walks of science (Boyd and Vandenberghe 2004). For example; *maximum-likelihood (ML)*, *least-squares (LS)* , robust *Huber loss*, and *penalised ML* estimators can all be discussed in this context. The principle idea is to suggest a mathematical, and therefore objectively

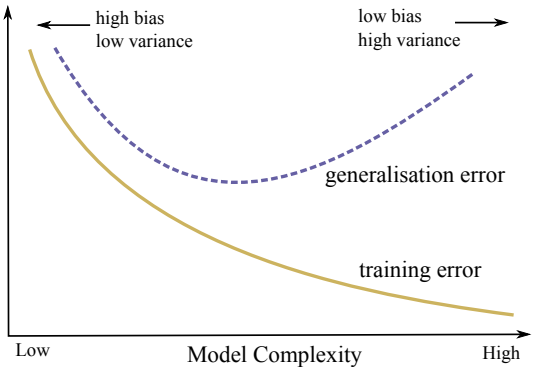


Figure 2.0.1 – Generalisation vs training error. Increasing model complexity may reduce training error, but perform poorly on out-of sample data.

communicable statement to the effect of *Occam's Razor*; given similar model-fit, one should prefer the simpler model (Swinburne 1997).

Figure 2.0.1 provides a graphical motivation for the Occam's Razor in the context of model estimation. If we plot the training and test error of a given set of models ($\mathcal{M}$) with respect to their complexity, we can draw curves corresponding to those in the figure. The key point to take away is that good performance on training data does not guarantee good performance on the *out-of-sample* test data. Occam's Razor and Eq. 2.0.1, therefore suggest we attempt to choose the model with the lowest generalisation error.

Depending on the specification of the functions $L(\cdot)$ and $R(\cdot)$ and associated model/parameter spaces, the problem in (2.0.1) can be either very easy or difficult to solve. Throughout this chapter the motivation for framing statistical estimation problems as *convex*[1] M-estimators is developed. In the next section, we start this discussion in the context of the canonical linear regression model. Specifically, we focus on the high-dimensional setting where the number of covariates is larger than the number of data-points and traditional estimation methods may fail to identify a model. We discuss why this happens, and how regularisation and formulation of M-estimators can enable model estimation, even in the relatively extreme case of high-dimensionality. After this,

[1]Special attention is given to M-estimators (2.0.1) where the constituent functions are convex as they result in relatively easy optimisation problems. Some properties of convex functions are given in the Appendix A.1.

a set of optimisation tools are introduced to help us solve practically solve M-estimation problems; this is followed by an introduction to graphical models. In the final section, a framework for theoretically analysing M-estimators is introduced alongside several key results from the literature.

2.1 Linear Regression

Linear regression is one of the most popular and simple statistical models in use today. Focussing on the predictive task, the model attempts to predict the value of an outcome $Y \in \mathcal{Y} \subseteq \mathbb{R}$, conditional on a set of input variables $\vec{X} \equiv (X_1, \ldots, X_p) \in \mathcal{X} \subseteq \mathbb{R}^p$. While most of this thesis is not directly concerned with the task of prediction, but rather descriptive modelling, it is still very important to understand the linear predictive model. In addition to providing a simple introduction to regularised M-estimation, we will later see how the linear model can be adapted for use in dependency and changepoint analysis.

In a statistical sense, the end goal of linear regression is to obtain the *posterior predictive* $P[Y = y_{\text{test}} | \vec{X} = \boldsymbol{x}_{\text{test}}, \hat{\boldsymbol{\theta}}]$, where $\hat{\boldsymbol{\theta}} \in \mathbb{R}^p$ are a set of model parameters estimated on some pre-observed training data. As the name suggests, linear regression assumes the function mapping the feature space to the labels $f(\boldsymbol{\theta}) : \mathcal{X} \mapsto \mathcal{Y}$ is linear in nature.

***Definition* 2.1.** *Linear Regression Model*

The linear regression model has two principle constructions. These are given as the fixed design *model where the covariates are* not *random variables*

$$Y^{(i)} = \theta_0 + \sum_{j=1}^{p} \theta_j x_j^{(i)} + \epsilon^{(i)} \quad \text{for } i = 1, \ldots, N \,, \tag{2.1.1}$$

and the random-design *model*

$$Y^{(i)} = \theta_0 + \sum_{j=1}^{p} \theta_j X_j^{(i)} + \epsilon^{(i)} \quad \text{for } i = 1, \ldots, N \,, \tag{2.1.2}$$

where $\epsilon^{(i)}$ is a zero-mean noise process (random variable). Additionally, it is usually assumed that the noise process is sampled independent, and is identically distributed (i.i.d).

Typically, one may assume that stochasticity is provided through a Gaussian random variable such that $\epsilon^{(i)} \overset{\text{i.i.d}}{\sim} \mathcal{N}(0, \sigma^2)$. For simplicity, let us consider the fixed design case where the intercept θ_0 is zero, and the covariates are centered and measured on the same scale[2]. For observational pairs $(y^{(i)}, \boldsymbol{x}^{(i)})$, the linear model (2.1.2) can be written in matrix-vector notation as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{\epsilon} \ ,$$

where $\boldsymbol{y} = (y^{(1)}, \dots, y^{(N)})^\top \in \mathbb{R}^N$ is the *response vector*, and $\boldsymbol{X} = (\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)})^\top \in \mathbb{R}^{N\times p}$ consists of measured covariates and is known as the *design matrix*. Traditionally, one may find estimates for the regression parameters, through either *least squares (LS)*; $\hat{\boldsymbol{\theta}}_{\text{LS}} := \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^N \|y^{(i)} - X_{i,\cdot}\boldsymbol{\theta}\|_2^2$, or if we associate a parameterised model to the errors, via *maximum likelihood estimation (MLE)*; $\hat{\boldsymbol{\theta}}_{\text{MLE}} := \arg\max_{\boldsymbol{\theta}} P[\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}]$.

Assuming that we have associated a Gaussian distribution function to our model, the likelihood $P(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})$ is given by

$$P[\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{\Sigma}] = (2\pi)^{-T/2}\big(\det(\boldsymbol{\Sigma})\big)^{-1/2}\exp\big\{-(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})/2\big\} \ .$$

Further, if one assumes errors are i.i.d we can set $\boldsymbol{\Sigma} = \sigma^2\boldsymbol{I}$ and the above simplifies to

$$P[\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}, \sigma] = (2\pi\sigma^2)^{-T/2}\exp[-(2\sigma^2)^{-1}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2] \ .$$

As the name suggests, for the MLE estimator one is required to maximise the above function with respect to $\boldsymbol{\theta}$. For example, say we wish to train the model to a set of observations $(\boldsymbol{y}, \boldsymbol{X}) \equiv (\boldsymbol{y}_{\text{train}}, \boldsymbol{X}_{\text{train}})$, then one can simply maximise the likelihood, or equivalently the log-likelihood

$$\{\hat{\boldsymbol{\theta}}_{\text{MLE}}, \hat{\sigma}^2\} := \arg\max_{\boldsymbol{\theta},\sigma^2}\big[\log P[\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}, \sigma^2]\big] = \arg\max_{\boldsymbol{\theta},\sigma^2}\left[-\frac{N}{2}\ln(\sigma^2) - \frac{1}{2\sigma^2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2\right] \ .$$

Differentiating the log-likelihood and equating to zero leads to the estimators

$$\hat{\boldsymbol{\theta}}_{\text{MLE}} = (\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\boldsymbol{y} \ ,$$

[2]This can generally be achieved through *z-scoring* the data such that $\bar{y} = N^{-1}\sum_{i=1}^N y^{(i)} = 0$ and $\hat{\sigma}_j^2 = N^{-1}\sum_{i=1}^N (x_j^{(i)} - \bar{x}_j) = 1$ for all $j = 1, \dots, p$.

$$\hat{\boldsymbol{\sigma}}^2_{\text{MLE}} = \frac{1}{N}(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\theta}}_{\text{MLE}})^\top(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\theta}}_{\text{MLE}}) \ .$$

There are several remarks worth making about the above result:

- The regression parameter estimates obtained through MLE are identical to those obtained via *Least Squares (LS)*, ie $\hat{\boldsymbol{\theta}}_{\text{LS}} := \arg\min_{\boldsymbol{\theta}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2 = \hat{\boldsymbol{\theta}}_{\text{MLE}}$.
- The MLE estimator for the variance differs from the unbiased estimator. Through Cochran's theorem we have; $N\hat{\sigma}^2_{\text{MLE}}/\sigma^2 \sim \mathcal{X}^2_{N-1} \implies \mathbb{E}[\hat{\sigma}^2_{\text{MLE}}] = \sigma^2(N-1)/N$.
- If $(\boldsymbol{X}^\top\boldsymbol{X})$ is singular and cannot be inverted then the MLE problem as above is an ill-posed problem. Additionally, if $(\boldsymbol{X}^\top\boldsymbol{X})$ is nearly singular, ie $\det(\boldsymbol{X}^\top\boldsymbol{X}) \approx 0$, then the estimates for $\hat{\boldsymbol{\theta}}_{\text{MLE}}$ will be very unstable.

Before proceeding, one may note that the matrix $\boldsymbol{X}^\top\boldsymbol{X}$ is related (in the random design case) to the empirical covariance estimator for the Gaussian distribution. If the covariates were drawn from a centered multivariate Gaussian distribution such that $\vec{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$, then the MLE estimator for the covariance is given as $\hat{\boldsymbol{S}} = N^{-1}\boldsymbol{X}^\top\boldsymbol{X}$. The covariance matrix $\hat{\boldsymbol{S}} \propto \boldsymbol{X}^\top\boldsymbol{X}$ is singular and thus non-invertible when there exists a linear inter-dependency between covariates. Specifically, if there is a linear dependency between columns, i.e. $\text{rank}(\boldsymbol{X}^\top\boldsymbol{X}) < p$, then the covariance matrix will be singular and a unique estimate $\hat{\boldsymbol{\theta}}_{\text{MLE}}$ cannot be obtained. With regards to the dimensions p, N we note; $\text{rank}(\boldsymbol{X} \in \mathbb{R}^{p\times q}) \leq \min(p, q)$, and thus $\text{rank}(\hat{\boldsymbol{S}}) \leq \min(p, N)$[3]. As such, in the high-dimensional setting where $p > N$, the covariance matrix becomes singular and the LS and MLE estimates are ill-defined.

Figure 2.1.1 provides a graphical illustration of how linear dependency between covariates can result in an inability to estimate parameters. While in the $N < p$ setting such singular behaviour is guaranteed, even in settings where $N > p$, the covariance can almost be singular, i.e. when the angle between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is very small.

It is well known that one can stabilise estimators in the high-dimensional setting by utilising prior knowledge relating to regression parameters. Two traditional approaches for this are discussed in the following sections. The first known as *ridge-regression* aims to shrink the estimates for our parameters

[3]Recall that $\text{rank}(\boldsymbol{AB}) \leq \min(\text{rank}(\boldsymbol{A}), \text{rank}(\boldsymbol{B}))$

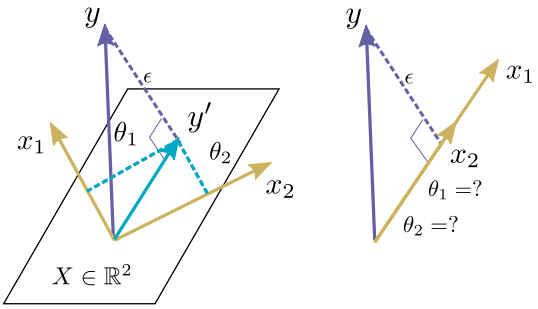


Figure 2.1.1 – Linear dependency between covariates provides a problem for estimating the two regression parameters θ_1, θ_2 as the plane defined by $\boldsymbol{x}_1, \boldsymbol{x}_2$ is no-longer defined.

towards zero. Alternatively, one may perform *subset selection* and regress onto a small subset of parameters. Interestingly, both of these approaches may be considered in the context of M-estimators, and understood in terms of placing priors on the model parameterisation.

2.1.1 Ridge Regression

One of the simplest priors we can adopt is to assume the parameters are drawn from a Gaussian distribution. For example, one may consider a zero-mean prior with identical variance σ_0^2 along each of the p parameters such that $\vec{\theta} \sim \mathcal{N}(\mathbf{0}, \sigma_0^2 \boldsymbol{I})$. If, for simplicity we assume a fixed design matrix, then the posterior over parameters is now given as

$$
\begin{aligned}
P[\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}] &= \frac{P[\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}, \sigma^2]}{\int P[\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}, \sigma^2] \mathrm{d}\boldsymbol{\theta}} P[\boldsymbol{\theta}] \,, \\
&\propto \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{N} (y^{(i)} - (\boldsymbol{x}^{(i)})^\top \boldsymbol{\theta} \right)^2 \exp\left(-\frac{1}{2\sigma_0^2} \sum_{j=1}^{P} \theta_j^2 \right) \,. \qquad (2.1.3)
\end{aligned}
$$

Taking the log-posterior; $\log P[\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}]$, we arrive at the familiar objective of ridge-regression (Hoerl et al. 1970)

$$
\mathcal{L}_{\text{Ridge}}(\boldsymbol{\theta}, \sigma^2/\sigma_0^2) := -\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2 - \frac{\sigma^2}{\sigma_0^2} \|\boldsymbol{\theta}\|_2^2 \,. \qquad (2.1.4)
$$

Maximising the log-posterior[4], or equivalently minimising the negative log-posterior, allows us to gain a unique estimate for the parameter

$$\hat{\boldsymbol{\theta}}_{\text{Ridge}} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \left[-\mathcal{L}_{\text{Ridge}}(\boldsymbol{\theta}, \lambda)\right] \ ,$$

where the quantity $\lambda = (\sigma/\sigma_0)^2$ is known as the *regularisation* parameter. Written in this form, the ridge-regression estimator can be directly related to the M-estimator framework of Eq. 2.0.1; consider the loss function $L(\boldsymbol{\theta}) \equiv \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2$, and complexity penalty $R(\boldsymbol{\theta}) \equiv \lambda\|\boldsymbol{\theta}\|_2^2$. Furthermore, since the ridge regression problem is convex it has a global mimima located at $\hat{\boldsymbol{\theta}}_{\text{Ridge}}$. Due to this convexity and smoothness, an explicit solution for the posterior mode is easily found by equating the partial derivatives of $\mathcal{L}_{\text{Ridge}}(\boldsymbol{\theta})$ to zero. The resulting estimate is given as

$$\hat{\boldsymbol{\theta}}_{\text{Ridge}} = (\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^\top\boldsymbol{y} \ .$$

Comparing this with the vanilla MLE solution we observe that the use of a Gaussian prior has added a term $\lambda\boldsymbol{I} \in \mathbb{R}^{p\times p}$ to the diagonal of the covariance matrix, hence the name ridge regression. Given this additional diagonal term, the solutions are stabilised even in the case where $N \ll p$. Unfortunately, this stability comes at the expense of adding a bias into our estimators. Nethertheless, by adjusting the regularisation strength λ we now have the ability to move along the bias/variance trade-off curve (Figure 2.0.1). One can interpret the use of a prior as adding extra *assumed* data to an estimation problem. In this case, adopting the Gaussian prior and likelihood with uniform variance means we are not letting this artificial data lie in any preferred direction. The bias imposed by ridge regression therefore does not preferentially select any parameters, but rather shrinks all the parameters together.

A further interesting way to interpret this prior, is to consider how it restricts estimates to what are known as feasible sets. Specifically, if one considers the regulariser $R_{\text{Ridge}}(\boldsymbol{\theta}, \lambda) := \lambda\|\boldsymbol{\theta}\|_2^2$ in conjunction with a certain loss function, then only solutions within a certain sub-space are allowed. We can see this more clearly by re-writing (2.1.4) as a constrained optimisation problem

[4]Similarly, one may consider the *maximum a-posteriori (MAP)* estimate. Maximising the log posterior is equivalent due to the monotonic nature of the log function.

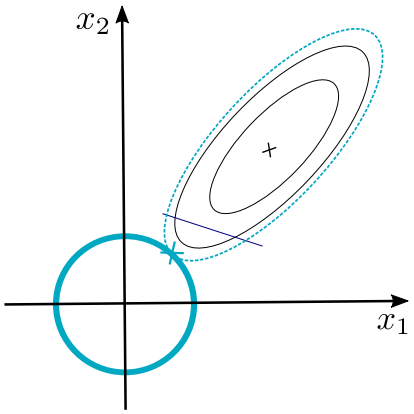


Figure 2.1.2 – Diagram of how the ridge regression estimator can be understood in terms of an explicitly constrained optimisation problem. Only when the contours of the least squares solution intersect the ℓ_2 ball are solutions feasible.

$$\begin{aligned} \hat{\boldsymbol{\theta}}_{\text{Ridge}} &= \arg\min_{\boldsymbol{\theta}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{\theta}\|_2^2 \leq l \ . \end{aligned} \tag{2.1.5}$$

For a given threshold l, such a formulation is equivalent to the ridge regression in (2.1.4). The appropriate value for l is defined by the regularisation parameter λ in conjunction with the size of the model fit term $\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2$. Figure 2.1.2 presents a graphical representation of both this projection, and other constraints considered in the sequel.

The idea of explicitly constraining estimates via such a norm ball, i.e. such that $\|\boldsymbol{\theta}\|_2^2 \leq l$ is extremely popular, not just in statistics, but also for general ill-posed inverse problems (Gockenbach 2016). In the next section, projections for different constraint sets that correspond to different prior assumptions are considered and result in new classes of M-estimators.

2.1.2 Subset Selection

A pervasive idea in high-dimensional statistics is that data may lie in some embedded low-dimensional structure or subspace (Meinshausen 2008; Negahban et al. 2012). Indeed, as figure 2.1.1 suggests, when the covariance matrix is singular it might be beneficial to reduce the dimensionality of the model.

For example, in the diagram, rather than regressing onto a plane, we could regress onto a line and choose a subset of features to utilise within the model.

A traditional approach to the subset selection problem is to start with no regression parameters (we assume they are inactive and equal to zero) and gradually add them as required. This method known as *forward selection* starts with an empty set $\mathcal{A} = \{\}$ and then iteratively activates parameters that appear to improve the model. However, we need a way to know when to stop adding parameters, "How far down the model complexity curve in Figure 2.0.1 do we allow our model to go?"

Similarly to the M-estimation idea, a popular way to achieve this is by penalising model fit with a complexity penalty. In particular, some popular penalties are the *Akikake Information Criteria* (Akaike 1973)

$$\text{AIC(k)} := 2k - 2L \ ,$$

or the Bayesian Information Criteria (Schwarz 1978)

$$\text{BIC}(k, N) := k(\ln(N) - \ln(2\pi)) - 2L \ ,$$

where $L \equiv -\log(\mathcal{L})$ is the negative log-likelihood, and k is the number of free parameters required to be estimated, in the forward selection case $k = |\mathcal{A}|$.

***Remark* 2.1.** *Complexity penalties in high-dimension*

Immediately, one notes that the AIC penalty is not dependent on N as it is derived in the asymptotic setting where $N \to \infty$. It would immediately seem somewhat inappropriate for use in the high-dimensional setting where we often have relatively small N. On the other hand, whilst BIC does depend on N, this also runs into trouble in the high-dimensional setting. For example, J. Chen et al. (2008) discuss issues of BIC not being able to adapt to high-dimensional model spaces. This is due to an implicit prior $P[M]$ in the formulation of BIC which assigns equal probability for all models M over model space $\mathcal{M}$. For regular problems where the number of parameters $k = p$ is fixed it is well known that BIC is consistent. However, in the high-dimensional setting where the number of covariates included $k = |\mathcal{A}|$ varies between models we see that regular BIC assigns probability according to the model class size. For example, consider the class of models $\mathcal{S}_1$ with $p = 100$, but only $k = |\mathcal{A}| = 1$ covariate, the number of models in this class is 100. Now, if we consider the class of

models with two covariates $\mathcal{S}_2$ we find this has size $100 \times 99/2$. Thus, the traditional BIC penalty when used to compare models with varying number of covariates assigns much greater probabilities to those with larger active sets. To overcome such inconsistencies J. Chen et al. (2008) suggest an extended BIC definition which actively accounts for the number of active covariates.

All of the above information criteria, including the extended BIC rely on us penalising the likelihood by some quantity proportional to the number of active elements. If we consider the problem in the context of linear regression, we would be required to construct an M-estimator of the form

$$\hat{\boldsymbol{\theta}}_{\mathcal{A}} = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2 + \lambda|\mathcal{A}(\boldsymbol{\theta})| \,, \tag{2.1.6}$$

where $|\mathcal{A}(\boldsymbol{\theta})| = k$ is the number of active non-zero components at a given point in the parameter space. In machine learning and engineering, the addition of this counting factor is often referred to as an "ℓ_0 norm", although this is not a norm in the traditional sense[5]. The reason for such abuse of terminology, is that when one considers a *q-norm*, often denoted ℓ_q of the form $\|\boldsymbol{\theta}\|_q := (\sum_i |\theta_i|^q)^{1/q}$. In the limit $q \to 0$ we obtain the ℓ_0 norm $\|\boldsymbol{\theta}\|_0 \equiv |\mathcal{A}(\boldsymbol{\theta})|$.

A particularly appealing property of subset selection, is that of *sparsity*, whereby there exist many zero elements in our estimate, i.e. $\|\hat{\boldsymbol{\theta}}\|_0 \equiv k \ll p$. Unlike the ridge regression setup, where $\|\hat{\boldsymbol{\theta}}_{\text{Ridge}}\|_0 = p$, when performing subset selection we have a bias which directs us to preferentially choose some key components. This appeals to us, both in terms of stabilising our estimator by adding bias, but also allowing some form of model interpretation through selection of coefficients. Given the discussion above, the challenge when using subset selection is not necessarily a statistical one, but a computational one.

In the previous section, where we discussed ridge-regression it was noted that the ℓ_2 norms led to an overall convex M-estimator as both constituent functions were convex (see 2.1.5). Unfortunately, this is not the case when one adopts the ℓ_0 *"norm"* (or any ℓ_q norm for $0 \leq q < 1$) as a penalty.

***Proposition* 2.1.** *Let $\boldsymbol{x} \in \mathbb{R}^p$, the ℓ_q norm $\|\boldsymbol{x}\|_q$ for $0 \leq q < 1$ is not convex.*

[5]For example, $\|c\boldsymbol{x}\|_0 = \|\boldsymbol{x}\|_0$ for all $c \neq 0$ and thus fails to satisfy the absolute homogeneity requirement of norms.

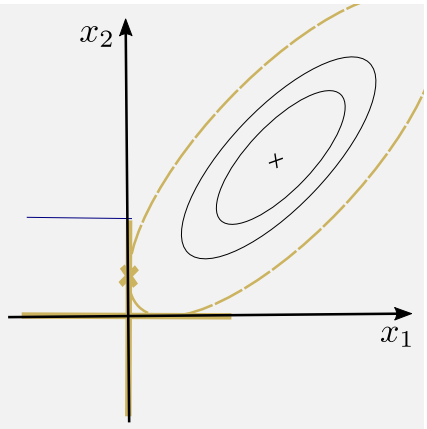


Figure 2.1.3 – As in the case of ridge-regression, the exact sub-set selection operator can be understood as an explicitly constrained estimate. In this case the ℓ_0 "norm" is confined to the axis, such that the estimator must select one of the directions, and is hence sparse. For any LS estimate in the grey region, the resultant regularised estimator will be sparse.

Proof. It is sufficient to demonstrate that the epigraph of $\|\boldsymbol{x}\|_q$ is not a convex set. See Appendix A.1.

Again, as in the ridge-regression setting, we may consider an explicitly constrained equivalent to (2.1.6) as graphically represented in Figure 2.1.3. Due to the addition of a non-convex function, the overall M-estimator (2.1.6) is also non-convex and it is therefore not guaranteed to have a single minima[6]. As a result, we are required to perform exhaustive search over the model space. The naive combinatorial cost for checking subsets scales as order $\mathcal{O}(2^p)$. Hence, while routines such as forward selection may provide a method to perform this search, they are only feasible for very small problems of size $p \approx 10 \to 40$.

In summary, the subset selection methods inspired by BIC can allow for a sparse selection of parameters; however, this comes at a high computational cost due to inherent non-convexity. In the next section, a third penalty function is introduced, which acts as a middle ground between the ridge regulariser and sub-set selection methods. Crucially, this new penalty enables a convex, and thus efficient search, whilst also possessing some parameter selection capabilities.

[6]Note: this can be seen graphically in Figure 2.1.3 where it is not possible to draw a straight line (or linearly move) from the ℓ_0 level set on one axis to another.

2.1.3 Least Absolute Shrinkage and Selection Operator

In this section, possibly one of the most significant methodological advances in modern statistics is introduced. Motivated by the desire to maintain convexity, while keeping the sparsity properties of an estimator, R. Tibshirani (1996) proposed the *least absolute shrinkage and selection operator*, or *lasso* estimator[7]. In practice, the lasso is simply another complexity penalty that can be used in conjunction with the linear-regression least-squares estimator. Written in the unconstrained form, the lasso is defined according to

$$\hat{\boldsymbol{\theta}}_{\text{Lasso}} := \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1 \,, \tag{2.1.7}$$

where $\|\boldsymbol{\theta}\|_1 := \sum_{i=1}^p |\theta_i|$ is known as the ℓ_1 norm. Crucially, unlike ridge regression, the lasso is capable of selecting parameters; while, unlike exact subset selection, it forms a convex problem. The lasso therefore takes a special middle ground in the regularisation hierarchy; the ℓ_1 norm is the limiting case for which an ℓ_p norm may be convex (c.f. Prop. 2.1).

Before further discussing the selection properties of the ℓ_1 regulariser, it is worth taking some time to consider how the lasso and ℓ_1 regularisation relate, not just to other ideas in this thesis, but also the wider literature. First of all, given the interesting convex and selection properties of the lasso, there are many many extensions to this method. In line with the M-estimation paradigm, these can generally be seen as incorporating different prior knowledge into the estimation of parameters. For example; the *group lasso* of M. Yuan et al. (2006), the *fused lasso* (R. Tibshirani et al. 2005), the *elastic net*[8] (Zou and Hastie 2005), and *generalised lasso*[9] (R.J. Tibshirani and J. Taylor 2011), all enable a practitioner to incorporate prior knowledge into an estimate. Furthermore, the development of the lasso, and more general shrinkage estimators is closely tied with work in signal processing where the problem (2.1.7) is often referred to as *basis pursuit denoising problem (BPDN)* (Candes et al. 2005)[10].

[7]The original lasso paper by Tibshirani had (according to Google) approximately 9500 citations in 2014, today, it has over 20,000.
[8]A linear combination of ℓ_1 and ℓ_2 penalties.
[9]The penalty is constructed of a linearly transformed parameter, i.e. $\|\boldsymbol{T}\boldsymbol{\theta}\|_1$.
[10]One can interpret the lasso problem as attempting to find a sparse (in the ℓ_1 sense) set of basis vectors in which to approximate $\boldsymbol{y}$.

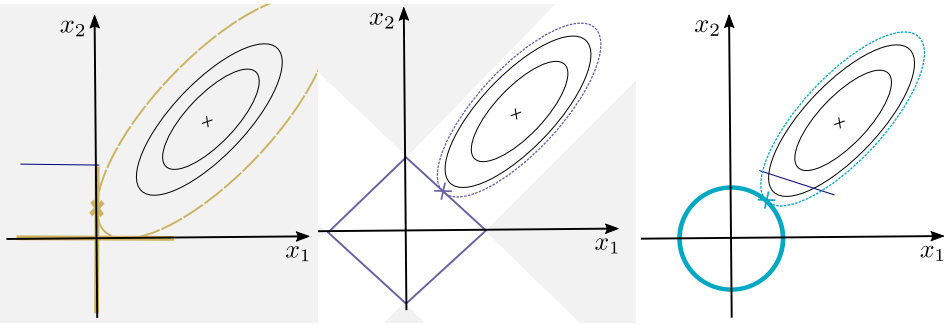


Figure 2.1.4 – Comparison of p-norm constraints, for $p = 0, 1, 2$ corresponding to exact, lasso and ridge regression respectively. Shaded areas represent areas in which if the MLE/LS estimate falls then only one of the two parameters (x_1, x_2) will be selected. The ℓ_2 norm does not select an explicit sparse subspace. In the ℓ_0 case we assume that the constraint level is set to $\|\boldsymbol{x}\|_0 \leq 1$.

In Chapter 5, these relationships are explored in more detail, in the context of wavelet denoising (D. L. Donoho et al. 1995; D. Donoho 1995; D. Donoho et al. 1994) and non-parametric smoothing (R.J. Tibshirani 2014). As hinted at earlier in the introduction, one of the main aims of this chapter is to motivate the formulation of convex M-estimators. Additionally, we note that the property of sparsity is desirable from an interpretability point of view. The lasso estimator provides the canonical M-estimator which possesses both convexity and sparsity properties. More importantly for this thesis, the lasso also has a counterpart which can be used for dependency analysis, known as the *graphical lasso (Banerjee et al. 2008; Friedman, Hastie, and R. Tibshirani 2008)*. This is further discussed in Section 2.3.5; however, has direct parallels with the ideas introduced here.

Let us now further consider the properties of the lasso estimator. As with the ridge-regression, the problem (2.1.7) can be written in an explicitly constrained form:

$$\begin{aligned} \hat{\boldsymbol{\theta}}_{\text{Lasso}} : &= \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{\theta}\|_1 \leq l \,. \end{aligned} \tag{2.1.8}$$

Again, as in (2.1.5) given $(\boldsymbol{y}, \boldsymbol{X})$ there is a one-one mapping between the threshold l and regularisation parameter λ. It seems then, that the lasso can

also be interpreted as constraining an estimator through projection onto a norm ball. In the case of the lasso we have the constraint that the optimal point $\hat{\boldsymbol{\theta}}^*$, should lie in a feasible *sub-level* set

$$\mathcal{C}_{\ell_1} = \{\boldsymbol{\theta} \in \text{dom}(\|\cdot\|_1) \mid \|\boldsymbol{\theta}\|_1 \leq l\}\,, \tag{2.1.9}$$

where $\hat{\boldsymbol{\theta}}^* \in \mathcal{C} \subseteq \mathbb{R}^p$. At first it is not clear how such a restriction enforces sparsity within the estimates. Again, such a property is perhaps best demonstrated geometrically. Figure 2.1.4 demonstrates the contrasting shapes of the constraint sets provided by the different p-norms considered so far. In both the ℓ_0 and ℓ_1 cases, corresponding to subset and lasso selection respectively, it is quite clear that the norm is not smooth at the boundaries between quadrants. This lack of smoothness mandates that within large regions of the parameter space, estimates will be constrained simply to a point. Since the example in Figure 2.1.4 considers a model with two parameters, one of the parameters will be shrunk exactly to zero, hence we obtain a sparse estimate.

The convexity of both the lasso and ridge regression (ℓ_1, ℓ_2) constraints is evident from Figure 2.1.4; one can easily see how it is possible to linearly move from point to point within the sets. As previously mentioned, in the exact-selection ℓ_0 case (2.1.6), we find the function $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_0$ is not convex; the sub-level set for the ℓ_0 function the set only contains entries at the axis where x_1 or x_2 are zero. However, while the ℓ_1 norm is convex, it is not continuously differentiable and possesses a discontinuity at the origin. In the next sections, an extended calculus which can deal with discontinuous functions is introduced. The enables us to minimise the lasso objective and further assess the behaviour of the estimator.

2.1.4 Optimality Conditions for the Lasso

The lasso allows us to encourage some level of sparsity and selectivity within our model parameterisation whilst maintaining convexity. However, unlike in the ridge-regression or least squares problems, the lasso problem involves a non-smooth $\|\boldsymbol{x}\|_1$ penalty term. There is a clear discontinuity at the origin, as seen in Figure 2.1.5. In order to evaluate the minima of the M-estimators,

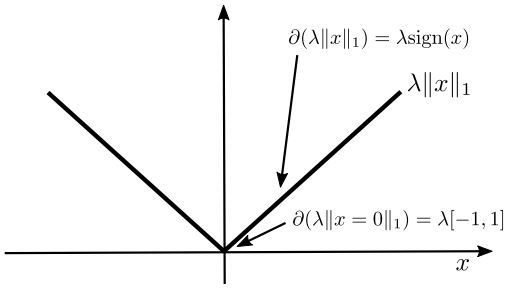


Figure 2.1.5 – The lasso regulariser term $R(\boldsymbol{x}) = \lambda\|\boldsymbol{x}\|_1$. Note, the gradient is classically well defined for all regions except the origin where there is a clear discontinuity.

we introduce a concept known as a subdifferential which provides a value, or a set of values, for the gradient at all points:

***Definition* 2.2.** *Let $f \in \text{conv}(\mathcal{V})$, and $\mathcal{V}^*$ be the dual space of $\mathcal{V}$. The vector $\boldsymbol{\psi} \in \mathcal{V}^*$ is called a* subgradient *of f at $\boldsymbol{x} \in \mathcal{V}$ if*

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \langle \boldsymbol{y} - \boldsymbol{x}, \boldsymbol{\psi} \rangle \quad \text{for all } \boldsymbol{y} \in \mathcal{V} \,. \tag{2.1.10}$$

The subdifferential *is defined as the set of subgradients at $\boldsymbol{x}$. For a convex function this is given as the closed interval $\partial f(\boldsymbol{x}) = [\boldsymbol{a}, \boldsymbol{b}]$ with one-sided limits:*

$$\boldsymbol{a} = \lim_{\boldsymbol{y} \to \boldsymbol{x}^-} \frac{f(\boldsymbol{y}) - f(\boldsymbol{x})}{\boldsymbol{y} - \boldsymbol{x}} \quad , \quad \boldsymbol{b} = \lim_{\boldsymbol{y} \to \boldsymbol{x}^+} \frac{f(\boldsymbol{y}) - f(\boldsymbol{x})}{\boldsymbol{y} - \boldsymbol{x}} \,. \tag{2.1.11}$$

In the above case, if the limits from both sides are equal then the subdifferential contains only a single entry and the function is classically differentiable at that point. Crucially, we now have the ability to deal with discontinuous functions through the notion of the subdifferential set $\partial f(\boldsymbol{x}) \in [\boldsymbol{a}, \boldsymbol{b}]$. For example, the subdifferential of the ℓ_1 norm over $\boldsymbol{x} \in \mathbb{R}^p$ is given as:

$$\partial\|\boldsymbol{x}\|_1 \in \begin{cases} \{\text{sign}(x_i)\} & \text{if } x_i \neq 0 \\ [-1, 1] & \text{if } x_i = 0 \end{cases} \text{, for } i = 1, \ldots, p. \tag{2.1.12}$$

The minima of the convex lasso problem can now be found by considering that the gradient evaluated an optimal point $\boldsymbol{\theta}^*$ must be zero

$$\begin{aligned} \mathbf{0} &\in \partial L(\boldsymbol{\theta}, \lambda)|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} \\ &= \nabla(\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2) + \lambda\partial\|\boldsymbol{x}\|_1 \\ &= -\boldsymbol{X}^\top\boldsymbol{y} + \boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\theta} + \lambda\partial\|\boldsymbol{x}\|_1 \ , \end{aligned} \tag{2.1.13}$$

Rearranging, we arrive at the so-called *Karush-Kuhn-Tucker (KKT)* optimality conditions for the lasso

$$\boldsymbol{X}^\top(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}^*) \in \lambda\partial\|\boldsymbol{x}\|_1 \ . \tag{2.1.14}$$

Note that there is no equality in the above statement, as even though the lasso problem (2.1.7) is convex, it is not always *strictly convex*, this is espcially the case when $p \gg N$. Generally, further analysis of the curvature of the loss function is required in order to show estimation stability. Further discussion of this is provided in Sec. 2.4. A requirement for strict convexity means the standard lasso problem saturates and can only select up to N variables (Bühlmann et al. 2011). If one wishes to include more variables, it is possible to use extensions such as the *elastic-net* (Zou and Hastie 2005) which utilise a combination of ℓ_2 and ℓ_1 penalties.

Due to the importance of the lasso estimator and ℓ_1 regularisation throughout the thesis, it is useful to discuss further properties of the estimators. In the next section, we discuss the application of lasso in the simple orthonormal design case, where closed form solutions are available. In the general case one requires an optimisation scheme which can deal with non-smooth functions. Some machinery for optimising such functions is introduced in Sec. 2.2.1.

2.1.5 Analysis in the Orthonormal Design Case (a relation to thresholding)

An intuitive understanding of the lasso is gained if one considers the orthonormal case where $\boldsymbol{X}^\top\boldsymbol{X} = \boldsymbol{I} \in \mathbb{R}^{p\times p}$. Clearly, in this case $\text{rank}(\boldsymbol{X}^\top\boldsymbol{X}) = p$, and thus we are not restricted by the stability considerations that we faced previously when dealing with design matrices with $p > N$. It is also worth

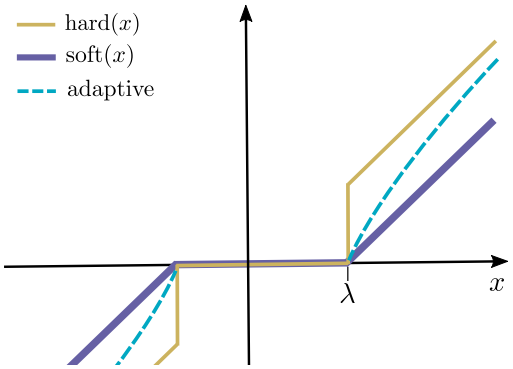


Figure 2.1.6 – Comparison of soft and hard thresholding operators, corresponding respectively to the ℓ_1 and ℓ_0 proximity operators. The dashed line indicates the shrinkage/selection of the adaptive lasso.

remarking that the lasso problem with an orthogonal design is directly related to the problem of denoising via wavelet shrinkage (see Chapters 5,6).

Proceeding to set $\boldsymbol{X}^\top \boldsymbol{X} = \boldsymbol{I}$ in Eq. 2.1.13, gives us the KKT conditions in the orthonormal case:

$$\boldsymbol{\theta} = \boldsymbol{X}^\top \boldsymbol{y} - \lambda \partial \|\boldsymbol{x}\|_1 \ .$$

In the orthogonal situation the least squares solution is given as $\hat{\boldsymbol{\theta}}_{\text{LS}} = \boldsymbol{X}^\top \boldsymbol{y}$. Given that the subdifferential is defined at the individual parameter level, then for $i = 1, \ldots, p$, for the active components (i.e $\theta_i \neq 0$) we now have

$$\theta_i = \hat{\theta}^{(i)}_{\text{LS}} - \lambda \text{sign}(\theta_i) \ .$$

Furthermore; if $\theta_i < 0$, then $\hat{\theta}^{(i)}_{\text{LS}} < -\lambda$; and if $\theta_i > 0$, then $\hat{\theta}^{(i)}_{\text{LS}} > \lambda$. Finally, in the case $|\hat{\theta}^{(i)}_{\text{LS}}| > \lambda$ we have $\text{sign}(\theta_i) = \text{sign}(\hat{\theta}^{(i)}_{\text{LS}})$; we thus obtain

$$\theta_i = \text{sign}(\hat{\theta}^{(i)}_{\text{LS}})(|\hat{\theta}^{(i)}_{\text{LS}}| - \lambda) \ .$$

In the alternate case, where $|\hat{\theta}^{i}_{\text{LS}}| < \lambda$, we have, from the subgradient; $\theta_i = \lambda[-1, 1] - \hat{\theta}^{i}_{\text{LS}} = 0$. Gathering the two cases together we find

$$\hat{\theta}^{(i)}_{\text{Lasso}} = \begin{cases} 0 & \text{if } |\hat{\theta}^{(i)}_{\text{LS}}| < \lambda \\ \hat{\theta}^{(i)}_{\text{LS}} - \lambda \text{sign}(\hat{\theta}^{(i)}_{\text{LS}}) & \text{if } |\hat{\theta}^{(i)}_{\text{LS}}| > \lambda \end{cases} .$$

The above can be re-written as below and is known as the soft-thresholding operator

$$\hat{\boldsymbol{\theta}}_{\text{Lasso}} = \text{soft}(\hat{\boldsymbol{\theta}}_{\text{LS}}; \lambda) := \text{sign}(\hat{\theta}^{(i)}_{\text{LS}})\max(|\hat{\theta}^{(i)}_{\text{LS}}| - \lambda, 0) \ , \tag{2.1.15}$$

for each element $i = 1, \ldots, p$. From the above, the solution of the lasso in the orthonormal case acts as a thresholding operator on the ordinary least squares solution.

Figure 2.1.6 demonstrates the effect of this operator in comparison to that of the hard-thresholding operator which arises in the ℓ_0 penalisation case. The lasso solution not only selects certain parameters when $|\hat{\theta}^{(i)}_{\text{LS}}| < \lambda$, but when parameters are non-zero this adds a shrinkage bias with respect to the LS estimates. The bias associated with shrinkage is often undesirable. For example, Bühlmann et al. (2011) discuss how this may lead the lasso to over-estimate the number of true active covariates.

2.2 Convex Optimisation

Whilst we have discussed solutions of the lasso in the orthonormal setting, methods to solve the general design case have not been discussed. In this section a modified approach to gradient descent is introduced that enables us to deal with non-smooth objectives. Additionally, a method for splitting objective functions up into simpler problems is introduced. For more details and a complete review on convex optimisation, the texts by Boyd, Parikh, et al. (2011) and Nesterov (2007) and Boyd and Vandenberghe (2004) are recommended.

2.2.1 Proximal gradient descent

In conventional (smooth) optimisation problems the canonical approach to obtaining minima is via gradient descent type algorithms. Extending gradient descent methods to cope with non-smooth objectives requires re-thinking how we step through parameter space when faced with discontinuous functions. One such method, known as proximal gradient descent revolves around minimising a surrogate function known as the *Moreau envelope*.

***Definition* 2.3.** *The* Moreau envelope *or* Moreau-Yoshida regularisation $M_{\lambda f}$ *of the function* λf *is defined as:*

$$M_{\lambda f}(\boldsymbol{v}) = \inf_{\boldsymbol{x}} \Big\{ f(\boldsymbol{x}) + \frac{1}{2\lambda} \|\boldsymbol{x} - \boldsymbol{v}\|_2^2 \Big\} \ . \tag{2.2.1}$$

The Moreau envelope can be interpreted as a smoothed form of f that has domain $\mathrm{dom}(M_{\lambda f}) \in \mathbb{R}^p$ and is continuously differentiable even when f is not. Furthermore, the sets of minimisers for f and M_f are the same (see Nesterov (2005)). This property can be useful for generalising gradient descent methods to non-smooth objectives, such as those found in the lasso.

***Proposition* 2.2.** Geometric Moreau

Let f *be convex and closed on the Hilbert space* $\mathcal{V} = \mathcal{H}$*, with dual* $\mathcal{V}^* = \mathcal{H}$*. Then for every* $\boldsymbol{z} \in \mathcal{H}$ *there is a unique decomposition*

$$\boldsymbol{z} = \hat{\boldsymbol{x}} + \boldsymbol{\psi} \quad \text{with } \boldsymbol{\psi} \in \partial f(\hat{\boldsymbol{x}}) \ , \tag{2.2.2}$$

and the unique $\hat{\boldsymbol{x}}$ *in this decomposition can be computed with the* proximal operator*:*

$$\mathrm{prox}_f(\boldsymbol{z}) := \underset{\boldsymbol{x} \in \mathcal{H}}{\arg\min} \Big\{ \frac{1}{2} \|\boldsymbol{x} - \boldsymbol{z}\|_{\mathcal{H}}^2 + f(\boldsymbol{x}) \Big\} \ . \tag{2.2.3}$$

Proof. See Parikh et al. (2013) and Rockafellar (1970).

Essentially, this result provides us with a way to move along our function by stepping in the direction given by the gradient (sub-gradient) $\boldsymbol{\psi}$. The Moreau envelope can now be related to the proximal operator, as the proximal operator returns the minimal value of $M_{\lambda f}$, such that

$$M_f(\boldsymbol{x}) = f(\mathrm{prox}_f(\boldsymbol{x})) + \frac{1}{2} \|\boldsymbol{x} - \mathrm{prox}_f(\boldsymbol{x})\|_2^2 \ .$$

***Proposition* 2.3.** *Let* $\hat{\boldsymbol{z}}$ *be a fixed point, such that;* $\hat{\boldsymbol{z}} = \mathrm{prox}_f(\hat{\boldsymbol{z}})$*. The minimiser of the functional* f *is thus* $\hat{\boldsymbol{z}}$*.*

Proof. This is a consequence of Moreau's theorem 2.2.2, where we have $\hat{\boldsymbol{z}} \in \mathrm{prox}_f(\hat{\boldsymbol{z}}) + \partial f(\hat{\boldsymbol{z}}) \iff \boldsymbol{0} \in \partial f(\hat{\boldsymbol{z}})$.

For a given point $\boldsymbol{x} = \text{prox}_{\lambda f}(\boldsymbol{z})$ and $\lambda > 0$ from 2.2.2 we have $\boldsymbol{z} \in \boldsymbol{x} + \partial\lambda f(\boldsymbol{x}) \iff \boldsymbol{x} \in \boldsymbol{z} - \lambda\partial f(\boldsymbol{x})$. The proximal operator therefore steps us through our parameter space with an implicit subgradient descent step of size λ. Iterating according to

$$\boldsymbol{x}_{k+1} = \text{prox}_{\lambda f}(\boldsymbol{x}_k) \,, \tag{2.2.4}$$

then leads to a convergent sequence until we arrive at a fixed point, at which the function f is minimised.

2.2.2 Alternating Directed Method of Multipliers (ADMM)

If one can calculate the proximity operator for the regulariser term $R(\boldsymbol{\theta})$, then Eq. 2.2.4 allows for a simple and effective gradient descent approach (Beck et al. 2009; Nesterov 2007; Wright et al. 2009). For some estimators such as those considered later in this thesis, a closed form solution for the proximity operators is not so obvious. A popular approach to solve problems with more complex regularisers is to attempt to split up the objective function into separate simpler optimisation problems. The *Alternating Directed Method of Multipliers (ADMM)* approach provides a way to do this splitting and is briefly introduced in this section. Specific versions of this algorithm are discussed in more detail in Chapters 3 and 6.

ADMM constitutes what is known as a dual-ascent algorithm, whereby the minima

$$\min_{\boldsymbol{u}} f(\boldsymbol{u}) \quad \text{subject to} \quad \boldsymbol{A}\boldsymbol{u} = \boldsymbol{b} \,, \tag{2.2.5}$$

can be obtained by maximising a different so-called *dual* function. Consider the Lagrangian function of the original constrained (2.2.5) problem

$$\mathcal{L}(\boldsymbol{u}, \boldsymbol{q}) := f(\boldsymbol{u}) + \langle \boldsymbol{q}, \boldsymbol{A}\boldsymbol{u} - \boldsymbol{b} \rangle \,,$$

where $\boldsymbol{q}$ may be referred to as Lagrange multipliers or dual variables. The *dual function* is then defined as the minimiser of the Lagrangian:

$$g(\boldsymbol{q}) := \inf_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{q}) \,.$$

An optimal value for the primal problem can be recovered from an optimal value for the dual $\boldsymbol{q}^* := \arg\max_{\boldsymbol{q}} g(\boldsymbol{q})$ according to $\boldsymbol{u}^* = \arg\min_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{q}^*)$.

Furthermore, if a property of the problem known as *strong-duality* holds, then the optimal values of the primal and dual problems are the same, i.e. $f(\boldsymbol{u}^*) = \max_{\boldsymbol{q}} g(\boldsymbol{q})$.

Dual ascent methods therefore work by iteratively maximising the dual and finding the mapping to the primal problem. For example, one may iteratively update the primal estimate $\boldsymbol{u}^{(k)}$ according to:

$$\begin{aligned} \boldsymbol{u}^{(k+1)} &:= \arg\min_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{q}^{(k)}) \\ \boldsymbol{q}^{(k+1)} &:= \boldsymbol{u}^{(k)} + \alpha^{(k)}(\boldsymbol{A}\boldsymbol{x}^{(k+1)} - \boldsymbol{b}) \ , \end{aligned} \tag{2.2.6}$$

where $\alpha^{(k)}$ is a step-size parameter. A simple modification to the above (known as the *method of multipliers)* introduces a regularisation term to the Lagrangian and solves the problem

$$\min f(\boldsymbol{u}) + \frac{\rho}{2}\|\boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}\|_2^2 \quad \text{subject to } \boldsymbol{A}\boldsymbol{u} = \boldsymbol{b} \ , \tag{2.2.7}$$

while when completing the dual update one sets $\alpha^{(k)} = \rho$. Clearly, when the constraint is met, the additional function is zero, and thus does not alter the minima. However, the additional term helps add curvature to the Lagrangian $\mathcal{L}_\rho(\boldsymbol{u}, \boldsymbol{q})$ at non-optimal points, and thus allows convergence under more general conditions (on f) than pure dual-ascent Boyd, Parikh, et al. (2011). To see how such methods help us split up M-estimation problems, consider the constrained optimisation problem:

$$\min_{\boldsymbol{u}} \left\{ L(\boldsymbol{u}) + R(\boldsymbol{v}) + \frac{\rho}{2}\|\boldsymbol{A}\boldsymbol{u} + \boldsymbol{B}\boldsymbol{v} - \boldsymbol{c}\|_2^2 \right\} \quad \text{such that } \boldsymbol{A}\boldsymbol{u} + \boldsymbol{B}\boldsymbol{v} = \boldsymbol{c} \ , \tag{2.2.8}$$

where $\boldsymbol{v}$ is known as an *auxiliary* variable. Specifically, if we set $\boldsymbol{A} = \boldsymbol{I}$, $\boldsymbol{B} = -\boldsymbol{I}$ and $\boldsymbol{c} = \boldsymbol{0}$, then the optimisation problem looks very similar to the M-estimator formulation (2.0.1). The augmented Lagrangian for the above is given as

$$\mathcal{L}_\rho(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{q}) := L(\boldsymbol{u}) + R(\boldsymbol{v}) + \frac{\rho}{2}\|\boldsymbol{A}\boldsymbol{u} + \boldsymbol{B}\boldsymbol{v} - \boldsymbol{c}\|_2^2 + \langle \boldsymbol{q}, \boldsymbol{A}\boldsymbol{u} + \boldsymbol{B}\boldsymbol{v} - \boldsymbol{c} \rangle \ ,$$

or equivalently

$$\mathcal{L}_\rho'(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{p}) := L(\boldsymbol{u}) + R(\boldsymbol{v}) + \frac{\rho}{2}\|\boldsymbol{A}\boldsymbol{u} + \boldsymbol{B}\boldsymbol{v} - \boldsymbol{c} + \boldsymbol{p}\|_2^2 \ , \tag{2.2.9}$$

where $\boldsymbol{p} = \rho^{-1}\boldsymbol{q}$ is a *scaled dual* variable. Performing dual ascent on (2.2.8) results in what is known as the *Alternating Directed Method of Multipliers* algorithm

$$\begin{aligned}\boldsymbol{u}^{(k+1)} &:= \arg\min_{\boldsymbol{u}} \mathcal{L}'_{\rho}(\boldsymbol{u}, \boldsymbol{v}^{(k)}, \boldsymbol{p}^{(k)}) \\ \boldsymbol{v}^{(k+1)} &:= \arg\min_{\boldsymbol{v}} \mathcal{L}'_{\rho}(\boldsymbol{u}^{(k+1)}, \boldsymbol{v}, \boldsymbol{p}^{(k)}) \\ \boldsymbol{p}^{(k+1)} &:= \boldsymbol{u}^{(k)} + \boldsymbol{A}\boldsymbol{u}^{(k+1)} + \boldsymbol{B}\boldsymbol{v}^{(k+1)} - \boldsymbol{c}\ .\end{aligned}$$

In the above scheme we note that the updates are separated across $\boldsymbol{u}, \boldsymbol{v}$ which gives rise to the alternating direction name. This can be contrasted with a pure dual-ascent approach which would solve $(\boldsymbol{u}^{(k+1)}, \boldsymbol{v}^{(k+1)}) = \arg\min_{\boldsymbol{u},\boldsymbol{v}} L_{\rho}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{p}^{(k)})$. In the formulation given above the objective is linear separable, i.e. L is only a function of $\boldsymbol{u}$ and R is only a function of $\boldsymbol{v}$. The ADMM algorithm can harness this property of the objective to separate the optimisation problem into two (potentially easier) problems. Furthermore, when considered in the context of $\mathcal{L}'_{\rho}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{p})$, the updates for $\boldsymbol{u}^{(k+1)}$, $\boldsymbol{v}^{(k+1)}$ take the form of proximity operators. For example, with the lasso $L(\boldsymbol{u}) = \propto \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{u}\|_2^2$ and $R(\boldsymbol{v}) \propto \|\boldsymbol{v}\|_1$ results in the updates

$$\begin{aligned}\boldsymbol{u}^{(k+1)} &= \text{prox}_{\rho\ell_2}(\boldsymbol{v}^{(k)} + \boldsymbol{p}^{(k)}) \\ \boldsymbol{v}^{(k+1)} &= \text{prox}_{(\lambda\rho)\ell_1}(\boldsymbol{u}^{(k+1)} + \boldsymbol{p}^{(k)})\ .\end{aligned}$$

Since in many cases the proximity updates have closed form solutions, the iterates of the ADMM routine may be computed with remarkable efficiency. In practice, there are many different variants of ADMM and proximal splitting algorithms (Combettes et al. 2011; Glowinski et al. 1989). For example, in the next chapter (Sec. B.3) an approach to splitting the objective into more than just two blocks proves useful (X. Wang et al. 2015). For proof of convergence for the ADMM iterates, the reader is directed towards Deng et al. (2012) and Lin et al. (2014). Furthermore, given linear separability of updates, one can often trivially distribute such optimisation algorithms (Boyd, Parikh, et al. 2011).

In summary, the ADMM algorithm presented above enable us to use very simple closed form proximity updates to solve a wide range of large scale optimisation problems. In relation to the rest of the thesis, throughout Sections

3.2 and 6.4.4 several variants of ADMM are utilised to break up large, otherwise computationally infeasible optimisation problems. In the next section the discussion switches back to statistical modelling where a link between the lasso and estimation for graphical models is made.

2.3 Graphical Models and Dependency Modelling

As a prelude to dependency modelling, the linear regression model was previously introduced and discussed in the high-dimensional setting. Linear regression forms what is commonly referred to as a discriminative model and aims to describe the conditional distribution $P[Y = y|\vec{X} = \boldsymbol{x}]$ over a target label $y \in \mathcal{Y}$ based on inputs $\boldsymbol{x} \in \mathcal{X}$. In particular, linear regression does not aim to model relationships between the covariates. Alternatively, one can use what are known as generative models to try and represent the joint distribution $P[Y = y, \vec{X} = \boldsymbol{x}]$. Not only can these models be used in a predictive task, i.e. to find the conditional $P[Y|X]$[11], but they can also describe statistical relationships between the covariates.

Unlike in linear regression, where each covariate can have an impact on the distribution for Y, in the generative model one needs to model not only these relations, but also assess the impact of the covariates on each others. As such, generative models generally possess more degrees of freedom when compared to their discriminative counterparts. Due to such model flexibility, in order to stabilise model estimation there is correspondingly greater requirements for prior knowledge. One form of prior knowledge, is that relationships between variables may be described in relation to a network or graph. In this section the concept of a graphical modelling is introduced in the context of both directed and undirected graphical models. The sub-class of relatively simplistic *Gaussian graphical models (GGM)* are then introduced which form the basis for many of the models and estimators in this thesis. Estimation of GGM and particularly the underlying graph structure is then discussed, in the high-dimensional setting clear parallels to the linear regression estimators can be drawn.

[11]One can simply condition on the covariates X by considering $P(Y|X) = P(Y, X)/P(X) = P(Y, X)/\int_{\mathcal{Y}} P(Y, X)dy$.

2.3.1 Directed Graphical Models

Let $\vec{X} \equiv (X_1, \ldots, X_p)^\top \sim \mathcal{D}(\boldsymbol{\theta})$ be set of random variables drawn from some distribution with parametric joint distribution given by $P[\vec{X} = \boldsymbol{x}] \equiv f_{\mathcal{D}}(\boldsymbol{x}, \boldsymbol{\theta})$.[12] Now let $G(V, E)$ define a graph with a set of $V = \{1, \ldots, p\}$ vertices and $E \subset V \times V$ edges. A directed graphical model aims to represent the joint distribution $f()$ as a product of functions defined with respect to conditionally dependent sets of variables. Specifically, if a variable X_i is conditionally dependent on another X_j then the graph $G(V, E)$ will contain a directed edge $(i, j) \in E$. The directed graph decomposes the joint distribution over these edges with functions relating to variables and their parent nodes such that

$$f_{\mathcal{D}}(\boldsymbol{x}, \boldsymbol{\theta}) = \prod_{v \in V} f_v(x_v, x_{\mathrm{Pa}(v)}, \boldsymbol{\theta}_v) \ ,$$

where f_v is a function for each vertex in the graph and $x_{\mathrm{Pa}(v)}$ are the values of nodes belonging to vertices that are parents of v. Figure 2.3.1 provides an example of such a graphical model where the joint distribution can be factorised according to $f_{\mathcal{D}}(x_1, x_2, x_3) = f_1(x_1) f_2(x_2) f_3(x_1, x_2)$.

The canonical example of such a directed graphical model is the so-called *Bayesian network*. In these models the underlying graph G takes the specific form of a *directed acyclic graph (DAG)*, the graph possesses no directed cycles such that there is no way to start at one vertex and follow edges to arrive back at that edge. Such models are particularly useful for representing causal relationships between variables. For example, if X_3 represents whether grass in your garden is wet, X_1, X_2 might be used describe whether it has rained, and whether the sprinkler has been turned on or off. These variables can then be used to predict, through the directed dependencies the state of the grass. For more details on directed graphical models the reader is directed to the excellent reviews of Jordan (2004) and Koller et al. (2007) .

[12] In the rest of this section no distinction is made between the label variable Y and the covariates X. Instead all variables under consideration are contained in the random vector X.

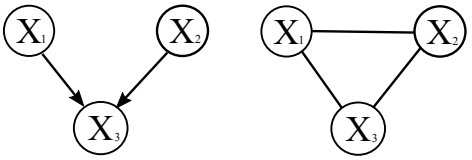


Figure 2.3.1 – Left: Example of a directed graph. Right: Example of an undirected graph.

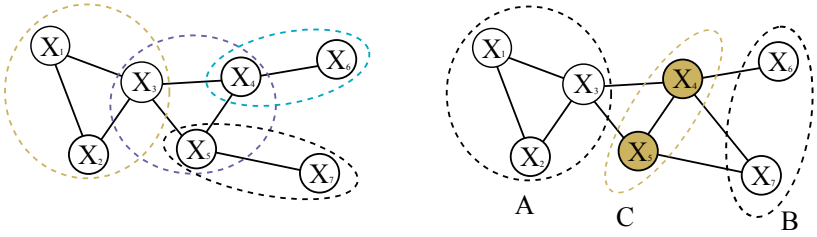


Figure 2.3.2 – Left: Example of a set of maximal cliques in relation to an undirected graph. Right: Example of a set of separating vertices C.

2.3.2 Undirected Graphical Models (UGM)

An alternative to directed graphical models is to look at distributions which factor over undirected graphs. In this case there can only be a maximum of one edge between two vertices. The edge can no longer encode a causal relationships as before, but instead describes a correlatory relationship between variables. Rather than decompose with respect to functions describing the conditional dependency between a variable and its parents, a UGM decomposes with respect to *potential functions* g_c over maximal cliques $c \in \mathcal{C}$ (see Figure 2.3.2)[13], such that

$$f(x_1, \ldots, x_p) = \frac{1}{Z} \prod_{c \in \mathcal{C}} g_c(x_c) \ ,$$

where x_c represents variables included in the clique c and Z is a noramlisation constant. If certain properties of these decompositions hold, then the graphical models are referred to as *Markov networks* or *Markov random fields.*

[13]A Maximal clique is a fully connected sub-graph, such that the inclusion of an additional vertex also requires additional edges.

***Definition* 2.4.** *Markov properties*

The distribution $\mathcal{D}$ satisfies the global Markov property *with respect to the undirected graph G if for any triple disjoint sets $A, B, C \subseteq V$ whereby C separates A, B the independence relation $X_A \perp X_B | X_C$ holds. A weaker condition is that the distribution satisfies the* local Markov property*, a special case of the above where $A = \{i\}$, $B = \{j\}$ and $C = V \backslash \{i, j\}$ for all unconnected vertices $i \neq j \in V$.*

This property allows us to link conditional dependencies between vertices based on separating vertices within the general set $C \subseteq V$. However, this does not guarantee that a conditional dependency $X_A \perp X_B | X_C$ in the distribution $\mathcal{D}$ necessarily results in a node separator in the graph. For this a further requirement is defined below.

***Definition* 2.5.** *Faithfullness*

The probability distribution $\mathcal{D}$ is faithful *to the graph G if for every triple of disjoint sets $A, B, C \subseteq V$,*

$$C \text{ seperates } A \text{ and } B \iff X_A \perp X_B | X_C$$

Faithfulness is a strong condition on a graphical model, such that all conditional dependencies permitted by $\mathcal{D}$ are represented by G. In general this is not true, and whilst the graphical model will permit us to represent some dependencies, the distribution $\mathcal{D}$ may permit independence relations not encoded by the graph. An example to demonstrate the consequences of unfaithfulness is given in the next section. Finally, it is worth noting that some directed graphical models can be related to undirected ones by confounding the effect of parent variables and representing dependencies according to the global Markov property.

***Definition* 2.6.** *Moralised DAG*

A moralised graph G_{M} is constructed by converting all edges in the DAG to be undirected and connecting the parents of any node with an undirected edge. For example X_1 and X_2 in Figure 2.3.1.

***Theorem* 2.1.** *Markov Moral Graph*

If a probability distribution factors with respect to a DAG G_{DAG}, then it obeys the global Markov Property with respect to its moralised undirected graph G_{M}.

2.3.3 Gaussian Graphical Models

Specifically, in this chapter (and throughout this thesis) the set of parametric Gaussian Graphical Models (GGM) is considered whereby the joint distribution follows a multivariate normal distribution, such that[14]

$$(X_1, \ldots, X_p)^\top \sim \mathcal{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{2.3.1}$$

The Gaussian nature of such models restricts them to modelling linear dependencies between continuous valued variables. However, they have several important benefits when it comes to identifying graphical model structure. In particular, with the GGM there is a direct connection between the second-order properties of the variables, i.e. covariances, and the conditional dependency structure (Lauritzen 1996).

To make a statement about the conditional independence properties and relate this to the edge set E, one must look at the partial correlation between variables. The partial covariance is defined as the covariance between two variables conditioned on the rest

$$\text{ParCov}(X_i, X_j, V\backslash\{i,j\}) := \text{Cov}(X_i|X_{V\backslash\{i,j\}},\ X_j|X_{V\backslash\{i,j\}})\ ,$$

where in this case $X_{V\backslash\{i,j\}}$ is the set of all variables excluding the i,jth elements.

A special property of the Gaussian distribution is that the global and local Markov properties are equivalent (see Lauritzen 1996 for proof). This property allows one to show that a pairwise independence $X_i \perp X_j|X_{V\backslash\{i,j\}}$ for an edge (i,j) implies its exclusion from the set E. In the Gaussian case the pairwise partial-covariance is encoded through entries in the precision matrix $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$, whereby

$$\text{ParCov}(X_i, X_j, V\backslash\{i,j\}) = 0 \iff \Theta_{i,j} = 0\ .$$

[14]For simplicity, in this thesis it is assumed the mean parameter is zero, i.e. $\boldsymbol{\mu} = \mathbf{0}$

A key corollary of this result is that if we can estimate accurately from data this precision matrix, and in particular the pattern of zeros, then we may infer GGM structure and highlight some dependencies between variables.

> Essentially, all models are wrong, but some are useful - (Box et al. 1986)

Nowhere is such a quote more valid than when implementing a GGM. Care should especially be taken when interpreting estimated graphical model structure. Recovering the dependency structure of a graphical model is a complicated task even in standard statistical settings, let alone in the high-dimensional settings we will consider in this thesis. As such, one should take the estimated graphical structure with some skepticism and model structure should be interpreted in conjunction with both common sense and scientific theory.

***Remark* 2.2.** *Example: Unfaithful GGM*

An example of an unfaithful GGM can be found in Figure 2.3.1. Specifically, consider the linear regression construction[15]

$$X_1 = \epsilon_1, X_2 = \alpha X_1 + \epsilon_2,\ X_3 = \beta X_1 + \gamma X_2 + \epsilon_3,$$

where $\epsilon_i \overset{iid}{\sim} \mathcal{N}(0,1)$*, and* $\epsilon_3 \perp \{X_2, X_1\}$*,* $\epsilon_2 \perp X_1$*. In the above setup, dependency between variables is described by the coefficients* (α, β, γ)*. If one sets* $\beta + \gamma\alpha = 0$*, then assuming Gaussianity we find* $(X_1, X_2, X_3) \sim \mathcal{N}_3(\mathbf{0}, \mathbf{\Sigma})$ *and* $\text{Cov}(X_1, X_3) = \Sigma_{1,3} = 0 \implies X_1 \perp X_3$*. However, examining the inverse covariance matrix (relating to partial correlation) we may generally find* $\Sigma^{-1}_{1,3} \neq 0 \implies X_1 \not\perp X_3 | X_2$*. There is no undirected graph which permits* $X_1 \perp X_3$ *but also allows* $X_1 \perp X_3 | X_2$ *thus in this case the GGM is said to be unfaithful.*

Such an example serves to remind us that while we can infer some, we may not in infer all of the dependencies from the graph structure. For a GGM to be faithful we need to encode an independence relation (i.e. no edge in the graph) when $\text{ParCov}(X_i, X_j, C) = 0$ *for some subset* $C \subseteq V$ *as opposed to all other variables* $C = V \backslash \{i, j\}$*. This is a stronger condition on the graphical model, and we would expect to obtain fewer edges in a faithful GGM. To this end, the work of Soh et al. (2014) provides an interesting direction, suggesting*

[15] See Bühlmann et al. 2011 p446-449 for more on this and other examples of unfaithful UGM.

it may be possible to test a GGM for faithfulness. In the proceeding sections, and throughout the rest of the thesis the issue of faithfulness is not considered, however it is worth keeping in mind when interpreting extracted graph structures.

***Remark* 2.3.** *Non-Gaussian graphical models*

It is beyond the scope of this chapter to fully review all forms of graphical model construction and estimation. However, it should be noted that there is a rich literature on this subject. In particular, whilst the precision matrix encodes the dependency structure of a GGM, generalisations to other distributions can be made by altering the loss function (as in the neighborhood selection case). Some notable examples include incorporating losses for binary variables via an Ising model (Ravikumar, Wainwright, and J.D. Lafferty 2010), count variables with a multivariate Poisson (E. Yang et al. 2013), or to a non-parametric setting via a non-paranormal (c.f. copula) model (J. Lafferty et al. 2012). Further to the above regularised approaches, one may also consider explicitly constrained approaches such as the CLIME estimator of Cai et al. (2011).

2.3.4 Estimation for GGM

In the previous section we defined GGM in a static (i.i.d) setting where a relationship between the precision matrix of the Gaussian distribution and the graphical edge structure was discussed. In this section we consider how one can practically and robustly estimate the sparsity structure within the precision matrix.

In the GGM case, learning appropriate structure for the graphical model can be linked with the general M-estimation framework of (2.0.1) through a ML or Maximum a-posteriori (MAP) paradigm. Assuming N observations $\boldsymbol{X} \in \mathbb{R}^{p\times N}$ drawn as i.i.d samples, the model fit function $L(\cdot)$ can be related to the likelihood specified by the multivariate Gaussian. Typically, one prefers to work with the log-likelihood, which is given by

$$\frac{1}{N}\log(P[\boldsymbol{X}|\boldsymbol{\Theta}]) = \frac{1}{2}\log\det(\boldsymbol{\Theta}) - \frac{1}{2}\text{trace}(\hat{\boldsymbol{S}}\boldsymbol{\Theta}) - \frac{p}{2}\ln(\pi)\ ,$$

where $\hat{\boldsymbol{S}} = N^{-1}\boldsymbol{X}\boldsymbol{X}^\top$ is often referred to as the empirical covariance matrix. Setting the loss function as $L(\cdot) = -\log\det(\boldsymbol{\Theta}) + \text{trace}(\hat{\boldsymbol{S}}\boldsymbol{\Theta})$ gives (in the setting where $N > p$) a well-behaved smooth, convex function describing how

well the distribution parameterised by $\boldsymbol{\Sigma}$ describes the data $\boldsymbol{X}$. If one considers Eq. 2.0.1 with the function $R(\cdot) = 0$, i.e. no complexity penalty, then the resultant problem gives a ML estimate for the precision matrix

$$\hat{\boldsymbol{\Theta}}_{\text{ML}} := \arg\min_{\boldsymbol{\Theta} \succeq 0} \left[-\log\det(\boldsymbol{\Theta}) + \text{trace}(\hat{\boldsymbol{S}}\boldsymbol{\Theta}) \right] . \tag{2.3.2}$$

We note some properties of this estimate:

- In general the estimator will be dense (not many zeros) and therefore the inferred GGM will be close to being complete.
- The estimator exhibits large variance when $N \approx p$ and is very sensitive to changes in observations leading to poor generalisation performance.
- In the high-dimensional setting ($p > N$), the sample estimator is rank deficient ($\text{rank}(\hat{\boldsymbol{S}}) < p$) and there is no unique inverse for $\hat{\boldsymbol{S}}$. This setting is is extremely important in dynamic graph estimation (Chapter 3).

2.3.5 Sparsity Assumptions

In order to avoid estimating a complete GGM graph where all nodes are connected to each other, one must actively select edges according to some criteria. In the asymptotic setting where $N \gg p$ one can test for the significance of edges by considering the asymptotic distribution of the empirical partial correlation coefficients ($\hat{\rho}_{ij} = -\hat{\Theta}_{ij}/\hat{\Theta}_{ii}^{1/2}\hat{\Theta}_{jj}^{1/2}$) (Drton et al. 2004). However, such a procedure cannot be performed in the high-dimensional setting as it requires that the empirical estimates be positive semi-definite. As discussed below, techniques for stabilising precision matrix estimation mirror those of the linear regression case.

BIC, AIC, ℓ_0-regularisation, hard-thresholding

An alternative approach to testing is to utilise some prior knowledge about the number of edges in the graph. If we assume a flat prior on the model[16] $\mathcal{M}$ and parameters $\boldsymbol{\Theta}(\mathcal{M})$, maximising the approximate posterior probability over models $P(\mathcal{M}|\boldsymbol{Y})$ leads to the Bayesian information criterion for GGM (Foygel et al. 2010)

$$\text{BIC}(\hat{\boldsymbol{\Theta}}_{\text{ML}}) = n(-\log\det(\hat{\boldsymbol{\Theta}}_{\text{ML}}) + \text{trace}(\hat{\boldsymbol{S}}\hat{\boldsymbol{\Theta}}_{\text{ML}})) + \hat{s}\log(n) , \tag{2.3.3}$$

[16]Note: the model here refers to the sparsity pattern, rather than the fact that the distribution is Gaussian.

where $\hat{s}$ is given by the number of unique non-zeros within the ML estimated precision matrix $\hat{\boldsymbol{\Theta}}_{\text{ML}}$. Unfortunately, as in the linear regression setting, the complexity penalty $R() = \hat{s}\log(n)$ is non-convex and thus requires an infeasible computational search.

Graphical Lasso, ℓ_1-regularisation, soft-thresholding

Similarly, to the lasso one may place a Laplace type prior on the precision matrix entries in an effort to directly shrink off-diagonal values (Friedman, Hastie, and R. Tibshirani 2008; J. Lafferty et al. 2012; H. Wang 2012; Zhou et al. 2010). While one could choose to perform full Bayesian inference for the posterior $P(\boldsymbol{\Theta}|\boldsymbol{X}, \gamma)$, as examined by H. Wang (2012), a computationally less demanding approach is to perform MAP estimation resulting in the *graphical lasso* problem (Friedman, Hastie, and R. Tibshirani 2008)

$$\hat{\boldsymbol{\Theta}}_{GL} := \arg\min_{\boldsymbol{\Theta}\succ 0} \left[-\log\det(\boldsymbol{\Theta}) + \text{trace}(\hat{\boldsymbol{S}}\boldsymbol{\Theta}) + \lambda\|\boldsymbol{\Theta}\|_1 \right] , \tag{2.3.4}$$

where $\|\boldsymbol{\Theta}\|_1 = \sum_{i\neq j}|\Theta_{i,j}|$ is the matrix ℓ_1 norm of $\boldsymbol{\Theta}$. As with the lasso, and convex M-estimators generally, there are many efficient optimisation techniques avialiable to solve such problems (see Banerjee et al. (2008) and X. Yuan (2011)). Additionally, due to the convexity of the problem it is easier to theoretically analyse estimator properties, as we will see in the following sections.

Neighborhood selection

A final alternative, and direct link to linear regression, is to attempt to split the problem up and study the edges connecting each variable separately. Such a *neighbourhood selection* process involves fitting a sparse regression model of each variables on the others and then iterating across nodes. Indexing the vector of data for variable i as $\boldsymbol{x}_i$ and the data without variable i as $\boldsymbol{X}_{\setminus i}$, then a sparse set of estimates can be obtained via the lasso

$$\hat{\boldsymbol{\theta}}^{(i)} := \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^{p-1}} (N^{-1}\|\boldsymbol{x}_i - \boldsymbol{X}_{\setminus i}\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1) \quad , \tag{2.3.5}$$

for $i = 1, \ldots, p$. The individual estimates $\hat{\theta}_j^{(i)}$ can the be related to a row or column in the precision matrix $\Theta_{i,j}$. For the static case, such a procedure is shown to be consistent for recovering the support of a precision matrix (Meinshausen

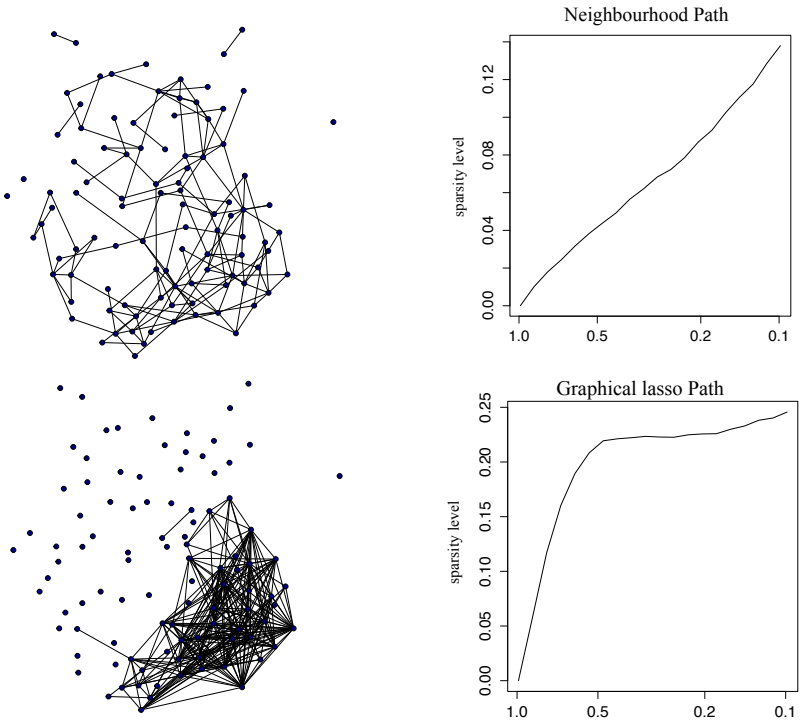


Figure 2.3.3 – Graphical models estimated from a set of log-return prices with glasso (top) and neighbourhood-selection (bottom).

and Bühlmann 2006). However, a neighbourhood selection approach does not necessarily result in positive semi-definite estimates for $\boldsymbol{\Theta}$ and therefore cannot be used to define a GGM. To estimate an actual probability distribution one would be required to find a positive semi-definite solution which is close to the graph selected via neighbourhood selection.

***Remark* 2.4.** *Global vs local graph estimation*

While the graphical lasso and neighbourhood selection approaches can both be used to recover GGM structure, their estimates in practice appear very different. As an example of this, Figure 2.3.3 displays a set of adjacency patterns estimated from 100 stock prices from the S&P500 *over 1258 trading days (2003-2008).*[17]*. To generate the estimates both glasso and neighbourhood selection methods were run (via R package* huge *T. Zhao et al. (2012)) over a path of 20 values of λ. The illustrated graphs correspond to estimates at points*

[17]The data-set is included within the *huge* package, and the 100 stocks chosen correspond to the first 100 of the listed 452 available through the packaged data-set.

in this path selected via an approach known as Stability Approach to Regularisation Selection (StARS) *(H. Liu et al. 2010)*[18]*. Whilst this simply constitutes one empirical example of graph estimation with the two methods, there are clear differences between the estimated graphs. Which is the better method to use will in practice depend on the application and in particular, whether a local (node-wise) or global view (graph-wise) of the dependency structure is prioritised.*

Given that most of this thesis deals with describing dependency structures, the ideas contained in this section will come up frequently; see Chapters 3, 4 and 7. In particular, one of the main novelties of this work is to extend these models to settings where samples are not assumed to be either identically or independently distributed. In the next, and final section of this introduction, we consider how one can analyse M-estimators in a theoretical sense. For example, it is possible to derive bounds on the estimation error under sampling from a presumed ground-truth distribution. Additionally, one may consider how well these estimators can theoretically recover the support of a model, in the context of a GGM how well can we recover the true graph.

2.4 Theory for Regularised estimation

So far, a set of various M-estimators for both linear regression, and precision matrix (GGM) estimation have been discussed. While such estimators have beneficial properties in that they enable high-dimensional estimation, they are intrinsically difficult objects to study statistically. For example, it is hard to derive distributional properties of general M-estimators. This section serves to introduce some recently developed mechanisms for analysing theoretical properties of M-estimators. While this does not give us a full distribution for the estimates, it can give us upper bounds on the estimation errors. In traditional statistical asymptotics the number of data-points can run to infinity, but the number of parameters is fixed. However, since we are applying M-estimators in high-dimensional settings, we are generally more interested in how the estimators behave when the number of data-points is fewer than the number of

[18]The StARS method forms an alternative post-regularisation procedure to BIC/AIC and considers selecting the smallest lambda for which the graph is to some sense sparse, and replicable (under re-sampling c.f. cross-validation).

parameters. To this end, the methods in this section provide a pathway for bounding estimation error even when the model dimensionality grows faster than the number of data-points.

The general framework described here is based on the work of many authors over the past decade, but is most succinctly described in Negahban et al. (2012). In what follows, I aim to give a concise description of the framework described in this paper.

2.4.1 M-estimators and decomposability

To start with, let us consider a more rigorous definition of the M-estimator as described in Eq. 2.0.1:

***Definition* 2.7.** Regularised M-Estimator

Given a convex and differentiable loss function $L(\boldsymbol{\theta}, \boldsymbol{X}_N) : \mathbb{R}^p \times \mathcal{X}^N \mapsto \mathbb{R}$, *where* $\boldsymbol{\theta} \in \mathbb{R}^p$ *is a parameter vector and* $\boldsymbol{X}_N = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$ *is a set of* N *i.i.d observations. A* Regulairsed M-estimator *is defined as:*

$$\hat{\boldsymbol{\theta}}_{\lambda_N} \in \underset{\boldsymbol{\theta} \in \mathbb{R}^P}{\arg\min} \big\{ L(\boldsymbol{\theta}; \boldsymbol{X}_N) + \lambda_N R(\boldsymbol{\theta}) \big\} , \tag{2.4.1}$$

where $\lambda_N > 0$ *is the familiar regularisation penalty, and* $R(\cdot) : \mathbb{R}^p \mapsto \mathbb{R}_+$ *is a norm.*

Now, let the $\bar{L}(\boldsymbol{\theta}) = \mathbb{E}_{X^{(N)}}[L(\boldsymbol{\theta}; X_N)]$ denote the expected loss over the population, and the optimal estimator be obtained by the as $\boldsymbol{\theta}_0 \in \arg\min_{\boldsymbol{\theta}} \{\bar{L}(\boldsymbol{\theta})\}$. The aim of the framework introduced in Negahban et al. (2012) is to derive bounds which hold in high probability, on the difference between the M-estimator $\hat{\boldsymbol{\theta}}_{\lambda_N}$ and the population parameter $\|\hat{\boldsymbol{\Delta}}\| = \|\hat{\boldsymbol{\theta}}_{\lambda_n} - \boldsymbol{\theta}_0\|$. Additionally, the theory provides for bounds on the regulariser norm, for example in the lasso setting we may readily obtain bounds for $\|\hat{\boldsymbol{\theta}}_{\lambda_N} - \boldsymbol{\theta}_0\|_1$.

The general idea of Negahban et al. (2012) is that one may capture error, as measured by the regulariser, by representing the population (denoted with bar) and estimated parameters in terms of so-called *active* (model) and *non-active* (perturbation) subspaces. To capture these different errors, let us introduce the *model subspace* $\mathcal{M} \subseteq \bar{\mathcal{M}} \in \mathbb{R}^p$ and *perturbation subspace*

$\bar{\mathcal{M}}^{\perp} := \{\boldsymbol{v} \in \mathbb{R}^p \,|\, \langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0 \; \forall \, \boldsymbol{u} \in \bar{\mathcal{M}}\}$. Note, that in general, for an appropriately constructed loss function, we will have alignment between the population and estimated spaces, such that $\bar{\mathcal{M}}^{\perp} = \mathcal{M}^{\perp}$. Let us now consider how the regulariser behaves in terms of these model spaces.

***Definition* 2.8.** Decomposable norm

A norm regulariser $R(\cdot)$ *is* decomposable *with respect to a pair of subspaces* $(\mathcal{M}, \bar{\mathcal{M}}^{\perp})$*, where* $\mathcal{M} \subseteq \bar{\mathcal{M}}^{\perp}$ *if:*

$$R(\boldsymbol{\theta} + \boldsymbol{\gamma}) = R(\boldsymbol{\theta}) + R(\boldsymbol{\gamma}) \quad \text{for all } \boldsymbol{\theta} \in \mathcal{M} \,, \boldsymbol{\gamma} \in \bar{\mathcal{M}}^{\perp} \,. \tag{2.4.2}$$

In some sense, if a norm is decomposable, it penalises perturbation maximally as for the model and permutation sub-sets the triangle inequality is required to hold in equality. As an example, the lasso reguariser $R(\boldsymbol{\theta}) \equiv \|\boldsymbol{\theta}\|_1$ obeys decomposability with respect to subspaces of the support of $\boldsymbol{\theta}$. Specifically, if we were to partition $\boldsymbol{\theta}$ into a) model subsets where the true parameters are *non-zero* and b) perturbation subsets where the true parameters are *zero*, then the ℓ_1 norm is decomposable with respect to the resultant subspaces (see Appendix A.1).

It turns out, that if the regulariser of the M-estimator has decomposability property, and its curvature (regulariser parameter λ_N) is appropriate, the estimation error can be restricted to a specific bounded set (for a graphical representation see Figure 2.4.1). To demonstrate this, let us introduce the dual norm of the regulariser:

$$R^*(\boldsymbol{v}) := \sup_{\boldsymbol{u} \in \mathbb{R}^p \setminus \{0\}} \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{R(\boldsymbol{u})} = \sup_{R(\boldsymbol{u}) \leq 1} \langle \boldsymbol{u}, \boldsymbol{v} \rangle \,. \tag{2.4.3}$$

As an example, in the ℓ_1 case the dual norm is given as the infinity norm; $R^*(\boldsymbol{v}) = \|\boldsymbol{v}\|_{\infty} := \max_{i=1,\ldots,p} |v_i|$. The result below, as demonstrated in Negahban et al. (2012) sets a specific condition on the regulariser such that it bounds the gradient of the likelihood in terms of the dual norm.

***Proposition* 2.4.** *Restricted error set*

Suppose that $\hat{\boldsymbol{\theta}}$ *is any optimal solution to the M-estimator in (2.4.1) with a regularisation parameter such that*

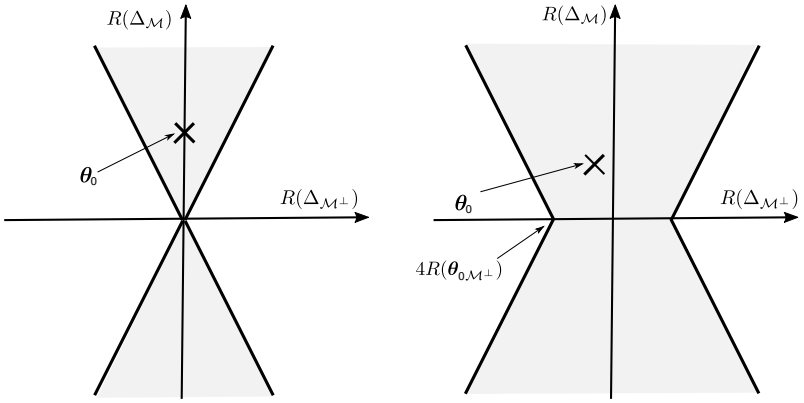


Figure 2.4.1 – Comparison of restricted error $\mathbb{C}(\mathcal{M}, \bar{\mathcal{M}}^\perp)$ sets for the ℓ_1 norm. In this case, a 2-dimensional example $\boldsymbol{\Delta} = (\Delta_1, \Delta_2) \in \mathbb{R}^2$, we have $S = \{2\}$, the model subspace is $\mathcal{M}(S) = \{\boldsymbol{\Delta} \in \mathbb{R}^2 | \Delta_1 = 0\}$ and the perturbation space $\mathcal{M}^\perp(S) = \{\boldsymbol{\Delta} \in \mathbb{R}^2 | \Delta_2 = 0\}$. The set is restricted according to the norms, see 2.4.5. Left: In the ideal case, the optimal point $\boldsymbol{\theta}_0 \in \mathcal{M}$, ie it lies on the y-axis. Right: In the case where the optimal parameters lie outside $\mathcal{M}$ the set is enlarged.

$$\lambda_n \geq 2R^*(\nabla L(\boldsymbol{\theta}_0; X_N)) \; . \tag{2.4.4}$$

Then for any $(\mathcal{M}, \bar{\mathcal{M}}^\perp)$ *for which* R *is decomposable (2.8), the error* $\hat{\boldsymbol{\Delta}} = \hat{\boldsymbol{\theta}}_{\lambda_n} - \boldsymbol{\theta}_0$ *belongs to the set*

$$\mathbb{C}(\mathcal{M}, \bar{\mathcal{M}}^\perp; \boldsymbol{\theta}_0) := \{\boldsymbol{\Delta} \in \mathbb{R}^p \mid R(\boldsymbol{\Delta}_{\bar{\mathcal{M}}^\perp}) \leq 3R(\boldsymbol{\Delta}_{\bar{\mathcal{M}}}) + 4R(\boldsymbol{\theta}_{0,\mathcal{M}^\perp})\} \; . \tag{2.4.5}$$

2.4.2 Restricted Strong Convexity

The difficulty of an estimation problem may be considered in terms of how much error we can expect between our estimate $\hat{\boldsymbol{\theta}}_{\lambda_N}$ and the ideal population parameter $\boldsymbol{\theta}_0$. For example, Figure 2.4.2 demonstrates both a strongly and weakly curved loss function; if the objective is not sufficiently curved, then it may be hard or impossible to recover the correct parameterisation.

Since we have assumed the loss function (2.4.1) is differentiable, we can study the first order Taylor expansion about the population parameter. The error in such a Taylor expansion is given by

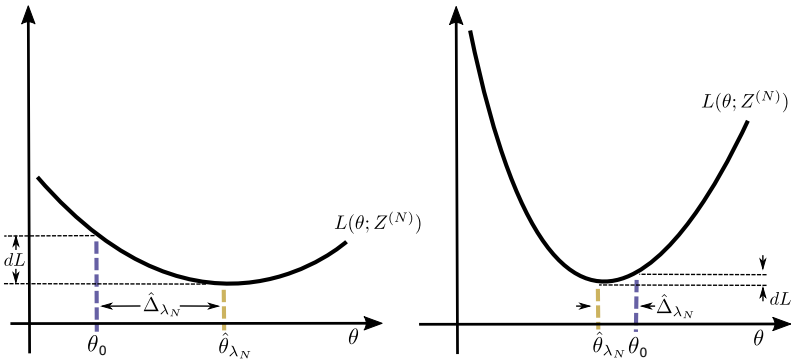


Figure 2.4.2 – The effect of curvature on estimation accuracy. If our loss function is strongly curved (right), then the error between estimated and the population parameterisation is small. The RSC condition places a constraint on the curvature, however, only in directions that are relevant in terms of contributing to $\hat{\boldsymbol{\Delta}} = \hat{\boldsymbol{\theta}}_{\lambda_N} - \boldsymbol{\theta}_0$.

$$\delta L(\boldsymbol{\Delta}, \boldsymbol{\theta}_0) := L(\boldsymbol{\theta}_0 + \boldsymbol{\Delta}) - L(\boldsymbol{\theta}_0) - \langle \nabla L(\boldsymbol{\theta}_0), \boldsymbol{\Delta} \rangle \ .$$

Classically, one way to ensure that the loss function is not too flat is through the requirement of strong convexity. Specifically, this means that the error in the expansion should be bounded from below such that $\delta L(\boldsymbol{\Delta}, \boldsymbol{\theta}_0) \geq \kappa \|\boldsymbol{\Delta}\|^2$, for some constant κ. A strongly convex loss function is also strictly convex, and therefore has a unique and global minima. In the high-dimensional setting where $p > N$, the standard least squares problem with $L \propto \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2$ is not strongly convex; the loss function in some directions will be completely flat and there is no unique minima[19]. Since the lasso uses the same loss function, we may expect that we could be faced with the same issue even in the regularised case. However, if we have a sparse support such that we can construct appropriate model sub-spaces, given appropriate regularisation we may restrict the error vector to the set $\hat{\boldsymbol{\Delta}}_{\lambda_N} \in \mathbb{C}(\mathcal{M}, \bar{\mathcal{M}}^{\perp}; \boldsymbol{\theta}_0)$. Therefore, instead of requiring sufficient curvature in all directions, we can restrict our requirements to the set $\mathbb{C}$ which leads to a so-called *restricted strong convexity* condition.

***Definition* 2.9.** *Restricted Strong Convexity*

The loss function satisfies the restricted strong convexity (RSC) condition with curvature $\kappa_L > 0$ *and* tolerance τ_L *if:*

[19] In the linear regression case we obtain the situation depicted in Figure 2.1.1.

$$\delta L(\boldsymbol{\Delta}, \boldsymbol{\theta}_0) \geq \kappa_L \|\boldsymbol{\Delta}\|^2 - \tau_L^2(\boldsymbol{\theta}_0) \quad \text{for all } \boldsymbol{\Delta} \in \mathbb{C}(\mathcal{M}, \bar{\mathcal{M}}^{\perp}; \boldsymbol{\theta}_0) \ . \tag{2.4.6}$$

The extra tolerance term τ_L is only required when $\boldsymbol{\theta}_0 \notin \mathcal{M}$ and is a consequence of the enhanced size of the set $\mathbb{C}$ at $\boldsymbol{\Delta} = 0$, see Fig 2.4.1. The reader is referred to Negahban et al. (2012) for further details of such cases. For the least squares loss of the lasso, or the log-det loss of the graphical lasso no tolerance function is required.

2.4.3 Bounds for M-estimators

To obtain bounds we introduce a function that consists of a difference of loss-functions and a difference of regularisers

$$F(\boldsymbol{\Delta}) := L(\boldsymbol{\theta}_0 + \boldsymbol{\Delta}) - L(\boldsymbol{\theta}_0) + \lambda_N \big(R(\boldsymbol{\theta}_0 + \boldsymbol{\Delta}) - R(\boldsymbol{\theta}_0) \big) \ . \tag{2.4.7}$$

given that $F(\mathbf{0}) = 0$, the optimal error $\hat{\boldsymbol{\Delta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0$ must maintain $F(\hat{\boldsymbol{\Delta}}) \leq 0$. An important stepping stone to bounding the estimation error is presented in the result below where the difference in objective $F(\cdot)$ can be used to bound the error.

***Lemma* 2.1. Negahban2011** *Lemma 4*

Let $\mathbb{K}(\epsilon) := \mathbb{C} \cap \{\|\boldsymbol{\Delta}\| = \epsilon\}$. *If* $F(\boldsymbol{\Delta}) > 0$ *for all vectors* $\boldsymbol{\Delta} \in \mathbb{K}(\epsilon)$, *then* $\|\hat{\boldsymbol{\Delta}}\| \leq \epsilon$.

The above result can then be used to find a general bound on the estimation error under appropriate regularisation. For example, once we know that the error vector is in the set $\mathbb{C}$ then by obtaining a lower bound on $F(\boldsymbol{\Delta})$ we can find a value for ϵ which bounds the size of the error. In order to relate the regulariser term to the RSC condition (which enables us to construct a lower bound for $F(\cdot)$) the *subspace compatibility constant* is introduced. For any subspace $\mathcal{M} \subseteq \mathbb{R}^p$, the *subspace compatibility constant* is defined, with respect to the pair $(R(\cdot), \|\cdot\|)$ as:

$$\Psi(\mathcal{M}) := \sup_{\boldsymbol{u} \in \mathcal{M} \setminus \{0\}} \frac{R(\boldsymbol{u})}{\|\boldsymbol{u}\|} \ . \tag{2.4.8}$$

The constant $\Psi(\mathcal{M})$ relates the distance measured by the regulariser to that of the defined norm $\|\cdot\|$. It does this in a maximal sense. For example, if we consider the lasso with $R(\boldsymbol{u}) = \|\boldsymbol{u}\|_1$ and we measure the error in the 2-norm $\|\boldsymbol{u}\|_{q=2}$, then for an s-dimensional subspace we have $\|\boldsymbol{u}\|_1 \leq \sqrt{s}\|\boldsymbol{u}\|_2$ and $\Psi(\mathcal{M}) = \sqrt{s}$.

Returning to Eq. 2.4.7, the RSC condition provides a bound for the difference in loss functions, while the set $\mathbb{C}$ provides a bound for the regulariser part. Combining these observations a bound for the error in the general M-estimator setting can be constructed as below:

Proposition* 2.5.** *Bounds for M-estimators (Negahban2011***)*

If the regulariser $R(\cdot)$ is decomposable (2.8) with respect to $(\mathcal{M}, \bar{\mathcal{M}}^{\perp})$, the loss function obeys restricted strong convexity (2.9), and $\lambda_N \geq 2R^*(\nabla L(\boldsymbol{\theta}_0))$. Then any solution $\hat{\boldsymbol{\theta}}_{\lambda_N}$ to the M-estimator problem (2.4.1) satisfies the bound:

$$\|\hat{\boldsymbol{\theta}}_{\lambda_N} - \boldsymbol{\theta}_0\|^2 \leq 9\frac{\lambda_N^2}{\kappa_L^2}\Psi^2(\bar{\mathcal{M}}) + \frac{\lambda_N}{\kappa_L}\left[2\tau_L^2(\boldsymbol{\theta}_0) + 4R^*(\boldsymbol{\theta}_{0\mathcal{M}^{\perp}})\right] . \tag{2.4.9}$$

Furthermore, as a result of the above, in the case where $\boldsymbol{\theta}^* \in \mathcal{M}$ one can obtain bounds on both the norm error, and the regulariser error, such that

$$\|\hat{\boldsymbol{\theta}}_{\lambda_N} - \boldsymbol{\theta}_0\| \leq 9\frac{\lambda_N^2}{\kappa_L}\Psi^2(\bar{\mathcal{M}}) \tag{2.4.10}$$

and $R(\hat{\boldsymbol{\theta}}_{\lambda_N}) \leq 12(\lambda_N/\kappa_L)\Psi^2(\bar{\mathcal{M}})$.

2.4.4 Bounds for the lasso

In the lasso problem (2.1.7), one can either assume that the regression coefficients $\boldsymbol{\theta}_0$ are exactly sparse, referred to as the *strong sparsity* setting, or that they can simply be approximated well by a sparse $\hat{\boldsymbol{\theta}}$, in the *weakly sparse* setting (Bunea et al. 2007). Both of these cases are studied in Negahban et al. (2012), however, only results for the strong sparsity setting are presented here. As further reference, there is a rich variety of work on the theoretical properties of lasso like estimators. These include results for; exact recovery (active predictors) given noiseless observations (Candes et al. 2005; D. Donoho 2006), prediction error consistency and consistency in various ℓ_p norms (Bunea

et al. 2007; Geer et al. 2009; C. Zhang et al. 2008), and variable selection consistency (Meinshausen and Bühlmann 2006; P. Zhao et al. 2006).

The Taylor expansion of the quadratic loss function that underlies the lasso is $\delta L(\boldsymbol{\Delta}, \boldsymbol{\theta}_0) = \langle \boldsymbol{\Delta}, N^{-1}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\Delta}\rangle = N^{-1}\|\boldsymbol{X}\boldsymbol{\Delta}\|_2^2$ and is thus independent of $\boldsymbol{\theta}_0$. In this much simplified case, to maintain the RSC it suffices to establish only a lower bound on $N^{-1}\|\boldsymbol{X}\boldsymbol{\Delta}\|_2^2$ that holds across an appropriately restricted subset of $\boldsymbol{\Delta} \in \mathbb{R}^p$. When $\boldsymbol{\theta}_0$ is exactly sparse, it is intuitive to select the subspace to be equal to the support set $S = \{i \mid (\theta_0)_i \neq 0\}$ (recall that the ℓ_1 regulariser is decomposable, Prop. A.1). One can view this as setting the model subspace $\mathcal{M}(S)$ to look at the components of $\boldsymbol{\theta}$ that relate to the non-zero components of $\boldsymbol{\theta}_0$. We can thus obtain error vectors for the allowed non-zero elements $\boldsymbol{\Delta}_S = \hat{\boldsymbol{\theta}}_S - \boldsymbol{\theta}_{0S}$ corresponding to $\mathcal{M}(S)$ and perturbation terms $\boldsymbol{\Delta}_{S^c} = \hat{\boldsymbol{\theta}}_{S^c} - \boldsymbol{\theta}_{0S^c}$ that correspond to $\bar{\mathcal{M}}^\perp(S)$. Given that $\boldsymbol{\theta}_0 \in \mathcal{M}$, we can consider the restricted set $\hat{\boldsymbol{\Delta}} \in \mathbb{C} = \{\boldsymbol{\Delta} \in \mathbb{R}^p | \ \|\boldsymbol{\Delta}_{S^c}\| \leq 3\|\boldsymbol{\Delta}_S\|_1\}$. In the lasso setting, the RSC condition translates into the well known restricted eigenvalue conditions (Geer et al. 2009; Raskutti et al. 2010):

***Corollary* 2.1.** Restricted Eigenvalue Condition

The RSC (Defintition 2.9) requires the design matrix $\boldsymbol{X}$ *satisfies a* restricted eigenvalue (RE) *condition*

$$\frac{\|\boldsymbol{X}\boldsymbol{\theta}\|_2^2}{N} \geq \kappa_L \|\boldsymbol{\theta}\|_2^2 \quad \text{for all } \boldsymbol{\theta} \in \mathbb{C}(S) \ . \tag{2.4.11}$$

or similarly $\frac{1}{N}\|\boldsymbol{X}\boldsymbol{\theta}\|_2^2 \geq \kappa_L^{'} \|\boldsymbol{\theta}\|_1^2/|S|$.

In many settings it is possible to prove with high-probability that the first order expansion of the loss function satisfies a lower bound. For example, in the lasso case with Gaussian design $X_{i,:} \overset{iid}{\sim} \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, Raskutti et al. (2010, 2011) prove that a bound of the form

$$\frac{\|\boldsymbol{X}\boldsymbol{\theta}\|_2^2}{N} \geq \kappa_1 \|\boldsymbol{\theta}\|_2^2 - \kappa_2 \frac{\log p}{N} \|\boldsymbol{\theta}\|_1^2, \tag{2.4.12}$$

holds with high-probability (greater than $1 - c_1 \exp(-c_2 N)$). Recalling that the RSC condition is only required to hold over the set $\mathbb{C}$. Consider $\boldsymbol{\theta} \in \mathbb{C}$ (2.4.5) with $\boldsymbol{\theta}_0 \in \mathcal{M}(S)$, utilising the subspace compatibility condition on R

we have

$$\|\boldsymbol{\theta}\|_1 \equiv R(\boldsymbol{\theta}) \leq 4\Psi(\bar{\mathcal{M}})\|\boldsymbol{\theta}\| = 4\sqrt{s}\|\boldsymbol{\theta}\|_2 \ ,$$

where $s = |S|$. Thus, given bounds of the form (2.4.12) we can show $N^{-1/2}\|\boldsymbol{X}\boldsymbol{\theta}\|_2 \geq (\kappa_1 - 16\kappa_2 s \log p/N)\|\boldsymbol{\theta}\|_2 \geq \kappa_L\|\boldsymbol{\theta}\|_2$, where $\kappa_L = \kappa_1/2$, and the last bound holds in the case when $N > 64(\kappa_1/\kappa_2)^2 s \log p$. This form of analysis enables us to state with high probability that when a certain amount of data is collected, the RSC condition will be met.

***Proposition* 2.6.** *Error bound for Lasso*

Consider the linear regression model $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta}_0 + \boldsymbol{\epsilon}$. *Assuming that the columns of the design matrix are normalised, ie* $N^{-1/2}\|X_{\cdot,j}\|_2 \leq 1$ *for all* $j = 1, \ldots, p$ *and the noise term* ϵ *posseses* sub-Gaussian *tails such that for a given scale factor* $\zeta < \infty$, $P(\exp(t\epsilon) \leq \exp(\zeta^2 t^2/2))$ *for all* $t \in \mathbb{R}$. *For a suitably chosen* $\lambda_N = 4\zeta\sqrt{(\log p)/N}$, *then with high-probability we recover the bound*

$$\|\hat{\boldsymbol{\theta}}_{\lambda_N} - \boldsymbol{\theta}_0\|_2^2 \leq \frac{64\zeta^2}{\kappa_L^2}\frac{s\log p}{N} \ . \tag{2.4.13}$$

Proof. The RSC condition can be demonstrated to hold in high probability using results similar to those of Eq. 2.4.12. One is also required to check that the regulariser is appropriately set. Specifically, this should satisfy $\lambda_N \geq 2R^*(\nabla L(\boldsymbol{\theta}_0))$. In the lasso case we obtain $2R^*(\nabla L(\boldsymbol{\theta}_0)) = 2\|N^{-1}\boldsymbol{X}^\top\boldsymbol{\epsilon}\|_\infty$, which can be bounded considering the sub-Gaussian error structure. For a full proof see Negahban et al. (2012).

As with the lasso, one can also obtain similar bounds for regularised covariance/precision estimation (Bühlmann et al. 2011; Lam et al. 2009; Ravikumar, Wainwright, and J.D. Lafferty 2010; Ravikumar, Wainwright, Raskutti, et al. 2011; Rothman et al. 2008; Saegusa et al. 2016). The following bound can be considered analogous to that of Prop. 2.6, except for the ℓ_1 penalised log-det problem (2.3.4), c.f. the graphical lasso:

***Proposition* 2.7.** *Bound for* ℓ_1 *Log-Det Estimation (Ravikumar, Wainwright, Raskutti, et al. 2011)*

If the rescaled $X_i/\sqrt{\Sigma^0_{ii}}$ *are sub-Gaussian, the precision matrix has* s *true* edges, *and a sample size of* $N = \Omega(d^2 \log p)$, *where* d *is the maximum node*

degree, then under suitable regularisation conditions the precision matrix is bounded as

$$\|\hat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_F = \mathcal{O}\left(\sqrt{\frac{(s+p)\log p}{N}}\right) , \tag{2.4.14}$$

with probability $1 - 1/p^{\tau-2} \to 1$*, where* $\tau > 2$*.*

Proof. The above result is a summarised version of Theorem 1 in Ravikumar, Wainwright, Raskutti, et al. (2011) specific to sub-Gaussian sampling for X_i. The parameter τ reflects the rate of convergence in probability, it affects the appropriate setting of both the regularsation constant λ and sample size required for the claims. A high τ results in high probability claims, but also an increased lower bound on the sample size. Again, as in the lasso case, one needs to check that the glasso estimator meets both an RSC condition and that there is sufficient regularisation. The specifics of such sample size and regularisation requirements are omitted here for readability.

It is worth noting that the above result holds for estimating the precision matrix of sub-Gaussian random variables. In the case of a GGM, the precision matrix elements can be considered as specification for a graph $G(V, E)$ as discussed in Section 2.3. In the more general case there is not such a clear interpretation of the off-diagonal precision matrix structure. The result is typical for high-dimensional graph selection problems. For similar related results see; Rothman et al. 2008, or Ravikumar, Wainwright, and J.D. Lafferty 2010 who also consider a binary Ising model, and Lam et al. 2009 who additionally consider non-convex penalties.

2.4.5 Support Recovery (Primal-Dual Witness)

In addition to bounding the error $\|\hat{\mathbf{\Delta}}\|$ of an M-estimator, it is also of interest to consider whether the methods can recover the true model structure. One measure of such recovery is known as *sign consistency,* and is defined with respect to the event

$$E_{\mathcal{M}}(\hat{\boldsymbol{\theta}}; \boldsymbol{\theta}_0) := \{\text{sign}(\hat{\theta}_i) = \text{sign}(\theta_{0,i}) \; \forall i \in S\} ,$$

where S is a set which indicates the true support of $\boldsymbol{\theta}_0$, i.e. $S = \text{supp}(\boldsymbol{\theta}_0)$. On the occurrence of $E_{\mathcal{M}}$, both the estimated parameter structure, in terms of its support, and the true parameter structure are the same. In fact, sign consistency is stronger than this, in that it requires both the support and the sign of the parameters to be successfully recovered. In an asymptotic sense, we may typically aim to demonstrate $P[E_{\mathcal{M}}(\hat{\boldsymbol{\theta}}; \boldsymbol{\theta}_0)] \rightarrow 1$ as $p, T \rightarrow \infty$.

A popular and fairly general approach to demonstrating such consistency is known as the *primal-dual witness* method (Wainwright 2009). Principally, this method works by deconstructing the KKT conditions (c.f. 2.1.14) of the M-estimator (2.4.1) into two blocks. Let us label these conditions $\text{KKT}(S, \partial R(\boldsymbol{\theta}_S))$ and $\text{KKT}(S^c, \partial R(\boldsymbol{\theta}_{S^c}))$, such that they respectively concern the true model components $\boldsymbol{\theta}_{0;S}$ and the compliment $\boldsymbol{\theta}_{0;S^c}$. The primal-dual witness approach consists of the following steps:

(1) Solve a restricted problem; $\tilde{\boldsymbol{\theta}} := \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}; X) + \lambda_N R(\boldsymbol{\theta})$, such that $\boldsymbol{\theta}_{S^c} = 0$. This constitutes a restricted estimation problem, whereby the compliment of the support is artificially fixed to be zero. It verifies that the $\text{KKT}(S, \partial R(\boldsymbol{\theta}_S))$ is satisfied under the block corresponding to the true support.
(2) Select w as the sub-differential of the regulariser $R(\cdot)$ evaluated at $\tilde{\boldsymbol{\theta}}$. Solve for the subgradient over the components S^c by using $\text{KKT}(\tilde{\boldsymbol{\theta}}, \boldsymbol{w})$
(3) Check that $E_{\mathcal{M}}$ occurs and that the sub-gradient in step (2) is sufficiently small $\|\boldsymbol{w}_{S^c}\|_\infty < \min(|a|, |b|)$. Note: in the lasso (and ℓ_1 penalty) case the limit is simply $\|\boldsymbol{w}_{S^c}\|_\infty < 1$ corresponding to Eq. 2.1.12.

To ensure the $E_{\mathcal{M}}$ occurs in high-probability, one needs to perform the above procedure in the presence of sampling noise via X. The form of this noise will be model dependent, however, typically one may assess recovery under the random design setting and assume the co-variates are drawn from a Gaussian (Bühlmann et al. 2011; Geer et al. 2009) . Again, in order to derive bounds on the recovery, one must make some assumptions on the problem design. Such conditions are often referred to as *incoherence* or *irrepresentability* conditions.

In the graphical structure learning setting these conditions act to limit correlation between edges and restrict the second order curvature of the loss

function. In the multivariate Gaussian case (log-det likelihood) the Hessian $\mathbf{\Gamma}^0 \equiv \nabla^2_{\mathbf{\Theta}} L(\mathbf{\Theta})|_{\mathbf{\Theta}_0}$ relates to the Fisher information matrix, such that $\Gamma^0_{(j,k)(l,m)} = \text{Cov}(X_j X_k, X_l X_m)$. Written in this form we can understand the Fisher matrix as relating to the covariance between *edge variables* defined as $Z_{(i,j)} = X_i X_j - \mathbb{E}[X_i X_j]$, where $i, j \in \{1, \ldots, p\}$:

***Definition* 2.10.** *Incoherence Condition*

Let S denote the set of components relating to true edges in the graph and S^c its compliment. For example, $\mathbf{\Gamma}^0_{SS}$ refers to the sub matrix of the Fisher matrix relating to edges in the true graph. The incoherence condition states that there exists some $\alpha \in (0, 1]$ such that

$$\max_{e \in S^c} \|\mathbf{\Gamma}^0_{eS} (\mathbf{\Gamma}^0_{SS})^{-1}\|_1 \leq (1 - \alpha)\,.$$

In the multivariate Gaussian case we have $\max_{e \in S^c} \|\mathbb{E}[Z_e Z_S^\top]\mathbb{E}[Z_S Z_S^\top]^{-1}\|_1 \leq (1 - \alpha)$. One can therefore interpret the incoherence condition as a statement on the correlation between edge variables which are outside the model subspace $Z_{(i,j)}$ such that $(i, j) \notin E$, to those contained in the true model $(i, j) \in E$. In practice, this sets bounds on the types of graph and associated covariance structures which estimators such as graphical lasso can recover (see the discussion Sec. 3.1.1 Ravikumar, Wainwright, Raskutti, et al. (2011) and Meinshausen (2008)). Again, as with the issue of faithfulness, such restrictions force us to be careful when interpreting the edge structure as parameterised via the estimated precision matrix. With such concerns in mind, one can utilise the primal-dual witness approach to derive finite sample, high probability bounds on model recovery.

***Proposition* 2.8.** *Model Selection Consistency (Ravikumar, Wainwright, Raskutti, et al. 2011)*

Let $\theta_{\min} \equiv \min_{i,j} |\Theta^0_{i,j}|$ *and* d *be the maximum degree of a the true graph. Given the incoherence condition of (2.10) and sufficient samples* $N = \Omega((d^2 + \theta^{-2}_{\min})\tau \log p)$ *, then under the same requirements as Prop. 2.7 one obtains the model consistency result*

$$P[E_{\mathcal{M}}(\hat{\mathbf{\Theta}}; \mathbf{\Theta}^0)] \geq 1 - 1/p^{\tau} \rightarrow 1 \ . \tag{2.4.15}$$

Proof. A full proof of the above can be found in Theorem 2 of Ravikumar, Wainwright, Raskutti, et al. (2011) and follows the primal-dual witness approach as discussed.

Results such as the above are typical when studying theoretical properties of M-estimators and many papers discuss finite sample bounds on the error, as per Eqs. 2.4.13 and 2.4.14, alongside support recovery akin to Prop. 2.8. In the following chapters, estimators such as the graphical lasso are extended to consider settings where the underlying distribution can change as a function of time. In Chapter 4 some asymptotic properties of these estimators are discussed. In order to provide statistical guarantees on these estimators in high-dimensions one may attempt to adapt the results of this section.

2.5 Summary

In this chapter we have reviewed several key ideas which are used throughout the rest of this thesis. As with the models so-far discussed, most regularised M-estimation problems are formulated in a setting where the underlying model is not expected to change as a function of where or how the samples are observed. Such assumptions relate to commonly used independent and identically distributed sampling schemes. However, in many applications this assumption seems suspect. For example, as discussed in Chapter 2, these assumptions often limit the insight we can get from statistical models. In the next chapter, we will discuss several ways in which we may extend graph estimation methods such as graphical lasso to dynamic settings. The concept of regularised M-estimation plays a key role in these extensions by giving us a principled way

to incorporate prior knowledge about model smoothness; i.e. how fast, and in which ways a model may change over time. In the next chapter, the discussion will primarily be focussed on different smoothness assumptions alongside the development of algorithmic methods that make it feasible to identify such dynamic models. To this end, a new class of ADMM algorithm is developed and some empirical properties of dynamic estimators are demonstrated. Chapter 4 considers theoretical analysis of dynamic graphical estimators, and extends the theoretical discussion of M-estimators (Sec. 2.4) to the dynamic setting.

Appendix A

A.1 Some properties of functions

For completeness, the properties of some commonly studied classes of functions are given below.

Definition. *A function $\|\cdot\| : X \mapsto \mathbb{R}$ is a norm if it obeys the following three properties for all $\boldsymbol{x}, \boldsymbol{y} \in X$ and $\lambda \in \mathbb{R}$*

- $\|\lambda \boldsymbol{x}\| = |\lambda| \|\boldsymbol{x}\|$ *(absolute homogeneity)*
- $\|\boldsymbol{x} + \boldsymbol{y}\| \leq \|\boldsymbol{x}\| + \|\boldsymbol{y}\|$ *(the triangle inequality)*
- *If $\|\boldsymbol{x}\| = 0$, then $\boldsymbol{x}$ is the zero vector (defines zero length)*

Definition. *A function $f : \mathbb{R}^p \to \mathbb{R}$ is convex if $\mathrm{dom} f$ is a convex set and it satisfies the condition*

$$f(a\boldsymbol{x} + b\boldsymbol{y}) \leq af(\boldsymbol{x}) + bf(\boldsymbol{y}) \ ,$$

for all $a, b \in \mathbb{R}$ and points $\boldsymbol{x}, \boldsymbol{y} \in \mathrm{dom} f$ with $a + b = 1$ and $a \geq 0$, $b \geq 0$.

Proposition. *Minima of convex functions*

Let f be a convex function $f \in \mathrm{conv}(\mathcal{V})$, then:

- *the set of minimisers $\arg\min_{\boldsymbol{x} \in \mathcal{V}} f(\boldsymbol{x})$ is convex (possibly empty).*
- *if $\hat{\boldsymbol{x}}$ is a local minimum of f, then $\hat{\boldsymbol{x}}$ is in fact a global minimum such that, $\hat{\boldsymbol{x}} \in \arg\min_{\boldsymbol{x} \in \mathcal{V}} f(\boldsymbol{x})$.*

Proposition. *Let $\boldsymbol{x} \in \mathbb{R}^p$, the ℓ_q norm $\|\boldsymbol{x}\|_q$ for $0 \leq q < 1$ is not convex.*

Proof. Proof of Prop. 2.1

It is sufficient to demonstrate that the epigraph of $\|\boldsymbol{x}\|_q$ is not a convex set. Let $S = \mathrm{epi}(\|\boldsymbol{x}\|_q)$. Given, $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$, convexity requires $(1-t)\boldsymbol{x}_1 + t\boldsymbol{x}_2 \in S$ for all $t \in [0, 1]$. Without loss of generality let $\boldsymbol{x}_1 = \boldsymbol{0} \in \mathbb{R}^p$, $\|\boldsymbol{x}_2\|_1 = 1$. The ℓ_1 norm defines a straight line between $x_{1,i}$ and $x_{2,i}$ (it is the limiting convex case) therefore $\|t\boldsymbol{x}_2\|_q > \|t\boldsymbol{x}_2\|_1$ implies non-convexity. In the case $\|\boldsymbol{x}\|_1 = 1$, we have $\|\boldsymbol{x}\|_1 - \|\boldsymbol{x}\|_q^q = 1 - \sum_{j=1}^p |x_j|^{q-1}$ and thus$\|\boldsymbol{x}\|_q = (\sum_{j=1}^p |x_j|^{q-1})^{1/q}$, noting that $a < 0$ and $0 < |x_j| < 1 \implies |x_j|^a > 1$ gives $\|\boldsymbol{x}\|_q > p^{1/q} > \|\boldsymbol{x}\|_1 = 1$.

□

Proposition. *Decomposability of ℓ_1 norm*

For each subset $S \subseteq \{1, \ldots, p\}$ defining subspace pairs as:

$$\mathcal{M}(S) := \{\boldsymbol{\theta} \in \mathbb{R}^p \mid \theta_j = 0 \text{ for all } j \in S^c\} , \tag{A.1}$$

$$\bar{\mathcal{M}}^{\perp}(S) := \{\boldsymbol{\theta} \in \mathbb{R}^p \mid \theta_j = 0 \text{ for all } j \in S\} , \tag{A.2}$$

then $\mathcal{M}^{\perp}(S) = \bar{\mathcal{M}}^{\perp}(S)$ and it follows that the ℓ_1 norm is decomposable:

$$\|\boldsymbol{\theta}_S + \boldsymbol{\theta}_{S^c}\|_1 = \|\boldsymbol{\theta}_S\|_1 + \|\boldsymbol{\theta}_{S^c}\|_1 , \tag{A.3}$$

for all $\boldsymbol{\theta}_S \in \mathcal{M}(S)$, $\boldsymbol{\theta}_{S^c} \in \bar{\mathcal{M}}^{\perp}(S)$.

Proof. This can be shown by construction as we have $\boldsymbol{\theta}_S = (\theta_S \in \mathbb{R}^s, \mathbf{0} \in \mathbb{R}^{p-s})$, $\boldsymbol{\theta}_{S^c} = (\mathbf{0} \in \mathbb{R}^s, \theta_{S^c} \in \mathbb{R}^{p-s})$, thus $\|\boldsymbol{\theta}_S + \boldsymbol{\theta}_{S^c}\| = \|(\theta_s, \mathbf{0}) + (\mathbf{0}, \theta_{S^c})\|_1 = \|\boldsymbol{\theta}_S\| + \|\boldsymbol{\theta}_{S^c}\|_1$.

□

Chapter 3

Dynamic Graphical Models

Graphical models provide a powerful mechanism for encoding properties of joint distributions over large sets of random variables. However, until now our discussion of graphical models has been limited to the setting where the graph and joint distribution is assumed identical for all samples. In applications such as functional connectivity analysis in neuroscience or modelling feature dependency across flows of network data, the assumption that an underlying generative distribution is constant over time is extremely limiting.

As demonstrated in Section 2.3, graphical models may be estimated in the stationary setting by combining appropriate loss functions with a sparsity inducing regulariser. For example, the *graphical lasso* (Banerjee et al. 2008; Friedman, Hastie, and R. Tibshirani 2008) penalizes a Gaussian likelihood with an ℓ_1 norm applied to the precision matrix (Eq. 2.3.4). Of particular interest in this chapter is how the graphical model evolves over time, and how prior knowledge of such dynamics can be exploited to constrain and estimate graph structure. It is worth remarking that the whole field of dynamic graph estimation is relatively young– examples of early work in this direction can be found in Ahmed et al. (2009) and Zhou et al. (2010). A simple way of extending static estimators to allow for model variation is through a localised moving window approach. However, such estimators are restricted in that they can only access local data, and assume the graph structure varies in a smooth manner. In this chapter, a more principled approach of regularised M-estimation

is investigated in order to generalise the moving window approach, and incorporate different smoothness assumptions. Notably, this allows us to not only detect smoothly varying graphical structures, but also sharp discontinuities, or *changepoints*, in the graphical structure.

The chapter proceeds as follows: We start by discussing the extension of multivariate likelihoods and sparsity inducing regularisers to the dynamic setting. Several smoothing methods are then discussed; these either involve local estimation via a moving window, or by introducing additional regularisers to constraint temporal model variation. One novel contribution of this thesis is then introduced in the form of a new type of group-fused regulariser that aims to discover systematic changes in graph structure. The flexibility of dynamic graphical models makes estimation a computationally challenging problem. In order to estimate dynamic graphs a new class of ADMM algorithm is developed. This is then used to examine empirical properties of some regularised dynamic graphical models in a synthetic setting. Finally, the chapter concludes with a set of example applications looking at the analysis of time-course microarray data and dependency of real-world network traffic metrics. This chapter contains sections of work derived from Gibberd, Evangelou, et al. (2016) and Gibberd and Nelson (2014a, 2016, 2017).

3.1 Model and Estimator Formulation

Before proceeding, it is worth noting that the estimators that follow utilise a loss-function primarily based on the Gaussian likelihood[1]. As discussed in the previous chapter (Sec. 2.3) the multivariate Gaussian model allows a specific interpretation of its parameters in terms of graphical model structure. Specifically, this enables us to estimate dependency structure from data by considering the pattern of non-zero entries in the precision matrix. In order to extend the model to allow dynamics in the model, we may simply allow the parameters of the Gaussian distribution to vary at each discrete time point

$$\vec{X}^{(t)} \sim \mathcal{N}(\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}) \ ,$$

[1]A brief discussion on pseudo-likelihood approaches to dynamic graph estimation can be found in Sec. 3.1.4.

for $t = 1, \ldots, T$. If the precision matrices $\boldsymbol{\Theta}^{(t)} := (\boldsymbol{\Sigma}^{(t)})^{-1}$ are well defined for each time-point then the dependency structure of the series can be captured by a dynamic Gaussian Graphical Model (GGM). This comprises a collection of time-indexed graphs $G^{(t)} = (V^{(t)}, E^{(t)})$, where the vertices $V^{(t)} = \{1, \ldots, p\}$ represent each variable $\vec{X}^{(t)} = (X_1^{(t)}, \ldots, X_p^{(t)})^\top$ and the edges $E^{(t)}$ represent conditional dependency relations between variables over time.

Traditionally, as discussed in the previous chapter, estimation of GGM's is usually performed under the assumption of stationarity, whereby we have *identically* distributed draws from a Gaussian model. Letting $\boldsymbol{X} = (\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(T)})$ be a set of observations and assuming $\vec{X}^{(t)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, one can construct an estimator for $\boldsymbol{\Sigma}^{-1}$ by maximizing the log-likelihood $\hat{\boldsymbol{\Theta}} := \arg\max_{\boldsymbol{U}} \big[\log\big(\det(\boldsymbol{U})\big) - \operatorname{tr}(\hat{\boldsymbol{S}}\boldsymbol{U})\big]$, where $\hat{\boldsymbol{S}} = \boldsymbol{X}\boldsymbol{X}^\top/2T$. However, if the distribution can change arbitrarily over time, because only one data-point may be observed at each node per time-step the traditional empirical covariance estimator $\hat{\boldsymbol{S}}^{(t)} \propto \boldsymbol{x}^{(t)}(\boldsymbol{x}^{(t)})^\top$ will be rank deficient. To this end, estimation of the precision matrix requires additional modeling assumptions. A strategy explored in several recent works (Ahmed et al. 2009; Danaher et al. 2013; Gibberd and Nelson 2014b) is to introduce priors in the form of regularized M-estimators, viz.

$$\hat{\boldsymbol{\Theta}}^{(t)} := \arg\min_{\boldsymbol{U}^{(t)} \succeq 0} \big[\mathcal{L}(\{\boldsymbol{U}^{(t)}\})\big], \tag{3.1.1}$$

with the cost function

$$\mathcal{L}(\{\boldsymbol{U}^{(t)}\}) = \sum_{t=1}^{T} L(\boldsymbol{U}^{(t)}, \boldsymbol{x}^{(t)}) + R_{\text{Shrink}}(\{\boldsymbol{U}^{(t)}\}) + R_{\text{Smooth}}(\{\boldsymbol{U}^{(t)}\}) \,. \tag{3.1.2}$$

As in the case of the graphical lasso, or lasso, the loss function $L(\boldsymbol{U}^{(t)}, \boldsymbol{x}^{(t)})$ may be considered as being proportional to the negative log-likelihood. The penalty terms R_{Shrink}, R_{Smooth} correspond respectively to prior shrinkage and smoothness assumptions. Typically, the smoothness term will be a function of the difference between estimates $\boldsymbol{U}^{(t)} - \boldsymbol{U}^{(t-1)}$, whereas the shrinkage term will act at specific time points, i.e. directly on $\boldsymbol{U}^{(t)}$. With regards to GGM, such an estimator not only allows us to place shrinkage priors on the individual precision matrices, akin to the graphical lasso (Banerjee et al. 2008; Friedman, Hastie, and R. Tibshirani 2008), but also control variation in this model structure over time. In what follows, several different forms of estimator are

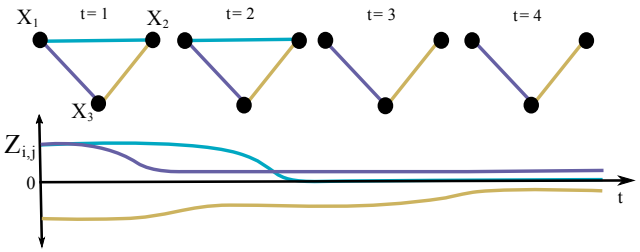


Figure 3.1.1 – Graphical depiction of continuously smooth varying graph structure as studied in Kolar et al. (2011) and Zhou et al. (2010).

introduced within this framework which allow for continuously smooth, and piecewise constant graphical models.

3.1.1 Smoothly Varying Graphical Lasso

Several approaches which incorporate dynamics in such graphical estimators have been suggested. Zhou et al. (2010) and Kolar et al. (2011) utilize a local estimate of the covariance in the term $L(\boldsymbol{U}^{(t)}; \boldsymbol{x}^{(t)})$ by replacing $\hat{\boldsymbol{S}}$ in the graphical lasso with a time-sensitive weighted estimator

$$\hat{\boldsymbol{S}}^t = \sum_{s=t-M}^{t+M} h_t^{(s)} \boldsymbol{x}^{(s)} (\boldsymbol{x}^{(s)})^\top / \sum_{s=t-M}^{t+M} h_t^{(s)}, \tag{3.1.3}$$

where $h_t^{(s)} = K(|s-t|/M)$ are weights derived from a symmetric non-negative smoothing kernel function $K(\cdot)$ with width M. The resulting graphs $\hat{G}^t$ are now representative of some temporally localized data. By making some smoothness assumptions on the underlying covariance matrix such a kernel estimator can be shown to be risk consistent (Zhou et al. 2010). Kolar et al. (2011) go further, and demonstrate that placing assumptions on the Fisher information matrix allows one to prove consistent estimation of graph structure in such dynamic GGM. The moving window approach does utilise the smoothing regulariser R_{smooth}. In the following sections, we will discuss how one can harness this additional complexity penalty to promote further, potentially more insightful, smoothness constraints.

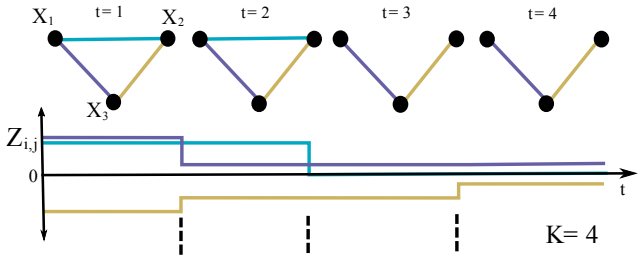


Figure 3.1.2 – Changepoints in graph structure are counted at the individual edge level. The works of Gibberd and Nelson (2014a) and Monti et al. (2014) can be considered under this model.

3.1.2 Independently Fused Graphical Lasso (IFGL)

There are many situations where we might expect continuous smoothness assumptions to be broken. For example, activity patterns in the brain may switch relatively rapidly when performing different tasks (Monti et al. 2014). Rather than adopt a continuously varying graphical model we now let it be piecewise constant– points in time where structure changes can be referenced by a set of K changepoints $\mathcal{T} = \{\tau_1, \ldots, \tau_K\}$ where $\tau_i \in \{1, \ldots, T\}$. When considering piecewise dynamics, there are a variety of smoothness measures that one could adopt especially when we are dealing with multivariate distributions. The discussion in this chapter focusses on two broad classes of *edge-wise*, and *graph-wise* smoothness.

In the edge-wise case, one can relate smoothness to the number of jumps or changepoints in the edge structure, i.e. entries in the precision matrix $\boldsymbol{\Theta}^t := (\boldsymbol{\Sigma}^t)^{-1}$. Such a model may be written as a constrained multivariate normal:

(3.1.4)

$$(X_1^{(t)}, \ldots X_p^{(t)})^\top \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^{(t)}) \quad , \quad \text{such that } \sum_{t=2}^{T-1} \|\boldsymbol{\Theta}_{\setminus ii}^{(t)} - \boldsymbol{\Theta}_{\setminus ii}^{(t-1)}\|_0 = 2K \ .$$

In the above, the ℓ_0 norm counts the number of non-zero elements of the differenced precision matrices, i.e. $\|\boldsymbol{X}\|_0 = |\{X_{i,j} \neq 0 \; \forall i \neq j\}|$. Such a constraint counts changes on each edge of the graphical model separately, hence the factor of 2 accounting for symmetry in the matrices. As discussed in Section 2.1.2, the ℓ_0 counting norm is non-convex. A penalised likelihood

approach to estimate $\{\boldsymbol{\Sigma}^{(t)}\}$ which uses the ℓ_0 penalty to count changepoints will therefore also be non-convex, resulting in a computationally infeasible problem. As with most regularisation problems, one may consider a convex relaxation. In this case, we relax the non-convex count on the changepoints, to a convex penalty over the difference of the precision matrices. The resulting objective function is henceforth referred to as the *independently fused graphical lasso (IFGL)* estimator, and takes the following form[2]:

$$\mathcal{L}(\{\boldsymbol{U}^{(t)}\}) = \overbrace{\sum_{t=1}^{T} \overbrace{\big(-\operatorname{logdet}(\boldsymbol{U}^{(t)}) + \operatorname{tr}(\hat{\boldsymbol{S}}^{(t)}\boldsymbol{U}^{(t)})\big)}^{L(\{\boldsymbol{U}^{(t)}\},\{\boldsymbol{x}^{(t)}\})}}^{\propto -\text{Likelihood}} + \overbrace{\lambda_1 \sum_{t=1}^{T} \|\boldsymbol{U}^{(t)}_{\setminus ii}\|_1}^{R_{\text{Shrink}}}}^{\ell_1 \text{ shrinkage}} \ldots$$
$$\ldots + \overbrace{\lambda_2 \sum_{t=2}^{T} \|\boldsymbol{U}^{(t)}_{\setminus ii} - \boldsymbol{U}^{(t-1)}_{\setminus ii}\|_1}^{R_{\text{Smooth}}}. \tag{3.1.5}$$
$$\ell_1 \text{ edge smoothing}$$

The works of Ahmed et al. (2009), Gibberd and Nelson (2014a), and Monti et al. (2014) can be considered in this setting. The work of Danaher et al. (2013) and S. Yang et al. (2015) also uses a similar smoothing strategy to the above, however, they do not consider the case of dynamics, and have a-priori blocks of observations that have a known size.

3.1.3 Group-Fused Graphical Lasso (GFGL)

In the previous section we considered that graph structure may change edge by edge, and changepoints were counted at an edge level. Alternatively, one may expect that the graph structure underlying a process may change systematically, such that the whole, or many of the edges change dependency structure in a short period of time. One of the main contributions of this thesis is to introduce estimators which are capable of detecting such graph dynamics.

Consider the Gaussian model as before, but where time variation is penalised across the graph whereby:

[2]Recall, the notation $\boldsymbol{U}_{\setminus ii}$ refers to a matrix with the iith elements removed, for example $\|\boldsymbol{U}^t_{\setminus ii}\|_1 = \sum_{i \neq j} |U^t_{i,j}|$.

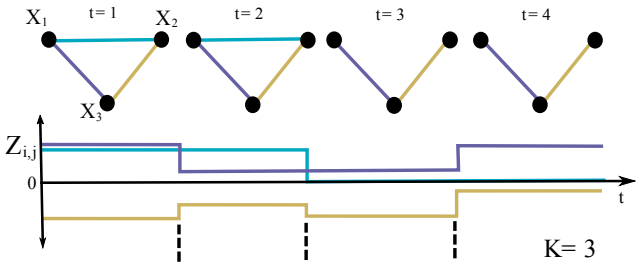


Figure 3.1.3 – Group-piecewise constant graph structure, edges are restricted so that changepoints are counted across the graph. It encodes an assumption that many edges may change simultaneously, as first introduced in (Gibberd and Nelson 2017).

(3.1.6)
$$(X_1^{(t)}, \dots X_p^{(t)})^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}^{(t)}) \quad , \quad \text{such that } \{\|\mathbf{\Theta}_{\setminus ii}^{(t)} - \mathbf{\Theta}_{\setminus ii}^{(t-1)}\| \neq 0\} = K \ .$$

In this case, the estimator assumes that many of the active edges may change at the same time and that changes in graphical structure are therefore syncronised. As in the IFGL case, the exact penalisation problem associated with (3.1.6) is non-convex due to the requirement to count changepoints. Solving such a problem is challenging. If the changepoints were known in advance then local graph estimation could be performed. However, the changepoints cannot be found without first estimating the graphs. Previous approaches (Angelosante et al. 2011) have resorted to using dynamic programming alongside the ℓ_1 graph learning approaches. Unfortunately, these are restricted to quadratic computational complexity as a function of the time-series length.

An alternative strategy, as with IFGL, is to construct a convex relaxation. Specifically, one may construct the *group-fused graphical lasso (GFGL)* estimator (Gibberd and Nelson 2017), with objective:

$$\mathcal{L}(\{\boldsymbol{U}^{(t)}\}) = \underbrace{\sum_{t=1}^{T} \overbrace{\big(-\text{logdet}(\boldsymbol{U}^{(t)}) + \text{tr}(\hat{\boldsymbol{S}}^{(t)}\boldsymbol{U}^{(t)})\big)}^{L(\{\boldsymbol{U}^{(t)}\},\{\boldsymbol{x}^{(t)}\})}}_{\propto -\text{Likelihood}} + \underbrace{\overbrace{\lambda_1 \sum_{t=1}^{T} \|\boldsymbol{U}_{\setminus ii}^{(t)}\|_1}^{R_{\text{Shrink}}} \dots}_{\ell_1 \text{ shrinkage}} \tag{3.1.7}$$

$$\overbrace{\ldots + \underbrace{\lambda_2 \sum_{t=2}^{T} \|\boldsymbol{U}_{\setminus ii}^{(t)} - \boldsymbol{U}_{\setminus ii}^{(t-1)}\|_F}_{\text{group } \ell_{2,1} \text{ smoothing}}}^{R_{\text{Smooth}}}.$$

In this chapter we describe how one can efficiently solve the GFGL problem and demonstrate two key properties consistent with the model formulation in (3.1.6), specifically:

(1) Estimated precision matrices encode a sparse dependency structure whereby many of the off axis entries are exactly zero, i.e. $\hat{\Theta}_{i,j}^{t} = 0$.
(2) Precision matrices maintain a piecewise constant structure where changepoints tend to be grouped *across* the precision matrix, such that for many edges indexed by (i, j) and (l, m) the estimated changepoints for the two edges are the same, viz. $\hat{\mathcal{T}}_{i,j} = \hat{\mathcal{T}}_{l,m}$ where $\hat{\mathcal{T}}_{i,j} = \{\hat{\tau}_{ij}^{(1)}, \ldots, \hat{\tau}_{ij}^{(\hat{K}_{ij})}\}$ represents the set of $\hat{K}_{ij}$ estimated changepoints $\tau_{ij}^{(k)}$ on the i, jth edge).

3.1.4 Fused Neighbourhood Selection

In the previous sections, we introduced M-estimators which operate in conjunction with a multivariate Gaussian likelihood. As in the i.i.d. setting, one can also consider local methods for graph estimation via pseudo-likelihood neighbourhood selection approaches (c.f. Section. 2.3.5). Such approaches have also been extended to the dynamic setting, examples of this can be seen in the works of Ahmed et al. (2009) and Kolar et al. (2012). Instead of a log-det problem, these authors frame the problem as a temporally sensitive neighbourhood selection problem extending the static case given in Eq. 2.3.5. By placing a *temporal-difference lasso* regulariser

$$R_{\text{td}} = 2\lambda_1 \sum_{t=1}^{T} \sum_{j \neq i=1}^{p} |u_{i,j}^{(t)}| + 2\lambda_2 \sum_{t=1}^{T} \|\boldsymbol{u}_i^{(t)} - \boldsymbol{u}_i^{(t-1)}\|_2 \ ,$$

on the node-wise loss $L(\boldsymbol{u}_i) = \sum_{t=1}^{T} (x_i^{(t)} - \sum_{j \neq i} x_j^{(t)} u_j^{(t)})^2$, it is possible to recover a piecewise structure in $u_i^{(\cdot)} \in \mathbb{R}^T$ from which changepoints can be obtained. The neighbourhood structure $\boldsymbol{\theta}_i^{(b)}$ for blocks $b = 1, \ldots, B = K + 1$ is obtained from the values of $\hat{\boldsymbol{u}}_i^{(t)}$ between changepoints. While these methods do not necessarily produce positive-semi-definite precision matrices, they have

potential to be adapted for modelling relationships between mixed types of data. For example, one may build models to encode dependency between count, and continuous data types. In the stationary setting, J. Lafferty et al. (2012) explored non-parametric graphical model estimation, and more recently Lee et al. (2015) studied models with mixed types of variable. The paper of Haslbeck et al. (2016) extends some of these methods to the dynamic case utilising a moving window approach. Extension of the regularised smoothing methods to the mixed variable case may form a direction for further work.

3.1.5 Summary of approaches

A summary of dynamic graph-estimation methods is presented in Table 1. Apart from GFGL most approaches penalise edge dynamics via an ℓ_1 norm. One notable exception to this can be found in the Varying-Coefficient Varying-Structure (VCVS) model of Kolar et al. (2012) who propose to select changepoints with an ℓ_2 type norm over the differences. The motivation in that work is similar to that presented here, except the authors formulate the graph-selection problem differently, utilising a node-wise regularised regression estimator rather than the multivariate Gaussian likelihood. While node-wise estimation can recover the conditional dependency graph, it does not in general result in a positive-definite precision matrix. The advantage of the GFGL/IFGL estimators, is that the output can directly be used to define a probability distribution via a GGM.

The aim in the following sections is to compare the effect of grouped (GFGL) and independent smoothing methods such as FMGL (S. Yang et al. 2015), TESLA (Ahmed et al. 2009), SINGLE/IFGL (Gibberd and Nelson 2014a; Monti et al. 2014) and JGL (Danaher et al. 2013). Rather than focusing on the smoothly evolving graph through the kernel covariance estimator $\hat{\boldsymbol{S}}^{(t)}$, we instead study the difference between the smoothing regularizer for IFGL and GFGL. Throughout the rest of this chapter a purely piecewise constant graph model is assumed and the empirical covariance is simply estimated with the data at time t according to $\hat{\boldsymbol{S}}^{(t)} = \boldsymbol{x}^{(t)}(\boldsymbol{x}^{(t)})^\top/2$. Effectively, this uses a Dirac-delta kernel for the covariance estimate, in (3.1.3) we set $h_t^{(s)} = \delta(s-t)$.

Table 1 – Overview of likelihood and smoothing approaches for dynamic graphical modeling. Shrinkage via an ℓ_1 term is common to all methods (in VCVS this is applied at the node-wise level) above when used for edge selection. This is usually applied to off-diagonal entries in the graph/precision matrix such that $R_{\text{Shrink}} = \lambda_1 \sum_{t=1}^{T} \|\boldsymbol{U}_{-ii}^{(t)}\|_1$. *Note: these methods are not specifically designed for time-series data but for building fused models over different $k = 1, \ldots, K$ classes/experiments each with n_k data-points.

Name	References	Likelihood L	Graph Smoothing R_{Smooth}
Dynamic Graphical Lasso	Zhou et al. (2010)	$\{\log(\det(\boldsymbol{U}^{(t)})) - \text{tr}(\hat{\boldsymbol{S}}^{(t)}\boldsymbol{U}^{(t)})\}_{t=1}^{T}$	via kernel (see Eq. 3.1.3)
Temporally smoothed ℓ_1 logistic regression (TESLA)	Ahmed et al. (2009)	$\sum_{t=1}^{T}\left[\log\left(1+\exp(\boldsymbol{x}_{-i}^{(t)}u_{\cdot,i}^{(t)})\right) - \boldsymbol{x}_{\setminus i}^{(t)}u_{\cdot,i}^{(t)}x_i^{(t)}\right]$	$\lambda_2 \sum_{t=2}^{T} \|\boldsymbol{u}_{\setminus i}^{(t)} - \boldsymbol{u}_{\setminus i}^{(t-1)}\|_1$
Joint Graphical Lasso (JGL)*	Danaher et al. (2013)	$\sum_{k=1}^{K}\left[n_k\left(\log(\det(\boldsymbol{U}^{(k)})) - \text{tr}(\hat{\boldsymbol{S}}^{(k)}\boldsymbol{U}^{(k)})\right)\right]$	$\lambda_2 \sum_{k<k'} \|\boldsymbol{U}^{(k)} - \boldsymbol{U}^{(k')}\|_1$
Fused Multiple Graphical Lasso (FMGL)*	S. Yang et al. (2015)	$\sum_{k=1}^{K}\left[n_k\left(\log(\det(\boldsymbol{U}^{(k)})) - \text{tr}(\hat{\boldsymbol{S}}^{(k)}\boldsymbol{U}^{(k)})\right)\right]$	$\lambda_2 \sum_{k=1}^{K} \|\boldsymbol{U}^{(k)} - \boldsymbol{U}^{(k-1)}\|_1$
SINGLE /IFGL	Gibberd and Nelson (2014a) and Monti et al. (2014)	$\sum_{t=1}^{T}\left[\log(\det(\boldsymbol{U}^{(t)})) - \text{tr}(\hat{\boldsymbol{S}}^{(t)}\boldsymbol{U}^{(t)})\right]$	$\lambda_2 \sum_{t=2}^{T} \|\boldsymbol{U}_{\setminus ii}^{(t)} - \boldsymbol{U}_{\setminus ii}^{(t-1)}\|_1$
VCVS Model	Kolar et al. (2012)	For each node $j = 1, \ldots, p$ $\sum_{t=1}^{T}(x_{t,i} - \sum_{j\neq i} x_{t,j}u_{j,t})^2$	$\lambda_2 \sum_{t=1}^{T} \|\boldsymbol{u}_{\cdot,t} - \boldsymbol{u}_{\cdot,t-1}\|_2$
GFGL	Gibberd and Nelson (2017)	$\sum_{t=1}^{T}\left[\log(\det(\boldsymbol{U}^{(t)})) - \text{tr}(\hat{\boldsymbol{S}}^{(t)}\boldsymbol{U}^{(t)})\right]$	$\lambda_2 \sum_{t=2}^{T} \|\boldsymbol{U}_{\setminus ii}^{(t)} - \boldsymbol{U}_{\setminus ii}^{(t-1)}\|_F$

It is of course possible, and potentially preferable, to combine kernel and regularised smoothing approaches. In the context of dynamic graphs, such an approach is considered in Monti et al. (2014), such a combination of kernel and regularised smoothing is also examined in Chapter 6. In practice, whether a kernel smoother is appropriate will depend on the application. However, in

this chapter the focus is on understanding the effect of the grouped vs independent smoothing assumptions encoded within the GFGL/IFGL regularisers.

In the next section, an ADMM approach to solving the convex optimisation problems associated with GFGL/IFGL is developed. This builds on the basic method discussed in Section 2.2.2 and enables us to practically estimate dynamic graphical models for reasonable problem sizes; $p \approx 10 - 100$, $T \approx 1000$. Following this, the methods are compared empirically on both synthetic experiments and real-world data.

3.2 Algorithms for GFGL/IFGL

Since the penalty function of IFGL approaches solely comprises ℓ_1 terms it is linearly separable. This permits block-coordinate descent approaches to optimisation whereby the precision matrix rows and columns are sequentially updated. For example, such methods are utilised in Friedman, Hastie, and R. Tibshirani (2008) and S. Yang et al. (2015). Unfortunately, the GFGL objective (3.1.7) does not have the same linear separability structure. This is due to the norm $\|\boldsymbol{U}^{(t)} - \boldsymbol{U}^{(t-1)}\|_F := (\sum_{i,j}(U_{i,j}^{(t)} - U_{i,j}^{(t-1)})^2)^{1/2}$ acting across the whole, or at least multiple rows/columns of the precision matrix. This lack of linear separability across the precision matrices precludes a block-coordinate descent strategy (Tseng et al. 2009). Instead, one can make use of the separability of the group norm (with respect to time) and utilise an ADMM algorithm.

It is worth noting that there are a variety of ways one can break down (3.1.7) as an ADMM problem. In this thesis two different forms of algorithm are presented: the first *(ADMM-D)* as developed in Gibberd and Nelson (2015b) uses a single set of auxiliary variables in combination with a proximal splitting mechanism, the second *(ADMM+),* utilises additional auxiliary variables, sometimes known as a multi-block ADMM construction. This simplifies the proximal updates and enables more efficient computation. For readability, and considering the experiments presented later in this chapter use ADMM-D, only this algorithm is presented in the main text. The ADMM+ scheme is detailed in Appendix B.3. Work is ongoing to assess the performance difference between ADMM-D and ADMM+. Reporting on this in more detail is left as future work.

3.2.1 ADMM + Dykstra Splitting (ADMM-D)

We here consider splitting the objective (3.1.7) into two separate but related problems. Minimizing $\mathcal{L}(\{\boldsymbol{U}^{(t)}\})$ in Eq. 3.1.1 is equivalent to solving the constrained problem:

$$\begin{aligned}\hat{\boldsymbol{\Theta}} \;:=\; & \underset{\{\boldsymbol{U}^{(t)},\boldsymbol{V}^{(t)}\}_{t=1}^{T}}{\arg\min} \Bigg[\sum_{t=1}^{T}\big(-\operatorname{logdet}(\boldsymbol{U}^{(t)})+\operatorname{tr}(\boldsymbol{S}^{(t)}\boldsymbol{U}^{(t)})\big)\dots \\ & \qquad \dots+\lambda_1\sum_{t=1}^{T}\|\boldsymbol{V}^{(t)}_{-ii}\|_1+\lambda_2\sum_{t=2}^{T}\|\boldsymbol{V}^{(t)}_{-ii}-\boldsymbol{V}^{(t-1)}_{-ii}\|_F\Bigg] \\ & \text{such that : } \boldsymbol{U}^{(t)}-\boldsymbol{V}^{(t)}=\boldsymbol{0}\,, \end{aligned} \tag{3.2.1}$$

where $\{\boldsymbol{U}^{(t)}\}$ and the auxiliary variables $\{\boldsymbol{V}^{(t)}\}$ are also constrained to be positive-semi-definite. An augmented Lagrangian for GFGL (in the rescaled form) is given as

$$\begin{aligned}\mathcal{L}(\{\boldsymbol{U}^{(t)}\},\{\boldsymbol{V}^{(t)}\},\{\boldsymbol{P}^{(t)}_V\}) : \;=\; & \sum_{t=1}^{T}\big(-\operatorname{logdet}(\boldsymbol{U}^{(t)})+\operatorname{tr}(\boldsymbol{S}^{(t)}\boldsymbol{U}^{(t)})\big)\dots \\ & \dots+\lambda_1\sum_{t=1}^{T}\|\boldsymbol{V}^{(t)}_{-ii}\|_1+\lambda_2\sum_{t=2}^{T}\|\boldsymbol{V}^{(t)}_{-ii}-\boldsymbol{V}^{(t-1)}_{-ii}\|_F\dots \\ & \dots+\frac{\gamma}{2}\sum_{t=1}^{T}\big(\|\boldsymbol{U}^{(t)}-\boldsymbol{V}^{(t)}+\boldsymbol{P}^{(t)}_V\|_F^2-\|\boldsymbol{P}^{(t)}_V\|_F^2\big)\,, \end{aligned}$$

where $\boldsymbol{P}^{(t)}_V$ is a rescaled dual variable. *Note: the construction above follows in a similar manner to that of Sec. 2.2.2 Eq. 2.2.9.* Let the solution at the nth iteration as $\{\boldsymbol{U}^{(t)}_n\} = \{\boldsymbol{U}^{(1)}_n,\dots\boldsymbol{U}^{(T)}_n\}$, the estimates are then updated according to to the three steps below; more detail on each step is provided in the sections that follow.

Likelihood Update (Sec. 3.2.2) :

$$\begin{aligned}\boldsymbol{U}^{(t)}_n \;=\; & \underset{\boldsymbol{U}^{(t)}}{\arg\min}\Bigg[-\operatorname{logdet}(\boldsymbol{U}^{(t)})+\operatorname{tr}(\hat{\boldsymbol{S}}^{(t)}\boldsymbol{U}^{(t)})\dots \\ & \qquad \dots+\frac{\gamma}{2}\|\boldsymbol{U}^{(t)}-\boldsymbol{V}^{(t)}_{n-1}+\boldsymbol{P}^{(t)}_{V;n-1}\|_F^2\Bigg]\,, \end{aligned} \tag{3.2.2}$$

Constraint Update (Sec. 3.2.3):

$$\{\boldsymbol{V}_n^{(t)}\} = \arg\min_{\{\boldsymbol{V}^{(t)}\}} \Bigg[\frac{\gamma}{2} \sum_{t=1}^{T} \|\boldsymbol{U}_n^{(t)} - \boldsymbol{V}^{(t)} + \boldsymbol{P}_{V;n-1}^{(t)}\|_F^2 + \lambda_1 \sum \|\boldsymbol{V}_{-ii}^{(t)}\|_1 \; \ldots$$

$$\ldots + \lambda_2 \sum_{t=2}^{T} \|\boldsymbol{V}_{-ii}^{(t)} - \boldsymbol{V}_{-ii}^{(t-1)}\|_F \Bigg] , \tag{3.2.3}$$

Dual Update (Sec. 3.2.4):

$$\boldsymbol{P}_{V;n}^{(t)} = \boldsymbol{P}_{V;n-1}^{(t)} + \big(\boldsymbol{U}_n^{(t)} - \boldsymbol{V}_n^{(t)}\big) . \tag{3.2.4}$$

3.2.2 Likelihood updates: An Eigen-decomposition

We can solve the update for $\boldsymbol{U}_n^{(t)}$ through an eigen-decomposition of terms in the covariance, auxiliary and dual variables (X. Yuan 2011). If we differentiate the objective (3.2.2) and set the result equal to zero we find

$$(\boldsymbol{U}^{(t)})^{-1} - \gamma \boldsymbol{U}^t = \hat{\boldsymbol{S}}^t - \gamma(\boldsymbol{V}_{(n-1)}^t - \boldsymbol{P}_{V(n-1)}^t) . \tag{3.2.5}$$

Noting that $\boldsymbol{U}^t$ and $\boldsymbol{S}^t - \gamma(\boldsymbol{V}_{n-1}^{(t)} - \boldsymbol{P}_{V;n-1}^{(t)})$ share the same eigenvectors, one can can update the eigenvalues of $\boldsymbol{U}^{(t)}$ based on the auxiliary variables. For each eigenvalue $\{u_h\}_{h=1}^P = \text{eigval}(\boldsymbol{U}^{(t)})$, and $\{s_h\}_{h=1}^P = \text{eigval}\big(\hat{\boldsymbol{S}}^{(t)} - \gamma(\boldsymbol{V}_{n-1}^{(t)} - \boldsymbol{P}_{V;n-1}^{(t)})\big)$, we can construct the quadratic equation $u_h^{-1} - \gamma u_h = s_h$. To obtain some level of interpretation, consider that the right hand side of (3.2.5) contains evidence from the data-set via $\hat{\boldsymbol{S}}^t$, but also takes into account the effect our priors encoded within $\boldsymbol{V}_{n-1}^{(t)}$. Solving the quadratic and updating the eigenvalues then enables us to gradually incorporate appropriate prior knowledge into our precision matrix estimates. Upon solving for u_h given s_h we find

$$u_h = \frac{1}{2\gamma}\big(- s_h + \sqrt{s_h^2 + 4\gamma}\big) .$$

The full precision matrix $\boldsymbol{U}^{(t)}$ can now be found through the eigen-decomposition:

$$\boldsymbol{U}^{(t)} = \begin{pmatrix} \boldsymbol{e}_1 \\ \\ \boldsymbol{e}_p \end{pmatrix} \begin{pmatrix} u_1 & & \\ & \ddots & \\ & & u_p \end{pmatrix} \begin{pmatrix} \boldsymbol{e}_1 \\ \\ \boldsymbol{e}_p \end{pmatrix}^\top .$$

where $\{\boldsymbol{e}_h\} = \text{eigenvec}(\hat{\boldsymbol{S}}^{(t)} - \gamma(\boldsymbol{Z}_{n-1}^{(t)} - \boldsymbol{U}_{n-1}^{(t)}))$. By choosing the positive solution for the quadratic, we ensure that $\boldsymbol{U}_{(n)}^{t}$ is positive-definite and thus produces a valid estimator for the precision matrix. Furthermore, since (3.2.2) refers to an estimation at each time-point separately, we can solve for each $\boldsymbol{U}_n^{(t)}$ independently for $t = 1, \ldots, T$ to yield the set $\{\boldsymbol{U}_n^{(t)}\}_{t=1}^{T}$. Indeed, this update can be computed in parallel, as appropriate.

3.2.3 Auxiliary Updates: The Group-Fused Signal Approximator

The main difference between the GFGL estimator and previous approaches is in the use of a grouped constraint. This becomes a significant challenge when updating the auxiliary variables $\{\boldsymbol{V}^{(t)}\}$ in Eq. 3.2.3. Unlike the calculation of $\{\boldsymbol{U}_n^{(t)}\}$ we cannot separate the optimization over each time-step. Instead, we must solve for the whole set of matrices $\{\boldsymbol{V}^{(t)}\}$ jointly. Furthermore, due to the Frobenius norm being used in GFGL we cannot separate the optimization across individual edges. For example, in contrast to independent penalization strategies like IFGL (Danaher et al. 2013; Monti et al. 2014) it is not possible to solve GFGL for $\{U_{ij}^{(t)}\}$ independently of $\{U_{i'j'}^{(t)}\}$, where $(i, j) \neq (i', j')$.

Since each $\boldsymbol{V}^{(t)}$ is symmetric about the diagonal we can reduce the number of elements by simply taking the elements above the diagonal $\boldsymbol{v}^{(t)} = (V_{i,j}^{(t)} | \, j > i, \;\; i = 1, \ldots, p)^{\top}$. We now construct a data-matrix of these elements such that $\boldsymbol{V} = (\boldsymbol{v}^{(1)}, \ldots, \boldsymbol{v}^{(T)})^{\top} \in \mathbb{R}^{T \times p(p-1)/2}$; row t of the matrix thus corresponds to values at time-step t. We perform similar transformations for $\boldsymbol{U}^{(t)} \mapsto \boldsymbol{U}$ and $\boldsymbol{P}_V^{(t)} \mapsto \boldsymbol{P}_V$, while also adjusting the regularisation parameters, setting[3] $\bar{\lambda}_1 = \lambda_1/\rho$, and $\bar{\lambda}_2 = \lambda_2/\rho$.

***Definition* 3.1.** *Group-Fused Lasso Signal Approximator*

Let $\boldsymbol{\Gamma}_V := \boldsymbol{U} + \boldsymbol{P}_V$*, then re-writing the objective (3.2.3) with these transformations yields the* group-fused lasso signal approximator (GFLSA). *Formally this is viewed as a proximity operator of the form* $\hat{\boldsymbol{V}}(\boldsymbol{\Gamma}; \bar{\lambda_1}, \bar{\lambda_2}) = \arg\min_{\boldsymbol{V}} G(\boldsymbol{V}; \bar{\lambda_1}, \bar{\lambda_2})$*, with cost*

[3]Note that, since we have essentially split the data in half (due to symmetry), we may wish to adjust the lambdas to be consistent with the original problem specification in Eq. 3.1.7.

$$G(\boldsymbol{V};\bar{\lambda_1},\bar{\lambda_2}) := \underbrace{\frac{1}{2}\|\boldsymbol{\Gamma}_V - \boldsymbol{V}\|_F^2}_{L(\boldsymbol{V})} + \underbrace{\bar{\lambda_1}\|\boldsymbol{V}\|_1}_{R_1(\boldsymbol{V})} + \underbrace{\bar{\lambda_2}\|\boldsymbol{D}\boldsymbol{V}\|_{2,1}}_{R_2(\boldsymbol{V})}, \tag{3.2.6}$$

where the $\ell_{2,1}$ norm is defined as the sum of ℓ_2 norms over the rows, i.e. $\|\boldsymbol{U}\|_{2,1} := \sum_t \|U_{t,\cdot}\|_2$ and $\boldsymbol{D} \in \mathbb{R}^{(T-1)\times T}$ is a backwards differencing matrix of the form $D_{i,i} = -1$, $D_{i,i+1} = 1$ for $i = 1,\ldots,T-1$ and the rest of the entries are zero.

While the GFLSA has a similar appearance to the previously proposed *fused lasso signal approximator (FLSA)* studied in H. Liu et al. (2010), it crucially incorporates a group $\ell_{2,1}$ norm rather than the ℓ_1 norm for performing smoothing. To understand the effect of the update on the auxiliary variables, consider that the GFLSA problem can also be thought of as a proximity operator of the form $\hat{\boldsymbol{V}}(\boldsymbol{\Gamma}_V;\bar{\lambda_1},\bar{\lambda_2}) \equiv \text{prox}_{R_1+R_2}(\boldsymbol{\Gamma})$. If $R_1(\cdot)$ and $R_2(\cdot)$ were indicator functions of two closed convex sets C and D respectively, then $\hat{\boldsymbol{V}}(\boldsymbol{\Gamma}_V;\bar{\lambda_1},\bar{\lambda_2})$ would find the best approximation to $\boldsymbol{\Gamma}_V$ restricted to the set $C \cap D$.

For any unconstrained optimal point $\boldsymbol{V}^* = \arg\min_{\boldsymbol{V}} L(\boldsymbol{V})$ there exists a set of parameters $(\lambda_1,\lambda_2) \in [0,\infty)$ which will act to move the optimal point of the regularized case $\boldsymbol{V}_R^* = \arg\min_{\boldsymbol{V}} G(\boldsymbol{V};\lambda_1,\lambda_2)$ such that $\boldsymbol{V}^* \neq \boldsymbol{V}_R^*$ where

$$\boldsymbol{V}_R^* := \arg\min_{\boldsymbol{V}} L(\boldsymbol{V}), \quad \text{subject to} \quad \|\boldsymbol{V}\|_1 \leq l_1 \quad \text{and} \quad \sum_{t=2}^{T} \|[\boldsymbol{D}\boldsymbol{V}]_{t,\cdot}\|_2 \leq l_2.$$

For a given likelihood term, we can obtain an l_1 sparse and l_2 smooth solution by solving a penalized problem instead of the explicitly constrained version above. Such a penalized form is found in Eq. 3.2.6 and, while $R_1(\lambda_1,\boldsymbol{V})$ and $R_2(\lambda_2,\boldsymbol{V})$ are not explicitly indicator functions (i.e. they do not take values ∞ outside some feasible region), there does exist a mapping between the values of the parameters $\lambda_1 \geq 0\,, \lambda_2 \geq 0$ to the corresponding l_1, l_2 sparsity and smoothness constraints. The intuition is similar to that of the lasso (Sec. 2.1.3), for a given constraint level l_1 and function $L(\boldsymbol{V})$, the size of the feasible set given by $C_{\lambda_1} = \{\boldsymbol{V} \mid \lambda_1\|\boldsymbol{V}\|_1 \leq l_1\}$ reduces as λ_1 increases. Thus sparsity is a monotonically non-decreasing function of λ_1. The same argument can be constructed for smoothing and the constraint set $D_{\lambda_2} = \{\boldsymbol{V} \mid \lambda_2 \sum_t \|(\boldsymbol{D}\boldsymbol{V})_{t,\cdot}\|_2 \leq l_2\}$.

Unlike FLSA which penalizes the columns of $\boldsymbol{V}$ independently, we find $\text{prox}_{R_1+R_2}(\boldsymbol{\Gamma}) \neq \text{prox}_{R_2}\big(\text{prox}_{R_1}(\boldsymbol{\Gamma})\big)$, thus in the case of GFGL one cannot apply the two-stage smooth-then-sparsify theorem of Friedman, Hastie, Hoefling, et al. (2007) and J. Liu et al. (2010) (specifically Theorem 1 of J. Liu et al. (2010)). An alternative approach is to follow the work of Alaíz et al. (2013) and adopt an iterative projection approach whereas Dykstra's method (Combettes et al. 2011) is used to find a feasible solution which simultaneously satisfies both the group fused $\ell_{2,1}$ and lasso ℓ_1 constraints. The *proximal Dykstra* algorithm, as outlined in Algorithm 1, provides a way to calculate a point $\boldsymbol{V}_r^* \in C_{\lambda_1} \cap D_{\lambda_2}$ that is, in the sense of the ℓ_2 distance, *close* or *proximal* to the unconstrained solution for $\arg\min_{\boldsymbol{V}} L(\boldsymbol{V}) = \boldsymbol{\Gamma}$. By iterating between the feasibility of a solution in C_{λ_1} and D_{λ_2} a solution can be found which is both suitably smooth and sparse. For more details on the proximal Dykstra algiorithm the reader is directed to Combettes et al. (2011).

Result: $\boldsymbol{V}_N = \text{prox}_{R_1+R_2}(\boldsymbol{\Gamma})$
$\boldsymbol{V}_0 = \boldsymbol{\Gamma}\,, \boldsymbol{U}_n = \boldsymbol{0}\,,\ \boldsymbol{Q}_n = \boldsymbol{0}$
while *not converged,* $n = 0, 1, \ldots$ **do**
  $\boldsymbol{Y}_n = \text{prox}_{R_2}(\boldsymbol{V}_n + \boldsymbol{U}_n)$
  $\boldsymbol{U}_{n+1} = \boldsymbol{V}_n + \boldsymbol{U}_n - \boldsymbol{Y}_n$
  $\boldsymbol{V}_{n+1} = \text{prox}_{R_1}(\boldsymbol{Y}_n + \boldsymbol{Q}_n)$
  $\boldsymbol{Q}_{n+1} = \boldsymbol{Y}_n + \boldsymbol{Q}_n - \boldsymbol{V}_{n+1}$
end

Algorithm 1: Dykstras iterative projection algorithm, the sequences of $\boldsymbol{V}_n$ converge to a feasible point.

Given that iterative projection can be used to find a feasible point, the challenge is now to compute the separate proximity operators for $R_1()$ and $R_2()$. For R_1 the proximity operator is simply the soft-thresholding function as derived in Sec. 2.1.5 (Eq. 2.1.15). Computing the group-fused term $\text{prox}_{R2}(\boldsymbol{\Gamma}; \lambda_2)$ is more involved and there is no obvious closed-form solution. Instead, we solve this through a block-coordinate descent approach similar to that considered by Bleakley et al. (2011) and M. Yuan et al. (2006). Specifically, Bleakley et al. (2011) show that it is possible to reformulate the GFSA as a group-lasso problem which can be solved via standard strategies.

***Proposition* 3.1.** *Group-Fused Signal Approximator*

The solution to the group-fused signal approximator

$$\text{prox}_{\text{GFSA}}(\boldsymbol{\Gamma}; \lambda_2) = \arg\min_{\boldsymbol{V} \in \mathbb{R}^{T \times p}} \frac{1}{2}\|\boldsymbol{V} - \boldsymbol{\Gamma}\|_F^2 + \lambda_2 \|\boldsymbol{D}\boldsymbol{V}\|_{2,1}, \tag{3.2.7}$$

may be written as a sum of estimates, such that

$$[\text{prox}_{\text{GFSA}}(\boldsymbol{\Gamma}; \lambda_2)]_{t,\cdot} = \begin{cases} \hat{\boldsymbol{\omega}} + \sum_{s=1}^{t-1} \hat{\Omega}_{s,\cdot} & t > 1 \\ \hat{\boldsymbol{\omega}} & t = 1 \end{cases}.$$

Let us define a linear partial sum operator $\boldsymbol{R} \in \mathbb{R}^{p \times T-1}$ *as* $R_{i,j} = 1$ *for* $i > j$ *and* 0 *otherwise. The estimated constant is given as* $\hat{\boldsymbol{\omega}} = \mathbf{1}_{1,T}(\boldsymbol{\Gamma} - \boldsymbol{V}\hat{\boldsymbol{\Omega}})/T$*, where the matrix of jumps* $\hat{\boldsymbol{\Omega}}$ *can then be obtained from solving the group-lasso problem with*

$$\hat{\boldsymbol{\Omega}} := \arg\min_{\boldsymbol{\Omega} \in \mathbb{R}^{(T-1) \times p}} \frac{1}{2}\|\bar{\boldsymbol{\Gamma}} - \bar{\boldsymbol{R}}\boldsymbol{\Omega}\|_F^2 + \lambda_2 \|\boldsymbol{\Omega}\|_{2,1} \ ,$$

where $\bar{\boldsymbol{X}}$ *is the column-centred version of* $\boldsymbol{X}$*.*

Proof. Simple variable substitution in (3.2.7). See Appendix B.1 for details.

Applying the above result in conjunction with a block-coordinate descent strategy allows us to compute the group-fused proximity operator $\text{prox}_{R_2}(\boldsymbol{\Gamma}; \lambda_2)$. The overall subproblem (3.2.6) can now be solved through iteratively applying the proximity operators according to Dykstra's algorithm.

3.2.4 Dual update and convergence

The final step in the ADMM-based method is to update the dual variable via Eq. 3.2.4. Convergence properties of general ADMM algorithms are analyzed in Glowinski et al. (1989). Importantly, the sequence of solutions $\{\boldsymbol{U}_n\}_{n \in \mathbb{N}}$ can be shown to converge (Eckstein et al. 1992) to the solution of the problem $\arg\min_{\boldsymbol{U}} \ f(\boldsymbol{U}) + g(\boldsymbol{L}\boldsymbol{U})$, under conditions that $\boldsymbol{L}^\top \boldsymbol{L}$ is invertible and the intersection between relative interiors of domains is non-empty $(\text{ri dom } g) \cap \text{ri } \boldsymbol{L}(\text{dom } f) \neq \emptyset$. In the GFGL and IFGL problems one simply sets $\boldsymbol{L} = \boldsymbol{I}$ in order to restrict $\boldsymbol{U} = \boldsymbol{V}$. Clearly in this case $\boldsymbol{I}^\top \boldsymbol{I}$ is invertible and $\text{dom } g = \boldsymbol{I}(\text{dom } f)$; thus the relative interiors intersect. Convergence of the Dykstra's algorithm is discussed in Combettes et al. (2011) and Bauschke

et al. (2008), it generally requires a minimal feasibility condition such that $\text{dom } R_1 \cap \text{dom } R_2 \neq \emptyset$ and a valid solution exists.

Whilst ADMM is guaranteed to converge to an optimal solution, in practice it converges relatively fast to a useful solution, but very slowly if high accuracy is required. Following the approach of Boyd, Parikh, et al. (2011) two convergence criteria are considered. The first, known as *primal feasibility,* relates to the optimality condition $\boldsymbol{U}^* - \boldsymbol{V}^* = \boldsymbol{0}$ where $\boldsymbol{U}^*$ is the optimal value of $\boldsymbol{U}$ and is measured according to:

$$r_{\text{prime}} = \sum_{t=1}^{T} \|\boldsymbol{U}_{(n)}^t - \boldsymbol{V}_{(n)}^t\|_F^2 \ ,$$

A second, *dual feasibility* criteria defined as:

$$r_{\text{dual}} = \sum_{t=1}^{T} \|\boldsymbol{V}_{(n)}^t - \boldsymbol{V}_{(n-1)}^t\|_F^2 \ ,$$

tracks the requirement that $\boldsymbol{0} \in \nabla F(\boldsymbol{U}^*) + \boldsymbol{P}_V^*$. The rate at which the algorithm converges is somewhat tunable through the γ parameter, however it is not clear how to find an optimal γ for a given problem. In practice, a value of order $\gamma = 10$ provides reasonably fast convergence with tolerances of order $r_{\text{prime}} < \epsilon_{\text{prime}} = 10^{-3}$ and $r_{\text{dual}} < \epsilon_{\text{dual}} = 10^{-3}$.

3.2.5 A solver for the Independent Fused Graphical Lasso

Whilst the ADMM-D algorithm was developed to solve the GFGL estimator, they can easily be adapted for the IFGL estimator by modifying the auxiliary updates corresponding to the non-smooth constraint projection. In place of (3.2.6), one may consider a fused lasso problem

$$G(\boldsymbol{V}; \bar{\lambda_1}, \bar{\lambda_2}) = \underbrace{\frac{1}{2}\|\boldsymbol{\Gamma} - \boldsymbol{V}\|_F^2}_{L(\boldsymbol{V})} + \underbrace{\bar{\lambda}_1\|\boldsymbol{V}\|_1}_{R_1(\boldsymbol{V})} + \underbrace{\bar{\lambda}_2\|\boldsymbol{D}\boldsymbol{V}\|_1}_{R_2(\boldsymbol{V})} \ , \tag{3.2.8}$$

where $\boldsymbol{\Gamma} = \boldsymbol{U} + \boldsymbol{P}_V$ and we replace the $\|\cdot\|_{2,1}$ norm of GFGL with a simple ℓ_1 penalty of IFGL. Since the ℓ_1 norm is linearly separable, i.e. $\|\boldsymbol{U}\|_1 = \sum_{ij} |U_{ij}|$, the objective can now be viewed as a series of $p(p-1)/2$ separate FLSA problems. This can be solved efficiently with gradient descent. In the IFGL case there is no need to apply the iterative Dykstra projection as one can

Data: $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(T)}$
Input: $\lambda_1,\ \lambda_2,\ \gamma,\ \epsilon_{\text{dual}},\ \epsilon_{\text{prime}}$
Result: $\{\hat{\boldsymbol{\Theta}}^{(1)}, \ldots, \hat{\boldsymbol{\Theta}}^{(T)}\}$
Calculate covariance matrix: $\hat{\boldsymbol{S}}^t = \boldsymbol{x}^{(t)}(\boldsymbol{x}^{(t)})^\top/2$ for $t = 1, \ldots, T$
Initialize: $\boldsymbol{V}^{(t)}_{(0)} = \boldsymbol{U}^{(t)}_{(0)} = \boldsymbol{P}^{(t)}_{V;(0)} = \boldsymbol{0}$
while *not converged* ($r_{\text{prime}} \geq \epsilon_{\text{prime}},\ r_{\text{dual}} \geq \epsilon_{\text{dual}}$), $n = 0, 1, \ldots$ **do**
  for *t=1,...,T* **do**
    Eigen-decomposition:
    $\{s_h,\ \boldsymbol{e}_h\}_{h=1}^P = \text{eigen}\big(\hat{\boldsymbol{S}}^{(t)} - \gamma(\boldsymbol{V}^{(t)}_{(n-1)} - \boldsymbol{P}^{(t)}_{V;(n-1)})\big)$
    $x_h = \big(-s_h + \sqrt{s_h^2 + 4\gamma}\big)/2\gamma$
    $\boldsymbol{E} = (\boldsymbol{e}_1, \ldots . \boldsymbol{e}_P),\ \boldsymbol{Q} = \text{diag}(x_1, \ldots, x_P)$
    Apply constraints: $\boldsymbol{U}^{(t)}_n = \boldsymbol{E}\boldsymbol{Q}\boldsymbol{E}^\top$
  end
  $\boldsymbol{V}_{(n)} = \text{prox}_{R1+R2}(\boldsymbol{U}_{(n)} + \boldsymbol{P}_{V;(n-1)}\ ; \lambda_1/\gamma, \lambda_2/\gamma)$ // *GFLSA* via Dykstras method*
  $\boldsymbol{P}^{(t)}_{V;(n)} = \boldsymbol{P}^{(t)}_{V;(n-1)} + \big(\boldsymbol{U}^{(t)}_{(n)} - \boldsymbol{V}^{(t)}_{(n)}\big)$, for $t = 1, \ldots, T$
  $r_{\text{prime}} = \sum_{t=1}^T \|\boldsymbol{U}^{(t)}_{(n)} - \boldsymbol{V}^{(t)}_{(n)}\|_F^2,\ r_{\text{dual}} = \sum_{t=1}^T \|\boldsymbol{V}^{(t)}_{(n)} - \boldsymbol{V}^{(t)}_{(n-1)}\|_F^2$
end
Return: $\{\hat{\boldsymbol{\Theta}}^{(t)} = \boldsymbol{U}^{(t)}, \ldots\}$

Algorithm 2: Outline of ADMM algorithm for GFGL. Note: to solve IFGL we simply replace the update (*) with $\boldsymbol{V}_{(n)} = \text{prox}_{R1+R3}(\boldsymbol{U}_{(n)} + \boldsymbol{P}_{V;(n-1)}\ ; \lambda_1/\gamma, \lambda_2/\gamma)$ which can be computed through the sub-gradient finding algorithm as proposed in H. Liu et al. (2010).

show the proximity operator can be calculated according to $\text{prox}_{R_1+R_2}(\boldsymbol{\Gamma}) \equiv \arg\min_{\boldsymbol{V}} G(\boldsymbol{V}; \bar{\lambda_1}, \bar{\lambda_2}) = \text{prox}_{R_2}\big(\text{prox}_{R_1}(\boldsymbol{\Gamma})\big)$ (H. Liu et al. 2010).

3.3 Synthetic Experiments

In this section, the IFGL and GFGL estimators are applied to simulated, piecewise stationary, multivariate time-series data. This provides an empirical comparison of their relative abilities to (i) recover the graphical structure and (ii) detect changepoints. Instead of assessing the speed of implementation, the experiments here aim to highlight the different qualitative features of graph estimation under the GFGL and IFGL penalisation schemes. An outline of the testing methodology is presented in Figure 3.3.1. In these experiments, the ADMM-D algorithm is utilised to implement the estimators, preliminary

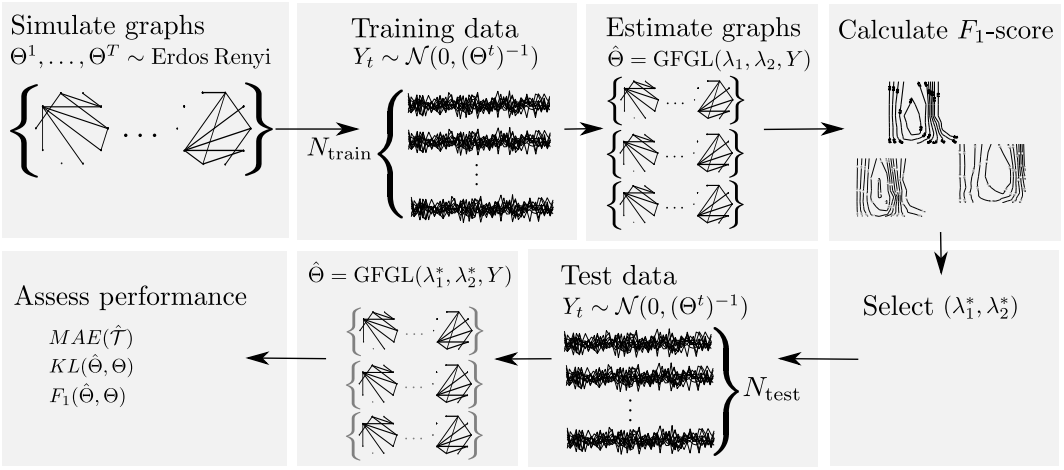


Figure 3.3.1 – Diagrammatic representation of synthetic experiments.

results, not reported here, suggest that the extended ADMM+ method may enable faster convergence in some situations. However, analysis of the ADMM+ algorithm requires further work and the focus in this section is on the empirical difference between the GFGL and IFGL estimators.

3.3.1 Data simulation

To validate the graphical recovery performance of estimators, data is simulated according to a known ground truth set of precision matrices $\{\boldsymbol{\Theta}_0^{(t)}\}_{t=1}^T$. The simulation is carried out such that, for a given number K^* of ground truth changepoints $\mathcal{T}^* = \{\tau_1, \ldots, \tau_{K^*}\}$, there are $K^* + 1$ corresponding graph structures. For each segment $k = 1, \ldots, K^* + 1$, graphical structure is simulated uniformly at random from the set of graphs with vertex size $|V_k| = P$ and $|E_k| = M_k$ edges, i.e. $G(V, E_k) \sim \text{ErdösRényi}(P, s_k)$. A draw of $G(V, E_k)$ can then be used to construct a valid GGM by equating the sparsity pattern of the adjacency matrix and precision matrix, i.e. $(i, j) \in E_k \iff \Theta_{i,j}^{(k)} \neq 0$.

Precision matrices are formed by taking a weighted identity matrix $\frac{1}{2}\boldsymbol{I} \in \mathbb{R}^{p \times p}$ and inserting off-diagonal elements according to edges E_k that are uniformly weighted in the range $[-1, -1/2] \cup [1/2, 1]$. The absolute value of these elements is then added to the appropriate diagonal entries to ensure positive semi-definiteness. To focus on the study of correlation structure between variables, the variance of the distributions are normalized such that $(\Theta_{ii}^{(t)})^{-1} = 1$ for $i = 1, \ldots, P$.

3.3.2 Hyper-parameter selection

With most statistical estimation problems there are a set of associated tuning parameters (common examples include; kernel width/shape, window sizes, etc.) which must be specified. In the GFGL and IFGL model, one can consider the regularizer terms $R_1(\lambda_1)$ and $R_2(\lambda_2)$ in Eq. (3.1.7) as incorporating prior knowledge into the model parameterization. Given this viewpoint, selection of tuning parameters (λ_1, λ_2) corresponds to specification of hyper-parameters for graph sparsity and smoothing.

The recovery performance will depend on the strength of priors employed. As such λ_1 and λ_2 must be tuned, or otherwise estimated, such that they are appropriate for a given data-set or task. In comparison to models which utilize only one regularizing term (for example, the graphical lasso of Banerjee et al. (2008)), the potential interplay between $R_1(\lambda_1)$ and $R_2(\lambda_2)$ sometimes conflates the interpretation of the different regularizers. For example, whilst λ_1 predominantly affects the sparsity of the extracted graphs, λ_2 can also have an implicit effect through smoothing.

In the synthetic data-setting, the availability of ground-truth or labeled data affords the opportunity to learn the hyper-parameters via a supervised scheme. In order to avoid repeated use of data, the simulations are split into test and training groups which share the same ground-truth structure $\{\boldsymbol{\Theta}_0^{(1)}, \ldots, \boldsymbol{\Theta}_0^{(T)}\}$, but are independently sampled. The IFGL and GFGL problems are then solved for each pair of parameters (λ_1, λ_2) over a search grid. Optimal hyper-parameters can then be selected according to a relevant measure of performance. Typically, one may consider either predictive risk (how well can the model represent the true distribution), or model recovery, i.e. estimation of the correct sparsity pattern (Zhou et al. 2010).

3.3.3 Model recovery performance

Considering the model recovery setting, the problem of selecting edges can be treated as a binary classification problem. One popular measure of performance for such problems is the F_β-score

$$F_\beta = \frac{(1+\beta^2)\mathrm{TP}}{(1+\beta^2)\mathrm{TP} + \beta^2\mathrm{FN} + \mathrm{FP}}\ , \tag{3.3.1}$$

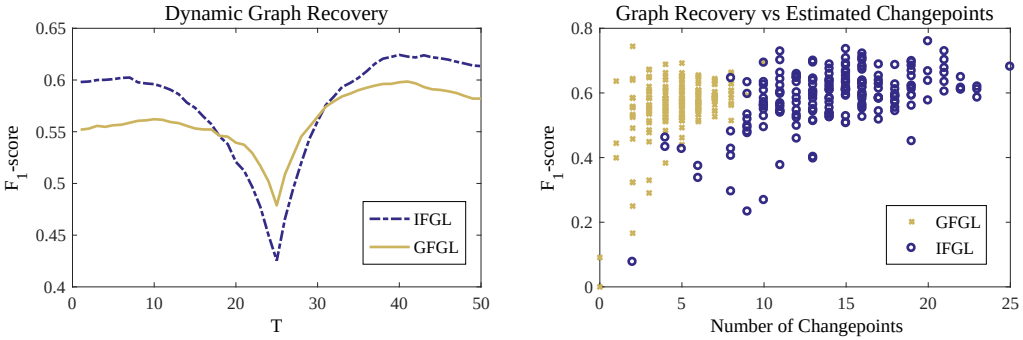


Figure 3.3.2 – Comparison of generalization performance between GFGL and IFGL, for a dataset of size $p = 10$, $T = 50$. Left: F_1-score as a function of time t, plotted lines are the averages over $N_{\text{test}} = 200$ time-series. Error-bars are omitted for presentation, with an estimated standard deviation in the F_1-scores of $\sigma_{\text{IFGL}} \approx 0.15$ and $\sigma_{\text{GFGL}} \approx 0.14$. Right: Demonstration of GFGL and IFGL graph recovery as a function of the number of estimated changepoints $|\hat{\mathcal{T}}|$.

where TP considers the number of correctly classified edges, whilst FP and FN relate to the number of *false positives* and *false negatives* (Type 1 and Type 2 errors) respectively (a score of $F_\beta = 1$ represents perfect recovery). Since dynamic network recovery is of interest, the average F_1-score is taken over each time-series to measure the effectiveness of edge selection.

For each training time-series an optimal set of parameters are chosen which maximise the F_1-score. Specifically, let us consider the set $\Gamma = \{(\lambda_1^*, \lambda_2^*)_i = \arg\max\ F_1(\lambda_1, \lambda_2)_i\}_{i=1}^{N_{\text{train}}}$, the final, optimal parameters λ_1^*, λ_2^*, are then computed as the median value over this training set. A hold-out test set of independently simulated time-series is then used to measure the generalization performance. Figure 3.3.2-left provides a typical comparison of the graph-recovery (F_1-score) performance between the IFGL and GFGL methods throughout the time-series duration. In this example it can be seen that IFGL tends to perform best at points far from the changepoint, whereas GFGL shows a benefit when estimating a graph close to the changepoint.

We note the primary difference between IFGL and GFGL is the number of edges effected at each changepoint. This is demonstrated more clearly in Figure 3.3.3. Here λ_1 is fixed and the number of edges which change at each time-point is plotted over a range of smoothing parameters λ_2. Clearly, GFGL results in a greater number of edges being effected at each changepoint. Due to

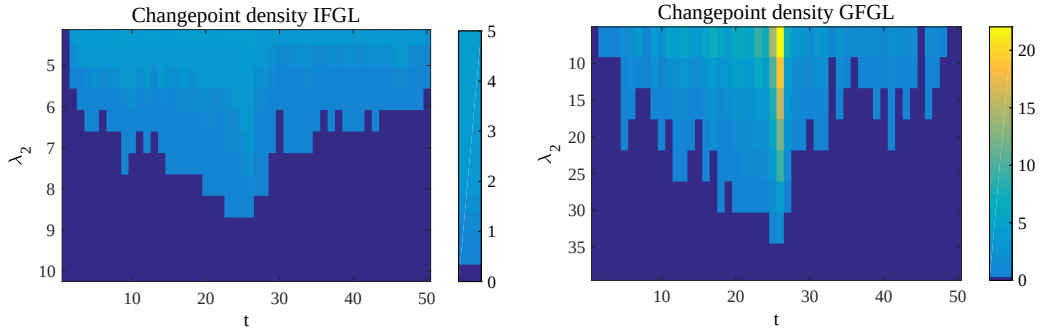


Figure 3.3.3 – Change-point density plots for IFGL and GFGL in the synthetic setting ($p = 10,\ T = 50,\ s = 10\ \lambda_1 = 0.2$), there is a simulated changepoint at $\mathcal{T} = 25$. Color represents the average number of edges (over $N = 100$ simulations) which experience a change at a given time point.

the grouped estimation of GFGL, a good graph recovery F_1-score performance is achievable with only a few changepoints (see Figure 3.3.2). In contrast, if one sets λ_2 to be large in the IFGL setting, only a few changepoints are selected - however, these represent changes in only very few edges (Figure 3.3.3-left). In this setting IFGL may perform well with regards to changepoint performance but this comes at the expense of poorer graph recovery as is evident from the F_1-scores. Where such grouped changepoint structure is present across many edges, GFGL enables one to recover changepoints without sacrificing as much graphical recovery performance.

3.3.4 Performance scaling

In this section, the recovery performance of the estimators is considered over a range of different problem sizes. In order to assess changepoint estimation performance and how this varies with scale, we may construct an error measure that monitors the average distance (in time) between estimated and true changepoints. The changepoints for a given edge (i, j) can be described by monitoring differences in the precision matrix, i.e. $\hat{\mathcal{T}}_{ij} = \left\{t \middle| |\hat{\Theta}_{ij}^{(t)} - \hat{\Theta}_{ij}^{(t-1)}| \neq 0,\ t = 2, \ldots, T\right\} =: \{\hat{\tau}_{ij}^{(k)}\}_{k=1}^{\hat{K}_{ij}}$, with $\hat{K}_{ij} = \left|\hat{\mathcal{T}}_{ij}\right|$. These are compared with the ground truth changepoints for the i, jth edge (τ_{ij}) from the changepoint set

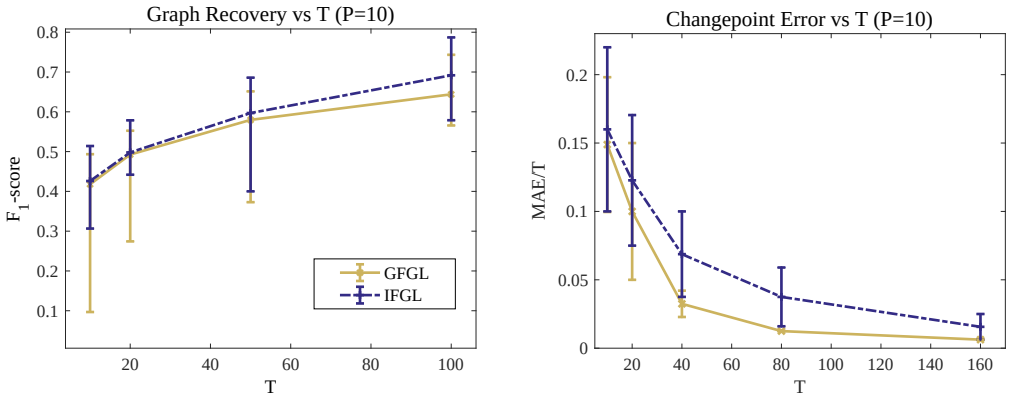


Figure 3.3.4 – Comparison of IFGL and GFGL estimator performance with increasing number of data-points T. Left: F_1-score, Right: Relative changepoint error vs time-series length. Error bars represent 67% confidence intervals as estimated from the empirical c.d.f. of $N = 200$ test examples.

$\mathcal{T}_{ij}$ via the mean absolute error measure, namely

$$\text{MAE} := \frac{1}{\hat{K}} \sum_{i,j} \sum_{k=1}^{\hat{K}_{ij}} \left| \hat{\tau}_{ij}^{(k)} - \tau_{ij} \right| ,$$

where[4] $\hat{K} = \sum_{ij} \hat{K}_{ij}$. In these experiments a single changepoint is shared across multiple edges at $\mathcal{T} = T/2$. To allow fair comparison between experiments at different time-series lengths, the same precision matrices are used on both sides of the changepoint. For example, under scaling $T \to 2T$, the number of data-points either side of the changepoint is simply doubled. When considering scaling with respect to dimension precision, matrices are simulated as in Section 3.3.1. However, the number of active edges scaled as $s = p$. Experiments were run with data-sets of size $N_{\text{train}} = 20$ and $N_{\text{test}} = 200$ and optimal lambdas were selected through F-score maximization.

Figure 3.3.6 presents the experimental results. As one may have expected, recovery performance improves as more data is made available (increasing T), but degrades as the problem task becomes more complex (increasing p). On average, IFGL performs slightly better at estimating the correct edges, whereas GFGL outperforms in the changepoint detection task. Such a result coincides

[4]One should note that $\hat{K} = |\hat{\mathcal{T}}|$ only when no changepoints occur simultaneously across multiple edges; i.e. $|\hat{\mathcal{T}}| = \hat{K} \iff |\bigcup_{ij} \hat{\mathcal{T}}_{ij}| = \sum_{ij} |\hat{\mathcal{T}}_{ij}|$.

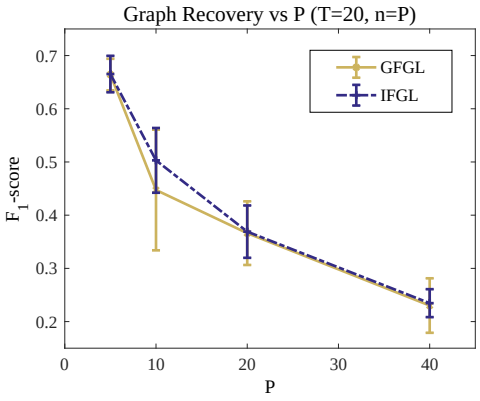


Figure 3.3.5 – F_1 score vs model dimension p.

with the performance demonstrated in Figure 3.3.2, where GFGL performs better in the vicinity of a changepoint. If grouped changepoints are present, the experiments suggest GFGL performs better in the changepoint estimation task without sacrificing much graph recovery performance.

The results here display how recovery performance scales with problem dimensionality. However, such performance will also depend on the structure of the ground-truth graph and precision matrices. As an example, in the stationary setting Ravikumar, Wainwright, Raskutti, et al. (2011) suggest that, for consistent recovery of graphs (with N data-points), one should bound the partial correlations, $[-1, -\alpha] \cup [\alpha, 1]$ such that $\alpha = \Omega(\sqrt{\log p / N})$. To enable better interpretation of experimental results, the scaling was fixed to $\alpha = 1/2$ in these examples. However, it is anticipated that changepoint and graph estimation may become more difficult as the true non-zero partial correlations $\Theta_{i,j}$ tend towards zero. A theoretical analysis of GFGL in Chapter 4 corroborates this intuition, for consistent recovery the non-zero elements should be appropriately lower bounded.

In order to investigate computational scalability, a series of experiments were performed on problems of various size (the experimental setup is the same as in Sec. 3.3.4). Results are summarised in Figure 3.3.6. In contrast with the quadratic time complexity for dynamic programming methods (Angelosante et al. 2011), it can be observed that the ADMM-D routine, as a whole, maintains roughly linear complexity with increasing T. However, when considering increases in the estimated number of changepoints $\hat{K}$, complexity

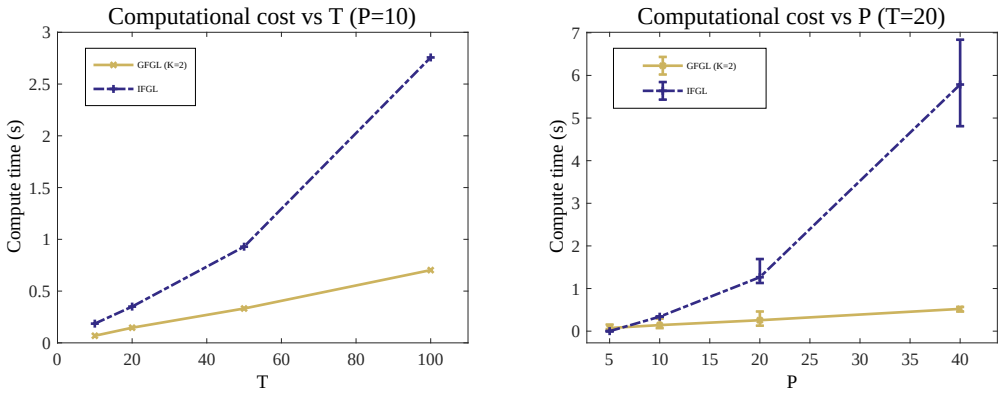


Figure 3.3.6 – Left: Compute-time vs time-series length T and Right: dimension for a fixed number of changepoints.

appears to follow the quadratic rate of *GFLseg*[5] which scales as $\approx \mathcal{O}(Tp^2\hat{K}^2)$ (Bleakley et al. 2011).

3.4 Applications

In this section, two example applications of the IFGL and GFGL estimators are given. In both cases, it is clear that modelling graph dynamics can provide additional insight into data-generating processes. In the first application, both the GFGL and IFGL methods are applied to a set of time-ordered genetic measurements. In this case the data-set has relatively large dimensionality in terms of the number of variables, but possesses a low number of time-points. In the second example, the IFGL model is used to estimate dynamic models that can represent dependency between features derived from computer network data. In this case, there are less variables to model and an increased number of data-points than in the genetic example. Additionally, given the increased volume of data, a cross-validation mechanism for estimating hyper-parameters is suggested.

3.4.1 Time Evolution of Genetic Dependency Networks

In recent years it has become increasingly common to construct experiments which sample gene-expression activity as a time-series. As an example

[5]In the implementation tested, the GFLseg algorithm is used to solve the Dykstra proximity update $\text{prox}_{R_2}(\mathbf{\Gamma}; \lambda_2)$. See Appendix B.2 for more details.

Figure 3.4.1 – Graphical depiction of the life-cycle stages of *Drosophila Melagonaster* (the common fruit fly). From left to right; embryo, larval, pupa, adult. At each stage throughout the life-cycle, genetic material can be analysed to assess the activity of genes.

of such data, we consider the genetic activity of a fruit-fly (*D. melanogaster*) from its embryonic birth to final adult state (for a visual representation see Figure 3.4.1). The dataset analyzed here is a subset of the data collected by Arbeitman et al. (2002), who measure gene expression patterns for 4096 genes, approximately one third of all *D. melanogaster* genes, over $T = 67$ time-points.

To aid interpretation of the results and for computational feasibility, we consider a smaller subset of genes ($p = 150$), which are understood to be linked to certain biological processes, in this case immune system response. The link between this subset of genes and biological function is motivated by considering conserved *co-domains* of a gene. Where such co-domains are shared between genes, one can often infer a similar biological function of the genes. This similarity can even be extended to other organisms if the genes are homologous (Forslund et al. 2011). In this case, our selection of genes is based on the Flybase Gene-ontology database (Attrill et al. 2016). Understanding the dependency between genes involved in a certain process is of interest to biologists who want to examine and understand why or how regulation of gene activity evolves over time, for example, after an intervention or treatment. Previous work on this data-set by Lèbre et al. (2010) considered estimating changepoints in a causal VAR-type model. In contrast to this work, the analysis here is concerned with estimating the contemporaneous relationships between genes. Specifically, the innovations $\boldsymbol{\epsilon}_t$ are modelled as a dynamic GGM, where $X^{(t)} = X^{(t-1)} + \boldsymbol{\epsilon}^{(t)}$ and $\boldsymbol{\epsilon}^{(t)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Theta}^{(t)})$.

Unlike in the synthetic experiments, the time-course data analyzed here was not replicated, meaning we only have one data-point at each time point in the fly's development. It is worth noting that more recent experiments involving time-course microarray data may produce replicated experiments. These are

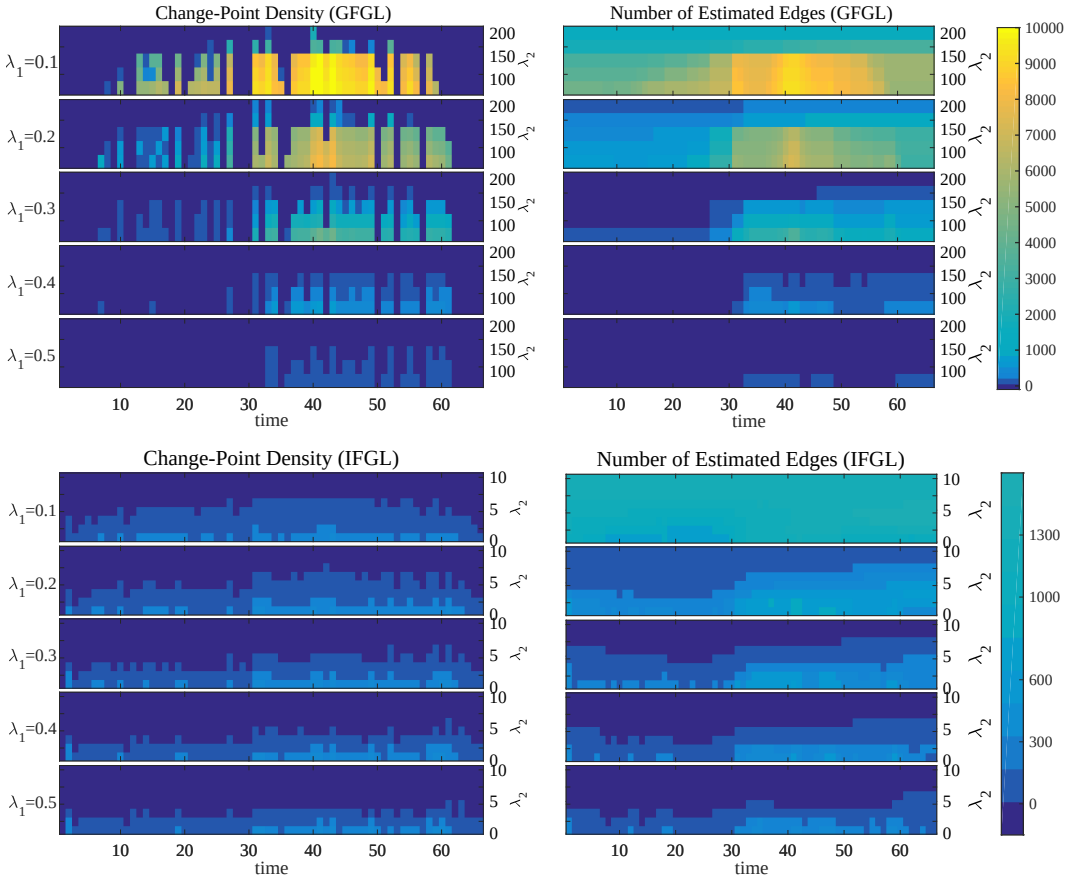


Figure 3.4.2 – Left: Change-point density. Right: number of edges as a function of regulariser parameters λ_1, λ_2. Top: GFGL, Bottom: IFGL.

thought to be particularly valuable, as it may allow us to gauge uncertainty due to variation in genetic populations and environmental factors. With such replicated experiments, there may also be a meaningful way to perform cross-validation to estimate the hyper-parameters. In the absence of replicates, the analysis approach here is an exploratory one whereby inferred structure is assessed over a wide range of regularization parameters. In particular, the sparsity parameter is assessed over the range $\lambda_1 = 0.1$ to 0.5 for both methods, with smoothing set to $\lambda_2 = 80$ to 200 for GFGL, and $\lambda_2 = 1$ to 10 in the IFGL case.

Figure 3.4.2 demonstrates how both the sparsity, number and position of changepoints in the solution behave as a function of λ_1, λ_2. One can clearly see that both smoothing (the number of changepoints) and sparsity (the number

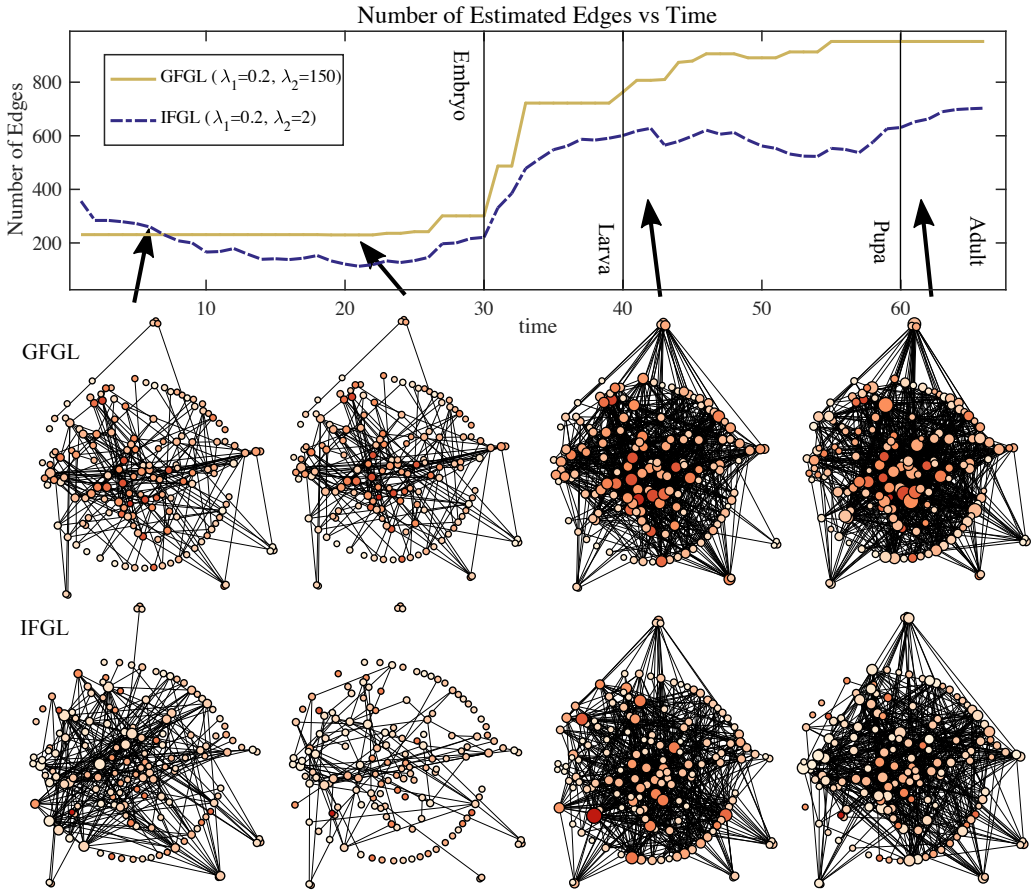


Figure 3.4.3 – An example of the graphical models recovered at one set of solutions. For comparison a set of λ_1, λ_2 which give roughly the same level of sparsity in the IFGL and GFGL models are selected.

of edges) are linked to (λ_1, λ_2) jointly. For a given selection of λ_1, λ_2, we obtain an estimate for the dynamic graph. Figure 3.4.3 presents some snapshots of these graphs.

In this example, the graphs are drawn such that gene-positions (vertices) are comparable both across time and between methods. This application to genetic data clearly illustrates the qualitative differences between the estimators in terms of extracted structure. In both methods we observe that more edges are detected in the later-half of the life-cycle, with a large change in structure inferred during the Larval stage of development. Unlike IFGL, which experiences *changepoints* at all time-points, GFGL clearly has more pronounced

jumps; i.e. more edges change at each changepoint (see Figure 3.4.2). Additionally, if one considers the varying size of node (proportional to degree), it appears that the degree of the GFGL estimates are more stable. Such a feature suggests that the particular GFGL estimate (in Figure 3.4.3) has fewer degrees of freedom than the IFGL estimate. Such a property may be appealing in the high-dimensional setting, where GFGL appears to permit similar graphical structure, but enhanced temporal stability in the graph.

3.4.2 Statistical modelling of Computer Network Traffic

As discussed in the introduction (Sec. 1.1.2), graphical models may provide a useful tool to model computer network traffic. It is well known that network traffic can exhibit non-stationary behaviour (Patcha et al. 2007; Scherrer et al. 2007) and thus one may expect dependencies in metrics derived from the traffic to change over time. In this section, we aim to quantify whether any such temporal variation in dependency exists on real network traffic. To achieve this, we utilise the IFGL model as discussed in previous sections. The GFGL model was not applied in this context due to the high-computational cost of the ADMM-D algorithm with respect to the number of changepoints (see Figure 3.3.6). Application of GFGL with the extended ADMM+ algorithm (B.3) is feasible, and provides an avenue for future research on changepoint detection. Dependencies between network extracted features are then represented as a set of time-varying graphs. The main purpose of the analysis is to uncover key relationships between features and quantify their variation over several days of real data. Although not developed here, one potential application of such models may be to provide anomaly detection functionality. For example, the extracted graphical models may be used to characterise different behaviours or activities conducted over the network. Once allowed activities are adequately mapped out, the models may be used to detect deviations from known activities and potentially alert to malicious activity. A more complete discussion of this application to network traffic modelling may be found in Gibberd, Evangelou, et al. (2016).

Table 2 – List of extracted NetFlow features used in this analysis. For further details on the construction of features see Evangelou et al. (2016).

Name	Description
No_Events	Number of events that start and end in bin
No_StartEvents	Number of events that start in bin but end outside
Bytes_Median	Median of packet size within bin
Bytes_MAD	Median absolute deviance (MAD) of packet size
Bytes_SUM	Total number of bytes in bin
Bytes_SD	Standard-deviation of bytes
Packets_Median	Median number of packets
Packets_MAD	MAD of packer distribution within bin
Packets_SUM	Number of packets within bin
Packets_SD	Standard-deviation of packet distribution in bin
BP_ratio_Median	Median of ratio between bytes and packet
BP_ratio_MAD	MAD of byte-packet ratio
BP_ratio_SUM	Sum of ratio within bin
BP_ratio_SD	Standard-deviation of byte-packet ratio

The Dataset

Before starting the analysis, one must first extract the features themselves from raw network data. Specifically, the dataset used here is constructed from a subset of 10 IP addresses within the Imperial College London (ICL) network[6]. The IP addresses (to be understood as devices) under study are kept constant throughout the study, data was collected for 13 consecutive days, including 4 weekend days which we discount in our analysis. Table 2 provides a summary of $p = 14$ features which quantify; the number of connections, packets, and size of packets traveling across the network. For more information on their construction the reader is referred to Evangelou et al. (2016).

The features we use in this study relate to various statistics of events contained within binned time intervals. Figure 3.4.4 demonstrates what one of these features (Number of events) looks like for different bin sizes. Clearly, the distribution of features within a bin changes depending on the size of the bin. In particular we note that many of the features we use are based on count

[6]This research would not be possible without my collaborators at Imperial College London. Specifically, I would like to thank Niall Adams, Marina Evangelou, and Andy Thomas, who curated the data-set and advised on feature construction and interpretation.

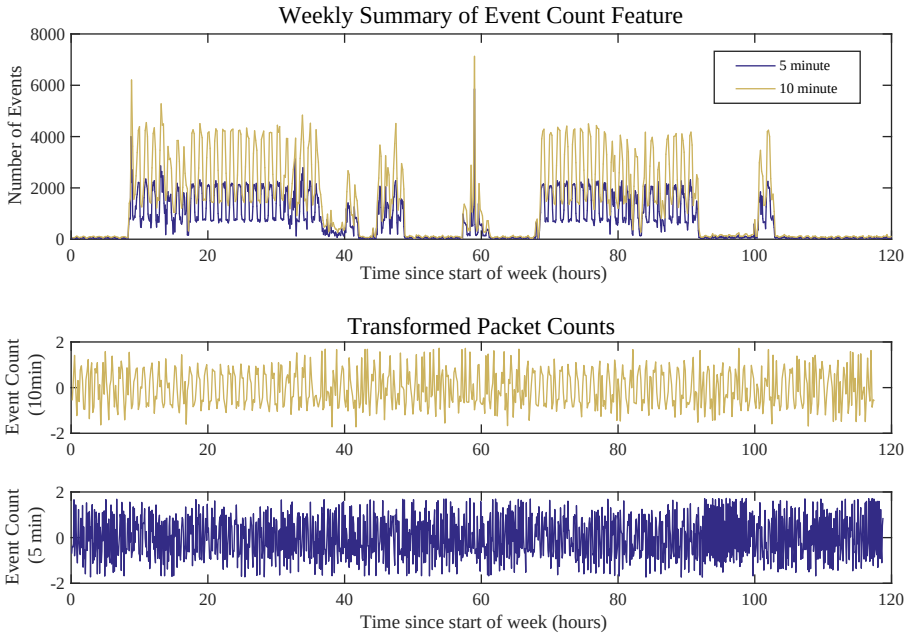


Figure 3.4.4 – Top: Raw packet counts aggregated over 5 and 10 minute intervals.

data, as the bin-size tends to zero these feature distributions will demonstrate an excess of zero values (Evangelou et al. 2016; Scherrer et al. 2007). This is undesirable for our particular analysis as we wish to model the features as continuous random variables. To avoid such problems, we perform our analysis with a relatively large bin-size of 10 minutes. In future work, one may consider looking at dynamic extensions of non-parametric or Poisson graphical models (J. Lafferty et al. 2012; E. Yang et al. 2013) to deal with count data.

To enhance the interpretation of dependency structure it is prudent to ensure that all features are measured on a similar scale level. To achieve this, a localised *z-scoring* procedure is used to approximately remove trends in the empirical mean and marginal variation of feature flows. The mean and variance are estimated locally, in a moving window of width m. We perform a local de-trending and variance stabilising transform (see Figure 3.4.4) to each feature flow according to $X_t = (X_t - \hat{\mu}_t)/\hat{\sigma}_t$ where $\hat{\mu}_t = (\sum_{i=t-h}^{t+h} X_i)/m$ and $\hat{\sigma}_t = \left(\sum_{i=t-h}^{t+h}(X_i - \hat{\mu}_t)^2/m - 1\right)^{1/2}$.

Results

In this experiment we aim to assess whether there is a common daily pattern to dependency dynamics. For example, one might hypothesize network behaviour corresponds to a day/night or work/home cycle. To test such a hypothesis, we run IFGL on each of the 9 working week-days within our dataset. For each day we find a solution path according to a grid of tuning parameters λ_1, λ_2 (similar to that of Figure 3.4.5). Each point in this solution path corresponds to a set of dynamic graphs with different sparsity and smoothness properties. Figure 3.4.5 provides a visualistation of this solution path over the whole week of activity. Qualitatively, this visualisation allows one to examine where and how dependencies appear to change. However, while the solution paths alone provide some insight into the dependency dynamics, ideally we would want to identify a model with appropriately chosen shrinkage and smoothness priors. Such a choice would correspond to taking a cross-section of the solution path for a specific combination of λ_1, λ_2. In the following, a method for selecting such hyper-parameters is developed.

Remark. *A pragmatic approach to cross-validation with dynamics*

As mentioned in the fruit-fly analysis, with time-series we often simply get one snapshot of data. For example, we may observe points $\boldsymbol{X} = (\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(T)})^\top$ *relating to gene-expression levels. However, unlike conventional i.i.d learning environments, the requirement to preserve order in a time-series prevents us re-sampling, or using conventional cross-validation procedures when estimating models. Unlike in the gene-expression example, when studying computer networks we have much more datapoints relative to the dimension p. Additionally, it may be appropriate to assume that the data generating process is in some sense locally stationary. For example, we may consider that dependency patterns may change throughout the day, but this activity may be similar across different days or weeks. In the following, we will make the assumption that there is some level of smoothness and sparsity that is consistent across days. With this assumption in mind, one may try to perform some form of cross-validation across different days of data. We can assume the different days of data constitute independent observations of an assumed generative process.*

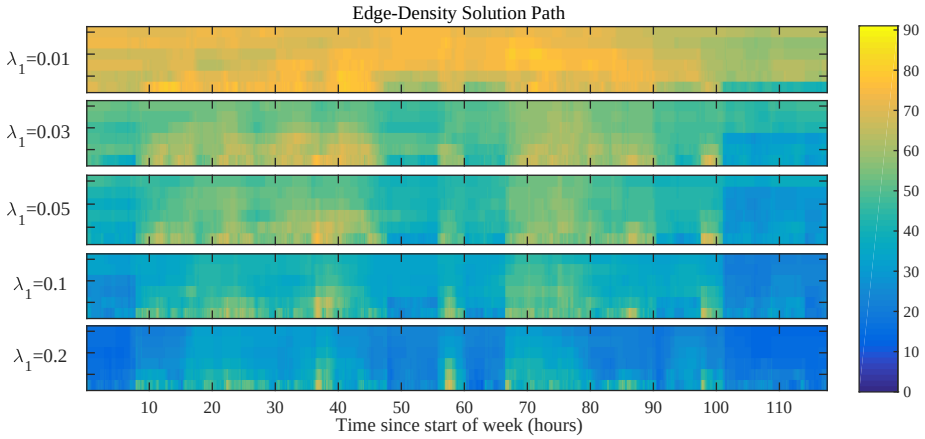


Figure 3.4.5 – Solution path for one week of analysis.

If the data from across different days were generated according to the same process, then a model trained on one day should be able to describe some of the behaviour of another day. The descriptive ability of a model may be formalised by constructing a measure of risk based on how well one *training* day can explain the data of a held out *test* day. Exchanging the test data-set across the days in a leave-one-out cross-validation fashion enables the risk to measure how well estimated models generalise between days. By minimising this risk surface one can estimate an optimal set of tuning parameters (λ_1, λ_2). To construct a risk function, we may adapt the idealised setting where we have knowledge of the ground-truth distribution. For a multivariate Gaussian distribution, the predictive risk for a pair of ground truth $\mathbf{\Sigma}_0$ and estimated $\hat{\boldsymbol{S}}$ covariance matrices is given as $R(\hat{\boldsymbol{S}}) = \mathrm{tr}((\hat{\boldsymbol{S}})^{-1}\mathbf{\Sigma}_0) + \log\det(\hat{\boldsymbol{S}})$. Zhou et al. (2010) observe that up to a constant $R(\hat{\boldsymbol{S}}) = -2E_X[\log(f_{\hat{\boldsymbol{S}}}(Z))]$, where $f_{\hat{\boldsymbol{S}}}$ is the probability density function corresponding to $\mathcal{N}(\mathbf{0}, \hat{\boldsymbol{S}})$ and the data is generated according to $\vec{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_0)$. The likelihood and the risk are thus related via the density function. In our case, this measure of risk is extended to cover all time-points. A leave one out cross-validation risk may then be constructed according to:

(3.4.1)

$$R_{\mathrm{loo}}(\{\hat{\boldsymbol{S}}^{(t)}\}_{t=1}^{T}) \equiv \sum_{t=1}^{T}\left(\frac{1}{N}\sum_{i_{\mathrm{test}}=1}^{N}\sum_{i\neq i_{\mathrm{test}}}\left[\mathrm{tr}(\hat{\mathbf{\Theta}}_i^{(t)}\boldsymbol{S}_{i_{\mathrm{test}}}^{(t)}) + \log\det\left((\hat{\mathbf{\Theta}}_i^{(t)})^{-1}\right)\right]\right),$$

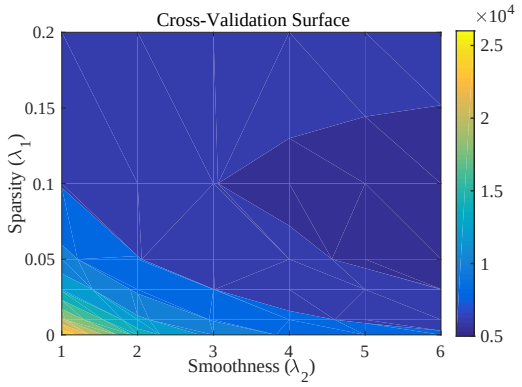


Figure 3.4.6 – Leave-one out cross-validation surface as averaged over $N = 9$ days of data.

where $\boldsymbol{S}^{(t)}_{i_{\text{test}}} = \boldsymbol{x}^{(t)}\boldsymbol{x}^{(t)\top}$ is an ill-conditioned estimate of the local empirical covariance. In effect, by averaging over the N different days of data, we can see how estimated models on one day, perform in terms of describing the data on days in a hold-out set. Such a procedure can be performed for various settings of λ_1 and λ_2 to build up a risk surface.

Figure 3.4.6, presents such a risk surface as estimated for the network traffic features. It tells us a lot about how the estimated dependency graphs generalise across days. In particular we note:

- Either a non-zero λ_1 or λ_2 improves generalisation performance as measured through the risk.
- The shrinkage inducing λ_1 appears to have a minima at around $\lambda_1 = 0.1$.
- The risk surface suggests that λ_2 should be set very large, there is no discernible minima with respect to λ_2.

The fact that the risk surface is minimised for very large λ_2 suggests that almost constant precision matrices should be preferred, i.e. there will be no dynamic structure estimated. This should be interpreted as evidence against the hypothesis that there is a regular daily cycle of dependency patterns, and no general level of smoothness to the graphs. At least in this data-set, there appears to be no consistent pattern to the dynamics across different days. A more interesting observation is the fact that *there is* a minima with respect to λ_1. Thus, while there are no regular temporal patterns this does suggest that there is an optimal level of sparsity which generalises across different days.

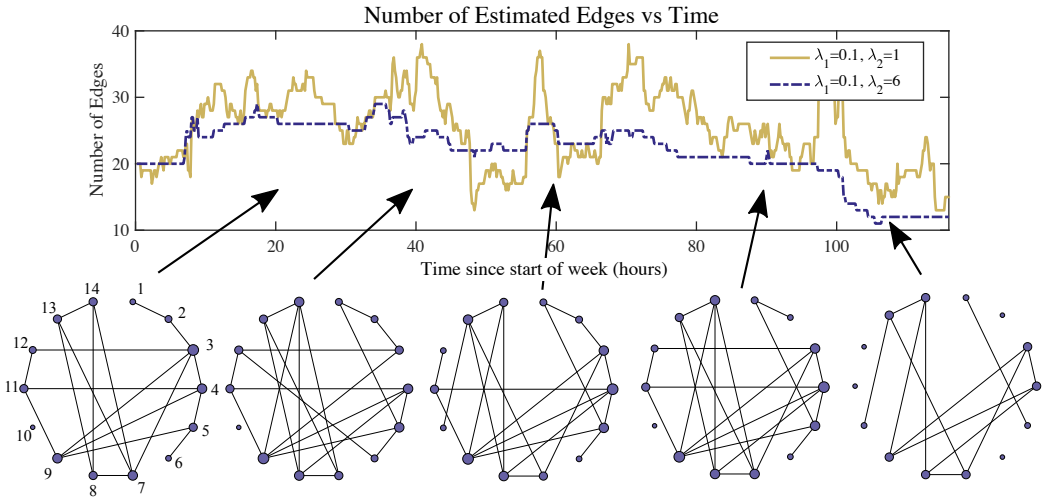


Figure 3.4.7 – Choosing $\lambda_1 = 0.1$ the number of edges is plotted as a function of time for two solutions, one with high smoothness $\lambda_2 = 6$ and one with low smoothness $\lambda_2 = 1$. Below this, some snapshots of graph structure are given at different points in the week (measured in hours). These graphs correspond to the solution with $\lambda_1 = 0.1$ and $\lambda_2 = 6$. *The features are indexed as: 1) bp_ratio_mad; 2) bp_ratio_median; 3) bp_ratio_sd; 4) bp_ratio_sum; 5) bytes_mad; 6) bytes_median; 7) bytes_sd; 8) bytes_sum; 9) no_events; 10) no_start_events; 11) packets_mad; 12) packets_mediam; 13) packets_sd; 14) packets_sum.*

Indeed, this lends support to the assumption that sparsity is appropriate for modelling network traffic dependencies.

While the cross-validation experiment did not provide evidence for a daily cycle of dependency activity, the results in Figure 3.4.5 clearly demonstrate temporal variation in the estimated sparsity patterns. For instance, we note that there seems to be a slightly more dense region within the periods $t \in [10, 50]$ and $t \in [70, 100]$. It is interesting to note that these appear to coincide with periods of increased activity as measured by overall event count (Figure 3.4.4). Rather than a daily cycle, this pattern suggests that meaningful dynamics might be detected over a period of several days. To investigate this structure further, we can harness the cross-validation analysis performed in the intra-day experiment to select an appropriate level of sparsity. Setting $\lambda_1 = 0.1$, we can then take a cross-sectional view of the solution path; Figure 3.4.7 plots the number of edges and snapshots of the estimated graph for two different values of λ_2. Again, as in the fruit-fly example, the extracted

dynamic graphical models can help us understand how features are related to each other. In this case, it appears that the features; *packets_sd*, *packets_sum*, *bytes_sum*, and *bytes_sd*, are somewhat inter-related. In this example, such relationships would intuitively make sense due to the construction of features; one may expect that the number of bytes measured in an interval is a function of the number of packets in the same period.

For privacy reasons, it is not clear in these experiments what specific processes were running on the devices attached to the network throughout the period of study. However, if one could simultaneously monitor what users are doing on their devices whilst measuring the resultant NetFlow data, then graphical models such as those applied here could be used to characterise the state of the network (with respect to these explicit labellings). Demonstrating this in practice would require significant further work, specifically in data-collection and classifying the graphical models, i.e. linking dependency graphs with network state. Given such data, then one may use the extracted graphs as features for classification. Since the dependency structure is encoded by the precision matrix, one could think of this as a similar approach to principle component based methods. In the PCA case, one would use features relating to the largest eigen-vectors of the covariance matrix; in the graphical model case, one would have features which correspond to graphs. It is interesting to note, that sparse extensions to PCA (c.f. Zou, Hastie, and R. Tibshirani (2006)) have been proposed to increase the interpretability of estimated principle components. Given, that incorporating sparsity enables a more robust estimate of the precision/covariance matrix, one may argue that the resultant features, if used in a classifier, would also be more robust, i.e. have lower variance under sampling. Although using the derived classes to construct features is not considered here, it is feasible that enhanced feature robustness may result in more stable classification. In an anomaly detection role, this may act to reduce the false positive rate when flagging anomalies, whilst crucially ensuring that detection ability is not compromised.

3.5 Summary

In this chapter, extensions to the stationary i.i.d graphical models have been developed. We discussed in some detail how different smoothness assumptions can be imposed on the graphical models in the context of M-estimators. Specifically, the IFGL and GFGL estimators were introduced in order to detect edge-wise and graph-wise changes in dependency structure. As these models are inherently extremely flexible, they rely in a very large part on having sufficient regularisation to add curvature within the resultant optimisation problems. Even though the proposed estimators are convex, they require optimisation over many parameters which motivated the development of a new class of ADMM algorithm. The estimation abilities of the methods were then assessed empirically in both a synthetic and real-world setting. In both the case of genetic and computer traffic analysis, it is clear that allowing for dynamics in graphical models can produce significant improvements in the level of insight obtained from observations. In the analysis of fruit-fly development, this enabled the time-localisation of changes in genetic dependency; in the study of computer networks, it enables us to highlight relationships between features and that these may depend on the process operating on the network. In the next chapter, the estimators introduced here are discussed in a more theoretical setting and we try to understand the circumstances in which GFGL may recover true changepoint structure.

Appendix B

B.1 Proof of Proposition 3.1

Proof. Consider the definition of the group-fused signal approximator proximity operator:

$$\text{prox}_{R2}(\boldsymbol{\Gamma}; \lambda_2) = \arg\min_{\boldsymbol{V}} \frac{1}{2}\|\boldsymbol{V} - \boldsymbol{\Gamma}\|_F^2 + \lambda_2 \|\boldsymbol{D}\boldsymbol{V}\|_{2,1}.$$

Re-writing the above with $\boldsymbol{\Omega} = \boldsymbol{D}\boldsymbol{V}$ and constructing $\boldsymbol{V}$ as a sum of differences via $V_{t,\cdot} = \boldsymbol{\omega} + \sum_{i=1}^{t-1} \Omega_{i,\cdot}$, where $\boldsymbol{\omega} = V_{1,\cdot}$, then one can interpret the proximal operator as a group lasso problem (Bleakley et al. 2011). Writing the re-parameterized problem in matrix form, one can show that solving for the jump parameters allows us to reconstruct an estimate for $\boldsymbol{V}$. This is formally equivalent to a group lasso (M. Yuan et al. 2006) class of problem:

$$\hat{\boldsymbol{\Omega}} := \underset{\boldsymbol{\Omega} \in \mathbb{R}^{(T-1)\times P(P-1)/2}}{\arg\min} \frac{1}{2}\|\bar{\boldsymbol{\Gamma}} - \bar{\boldsymbol{R}}\boldsymbol{\Omega}\|_F^2 + \lambda_2 \|\boldsymbol{\Omega}\|_{2,1} \ , \tag{B.1}$$

where a bar $\bar{\boldsymbol{X}}$ denotes a column centered matrix and $\boldsymbol{R} \in \mathbb{R}^{T\times(T-1)}$ is a matrix with entries $R_{i,j} = 1$ for $i > j$ and 0 otherwise, i.e.

$$\boldsymbol{R} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} .$$

The problem (B.1) can be solved through a block-coordinate descent strategy, sequentially updating the solution for each block $\Omega_{t,\cdot}$ for $t = 1, \ldots, T-1$ (see Sec. B.2). We can then construct a solution for $\hat{\boldsymbol{V}}$ by summing the differences and noting that the optimal value for $\boldsymbol{\omega}$ is given by $\hat{\boldsymbol{\omega}} = \boldsymbol{1}_{1,T}(\boldsymbol{\Gamma} - \boldsymbol{R}\hat{\boldsymbol{\Omega}})$. Correspondingly, the proximal operator for R_2 is constructed as

$$\text{prox}_{R2}(\boldsymbol{\Gamma}; \lambda_2) = \big(\hat{\boldsymbol{\omega}}^\top, (\hat{\boldsymbol{\omega}} + \hat{\Omega}_{1,\cdot})^\top, \ldots, (\hat{\boldsymbol{\omega}} + \sum_{i=1}^{T-1} \hat{\Omega}_{i,\cdot})^\top\big)^\top. \tag{B.2}$$

□

B.2 Group-Fused Lasso solver (a note on GFL-Seg)

To solve the group lasso problem in the GFGL subroutine I use the *GFLseg* algorithm developed in Bleakley et al. (2011). This algorithm utilizes a natural block structure in the group lasso problem:

$$\hat{\mathbf{\Gamma}} := \underset{\mathbf{\Gamma}\in\mathbb{R}^{(T-1)\times P}}{\arg\min} \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{X}\mathbf{\Gamma}\|_2^2 + \lambda_2\|\mathbf{\Gamma}\|_{2,1} \ ,$$

where $\boldsymbol{Y}$ is a data or target matrix and $\boldsymbol{X}$ is referred to as the design matrix. We see that the group lasso problem as formulated above is linearly separable across the groups, given by rows in $\mathbf{\Gamma}$. Writing the regularizer as $\|\mathbf{\Gamma}\|_{2,1} = \sum_{T-1}\|\Gamma_{t,\cdot}\|$ one notes that the sum of squared term can also be decomposed across such groups (in our application the groups refer to time slices).

The update for block t can be found according to (Bleakley et al. 2011)

$$\Gamma_{t,\cdot} \leftarrow \frac{1}{\|X_{\cdot,t}\|^2}\left(1 - \frac{\lambda_2}{\|\boldsymbol{e}_t^{-t}\|}\right)_+ \boldsymbol{e}_t^{\setminus t} \ ,$$

where $\boldsymbol{e}_t^{\setminus t} := X_{\cdot,t}^\top(\boldsymbol{Y} - \boldsymbol{X}\mathbf{\Gamma}_{\setminus t})$, and $\mathbf{\Gamma}_{\setminus t}$ denotes the matrix $\mathbf{\Gamma}$ with the t-th row set to zero. If one applies the above update scheme then the estimates are guaranteed to converge (M. Yuan et al. 2006). To speed up the algorithm, Bleakley et al. (2011) adopt an active set strategy which takes advantage of the fact we expect only few active blocks (which would correspond to changepoints). One iterates between adding blocks to the active set $\mathcal{A}$, according to maximal violation of the KKT conditions, and updating blocks in $\mathcal{A}$ according to the above. The KKT conditions for the group lasso are given as:

$$\begin{aligned} -\boldsymbol{e}_t + \frac{\lambda_2\Gamma_{t,\cdot}}{\|\Gamma_{t,\cdot}\|} = 0 \quad \forall\, \Gamma_{t,\cdot} \neq 0 \ , \\ \|-\boldsymbol{e}_t\| \leq \lambda_2 \quad \forall\, \Gamma_{t,\cdot} = 0 \ , \end{aligned}$$

where $\boldsymbol{e}_t = X_{\cdot,t}^\top(\boldsymbol{Y} - \boldsymbol{X}\mathbf{\Gamma})$ is the residual projected along the t-th group. The performance of the ADMM-D algorithm in practice appears to follow the computational complexity as discussed in Bleakley et al. (2011). Numerically, we observe that computational complexity appears to be quadratic in the number of changepoints, see Figure B.1.

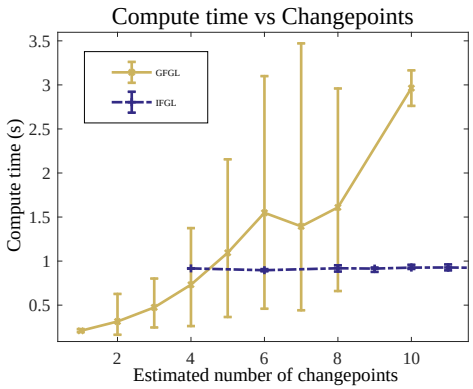


Figure B.1 – Compute-time vs number of estimated changepoints $|\hat{\mathcal{T}}|$.

B.3 Extended ADMM Solver

The algorithm proposed in the main text (as used in Gibberd and Nelson (2017)) comprises an ADMM algorithm that only has one set of auxiliary variables $\{\boldsymbol{V}^t\}$. In the review for that paper it was suggested that a scheme with more auxiliary variables may provide a more robust approach. Such an approach is implemented here with three auxiliary variables $\{\boldsymbol{V}_1^{(t)}, \boldsymbol{V}_2^{(t)}, \boldsymbol{W}^{(t)}\}$. In the process of completing this work, the work of Hallac et al. (2017) was made available via arXiv. The authors propose a similar methodology for recovering time-varying graphical models as given here. Specifically, they introduce auxiliary variables and constraints according to $\boldsymbol{V}_1 = \{\boldsymbol{\Theta}^{(t)}\}, \boldsymbol{V}_2 = \{\boldsymbol{\Theta}^{(t)}\}, \boldsymbol{V}_3 = \{\boldsymbol{\Theta}^{(t-1)}\}$ before proceeding with a linearized ADMM scheme. In the analysis presented here we forgo this extra variable $\boldsymbol{V}_3$ and instead work with the difference defined as $\boldsymbol{W}^t = \boldsymbol{V}_1^t - \boldsymbol{V}_2^{t-1}$. Arguably, the variable splitting proposed here allows for simpler updates of the auxiliary variables. However, given that in both cases proximal updates can be obtained in closed form, for example via thresholding, it should be expected that performance of both algorithms will be similar.

As previously, an ADMM Lagrangian for the class of fused graphical models can be constructed as a function $\mathcal{L}(U, V_1, V_2, W, P_{V_1}, P_{V_2}, P_W)$; where the

variables are grouped into sets corresponding to primal U, auxiliary V, transformations of auxiliary variables W, and dual variables P. The re-scaled Lagrangian is written as:

$$\mathcal{L}(U, V_1, V_2, W, P_{V_1}, P_{V_2}, P_W) := \sum_{t=1}^{T} \Big(-\log\det(\boldsymbol{U}^{(t)}) + \mathrm{tr}(\boldsymbol{U}^{(t)}\boldsymbol{S}^{(t)}) \Big) \dots$$
$$+ \lambda_1 \sum_{t=1}^{T} R_1(\boldsymbol{V}_1^{(t)}) + \lambda_2 \sum_{t=2}^{T} R_2(\boldsymbol{W}^{(t)}) + \frac{\gamma_{V_1}}{2} \Big(\sum_{t=1}^{T} \|\boldsymbol{U}^{(t)} - \boldsymbol{V}_1^{(t)} + \boldsymbol{P}_{V_1}\|_F^2 - \|\boldsymbol{P}_{V_1}\|_F^2 \Big) \dots$$
$$+ \frac{\gamma_{V_2}}{2} \Big(\sum_{t=1}^{T-1} \|\boldsymbol{U}^{(t)} - \boldsymbol{V}_2^{(t)} + \boldsymbol{P}_{V_2^{(t)}}\|_F^2 - \|\boldsymbol{P}_{V_2^{(t)}}\|_F^2 \Big) \dots$$
$$+ \frac{\gamma_W}{2} \Big(\sum_{t=2}^{T} \|(\boldsymbol{V}_1^{(t)} - \boldsymbol{V}_2^{(t-1)}) - \boldsymbol{W}^{(t)} + \boldsymbol{P}_{W^{(t)}}\|_F^2 - \|\boldsymbol{P}_W\|_F^2 \Big) .$$

where $R_1(\cdot)$ and $R_2(\cdot)$ are regulariser functions acting respectively on the precision matrices and their differences. Note, the particular benefit of this formulation is that using an additional set of auxiliary variables $\boldsymbol{W}^t = \boldsymbol{V}_1^t - \boldsymbol{V}_2^{t-1}$ enables us to decouple the update for $\{\boldsymbol{U}\}$ from the differencing terms. Similarly to the ADMM-D algorithm, the $R_2(\cdot)$ regulariser can take the form of either ℓ_1 (IFGL) or $\ell_{2,1}$ (GFGL) smoothing, both options can either include or exclude the smoothing of diagonal elements. For example, in the group-smoothed without diagonal elements we set the regularisers; $R_1(\boldsymbol{V}_1^{(t)}) \equiv \|\boldsymbol{V}_{1\setminus ii}^{(t)}\|_1$, and

$$R_2(\boldsymbol{W}^{(t)}) \equiv \|\boldsymbol{W}^{(t)}\|_F = \|\boldsymbol{V}_1^{(t)} - \boldsymbol{V}_2^{(t-1)}\|_F \equiv \|\boldsymbol{U}_{\setminus ii}^{(t)} - \boldsymbol{U}_{\setminus ii}^{(t-1)}\|_F .$$

The optimisation strategy is now to minimise the Lagrangian $\mathcal{L}(\cdot)$ with respect to U, V_1, V_2, W whilst maximising it with respect to the dual variables P_{V_1}, P_{V_2}, P_W. The updates for the primary, auxiliary, and smoothing terms are given below. The updates for the dual variables follow in the standard way (c.f. Eq. 3.2.4).

Update the primary terms $\{U\}$

This step is similar to that in (Sec. 3.2.2), whereby we solve a semi-definite program

(B.3)

$$\arg \min_{\{\boldsymbol{U}^{(t)} \succeq 0\}_{t=1}^{T}} \Bigg[\sum_{t=1}^{T} \Big(-\log\det(\boldsymbol{U}^{(t)}) + \operatorname{tr}(\boldsymbol{U}^{(t)}\boldsymbol{S}^{(t)}) \Big) \dots + \frac{\gamma_{V_1}}{2} \sum_{t=1}^{T} \|\boldsymbol{U}^{(t)} - \boldsymbol{V}_1^{(t)} + \boldsymbol{P}_{V_1^{(t)}}\|_F^2 + \frac{\gamma_{V_2}}{2} \sum_{t=1}^{T-1} \|\boldsymbol{U}^{(t)} - \boldsymbol{V}_2^{(t)} + \boldsymbol{P}_{V_2^{(t)}}\|_F^2 \Bigg] .$$

This problem is considerably simplified by the introduction of a third auxiliary variable $\boldsymbol{V}_2^t$ which means we do not have to consider time-coupling in this update. The problem is separable w.r.t time and can be trivially parallelised.

Remark. *A useful relation for manipulating weighted Frobenius terms is*

$$\text{(B.4)} \quad \frac{a}{2}\|\boldsymbol{A}-\boldsymbol{B}\|_F^2 + \frac{b}{2}\|\boldsymbol{A}-\boldsymbol{C}\|_F^2 = \frac{(a+b)}{2}\|\boldsymbol{A} - \frac{(a\boldsymbol{B}+b\boldsymbol{C})}{a+b}\|_F^2 + f(\boldsymbol{B},\boldsymbol{C}),$$

where in general we are not interested in the form of $f(\cdot)$.

Minimising (B.3) with respect to $\boldsymbol{U}^{(t)}$ requires us to minimise the following

$$\arg\min_{\boldsymbol{U}^{(t)}} \Big[-\log\det(\boldsymbol{U}^{(t)}) + \operatorname{tr}(\boldsymbol{U}^{(t)}\boldsymbol{S}^{(t)}) + (\gamma_{V_1}+\gamma_{V_2})/2\|\boldsymbol{U}^{(t)} - \boldsymbol{\Gamma}_V^{(t)}\|_F^2 \Big] ,$$

where

$$\boldsymbol{\Gamma}_U^{(t)} := \big(\gamma_{V_1}(\boldsymbol{V}_1^{(t)} - \boldsymbol{P}_{V_1^{(t)}}) + \gamma_{V_2}(\boldsymbol{V}_2^{(t)} - \boldsymbol{P}_{V_2^{(t)}})\big)/(\gamma_{V_1}+\gamma_{V_2}) ,$$

For notational simplicity, let $\bar{\gamma} = (\gamma_{V_1}+\gamma_{V_2})/2$. The solution to this is given via eigen-decomposition, where the regularising term $\bar{\gamma}\|\boldsymbol{U}^{(t)} - \boldsymbol{\Gamma}_U^{(t)}\|_F^2$ acts to update the eigen-values of the resultant matrix $\boldsymbol{U}^{(t)}$. Consider

$$\begin{aligned} \gamma\|\boldsymbol{U}^{(t)} - \boldsymbol{\Gamma}_U^{(t)}\|_F^2 &= \bar{\gamma}\operatorname{tr}((\boldsymbol{U}^{(t)} - \boldsymbol{\Gamma}_U^{(t)})^\top(\boldsymbol{U}^{(t)} - \boldsymbol{\Gamma}_U^{(t)})) \\ &\propto \bar{\gamma}\operatorname{tr}((\boldsymbol{U}^{(t)})^\top\boldsymbol{U}^{(t)}) - 2\bar{\gamma}\operatorname{tr}((\boldsymbol{\Gamma}_U^{(t)})^\top\boldsymbol{U}^{(t)}) . \end{aligned}$$

The gradient of the above gives $-(\boldsymbol{U}^{(t)})^{-1} + \boldsymbol{S}^{(t)} + 2\bar{\gamma}\boldsymbol{U}^{(t)} - 2\bar{\gamma}\boldsymbol{\Gamma}_U^{(t)} = \boldsymbol{0}$, and therefore; $(\boldsymbol{U}^{(t)})^{-1} - 2\bar{\gamma}\boldsymbol{U}^{(t)} = \boldsymbol{S}^{(t)} - 2\bar{\gamma}\boldsymbol{\Gamma}_U^{(t)}$. A solution for $\boldsymbol{U}^{(t)}$ is constructed

by equating the eigenvectors of the left and right hand-sides. The eigenvalues of each side, respectively $\{u_h\}_{h=1}^p$ and $\{s_h\}_{h=1}^p$ obey the quadratic $u_h^{-1} - 2\bar{\gamma}u_h = s_h$. Given the eigenvectors $\{\boldsymbol{v}_h\} := \text{eigenvec}(\boldsymbol{S}^{(t)} - 2\bar{\gamma}\boldsymbol{\Gamma}_U^{(t)})$ and corresponding eigenvalues $\{s_h\}$, an update for u_h can be obtained by solving the quadratic

$$u_h = -\frac{1}{4\bar{\gamma}}\left(s_h \pm \sqrt{s_h^2 + 8\bar{\gamma}}\right) \quad \text{for } h = 1, \ldots, p\ .$$

A positive-semi-definite update for $\boldsymbol{U}^{(t)}$ can now be constructed according to

$$\boldsymbol{U}^{(t)} = \begin{pmatrix} \boldsymbol{v}_1 \\ \\ \boldsymbol{v}_p \end{pmatrix} \begin{pmatrix} u_1 & & \\ & \ddots & \\ & & u_p \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_1 \\ \\ \boldsymbol{v}_p \end{pmatrix}^\top .$$

Updating the auxiliary precision $\{V_1, V_2\}$

The next step is to update the auxiliary variables corresponding to the original (primal variable). These are used in the differencing term and also the sparsity update. Assuming that the shrinkage $R_1(\cdot)$ only applies to off-diagonal entries, we obtain the following soft-thresholding updates for all $i \neq j$:

$$[\boldsymbol{V}_1^{(t)}]_{i,j} = \begin{cases} \text{sign}([\boldsymbol{\Gamma}_{V_1}^{(t)}]_{i,j}) \odot \max(|[\boldsymbol{\Gamma}_{V_1}^{(t)}]_{i,j}| - \frac{\lambda_1}{\gamma_{V_1}+\gamma_W}, \boldsymbol{0}) & \text{for } t > 1 \\ \text{sign}([\boldsymbol{U}^{(t)} + \boldsymbol{P}_{V_1^{(t)}}]_{i,j}) \odot \max(|[\boldsymbol{U}^{(t)} + \boldsymbol{P}_{V_1^{(t)}}]_{i,j}| - \lambda_1/\gamma_{V_1}, \boldsymbol{0}) & \text{for } t = 1 \end{cases},$$

where $\boldsymbol{\Gamma}_{V_1}^{(t)} := \gamma_{V_1}(\boldsymbol{U}^{(t)} + \boldsymbol{P}_{V_1^{(t)}}) + \gamma_W(\boldsymbol{V}_2^{(t-1)} + \boldsymbol{W}^{(t)} - \boldsymbol{P}_{W^{(t)}})/(\gamma_{V_1} + \gamma_W)$. The diagonal updates are given by $\text{diag}(\boldsymbol{V}_1^{(t)}) = \text{diag}(\boldsymbol{\Gamma}^{(t)})$ for $t = 2, \ldots, T$, and $\text{diag}(\boldsymbol{V}_1^{(1)}) = \text{diag}(\boldsymbol{U}^{(1)} + \boldsymbol{P}_{V_1^{(1)}})$ for $t = 1$. The update for V_2 is even simpler and can be found in closed form as

$$\boldsymbol{V}_2^{(t)} = (\gamma_W + \gamma_{V_2})^{-1}\left[\gamma_W(\boldsymbol{V}_1^{(t+1)} - \boldsymbol{W}^{(t+1)} + \boldsymbol{P}_{W^{(t+1)}}) + \gamma_{V_2}(\boldsymbol{U}^{(t)} + \boldsymbol{P}_{V_2^{(t)}})\right] .$$

Updating the auxiliary difference $\{W\}$

Taking the minima of $\mathcal{L}(\{\boldsymbol{U}\}, \{\boldsymbol{V}, \boldsymbol{W}\}, \{\boldsymbol{P}\})$ with respect to $\{\boldsymbol{W}\}$ requires us to solve

$$\arg\min_{\{\boldsymbol{W}^{(t)}\}} \left[\lambda_2 \sum_{t=2}^{T} R_2(\boldsymbol{W}^{(t)}) + \frac{\gamma_W}{2}\left(\sum_{t=2}^{T} \|\boldsymbol{W}^{(t)} - \boldsymbol{\Gamma}_W^{(t)}\|_F^2\right)\right] ,$$

where $\boldsymbol{\Gamma}_W^{(t)} := \boldsymbol{V}_1^{(t)} - \boldsymbol{V}_2^{(t-1)} + \boldsymbol{P}_W$. With $R_2(\boldsymbol{W}^{(t)}) \equiv \|\boldsymbol{W}^{(t)}\|_F$. This update is equivalent to a group lasso update as per the below theorem:

Proposition. *Group Norm Projection*

For matrices $\boldsymbol{W}, \boldsymbol{\Gamma} \in \mathbb{R}^{p\times p}$ the solution to the group-Frobenius projection

$$\arg\min_{\boldsymbol{W}\in\mathbb{R}^{p\times p}} \left[\frac{1}{2}\sum_{t=2}^{T}\|\boldsymbol{\Gamma}^{(t)} - \boldsymbol{W}^{(t)}\|_F^2 + \frac{\lambda_2}{\gamma_W}\sum_{t=2}^{T}\|\boldsymbol{W}^{(t)}\|_F\right] ,$$

is given by the soft-thresholding operator

$$[\hat{W}^{(t)}]_{i,j} = \begin{cases} \frac{\Gamma_{i,j}^{(t)}}{\|\boldsymbol{\Gamma}^{(t)}\|_F}(\|\boldsymbol{\Gamma}^{(t)}\|_F - \frac{\lambda_2}{\gamma_W}) & \text{if } \|\boldsymbol{\Gamma}^{(t)}\|_F > \lambda_2/\gamma_W \\ 0 & \text{otherwise} \end{cases} .$$

Proof. The above can be derived by relating the group-lasso to the optimality conditions required for the lasso. Principally this is achieved by vectorising the matrices $\{\boldsymbol{W}^{(2)}, \ldots, \boldsymbol{W}^{(T)}\}$ and extending the definition of the sign() function. Recall the lasso problem $\arg\min_{\boldsymbol{u}\in\mathbb{R}^T} \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{u}\|_2^2 + \lambda\|\boldsymbol{u}\|_1$ in the signal approximation setting. Now consider creating a partitioned target vector $\tilde{\boldsymbol{u}} \in \mathbb{R}^{(T-1)p^2}$, where $\tilde{\boldsymbol{u}} = (\tilde{\boldsymbol{u}}_1, \ldots, \tilde{\boldsymbol{u}}_{T-1})$ such that $\tilde{\boldsymbol{u}}_i$ represent groups of variables. In particular, consider setting this modified vector to the Frobenius term $\tilde{\boldsymbol{u}}_t = \text{vec}(\boldsymbol{W}^t)$. In the standard lasso setting the thresholding operation is given as

$$\text{soft}(\boldsymbol{u};\lambda) = \text{sign}(\boldsymbol{u}^{(t)})\max(|\boldsymbol{u}^{(t)}| - \lambda, 0)$$

For the group-lasso case, consider extending the notation for the sign function to the matrix

$$[\text{sign}(\boldsymbol{W}^{(t)})]_{i,j} = W_{i,j}^{(t)}/\|\boldsymbol{W}^{(t)}\|_F .$$

Note as per Van Den Berg et al. (2008), the usual properties of the sign function apply:

$$\text{sign}(\alpha\boldsymbol{W}^{(t)}) = \text{sign}(\boldsymbol{W}^{(t)}) \text{ for all } \alpha > 0$$
$$\|\text{sign}(\boldsymbol{W}^{(t)})\|_F < 1 .$$

In the partitioned vector, define the 1-norm as $\|\tilde{\boldsymbol{u}}\|_1 = \sum_{t=1}^{T-1}\left|\|\tilde{\boldsymbol{u}}_t\|_2\right| \equiv \|\boldsymbol{u}\|_{2,1}$, one can now equivalently minimise the objective

$$\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{u}\|_2^2 + \lambda\|\boldsymbol{u}\|_{2,1} \quad \text{or} \quad \frac{1}{2}\|\tilde{\boldsymbol{y}}-\tilde{\boldsymbol{u}}\| + \lambda\|\tilde{\boldsymbol{u}}\|_1 .$$

Using the optimality conditions for the lasso (see Sec. 2.1.4), and using the extended definition of the sign($\boldsymbol{W}^{(t)}$) function leads to the soft-thresholding operation as stated in the proposition (set $\tilde{\boldsymbol{y}} = (\text{vec}(\boldsymbol{\Gamma}^{(2)}), \ldots, \text{vec}(\boldsymbol{\Gamma}^{(T)})) \in \mathbb{R}^{(T-1)p^2}$). For a more complete analysis in the group-lasso setting see Van Den Berg et al. (2008).

□

Chapter 4

Estimation Theory for GFGL

In the previous chapter two methods for estimating dynamic graphical models were introduced, namely the IFGL and GFGL estimators. The synthetic experiments and applications suggest that these estimators can recover graphical models to varying degrees of success. In particular, the synthetic experiments are aligned with what we might intuitively expect, i.e. performance increases with increasing data, or reduced dimensionality. In this chapter the recovery properties of the GFGL estimator are studied from a theoretical perspective. One of the advantages of convex M-estimators such as IFGL and GFGL is that the curvature around the global minima can readily be assessed in a statistical sense.

In this chapter a set of stationarity conditions for the GFGL estimator are derived and then utilised to demonstrate that in high-probability GFGL can consistently estimate multiple changepoints in a GGM. To my knowledge, this is the first result of this kind for a group-fused maximum-likelihood estimator. However, the proof technique follows that of others, primarily Harchaoui et al. (2010), and Kolar et al. (2012). In addition to proving convergence in the standard dimensional setting, an extension to high dimensional settings is discussed. In particular, it is considered how one can reconcile the general framework of Negahban et al. (2012) with the more specialised changepoint consistency approaches of Harchaoui et al. (2010). One should note that the focus of this chapter is on changepoint consistency, as opposed to consistent recovery of graph structure. However, to some extent, the problem of precision

matrix and changepoint estimation go hand in hand and this is discussed where appropriate.

4.1 Preliminaries

As previously discussed, several different forms of cost function have been proposed for precision matrix estimation (see Table 1 and the review in Sec. 2.3.5). Typically, these are based around either Markov-random field (MRF) pseudo likelihoods (Meinshausen and Bühlmann 2006; Ravikumar, Wainwright, and J.D. Lafferty 2010), or the Gaussian likelihood directly (Friedman, Hastie, and R. Tibshirani 2008). The focus in this chapter, is on Gaussian models and specifically the GFGL cost function. In the i.i.d setting where structure does not change over time an analogous estimator would look something like the graphical lasso. For example, one may consider the regularised negative log-likelihood cost

$$\mathcal{L}(\hat{\boldsymbol{S}}, \boldsymbol{U}) = -\log\det(\boldsymbol{U}) + \operatorname{tr}(\hat{\boldsymbol{S}}\boldsymbol{U}) + R(\boldsymbol{U}) \ ,$$

where $R(\boldsymbol{U})$ is a penalty term and $\hat{\boldsymbol{S}} := T^{-1}\sum_{t=1}^{T} \boldsymbol{X}^{(t)}(\boldsymbol{X}^{(t)})^\top$ is the empirical covariance matrix. Lam et al. (2009) demonstrate that for both non-convex $R_{\text{hard}}(\boldsymbol{U}) := \lambda^2 - (|U_{ij}| - \lambda)^2\mathbf{1}_{|U_{i,j}|<\lambda}$ and convex penalties $R_{\ell 1}(\boldsymbol{U}) := \|\boldsymbol{U}\|_1$, a penalised maximum likelihood estimator can recover the true network structure of a GGM. The convergence of this class of estimators has been analysed in the high-dimensional case, where an error bound of order $\|\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}_0\|_F = \mathcal{O}_P(\sqrt{s\log(p)/T})$ can be obtained (Lam et al. 2009; Ravikumar, Wainwright, Raskutti, et al. 2011) (see also the results in Sec. 2.4). However, for sparsistency i.e. the recovery of the true network pattern, Lam et al. (2009) suggest that the ℓ_1 estimator requires the number of non-zero elements to be $s = \mathcal{O}(p)$, the non-convex R_{hard} loss does not have this restriction. As discussed in the introduction, in the high-dimensional setting there is often a tradeoff between algorithmic and statistical efficiency. For example, statistically, one may prefer a non-convex model-selection paradigm, although in practice the algorithms one implements may not be able to recover the global minimiser which achieves this performance. In the i.i.d setting, one can utilise the framework discussed in Sec. 2.4 to obtain finite sample bounds for the graphical lasso, or neighbourhood selection methods.

Very recently some authors have considered expanding graphical models to non-identical (but still high-dimensional) settings. For example, the work of Danaher et al. (2013) considers estimation across multiple classes of GGM. The recent work of Saegusa et al. (2016) obtains finite sample bounds for such non-identical models and makes use of a framework similar to that of Negahban et al. (2012). Theoretical analysis of such models is particularly challenging due to the expanded parameter space required to describe heterogeneous populations. Additionally, one has to account for the affect of the smoothing regulariser in conjunction with sparsity inducing penalties.

In the changepoint literature, there are several works which assess changepoint recovery performance with regularised models. A seminal paper being that of Harchaoui et al. (2010), who demonstrate that total-variation denoising can consistently recover changepoints such that $P[\max_k |\tau_k - \hat{\tau}_k| > \delta_T T] \to 0$ for some decreasing sequence δ_T. While this work focusses on the univariate setting, a similar proof technique is used by Kolar et al. (2012) to demonstrate changepoint consistency with a neighbourhood selection network estimation approach. Additionally, the work of B. Zhang et al. (2015) demonstrates changepoint consistency in a time-varying lasso model. The work of Roy et al. (2016) studies a neighbourhood selection method to estimate graphical models either side of a single changepoint, which is selected over a grid of candidate changepoints. While this is only directly applicable to the estimation of a single changepoint, the work provides an interesting application of the high-dimensional framework of Negahban et al. (2012) to the changepoint estimation setting.

In this work, changepoint consistency of the GFGL estimator is first examined in standard dimensional settings via the application of techniques from Harchaoui et al. (2010). Firstly, it is demonstrated that changepoint consistency can be achieved in standard dimensional settings $p < \min_k\{\tau_k - \tau_{k-1}\} < T$. It is then discussed how these bounds may be expanded to high-dimensional finite sample settings. To my knowledge such a setting has not yet been examined in the literature with respect to multiple changepoint estimation in graphical models. Some of the works mentioned above also give results on estimation error, i.e. the recovery of the precision matrices; however, in the

GFGL case this is left as further work. In the discussion section, some suggestions for how such results may be obtained are given.

Finally, in the proceeding analysis it is assumed that the actual number of changepoints to be estimated is known, and that a set of λ_1, λ_2 can be adequately set to segment into this number of blocks. Consistent estimation for the number of blocks in fused models is still an open problem, and one that I do not aim to tackle here. Pragmatically, one can obtain regularisers through cross-validation if we have repeated time-series (c.f. the NetFlow example in Sec. 3.4.2). An alternative strategy is to choose a set of regularisers that over-segments. Typically, many of the estimated changepoints will then cluster around a true changepoint which can then be recovered by post-processing the raw regularised estimator, such approaches are also suggested and examined in Harchaoui et al. (2010) and Kolar et al. (2012).

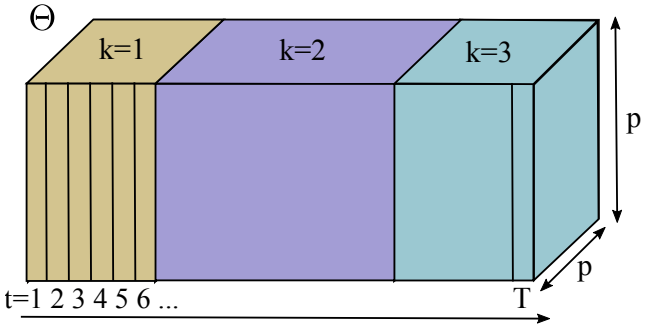
4.1.1 Notation

Demonstrating estimator consistency in dynamic settings can be challenging as one needs to keep track of how both the model and samples are growing with increasing time. As a result, analysis can be notationally complicated. A reference for notation in his chapter can be found in Table 1. Throughout this chapter I will use a form of notational overloading whereby $\mathbf{\Theta}^{(k)} \in \mathbb{R}^{p \times p}$ refers to a *block* indexed precision matrix and $\mathbf{\Theta}^{(t)}$ refers to a *time*-indexed one $t = 1, \ldots, T$. Note that each block will in general contain multiple time-points. It is expected that many $\mathbf{\Theta}^{(t)}$ will be the same, whereas only a few $\mathbf{\Theta}^{(k)}$ will be similar. Within this section, the index k is reserved for indexing such blocks and the changepoints which separate them.

4.1.2 Model and Estimator Definition

Before beginning the analysis we need to define the theoretical settings and model structure against which consistency is defined. In the following, it is assumed that the data-generating process is a block-constant GGM. We also recall the definition of the GFGL estimator.

Table 1 – Summary of mathematical notation for this chapter. The diagram below depicts how a set of precision matrices $\{\boldsymbol{\Theta}^{(t)}\}_{t=1}^{T}$ may also be referenced by their corresponding blocks $\{\boldsymbol{\Theta}^{(k)}\}_{k=1}^{K+1}$ and can be indexed respectively either by block $k = 1, \ldots, K+1$ or time $t = 1, \ldots, T$.



Notation	Description	Example
$[K]$	The set $\{1, \ldots, K\}$	
$\|\boldsymbol{\Sigma}^{(k)}\|_F$	Frobenius norm of covariance at block k	$\sum_{i,j} \lvert\Sigma_{i,j}^{(k)}\rvert^2$
$\left\|\left\|\left\|\boldsymbol{\Sigma}^{(k)}\right\|\right\|\right\|_2$	ℓ_2, ℓ_2 norm, or spectral norm of covariance at block k	$\max_{\|\boldsymbol{x}\|_2=1} \|\boldsymbol{\Sigma}^{(k)}\boldsymbol{x}\|_2$
$\|\boldsymbol{A}\|_\infty$	Largest element in matrix	$\max_{1\le i,j\le p} \lvert A_{i,j}\rvert$
$\phi_\Theta^{(k)}$	Maximum eigenvalue of precision matrix at block k	
ϕ_Θ	Largest eigenvalue in all precision matrices	$\max_{1\le k\le K}\{\phi_\Theta^{(k)}\}$
$\eta_{\min}^{F}$	Minimum jump in Frobenius norm	$\min_{1\le k,k'\le K} \|\boldsymbol{\Sigma}^{(k)} - \boldsymbol{\Sigma}^{(k')}\|_F$
$\eta_{\max}^{\infty}$	Maximum element wise jump	$\max_{1\le k,k'\le K} \|\boldsymbol{\Sigma}^{(k)} - \boldsymbol{\Sigma}^{(k')}\|_\infty$
$d_{\min}$	Minimum true interval	$\min_{k\in K}\{\lvert\tau_k - \tau_{k-1}\rvert\}$
$\hat{d}_{\min}$	Minimum estimated intervals	$\min_{k\in K}\{\lvert\hat{\tau}_k - \hat{\tau}_{k-1}\rvert\}$
n_k	Length of estimated block k	$\hat{\tau}_k - \hat{\tau}_{k-1}$
n_{lk}	Length of overlapping estimated k and ground-truth block l	$\min\{\min\{\hat{\tau}_k, \tau_{l+1}\} - \max\{\hat{\tau}_{k-1}, \tau_l\}, 0\}$
s_k	Sparsity (number of non-zero off-diagonal entries) of block k	
s_∞	Maximum sparsity across blocks	$\max_{1\le k\le K}\{s_k\}$

***Definition* 4.1.** *Block-Constant Gaussian Graphical Model*

Let the set $B^{(k)} = \{\tau_{k-1}, \dots, \tau_k - 1\}$ *denote the block of time-points before the current changepoint* $\tau_k \in [T]$ *but after the previous one* τ_{k-1}*. A* block-constant GGM *is then defined as*

$$X_t \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}^{(k)}) \quad , \; t \in B^{(k)} \; , \tag{4.1.1}$$

where $t \in [T]$ *indexes the time of the observed data-point, and* $k \in [K+1]$ *indexes the blocks resulting from changepoints* $\{\tau_k\}_{k=1}^K$ *at which point the covariance matrices* $\mathbf{\Sigma}^{(k)}$ *change.*

***Definition* 4.2.** *Group-Fused Graphical Lasso Estimator*

Let $\hat{\boldsymbol{S}}^{(t)} := X^{(t)}(X^{(t)})^\top$ *be the localised empirical covariance estimator. The GFGL estimator is constructed by minimising the penalised log-likelihood, such that*

$$\begin{aligned}\{\hat{\mathbf{\Theta}}^{(t)}\}_{t=1}^T = \arg\min_{\{\boldsymbol{U}^{(t)}\}_{t=1}^T} \Bigg\{ \sum_{t=1}^T \big(-\log\det(\boldsymbol{U}^t) + \operatorname{trace}(\hat{\boldsymbol{S}}^{(t)}\boldsymbol{U}^{(t)})\big) + \lambda_1 \sum_{t=1}^T \sum_{i\neq j} |U_{i,j}^{(t)}| \\ \dots + \lambda_2 \sum_{t=2}^T \sqrt{\sum_{i,j=1}^P (U_{i,j}^{(t)} - U_{i,j}^{(t-1)})^2} \Bigg\} \; .\end{aligned} \tag{4.1.2}$$

Once the precision matrices have been estimated, changepoints may be recovered by identifying time-points where the estimated precision matrices change. Specifically, we may construct the set:

$$\hat{\mathcal{T}} := \{\hat{\tau}_1, \dots, \hat{\tau}_{\hat{K}}\} \equiv \{t \,|\hat{\mathbf{\Theta}}^{(t)} - \hat{\mathbf{\Theta}}^{(t-1)} \neq \mathbf{0}\} \; ,$$

where $\hat{\tau}_1, \dots, \hat{\tau}_{\hat{K}}$ *are the estimated changepoint locations. Additionally, we define the augmented changepoint set* $\hat{\mathcal{T}}' := \{1\} \cup \hat{\mathcal{T}} \cup \{T+1\}$*.*

The task in this work is to assess how well, or indeed if, the GFGL estimator can recover the changepoint positions $\{\tau_1 \dots, \tau_K\}$. To understand properties of the estimator when applied to the block-GGM model one needs to consider the curvature of the loss function in Eq. 4.1.2 around identified minima. The

following lemma considers what happens when the gradient of the GFGL estimator is set to zero, as is the case when the objective is minimised. It should be noted that the equality in the optimality condition holds with probability one, i.e. it is a deterministic result. In the sequel, the stochastic contribution to the estimator is considered (via $\hat{\boldsymbol{S}}^t$) from which the performance of the estimator can be assessed in conjunction with the generative model.

***Lemma* 4.1.** *Optimality Conditions for GFGL*

Let $\vec{\boldsymbol{\Psi}}^{(t)} := \hat{\boldsymbol{S}}^{(t)} - \boldsymbol{\Sigma}^{(t)}$ *be the difference between a ground-truth covariance and the empirical (single sample) covariance matrix* $\hat{\boldsymbol{S}}^{(t)} := \vec{X}^{(t)}(\vec{X}^{(t)})^\top$*, where* $\vec{X}^{(t)}$ *has the distribution given by the model in (4.1.1). Let us introduce a matrix quantity corresponding to the differences in precision matrices:*

$$\boldsymbol{\Gamma}^{(t)} = \begin{cases} \boldsymbol{\Theta}^{(t)} & \text{for } t = 1 \\ \boldsymbol{\Theta}^{(t)} - \boldsymbol{\Theta}^{(t-1)} & \text{otherwise} \end{cases} .$$

Furthermore, let $\hat{\boldsymbol{R}}_1^{(t)}$ *be the sub-gradient of the* ℓ_1 *penalty and* $\hat{\boldsymbol{R}}_2^{(t)}$ *the sub-gradient of the group penalty, such that*

$$\hat{R}_{1(i,j)}^{(t)} = \begin{cases} \text{sign}(\sum_{s\leq t} \Gamma_{i,j}^{(s)}) & \text{if } \sum_{s\leq t} \Gamma_{i,j}^{(s)} \neq 0 \\ [-1,1] & \text{otherwise} \end{cases} \quad \text{and } \hat{\boldsymbol{R}}_2^{(t)} = \begin{cases} \frac{\hat{\boldsymbol{\Gamma}}^{(t)}}{\|\hat{\boldsymbol{\Gamma}}^{(t)}\|_F} & \text{if } \hat{\boldsymbol{\Gamma}}^{(t)} \neq \boldsymbol{0} \\ \in \mathcal{B}_F(0,1) & \text{otherwise} \end{cases} ,$$

where $\mathcal{B}_F(0,1)$ *is the unit ball. The minimiser* $\{\hat{\boldsymbol{\Theta}}^{(t)}\}_{t=1}^T$ *of the GFGL objective satisfies the following*

$$\sum_{t=l}^{T} \left((\boldsymbol{\Theta}^{(t)})^{-1} - (\hat{\boldsymbol{\Theta}}^{(t)})^{-1} \right) - \sum_{t=l}^{T} \vec{\boldsymbol{\Psi}}^{(t)} + \lambda_1 \sum_{t=l}^{T} \hat{\boldsymbol{R}}_1^{(t)} + \lambda_2 \hat{\boldsymbol{R}}_2^{(l)} = \boldsymbol{0} ,$$

for all $l \in [T]$ *and* $\hat{\boldsymbol{R}}_2^{(1)} = \hat{\boldsymbol{R}}_2^{(T)} = \boldsymbol{0}$.

Proof. The result follows from considering the gradient of Eq. 4.1.2, see Appendix C.2.

4.2 Changepoint Consistency

In this section, a result for changepoint consistency in standard dimensions is stated. The approach for demonstrating consistency used here is similar to that of Harchaoui et al. (2010) and Kolar et al. (2012). The work of Kolar et al. (2012) is perhaps the most related to this analysis as the authors in that paper also attempt to find changepoints in a graphical model with a group-fused penalty. However, there are several key differences between the GFGL estimator and the neighbourhood selection approach studied in that work:

***Remark* 4.1.** *GFGL vs Group-Fused Neighbourhood Selection*

Where the neighbourhood selection acts at a row/column wise level, the GFGL estimator groups changes in edge structure across the full graph. As discussed in Section 3.1.4, neighbourhood selection is the result of a set of p group-fused optimisation problems; the changepoints in the graph are thus related to each node and its neighbours, i.e. they don't encode changes across the whole graph. In the case of GFGL the estimator is obtained by optimising over the whole set of precision matrices jointly.

In the context of changepoint detection for the block-constant GGM (4.1.1) the results of Kolar et al. (2012) require one to use a union bound over the p separate nodes. The bulk of the theory in their paper operates at the node level, with the argument being that when one has successfully recovered the local structure then global structure can be recovered by combining results over nodes. This contrasts with the approach taken here where the stationarity conditions apply to the whole set of p nodes jointly.

Unlike Kolar et al. (2012) who utilise a neighbourhood selection approach the constraints on the model are now enforced at a graph-wise rather than node-wise level. These are formalised with regards to the following quantities:

- Let $d_{\min} := \min_{k \in [K+1]} |\tau_k - \tau_{k-1}|$ be the minimum distance between changepoints. In the sequel, it is useful to consider this as a proportion of T. Specifically, let us define $\gamma_{\min} = d_{\min}/T$, i.e. a constant proportion of T.
- The minimum jump size is denoted $\eta_{\min} := \min_{k \in [K]} \|\mathbf{\Sigma}^{(k+1)} - \mathbf{\Sigma}^{(k)}\|_F$

Unlike neighbourhood based estimators, GFGL considers changepoints at the full precision matrix scale, and one might therefore expect that the minimum jump size is greater than that utilised in the neighbourhood selection case. For example, in the nodewise case, consider the analogous quantity

$$\eta^{\text{NS}}_{\text{min}}(a) := \min_{k\in[K]} \|\Sigma^{(k+1)}_{a,\cdot} - \Sigma^{(k)}_{a,\cdot}\|_2$$

for nodes[1] $a = 1, \ldots, p$. Summing over nodes, the neighbourhood selection jump size can now be related to the mixed (group) norm $\|\mathbf{\Sigma}^{(k+1)} - \mathbf{\Sigma}^{(k)}\|_{2,1} = \sum_a \|\Sigma^{(k+1)}_{a,\cdot} - \Sigma^{(k)}_{a,\cdot}\|_2$. Furthermore, if the smallest jump occurs at the neighbourhood for all blocks, i.e. $\arg\min_{k\in[K]} \|\Sigma^{(k+1)}_{a,\cdot} - \Sigma^{(k)}_{a,\cdot}\|_2 = \arg\min_{k\in[K]} \|\Sigma^{(k+1)}_{a',\cdot} - \Sigma^{(k)}_{a',\cdot}\|_2$ for all $a \neq a' \in [p]$, then $\sum_a \eta^{\text{NS}}_{\text{min}}(a) = \min_k \|\mathbf{\Sigma}^{(k+1)} - \mathbf{\Sigma}^{(k)}\|_{2,1}$. Using the inequality (for $\boldsymbol{x} \in \mathbb{R}^n$) $\|\boldsymbol{x}\|_2 \leq \|\boldsymbol{x}\|_1 \leq \sqrt{n}\|\boldsymbol{x}\|_2$, the jumps as measured through the group-norm can be related to those measured in a Frobenius sense, such that

$$\eta_{\text{min}} \leq \sum_a \eta^{\text{NS}}_{\text{min}}(a) \leq \sqrt{p}\eta_{\text{min}} \ .$$

Thus, even though the minimum jump size in the GFGL case is greater, i.e. $\eta_{\text{min}} > \eta^{\text{NS}}_{\text{min}}(a)$, it is not proportionally greater, i.e. when one considers summing over nodes. In the proceeding analysis it should be noted that consistent recovery of changepoints requires a tradeoff between the minimum jump-size η_{min} and the amount of data T. For example, a smaller minimum jump-size will generally require more data; as expected it is harder to detect small jumps. The relation $\eta_{\text{min}} \leq \sum_{a=1}^{p} \eta^{\text{NS}}_{\text{min}}(a)$ suggests that the minimum jump-size at a graph-wide (precision matrix wide) level is smaller when measured in the Frobenius norm, rather than at a node-wise level. As a result, for equivalent scaling of η_{min} and $\eta^{\text{NS}}_{\text{min}}$ the graph-wide GFGL method will be able to detect smaller (graph-wide) jumps with an equivalent level of data. Conversely, if the jumps one is interested in occur at the neighbourhood level the neighbourhood based method would be more appropriate, however, this is generally not the case with the block-constant GGM model (4.1.1).

In order to control the variance in the underlying model it is required to introduce several assumptions on the generating process:

[1]It is worth noting that the work of Kolar et al. (2012) requires a minimum jump on the precision matrix elements rather than the covariance. In the proof here we work directly with the covariance due to the form of the optimality conditions (Lemma 4.1).

***Assumption* 4.1.** *Bounded Eigenvalues*

There exist two constants $\phi_{\min} > 0, \phi_{\max} < \infty$ that give the minimum and maximum eigenvalues of the true covariance matrix (across all blocks) such that

$$\phi_{\min} = \min\{\Lambda_{\min}(\mathbf{\Sigma}^{(k)})|\ k \in [K+1]\} \quad \text{and} \quad \phi_{\max} = \max\{\Lambda_{\max}(\mathbf{\Sigma}^{(k)})|\ k \in [K+1]\}\ .$$

***Assumption* 4.2.** *Variables are scaled such that $\Sigma^{(k)}_{ii} = 1$ for all $k \in [K+1]$ and all $i \in V$.*

***Assumption* 4.3.** *Finite Jumps*

There exists a constant $M > 0$ such that the difference between any two blocks is bounded:

$$\max_{k,k' \in [K+1]} \|\mathbf{\Sigma}^{(k')} - \mathbf{\Sigma}^{(k)}\|_F \leq M\ .$$

The above assumptions essentially require the variance of the underlying process to be well behaved. The eigenvalue conditions represent constraints on the minimum and maximum variance of the process. Assumption 4.3 requires that the difference between covariance matrices is bounded by a constant. It is generally satisfied automatically by Assumption 4.1[2].

The changepoint consistency result presented here will take the form of an upper bound on the maximum error. To demonstrate this, let us introduce the quantity $\{\delta_T\}_{T \geq 1}$ as a non-increasing positive sequence that converges to zero as $T \to \infty$. Note: this should converge at a rate which ensures an increasing absolute quantity $T\delta_T \to \infty$ as $T \to \infty$. Specific settings of this quantity are discussed after statement of the main result.

***Assumption* 4.4.** *Minimum jump size*

The tradeoff between jump-size and data quantity is formalised through the assumption that

$$\eta_{\min}\sqrt{T\delta_T}/\phi_{\max} \to \infty \tag{4.2.1}$$

as $T \to \infty$.

[2]A finite bound on the covariance also implies a bound on the precision.

These conditions ensure that there are sufficient samples to estimate the empirical covariance around the estimated changepoints.

***Theorem* 4.1.** *Changepoint Consistency (Standard Dimensions)*
Given Assumptions 4.1,4.2,4.3, 4.4, and appropriate regularisers λ_1 and λ_2 such that:

$$\begin{aligned}\beta_1 &:= (\eta_{\min}\gamma_{\min}T)^{-1}\lambda_2 \to 0 \ , \\ \beta_2 &:= \eta_{\min}^{-1}\lambda_1\sqrt{p(p-1)} \to 0 \ , \\ \beta_3 &:= (\eta_{\min}T\delta_T)^{-1}\lambda_2 \to 0 \ ,\end{aligned}$$

as $T \to \infty$ and $|\hat{K}| = K$. With a finite but large sample T such that $\beta_1^{-1} > 32$, $\beta_2^{-1} > 8$ and $\beta_3^{-1} > 3$, then the changepoint error is bounded according to probability

$$P[\max_{k\in[K]} |\tau_k - \hat{\tau}_k| \leq T\delta_T] \geq 1 - C_K \exp\{-\eta_{\min}\sqrt{T\delta_T}/40\phi_{\max}\} \ , \tag{4.2.2}$$

where $C_K = K(K^2 2^{K+1} + 4)$. Furthermore, if $\eta_{\min}\sqrt{T\delta_T}/\phi_{\max} \to \infty$ (as per Ass. 4.4) for $T \to \infty$ the changepoint error is bounded such that

$$P[\max_{k\in[K]} |\tau_k - \hat{\tau}_k| \leq T\delta_T] \to 1 \ .$$

Unlike in high-dimensional settings, the regularisation parameters have less strict requirements on their form. As suggested by Kolar et al. (2012) the form

$$\lambda_1 \asymp \lambda_2 = \mathcal{O}(\sqrt{\log(T)/T}) \ .$$

enables convergence in probability with the following quantities

$$\delta_T = \log(T)^\alpha/T \quad \text{and} \quad \eta_{\min} = \Omega((\log T)^{(1-\alpha)/2}) \ . \tag{4.2.3}$$

Under such regularisation, the conditions in the theorem are met and the exponential bound of (4.2.2) takes the form $\exp(-c_1^{-1}\sqrt{\log T})$. While this rate is relatively slow, this does not depend on α, so holds regardless of whether $\eta_{\min}$ varies with T or not.

Alternatively, one may consider the polynomial quantities

$$\eta_{\min} = \Omega(T^{-b}) \quad \text{and} \quad \delta_T = T^{-a} \ ,$$

where $0 < a < 1$, $0 < b < 1/2$ and $a + 2b = c < 1$. In this case $T\delta_T > 0$ still increases with T, however, we obtain the exponential bound $\exp(-c_1^{-1}T^{(1-c)/2})$. Unlike in (4.2.3), when using the polynomial scaling there is a clear trade-off between the minimum jump size $\eta_{\min}$, and the amount of data $T\delta_T$ required to gain a certain level of changepoint consistency. For example, considering the case where $b = 0$ (and thus $\eta_{\min}$ is a constant as $T \to \infty$) enables, for a fixed value of c, a larger value of $a = c$ and therefore changepoints may be recovered with greater accuracy for the same quantity of data, i.e. $T\delta_T\big|_{a=c} < T\delta_T\big|_{a=c-2b}$.

Although the focus of this work is on changepoint, as opposed to structural recovery, as we will see in the proof these problems go somewhat hand-in-hand. Typically, one attempts to assess structural recovery in terms of a normed error $\|\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}_0\|$, or via sparsistency (graph recovery) results as in Sec. 2.4.5. In the following proof, such results are not derived, rather the stationarity conditions themselves can be used to bound the estimation error (in standard dimension). In the high-dimensional setting one must check that sufficient curvature exists in the estimator to recover the correct structure. This is further discussed in Sec. 4.4.

4.3 Proof of Changepoint Consistency

We relate the proof bounding the maximum deviation between estimated and true changepoints to the probability of an individual changepoint breaking the bound. Following Harchaoui et al. (2010) and Kolar et al. (2012), we utilise the union bound

$$P[\max_{k\in[K]} |\tau_k - \hat{\tau}_k| \geq T\delta_T] \leq \sum_{k\in[K]} P[|\tau_k - \hat{\tau}_k| \geq T\delta_T] \ .$$

Note: the compliment of the event on the LHS is equivalent to the target of proof; we wish to demonstrate $P[\max_{k\in[K]} |\tau_k - \hat{\tau}_k| \leq T\delta_T] \to 1$. In order to show this, we need to show the LHS above goes to zero as $T \to \infty$. It is sufficient, via the union bound, to demonstrate that the probability of the (rather bad) events:

$$A_{T,k} := \{|\tau_k - \hat{\tau}_k| > T\delta_T\} \ ,$$

go to zero for all $k \in [K]$. The strategy presented here separates the probability of $A_{T,k}$ occurring across complimentary events. In particular, let us construct what can be thought of as a good event, where the estimated changepoints are within a region of the true ones:

$$C_T := \left\{ \max_{k \in [K]} |\hat{\tau}_k - \tau_k| < \frac{d_{\min}}{2} \right\} .$$

The task is then to show that $P[A_{T,k}] \to 0$ by showing $P[A_{T,k} \cap C_T] \to 0$ and $P[A_{T,k} \cap C_T^c] \to 0$ as $T \to 0$.

4.3.1 Stationarity induced bounds

As a first step let us introduce some rough bounds (which occur in probability one) based on the optimality conditions. From here, a set of events can be constructed that occur when the stationarity conditions are met. These can be intersected with the required events to break up the probabilities for $\mathbb{P}[A_{T,k} \cap C_T]$ which can then be separately bounded towards zero.

Without loss of generality, consider the stationarity equations (4.1) with changepoints $l = \tau_k$ and $l = \hat{\tau}_k$ such that[3] $\hat{\tau}_k < \tau_k$. Taking the differences between the equations we find (see Appendix C.3 for more details):

$$\| \sum_{t=\hat{\tau}_k}^{\tau_k - 1} \left((\boldsymbol{\Theta}^{(t)})^{-1} - (\hat{\boldsymbol{\Theta}}^{(t)})^{-1} \right) - \sum_{t=\hat{\tau}_k}^{\tau_k - 1} \vec{\boldsymbol{\Psi}}^{(t)} + \lambda_1 \sum_{t=\hat{\tau}}^{\tau_k - 1} \hat{\boldsymbol{R}}_1^{(t)} \|_F \leq 2\lambda_2 . \tag{4.3.1}$$

The gradient from the ℓ_1 term $\sum_{t=\hat{\tau}_k}^{\tau_k - 1} \lambda \hat{\boldsymbol{R}}_1^{(t)}$ can obtain a maximum value of $\pm\lambda_1(\tau_k - \hat{\tau})$ for each entry in the precision matrix, transferring this to the RHS we obtain:

$$\| \sum_{t=\hat{\tau}_k}^{\tau_k - 1} \left((\boldsymbol{\Theta}^{(t)})^{-1} - (\hat{\boldsymbol{\Theta}}^{(t)})^{-1} \right) - \sum_{t=\hat{\tau}_k}^{\tau_k - 1} \vec{\boldsymbol{\Psi}}^{(t)} \|_F \leq 2\lambda_2 + \lambda_1 \sqrt{p(p-1)} (\tau_k - \hat{\tau}_k) .$$

In order to examine the conditions in detail, it is prudent to split the LHS of the above display into components relating to the ground-truth, minimiser, and stochastic terms. Let us re-write the above as:

[3] We can use $\hat{\tau}_k < \tau_k$ to get inequalities for this configuration of changepoint estimator. However, the reverse situation $\tau_k > \hat{\tau}_k$ follows through symmetry.

(4.3.2)
$$\|\sum_{t=\hat{\tau}_k}^{\tau_k-1}\left((\mathbf{\Theta}^{(t)})^{-1}-(\hat{\mathbf{\Theta}}^{(t)})^{-1}\right)\|_F-\|\sum_{t=\hat{\tau}_k}^{\tau_k-1}\vec{\mathbf{\Psi}}^{(t)}\|_F\leq 2\lambda_2+\lambda_1\sqrt{p(p-1)}(\tau_k-\hat{\tau}_k)\ .$$

The next step is to replace the time indexed inverse precision matrices $\mathbf{\Theta}^{(t)}$ with the block-covariance matrices indexed $\mathbf{\Sigma}^{(k)}$ and $\mathbf{\Sigma}^{(k+1)}$. We can re-express the difference in precision matrices as the sum of a difference between true values before τ_k , i.e. $\mathbf{\Sigma}^{(k+1)}-\mathbf{\Sigma}^{(k)}$, and the difference between the next $(k+1)$st true block and estimated block, i.e. $\hat{\mathbf{\Sigma}}^{(k+1)}-\mathbf{\Sigma}^{(k+1)}$. After some algebra (Appendix C.3) we obtain the bound

$$\begin{aligned}\lambda_2+\lambda_1\sqrt{p(p-1)}(\tau_k-\hat{\tau}_k)\quad\geq\quad&\underbrace{\|\sum_{t=\hat{\tau}_k}^{\tau_k-1}\mathbf{\Sigma}^{(k)}-\mathbf{\Sigma}^{(k+1)}\|_F}_{\|R_1\|_F}-\underbrace{\|\sum_{t=\hat{\tau}_k}^{\tau_k-1}\hat{\mathbf{\Sigma}}^{(k+1)}-\mathbf{\Sigma}^{(k+1)}\|_F}_{\|R_2\|_F}\dots\\ &\dots-\underbrace{\|\sum_{t=\hat{\tau}_k}^{\tau_k-1}\vec{\mathbf{\Psi}}^{(t)}\|_F}_{\|R_3\|_F}\ ,\end{aligned}\tag{4.3.3}$$

which holds with probability one. Define the events:

$$\begin{aligned}E_1:=&\{\lambda_2+\lambda_1\sqrt{p(p-1)}(\tau_k-\hat{\tau}_k)\geq\frac{1}{3}\|R_1\|_F\}\\ E_2:=&\{\|R_2\|_F\geq\frac{1}{3}\|R_1\|_F\}\\ E_3:=&\{\|R_3\|_F\geq\frac{1}{3}\|R_1\|_F\}\end{aligned}$$

Since we know that the bound (4.3.3) occurs with probability one, then the union of these three events must also occur with probability one, i.e. $P[E_1\cup E_2\cup E_3]=1$.

4.3.2 Bounding the Good Cases

One of the three events above are required to happen, either together, or separately. We can thus use this to bound the probability of both the good C_T and bad $A_{T,k}$ events. Similarly to Harchaoui et al. (2010) and Kolar et al.

(2012) we obtain

$$P[A_{T,k} \cap C_T] \leq P[\overbrace{A_{T,k} \cap C_T \cap E_1}^{A_{T,k,1}}] + P[\overbrace{A_{T,k} \cap C_T \cap E_2}^{A_{T,k,2}}] + P[\overbrace{A_{T,k} \cap C_T \cap E_3}^{A_{T,k,3}}]$$

The following sub-sections describe how to separately bound these sub-events.

Bound on $A_{T,k,1}$

Unlike in the work of Kolar et al. (2012), there is no stochastic element (related to the data $\vec{X}_t$) within the first event $A_{T,k,1}$. We can bound the probability of $P[A_{T,k,1}]$ by considering the event $\{\frac{1}{3}\|R_1\|_F \leq \lambda_2 + \lambda_1\sqrt{p(p-1)}(\tau_k - \hat{\tau}_k)\}$. Given $\|R_1\|_F = \|\sum_{t=\hat{\tau}_k}^{\tau_k - 1} \mathbf{\Sigma}^{(k)} - \mathbf{\Sigma}^{(k+1)}\|_F \geq (\tau_k - \hat{\tau}_k)\eta_{\min}$ we therefore obtain the bound

$$P[A_{T,k,1}] \leq P[(\tau_k - \hat{\tau}_k)\eta_{\min}/3 \leq \lambda_2 + \lambda_1\sqrt{p(p-1)}(\tau_k - \hat{\tau}_k)] \ .$$

When the events $C_T, A_{T,k}$ occur we have $T\delta_T < \tau_k - \hat{\tau}_k \leq d_{\min}/2$ to ensure the event $A_{T,k,1}$ does not occur, we need:

$$\frac{\eta_{\min} T\delta_T}{\lambda_2} > 3 \quad \text{and} \quad \frac{\eta_{\min}}{\lambda_1\sqrt{p(p-1)}} > 3 \ . \tag{4.3.4}$$

This occurs asymptotically if

$$3\lambda_2(\eta_{\min} T\delta_T)^{-1} \to 0 \quad \text{and} \quad 3\lambda_1\sqrt{p(p-1)}\eta_{\min}^{-1} \to 0 \ ,$$

as $T \to \infty$. Note that for a large enough T, we can show that the probability $P[A_{T,k,1}] = 0$, the size of this T depends on the quantities in Eq. 4.3.4.

Bound on $A_{T,k,2}$

Consider the quantity $\bar{\tau}_k := \lfloor(\tau_k + \tau_{k+1})/2\rfloor$. On the event C_n, we have $\hat{\tau}_{k+1} > \bar{\tau}_k$ so $\hat{\mathbf{\Sigma}}^{(t)} = \hat{\mathbf{\Sigma}}^{(k+1)}$ for all $t \in [\tau_k, \bar{\tau}_k]$. Using the optimality conditions (Lemma 4.1) with changepoints at $l = \bar{\tau}_k$ and $l = \tau_k$ we obtain the expression:

$$2\lambda_2 + \lambda_1\sqrt{p(p-1)}(\bar{\tau}_k - \tau_k) \geq \|\sum_{t=\tau_k}^{\bar{\tau}_k - 1} \hat{\mathbf{\Sigma}}^{(k+1)} - \mathbf{\Sigma}^{(k+1)}\|_F - \|\sum_{t=\tau_k}^{\bar{\tau}_k - 1} \vec{\mathbf{\Psi}}^{(t)}\|_F$$

$$\|\hat{\mathbf{\Sigma}}^{(k+1)} - \mathbf{\Sigma}^{(k+1)}\|_F \leq \frac{4\lambda_2 + 2\lambda_1\sqrt{p(p-1)}(\bar{\tau}_k - \tau_k) + 2\|\sum_{t=\tau_k}^{\bar{\tau}_k - 1} \vec{\mathbf{\Psi}}^{(t)}\|_F}{\tau_{k+1} - \tau_k}$$

We now combine the bounds for events E_1 and E_2, via $E_2 := \{\|R_2\|_F \geq \frac{1}{3}\|R_1\|_F\}$ and the bounds $\|R_1\|_F \geq (\tau_k - \hat{\tau}_k)\eta_{\min}$ and $\|R_2\|_F \leq (\tau_k - \hat{\tau}_k)\|\hat{\mathbf{\Sigma}}^{k+1} - \mathbf{\Sigma}^{k+1}\|_F$. Substituting in (4.3.5) we have

(4.3.6)
$$P[A_{T,k,2}] \leq P[E_2] = P\left[\eta_{\min} \leq \frac{12\lambda_2 + 6\lambda_1\sqrt{p(p-1)}(\bar{\tau}_k - \tau_k) + 6\|\sum_{t=\tau_k}^{\bar{\tau}_k - 1}\vec{\mathbf{\Psi}}^{(t)}\|_F}{\tau_{k+1} - \tau_k}\right] .$$

Splitting the probability into three components, we obtain
(4.3.7)
$$P[A_{T,k,2}] \leq P[\eta_{\min} d_{\min} \leq 12\lambda_2] + P[\eta_{\min} \leq 3\lambda_1\sqrt{p(p-1)}] + P\left[\eta_{\min} \leq \frac{6\|\sum_{t=\tau_k}^{\bar{\tau}_k - 1}\vec{\mathbf{\Psi}}^{(t)}\|_F}{\tau_{k+1} - \tau_k}\right] .$$

Convergence of the first two terms follows as in $A_{T,k,1}$, the second is exactly covered in $A_{T,k,1}$; however, the third term $\eta_{\min} \leq 3\|\sum_{t=\tau_k}^{\bar{\tau}_k - 1}\vec{\mathbf{\Psi}}^{(t)}\|_F/(\bar{\tau}_k - \tau_k)$ requires some extra treatment. As $\bar{\tau}_k < \tau_{k+1}$, we can relate the covariance matrix of the ground-truth (time-indexed) and block (indexed by k) such that $\mathbf{\Sigma}^{(t)} = \mathbf{\Sigma}^{(k)}$ for all $t \in [\tau_k, \tau_{k+1}]$. One can now write the sampling error across time into one which relates to blocks k as

$$\|\sum_{t=\tau_k}^{\bar{\tau}_k - 1}\vec{\mathbf{\Psi}}^{(t)}\|_F \equiv (\bar{\tau}_k - \tau_k)\|\vec{\boldsymbol{W}}_k^{(\bar{\tau}_k - \tau_k)}\|_F ,$$

where the block level sampling error is given as

$$\vec{\boldsymbol{W}}_k^{(n)} := \left(\frac{1}{n}\sum_{t=1}^{n}\vec{Z}^{(t)}(\vec{Z}^{(t)})^\top\right) - \mathbf{\Sigma}_0^i \quad , \quad \vec{Z}^{(t)} \sim \mathcal{N}(0, \mathbf{\Sigma}_0^{(k)}) . \tag{4.3.8}$$

The probability we desire to bound is $P[\eta_{\min} \leq 3\|\vec{\boldsymbol{W}}_k^{(\tau_k - \bar{\tau}_k)}\|_F]$, or equivalently

$$P[\|\vec{\boldsymbol{W}}_k^{(\tau_k - \bar{\tau}_k)}\|_F > \eta_{\min}/3] . \tag{4.3.9}$$

***Lemma* 4.1.** *Frobenius Bound on Sample Error (SD)*
In standard dimensions $p < \sqrt{d_{\min}}$ we have

$$P[\|\vec{\boldsymbol{W}}_k^{(\bar{\tau}_k - \tau_k)}\|_F > \eta_{\min}/3] \leq 2\exp\left\{-\eta_{\min}\sqrt{d_{\min}}/24\sqrt{2}\phi_{\max}\right\} .$$

Proof. See Appendix C.4.

Gathering the terms in 4.3.7, asymptotically only the last term occurs with non-zero probability such that

$$P[A_{T,k,2}] \leq 2\exp\left\{-\eta_{\min}\sqrt{d_{\min}}/24\sqrt{2}\phi_{\max}\right\} . \tag{4.3.10}$$

To maintain convergence $P[\|\vec{\boldsymbol{W}}_k^{(\bar{\tau}_k-\tau_k)}\| > \eta_{\min}/3] \to 0$, it is simply required that $\eta_{\min}\phi_{\max}^{-1} > (\gamma_{\min}T)^{-1/2}$ as $T \to \infty$, as per Assumption 4.4.

Bound on $A_{T,k,3}$

Recall $P[A_{T,k,3}] := P[A_{T,k} \cap C_T \cap E_3] := P[A_{T,k} \cap C_T \cap \{\|\sum_{t=\hat{\tau}_k}^{\tau_k-1} \vec{\boldsymbol{\Psi}}^{(t)}\|_F \geq \frac{1}{3}\|R_1\|_F\}]$. Given that $\|R_1\|_F \geq (\tau_k - \hat{\tau}_k)\eta_{\min}$ with probability 1, an upper bound on $P[A_{T,k,3}]$ can be found according to

$$\begin{aligned} P[A_{T,k,3}] &\leq P\left[\|\vec{\boldsymbol{W}}_k^{(\tau_k-\hat{\tau}_k)}\|_F > \eta_{\min}/3\right] \\ &\leq 2\exp\left\{-\eta_{\min}\sqrt{T\delta_T}/24\phi_{\max}\right\} . \end{aligned} \tag{4.3.11}$$

The above is similar to (4.3.9), except where the interval we integrate over is given as $T\delta_T < \tau_k - \hat{\tau}_k \leq d_{\min}/2$ (recall we condition on $A_{T,k}$ and C_T). In a similar manner to $A_{T,k,2}$ the probability converges if $\eta_{\min}\phi_{\max}^{-1} > (T\delta_T)^{-1/2}$, since $\delta_T < \gamma_{\min}$ this probability is the main limiting factor for convergence as $T \to \infty$, as noted in Assumption 4.4.

4.3.3 Bounding the Bad Cases

In order to complete the proof, we need to demonstrate that $P[A_{T,k} \cap C_T^c] \to 0$. The argument below follows that of Harchaoui et al. (2010), whereby the bad case is split into several events:

$$\begin{aligned} D_T^{(l)} : &= \{\exists k \in [K],\ \hat{\tau}_k \leq \tau_{k-1}\} \cap C_T^c, \\ D_T^{(m)} : &= \{\forall k \in [K],\ \tau_{k-1} < \hat{\tau}_k < \tau_{k+1}\} \cap C_T^c, \\ D_T^{(r)} : &= \{\exists k \in [K],\ \hat{\tau}_k \geq \tau_{k+1}\} \cap C_T^c, \end{aligned}$$

where $C_T^c = \{\max_{k\in[K]} |\hat{\tau}_k - \tau_k| \geq d_{\min}/2\}$ is the compliment of the good event. The events above correspond to estimating a changepoint; a) before

the previous true changepoint ($D_T^{(l)}$); b) between the previous and next true changepoint ($D_T^{(m)}$), and c) after the next true changepoint ($D_T^{(r)}$). The events $D_T^{(l)}$ and $D_T^{(r)}$ appear to be particularly bad as the estimated changepoint is very far from the truth, due to symmetry we can bound these events in a similar manner. Focussing on the middle term $P[A_{T,k} \cap D_T^{(m)}]$, let us again assume $\hat{\tau}_k < \tau_k$, the reverse arguments hold by symmetry.

***Lemma* 4.2.** *Upper bound for* $P[A_{T,k} \cap D_T^{(m)}]$

The probability of the intersection of $A_{T,k}$ *and* $D_T^{(m)}$ *can be bounded from above by considering the events*

$$E_k^{'} := \{(\hat{\tau}_{k+1} - \tau_k) \geq d_{\min}/2\} , \tag{4.3.12}$$

$$E_k^{''} := \{(\tau_k - \hat{\tau}_k) \geq d_{\min}/2\} . \tag{4.3.13}$$

In particular, one can demonstrate that:

$$P[A_{T,k} \cap D_T^{(m)}] \leq P[A_{T,k} \cap E_k^{'} \cap D_T^{(m)}] + \sum_{j=k+1}^{K} P[E_j^{''} \cap E_j^{'} \cap D_T^{(m)}] . \tag{4.3.14}$$

Proof. The result follows from expanding events based on neighbouring changepoints (see Appendix C.5 for detail).

Bound on $P[A_{T,k} \cap D_T^{(m)} \cap E_k^{'}]$

Consider the stationarity conditions (4.3.2) and set start/end points as $l = \hat{\tau}$, $l = \tau_k$ and $l = \hat{\tau}_k, l = \tau_{k+1}$, we respectively obtain:

(4.3.15)

$$|\tau_k - \hat{\tau}_k| \| \left(\mathbf{\Sigma}^{(k)} - \hat{\mathbf{\Sigma}}^{(k+1)} \right) \|_F \leq 2\lambda_2 + \lambda_1 \sqrt{p(p-1)} (\tau_k - \hat{\tau}_k) + \| \sum_{t=\hat{\tau}_k}^{\tau_k - 1} \vec{\mathbf{\Psi}}^{(t)} \|_F$$

(4.3.16)

$$|\tau_k - \hat{\tau}_{k+1}| \| \left(\mathbf{\Sigma}^{(k+1)} - \hat{\mathbf{\Sigma}}^{(k+1)} \right) \|_F \leq 2\lambda_2 + \lambda_1 \sqrt{p(p-1)} (\hat{\tau}_{k+1} - \tau_k) + \| \sum_{t=\tau_k}^{\hat{\tau}_{k+1} - 1} \vec{\mathbf{\Psi}}^{(t)} \|_F .$$

The next step is to define an event that can bound $\mathbb{P}[A_{T,k} \cap E_k^{'} \cap D_T^{(m)}]$. Using the triangle inequality we bound $\|\mathbf{\Sigma}^{(k+1)} - \mathbf{\Sigma}^{(k)}\|_F$ whilst noting that we have

$E_k^{'} := \{(\hat{\tau}_{k+1} - \tau_k) \geq \frac{d_{\min}}{2}\}$ and $A_{T,k} := \{|\tau_k - \hat{\tau}_k| > T\delta_T\}$. One can therefore construct the event

$$H_T^{\Sigma} : = \left\{ \|\mathbf{\Sigma}_{k+1} - \mathbf{\Sigma}_k\|_F \leq 2\lambda_1\sqrt{p(p-1)} + 2\lambda_2\left(\frac{1}{T\delta_T} + \frac{2}{d_{\min}}\right)\ldots \right.$$
$$\left. + \frac{\|\sum_{t=\hat{\tau}_k}^{\tau_k - 1} \vec{\mathbf{\Psi}}^{(t)}\|_F}{\tau_k - \hat{\tau}_k} + \frac{\|\sum_{t=\tau_k}^{\hat{\tau}_{k+1}-1} \vec{\mathbf{\Psi}}^{(t)}\|_F}{\hat{\tau}_{k+1} - \tau_k} \right\} \tag{4.3.17}$$

which bounds the first term of (4.3.14) such that $P[A_{T,k} \cap E_k^{'} \cap D_T^{(m)}] \leq P[H_T^{\Sigma} \cap \{\tau_k - \hat{\tau}_k \geq T\delta_T\} \cap E_k^{'}]$. Splitting the intersection of events we now have 5 terms to consider

$$\begin{aligned}
& P[A_{T,k} \cap E_k^{'} \cap D_T^{(m)}] \\
\leq\ & P[H_T^{\Sigma} \cap \{\tau_k - \hat{\tau}_k \geq T\delta_T\} \cap E_k^{'}] \\
\leq\ & P\left[2\lambda_1\sqrt{p(p-1)} \geq \frac{\eta_{\min}}{5}\right] + P\left[\frac{2\lambda_2}{T\delta_T} \geq \frac{\eta_{\min}}{5}\right] + P\left[\frac{4\lambda_2}{d_{\min}} \geq \frac{\eta_{\min}}{5}\right] \\
& + P\left[\left\{\frac{\|\sum_{t=\hat{\tau}_k}^{\tau_k-1} \vec{\mathbf{\Psi}}^{(t)}\|_F}{\tau_k - \hat{\tau}_k} \geq \frac{\eta_{\min}}{5}\right\} \cap \left\{(\tau_k - \hat{\tau}_k \geq T\delta_T)\right\}\right] \\
& + P\left[\left\{\frac{\|\sum_{t=\tau_k}^{\hat{\tau}_{k+1}-1} \vec{\mathbf{\Psi}}^{(t)}\|_F}{\hat{\tau}_{k+1} - \tau_k} \geq \frac{\eta_{\min}}{5}\right\} \cap \left\{\left(\hat{\tau}_{k+1} - \tau_k \geq \frac{d_{\min}}{2}\right)\right\}\right] .
\end{aligned}$$

The stochastic error terms (containing $\vec{\mathbf{\Psi}}^{(t)}$) can then be shown to converge similarly to $P[A_{T,k} \cap C_T]$ c.f. Eq. (4.3.6). Again, it is worth noting that the term involving $T\delta_T$ will be slowest to converge, as $d_{\min} = \gamma_{\min}T > \delta_T T$ for large T. The first three terms are bounded through the assumptions on $d_{\min}, \lambda_1, \lambda_2$, and δ_T as required by the theorem (and enforce a similar requirement to those used to bound $P[A_{T,k,1}]$ in Eq. 4.3.4). The other terms in (4.3.14), i.e. $\sum_{j=k+1}^{K} P[E_j^{''} \cap E_j^{'} \cap D_T^{(m)}]$ can be similarly bounded. Instead of using exactly the event H_T^{Σ} one simply replaces the term $1/T\delta_T$ in (4.3.17) with $2/d_{\min}$.

Bound on $D_T^{(l)}$

Recall $D_T^{(l)} := \{\exists k \in [K],\ \hat{\tau}_k \leq \tau_{k-1}\} \cap C_T^c$. The final step of the proof is to show that the bound on $A_{T,k} \cap D_T^{(l)}$, and similarly $A_{T,k} \cap D_T^{(r)}$ tends to zero:

***Lemma* 4.3.** *Union bound for $D_T^{(l)}$*

The probability of $D_T^{(l)}$ is bounded by

$$P[D_T^{(l)}] \leq 2^K \sum_{k=1}^{K-1} \sum_{l \geq k}^{K-1} P[E_l^{''} \cap E_l^{'}] + 2^K P[E_K^{'}] \, .$$

Proof. This is based on a combinatorial argument for the events that can be considered on addition of each estimated changepoint. For details see Appendix C.6.

In order to bound the above probabilities we relate the events $E_l^{''}$ and $E_l^{'}$ to the stationarity conditions as before (via Eqs. 4.3.15, 4.3.16). Setting $k = l$ and invoking the triangle inequality gives us

$$\begin{aligned} \Bigg\{ \|\mathbf{\Sigma}_{l+1} - \mathbf{\Sigma}_l\|_F \quad &\leq \quad 2\lambda_1 \sqrt{p(p-1)} + \overbrace{2\lambda_2 \left(\frac{1}{|\tau_l - \hat{\tau}_l|} + \frac{1}{|\hat{\tau}_{l+1} - \tau_l|} \right)}^{\Gamma} \dots \\ &\quad + \frac{\| \sum_{t=\hat{\tau}_l}^{\tau_l - 1} \vec{\mathbf{\Psi}}^{(t)} \|_F}{|\tau_l - \hat{\tau}_l|} + \frac{\| \sum_{t=\tau_l}^{\hat{\tau}_{l+1}-1} \vec{\mathbf{\Psi}}^{(t)} \|_F}{|\hat{\tau}_{l+1} - \tau_l|} \Bigg\} \, . \end{aligned}$$

Conditioning on the event $E_l^{''} \cap E_l^{'}$ implies that $\Gamma = 8\lambda_2 / d_{\min}$. We can thus write

$$\begin{aligned} P[E_l^{''} \cap E_l^{'}] \quad &\leq \quad P\left[\frac{\eta_{\min}}{8} \leq \lambda_1 \sqrt{p(p-1)} \right] + P\left[\frac{\eta_{\min}}{32} \leq \frac{\lambda_2}{d_{\min}} \right] \\ &\quad + P\left[\left\{ \frac{\eta_{\min}}{4} \leq \frac{\| \sum_{t=\hat{\tau}_l}^{\tau_l - 1} \vec{\mathbf{\Psi}}^{(t)} \|_F}{|\tau_l - \hat{\tau}_l|} \right\} \cap \left\{ \tau_l - \hat{\tau}_l \geq \frac{d_{\min}}{2} \right\} \right] \\ &\quad + P\left[\left\{ \frac{\eta_{\min}}{4} \leq \frac{\| \sum_{t=\tau_l}^{\hat{\tau}_{l+1}-1} \vec{\mathbf{\Psi}}^{(t)} \|_F}{|\hat{\tau}_{l+1} - \tau_l|} \right\} \cap \left\{ \hat{\tau}_{l+1} - \tau_l \geq \frac{d_{\min}}{2} \right\} \right] \, . \end{aligned}$$

Finally, the term corresponding to the last changepoint can be bounded by noting that when $k = K$ we have $\Gamma = 6\lambda_2 / d_{\min}$.

$$P[E_K^{''}] \leq P\left[\frac{\eta_{\min}}{8} \leq \lambda_1 \sqrt{p(p-1)} \right] + P\left[\frac{\eta_{\min}}{24} \leq \frac{\lambda_2}{d_{\min}} \right]$$

$$+ P\left[\left\{\frac{\eta_{\min}}{4} \leq \frac{\|\sum_{t=\hat{\tau}_K}^{\tau_K-1} \vec{\mathbf{\Psi}}^{(t)}\|_F}{|\tau_K - \hat{\tau}_K|}\right\} \cap \{\tau_K - \hat{\tau}_K \geq \frac{d_{\min}}{2}\}\right]$$

$$P\left[\frac{\eta_{\min}}{4} \leq \frac{\|\sum_{t=\tau_K}^{T} \vec{\mathbf{\Psi}}^{(t)}\|_F}{|T+1-\tau_K|}\right] . \tag{4.3.18}$$

4.3.4 Summary

The bounds derived above demonstrate that $P[A_{T,k}] \to 0$ since $P[A_{T,k} \cap C_T] \to 0$ and $P[A_{T,k} \cap C_T^c] \to 0$ as $T \to 0$. In particular, once the sample size T is large enough then the bound $P[\max_{k\in[K]} |\tau_k - \hat{\tau}_k| \geq T\delta_T] \leq K \max_{k\in[K]} P[|\tau_k - \hat{\tau}_k| \geq T\delta_T]$ can be obtained. The event $E_l'' \cap E_l'$ establishes a minimal condition on T, such that $\eta_{\min} d_{\min}/\lambda_2 > 32$ and $\eta_{\min}/\lambda_1\sqrt{p(p-1)} > 8$. A final condition for $A_{T,k,1}$ requires $\eta_{\min} T\delta_T/\lambda_2 > 3$. Once T is large enough to satisfy these conditions, the probabilistic bound is determined either by the smallest block size $d_{\min} = \gamma_{\min} T$ or by the minimum error $T\delta_T$. Summing the probabilities, one obtains the upper bound:

$$\begin{aligned} P[|\tau_k - \hat{\tau}_k| \geq T\delta_T] \leq & 2 \times 2^K\big((K-1)^2+1\big)\left(2\exp\left\{-\frac{\eta_{\min}\sqrt{d_{\min}/2}}{32\phi_{\max}}\right\}\right) \\ & + 2\exp\left\{-\frac{\eta_{\min}\sqrt{T\delta_T}}{40\phi_{\max}}\right\} \\ & + 2\exp\left\{-\frac{\eta_{\min}\sqrt{T\delta_T}}{24\phi_{\max}}\right\} , \end{aligned}$$

where the different rows correspond to events; top) $D_T^{(l)}$, $D_T^{(r)}$; middle) $D_T^{(m)}$, and bottom) $A_{T,k,2}$ and $A_{T,k,3}$. Since $\delta_T T < \gamma_{\min} T$ the above bounds will be dominated by the $\exp\{-\eta_{\min}\sqrt{T\delta_T}/40\phi_{\max}\}$ term. A suitable (although not particularly tight) overall bound on the probability is

$$\begin{aligned} P[\max_{k\in[K]} |\tau_k - \hat{\tau}_k| \geq T\delta_T] \leq & K^3 2^{K+1} \exp\{-\eta_{\min}\sqrt{d_{\min}/2}/32\phi_{\max}\} \ldots \\ & \ldots + K(4\exp\{-\eta_{\min}\sqrt{T\delta_T}/40\phi_{\max}\} \\ & \leq C_K \exp\{-\eta_{\min}\sqrt{T\delta_T}/40\phi_{\max}\} , \end{aligned}$$

where $C_K = K(K^2 2^{K+1} + 4)$. We thus arrive at the result of Theorem 4.1.

□

4.4 Changepoint Consistency (in High-Dimensions)

While Theorem 4.1 asserts consistency in terms of changepoint estimation, it relies strongly on sampling bounds derived in standard-dimensional settings (Lemmas C.1, C.2). In the context of the changepoint proof, the dimensionality[4] (in relation to sampling) concerns the quantities $p < T\delta_T$ and $p < \sqrt{d_{\min}}$. However, in cases where $p > T$, we find ourselves firmly in a high-dimensional setting and we can never attain $p < T\delta_T$. As a minimum we are required to obtain alternative sampling bounds for Eqs. 4.3.10, 4.3.11. Additionally, experience from the canonical example of the lasso (Sec. 2.4) suggests that in high-dimensions we are required to ensure there is sufficient curvature around the optima. The remainder of this section discusses these two issues within the context of GFGL for the estimation of block-constant GGM. In particular, the GFGL problem is discussed in the context of a regularised M-estimator under the framework of Negahban et al. (2012). The work of Saegusa et al. (2016) is of interest here as the authors also consider a fused estimator of a form similar to GFGL. However, unlike GFGL, in that work, the block partitions are known a-priori and there is no need to estimate changepoints. As will be demonstrated in the following sections, such joint estimation of both changepoints and precision matrix structure requires careful thought over how to deal with mis-specification. The discussion that follows does not constitute a proof of consistency in high-dimensions but simply suggests a pathway towards such results. In particular, we discuss how one may account for sampling error and estimation error in high-dimensions.

4.4.1 Sampling

Lemmas C.1 and C.2 are not sufficient in the high-dimensional setting where $p > T\delta_T$. However, an alternative upper bound for the sampling error between the empirical and ground-truth covariance matrix can be obtained as below (Ravikumar, Wainwright, and J.D. Lafferty 2010).

***Lemma* 4.1.** *Frobenius Sampling Bound (HD)*

[4]However, even under these conditions, the GFGL model may still be considered in some sense high-dimensional as the number of model parameters scales as $\mathcal{O}(Tp^2)$.

Consider a zero-mean p-variate random vector $\vec{Z}^{(t)} \sim \mathcal{D}(\mathbf{0}, \mathbf{\Sigma}_0^{(k)})$ *with covariance* $\mathbf{\Sigma}_0^{(k)} \in \mathbb{R}^{p\times p}$ *such that each* $\vec{Z}_i^{(t)}/[\Sigma_0^{(k)}]_{ii}^{1/2}$ *is sub-Gaussian with parameter* ζ_k*. The error of the empirical covariance estimator*

$$\vec{\boldsymbol{W}}_k^{(n)} := \left(\frac{1}{n}\sum_{t=1}^{n} \vec{Z}^{(t)}(\vec{Z}^{(t)})^\top\right) - \mathbf{\Sigma}_0^{(k)} \ ,$$

is bounded according to

$$P\left[\|\vec{\boldsymbol{W}}_k^{(n)}\|_F > \epsilon\right] \leq p^2 4\exp\left\{-\frac{n\epsilon^2}{2a_k^2p^2}\right\} \ ,$$

where $a_k = 8(1+4\zeta_k^2)$*. Note: In the block-constant GGM model* $\zeta_k = 1$*.*

Proof. See Appendix C.5.

Using the above lemma allows us to bound events which depend on sampling, for example in $A_{T,k,2}$ (see Eq. 4.3.10), one may obtain $P[\|\boldsymbol{W}_k^{(\bar{\tau}_k-\tau_k)}\|_F > \eta_{\min}/3] \leq p^2 4\exp(-d_{\min}\eta_{\min}^2/36a^2p^2)$ (see Appendix **??**). As such, it is clearly possible to obtain some level of control on the sampling noise, even in the high-dimensional setting.

***Remark* 4.2.** *Non-Gaussian processes*

While in Theorem 4.1 we considered a process which is strictly Gaussian, it may be possible to extend analysis to non-Gaussian situations via Lemma 4.1. For example, the tail bounds derived in Ravikumar, Wainwright, Raskutti, et al. (2011) (which Lemma 4.1 is based on) are used to bound the estimation error of the estimated covariance for sub-Gaussian sampling schemes. Additionally, it is possible to utilise polynomial tail bounds (c.f. Ravikumar, Wainwright, Raskutti, et al. (2011) and Saegusa et al. (2016)) to analyse an even wider range of processes. However, one has to bear in mind, that when the process is not Gaussian the precision matrix may require a different interpretation. For example, the sparsity structure may no longer encode conditional dependency relationships.

4.4.2 Curvature

As discussed in Section 2.4, the gradient of a loss function may be flat in many directions when considering high-dimensional estimation. The advantage, and purpose of regularisation is to give added curvature to statistical estimation problems. In Lemma 4.1 the optimality conditions for the GFGL estimator were derived, and it should be noted at this point that these conditions hold independently of dimension. However, in the high-dimensional setting the specific form of the regularisers are more important, in that they act to give curvature in certain directions. If this curvature is appropriate, then one may conjecture model structure can be consistently estimated, even as $p > T$ and $p, T \to \infty$. The form of curvature required is formalised by the restricted strong convexity (RSC) condition as laid out in Section 2.4.2 (Def. 2.9).

To reconcile the analysis of the previous section (and that of Harchaoui et al. (2010) and Kolar et al. (2012)) with the M-estimation framework of Negahban et al. (2012), one needs to check that the RSC condition holds with respect to an appropriate sub-space of model parameters. The following discussion forms some preliminary work in this direction and suggests an alternative method to bound estimation error $\|\hat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|$, and thus changepoint error, even in high-dimensions. Such bounds may then be used in place of (or in conjunction with) the bounds derived from GFGL stationarity conditions given in Eqs. 4.3.16, 4.3.15. In particular, the results provide a pathway to choosing a set of λ_1 and λ_2 which appropriately add curvature in high-dimensions.

In the full GFGL model there are potentially $\mathcal{O}(p^2T)$ parameters, however, in the block-constant GGM there are only $\mathcal{O}((K+1)p^2)$. Under the correct identification of changepoints $\{\tau_1, \ldots, \tau_K\}$, the GFGL model has many precision matrices which are equal and can thus be compared to the underlying GGM model. For example, from the GFGL estimator with K changepoints one can identify $K+1$ precision matrices $\{\hat{\mathbf{\Theta}}^{(k)}\}_{k=1}^{K+1}$. These can then be compared to the GGM parameterisation $\{\mathbf{\Theta}_0^{(k)}\}_{k=1}^{K+1}$. Analysis of the curvature in the estimator can therefore be assessed over a model sub-space parameterised in terms of the true B block model. Concatenating the parameterisation across the B blocks gives rise to a block-diagonal matrix $\mathbf{\Theta}_0 \in \tilde{\mathbb{R}}^{(K+1)p \times (K+1)p}$ of the

form

$$\boldsymbol{\Theta} = \begin{pmatrix} \boldsymbol{\Theta}^{(1)} & & \\ & \ddots & \\ & & \boldsymbol{\Theta}^{(K+1)} \end{pmatrix} ,$$

where the block-diagonal sub-matrices are the precision matrices of individual blocks $\boldsymbol{\Theta}^{(k)} \in \mathbb{R}^{p\times p}$. If the changepoints were identified correctly such that $\hat{\mathcal{T}} = \mathcal{T}$, we may utilise the standard GFGL likelihood

$$L_B(\boldsymbol{\Theta}) = \sum_{k=1}^{K+1}(\tau_k - \tau_{k-1})\big(- \log\det(\boldsymbol{\Theta}^{(k)}) + \operatorname{trace}(\hat{\boldsymbol{S}}^{(k)}\boldsymbol{\Theta}^{(k)})\big) , \tag{4.4.1}$$

where $\hat{\boldsymbol{S}}^{(k)} = (\tau_k - \tau_{k-1})^{-1}(\sum_{t=\tau_{k-1}}^{\tau_k} \vec{X}_k^{(t)}(\vec{X}_k^{(t)})^\top)$ and $\vec{X}_k^{(t)} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_0^{(k)})$ for $t = \tau_{k-1}, \ldots, \tau_k$ and $k = 1, \ldots, K+1$. However, such a case is not guaranteed and in general the estimator will exhibit some form of mis-specification relating to estimation error in the changepoint positions. In this setting, it is inappropriate to analyse curvature (especially when considering sampling) with respect to the GFGL likelihood (4.4.1). Instead, we should consider the mis-specified likelihood:

$$L_{\hat{K}}(\boldsymbol{\Theta}) = \sum_{k=1}^{\hat{K}+1}\left((\hat{\tau}_k - \hat{\tau}_{k-1})(-\log\det(\boldsymbol{\Theta}^{(k)}) + \sum_{l=1}^{\hat{K}+1} n_{lk}\operatorname{tr}(\hat{\boldsymbol{S}}^{(k,l)}\boldsymbol{\Theta}^{(k)})\right) , \tag{4.4.2}$$

where $n_{lk} = \max\{\min\{\hat{\tau}_k, \tau_{l+1}\} - \max\{\hat{\tau}_{k-1}, \tau_l\}, 0\}$, this represents the proportion of ground-truth block l mixing with estimate block k, and empirical covariance:

$$\hat{\boldsymbol{S}}^{(k,l)} = \begin{cases} \sum_{t=\max\{\hat{\tau}_{k-1},\tau_l\}}^{\min\{\hat{\tau}_k,\tau_{l+1}\}} \left(\frac{\vec{X}_l^{(t)}(\vec{X}_l^{(t)})^\top}{\min\{\hat{\tau}_k,\tau_{l+1}\}-\max\{\hat{\tau}_{k-1},\tau_l\}}\right) & \text{if } l \in \mathcal{Q}^{(k)} := \{i \mid \hat{\tau}_{k-1} \le \tau_i \le \hat{\tau}_k\} \\ \mathbf{0} & \text{otherwise} \end{cases} .$$

Note: the role of the set $\mathcal{Q}^{(k)}$ in the above is to indicate changepoint indexes which need to be considered in the mis-specification for each estimated block $k = 1, \ldots, \hat{K}+1$. In this case $\hat{\boldsymbol{S}}^{(k,l)}$ describes the contribution of sampling from the ground-truth covariance into the mis-specified likelihood. Within the context of regularised M-estimation (Sec. 2.4 Def. 2.7) the GFGL regulariser

can be written in the form[5]

$$R(\boldsymbol{\Theta}) \equiv \left(\sum_{k=1}^{\hat{K}+1} \|\boldsymbol{\Theta}^{(k)}_{-ii}\|_1 + \frac{\lambda_2}{\lambda_1} \sum_{k=2}^{\hat{K}+1} \|\boldsymbol{\Theta}^{(k)} - \boldsymbol{\Theta}^{(k-1)}\|_F \right) .$$

Following the framework of Section. 2.4, let $\mathcal{S}^{(k)} = \{(i,j) \mid [\Theta_0^{(k)}]_{i,j} \neq 0 \; i,j \in [p]\}$ represent the support (non-zero elements) of $\boldsymbol{\Theta}_0^{(k)}$. Analogously, let $\mathcal{S}$ is the support of the full matrix $\boldsymbol{\Theta}_0 \in \tilde{\mathbb{R}}^{(K+1)p \times (K+1)p}$. If, in a simillar way to the lasso we construct meaningful model subspaces in the context of the true sparsity structure, then we can assess the decomposability properties of the GFGL estimator. For instance, in the block-wise parameterisation, one may be interested in defining the subspaces:

$$\mathcal{M} = \{\boldsymbol{\Theta} \in \tilde{\mathbb{R}}^{(K+1)p \times (K+1)p} \, | \Theta_{i,j} = 0 \, , (i,j) \notin \mathcal{S}\} \, ,$$

$$\mathcal{M}_\perp = \{\boldsymbol{\Theta} \in \tilde{\mathbb{R}}^{(K+1)p \times (K+1)p} \mid \Theta_{i,j} = 0 \, , (i,j) \in \mathcal{S}\} \, .$$

If the regulariser $R(\cdot)$ is decomposable over these subspaces, then from Lemma 2.4, conditional on $\lambda \geq 2R^*(\nabla L_{\hat{K}}(\boldsymbol{\Theta}_0; \vec{X}^{(1:T)}))$, the error $\hat{\boldsymbol{\Delta}} := \hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}_0$ belongs to the set

$$\mathbb{C}(\mathcal{M}, \bar{\mathcal{M}}^\perp; \boldsymbol{\Theta}_0) := \left\{ \boldsymbol{\Delta} \in \tilde{\mathbb{R}}^{Bp \times Bp} | \; R(\boldsymbol{\Delta}_{\mathcal{M}^\perp}) \leq 3R(\boldsymbol{\Delta}_{\mathcal{M}}) \right\} . \tag{4.4.3}$$

As demonstrated in Sec. 2.4, if $\hat{\boldsymbol{\Delta}} \in \mathbb{C}$ and the RSC condition (Def. 2.9) is met, then one may upper bound the estimation error $\|\hat{\boldsymbol{\Delta}}\|$. However, in order to ensure $\hat{\boldsymbol{\Delta}} \in \mathbb{C}$, one is required to set an appropriate regularisation, and therefore needs to examine the dual norm of the likelihood gradient $R^*(\nabla L_{\hat{K}}(\boldsymbol{\Theta}_0; \vec{X}^{(1:T)}))$. If this can be bounded by λ_1 in high probability for a given amount of data, then the estimation error $\|\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}_0\|$ may also be bounded. The specifics of such an analysis provide a direction for future work and mark a path towards obtaining bounds on the changepoint error even in high-dimensions.

[5]The total cost function can therefore be written in the form $L_{\hat{K}}(\boldsymbol{\Theta}) + \lambda_1 R(\boldsymbol{\Theta})$.

4.5 Summary

In this chapter the GFGL estimator was analysed for consistency properties, both in the standard and high-dimensional settings. In standard dimensions, changepoint consistency was demonstrated as long as $p < \delta_T T$. In order to extend consistency results to the high-dimensional setting, the GFGL estimator was then broken down as a M-estimator under the framework of Negahban et al. (2012) with the aim of bounding the estimation error $\|\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}_0\|$.

Traditionally, for example with the lasso, the loss-function does not have to account for mis-specification. One typically assumes that the data is identically drawn and there are no changepoints that need to be estimated. In the GFGL class of estimators one must not only characterise the curvature (in a number of directions identified by the model-subspace), but also consider this curvature in the presence of mis-specification of the likelihood (or loss function $L(\cdot)$). To my knowledge such a mis-specified likelihood has not to-date been analysed in terms of a high-dimensional M-estimator. It may appear that the stationarity condition induced bounds c.f. Eqs. 4.3.16 and 4.3.15 demonstrate sufficient curvature. However, this is not necessarily the case, as when operating in high-dimensions the loss function may be flat in many directions. This contrasts with the RSC requirements that sufficient curvature is present in certain directions, recall we desire $\delta L(\boldsymbol{\Delta}, \boldsymbol{\theta}_0) \geq \kappa_L \|\boldsymbol{\Delta}\|^2$ for all $\boldsymbol{\Delta} \in \mathbb{C}(\mathcal{M}, \bar{\mathcal{M}}^{\perp}; \boldsymbol{\theta}_0)$. Such specific curvature requirements place more stringent requirements on λ_1, λ_2, which require analysis of the mis-specified likelihood $L_{\hat{K}}(\cdot)$.

There are still several hurdles to overcome in order to obtain a high-dimensional bound and this provides an important direction for future work. In particular, it still remains to find probabilistic bounds on the expression $\lambda_1 \geq 2R^*(\nabla L_{\hat{K}}(\boldsymbol{\Theta}_0; \vec{X}^{(1:T)}))$, although the sampling bounds of Lemma C.5 provide a pathway to achieve this. If changepoint consistency can be demonstrated in high-dimensions (with asymptotics in p and T), the general mechanism of combining a mis-specified likelihood and the framework of Negahban et al. (2012) would provide a valuable tool for theoretical analysis of a very wide class of high-dimensional non-stationary models.

In the following chapters we take a step back from estimation theory and focus again on model construction. In particular, we will consider how one may describe non-stationary processes which are not just dependent across streams, but can also across time/space. The concepts of model sparsity and smoothness will again play a large role in estimation for the proposed models. It is plausible that the theoretical analysis developed in this chapter may one-day be extended to M-estimators for this larger class of models. More specifics of such an extension are discussed in the conclusion (Chapter 8).

Appendix C

C.1 Some known results

Tail bounds in standard-dimensional settings

Below are a collection of results from Wainwright (2009) who studies thresholding with the lasso. In the standard, or low-dimensional setting, where $p < \eta_{\min}/2$. These results can be used to bound the sampling error terms c.f. Eq. 4.3.5.

***Lemma* C.1.** *Concentration relative to Identity (Wainwright 2009)*

For $p \leq n$, let $\vec{\boldsymbol{Q}} \in \mathbb{R}^{n \times p}$ be a random matrix from the standard Gaussian ensemble, such that $Q_{ij} \overset{iid}{\sim} \mathcal{N}(0,1)$, then for all $\sigma > 0$, we have:

$$P\left[\left|\left|\left|\frac{1}{n}\vec{\boldsymbol{Q}}^{\top}\vec{\boldsymbol{Q}} - \boldsymbol{I}_{p\times p}\right|\right|\right|_2 \geq \delta(n,p,\sigma)\right] \leq 2\exp(-n\sigma^2/2) \ ,$$

where

$$\delta(n,p,\sigma) := 2\left(\sqrt{\frac{p}{n}} + \sigma\right) + \left(\sqrt{\frac{p}{n}} + \sigma\right)^2 \ .$$

Proof. For proof see Davidson et al. (2001).

***Lemma* C.2.** *Concentration of spectral norm and eigenvalues (Wainwright 2009)*

For $p \leq n$ let $\vec{\boldsymbol{Z}} \in \mathbb{R}^{n \times p}$ have i.i.d rows such that the ith row is generated as $\vec{Z}^{(i)} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ for $i = 1, \ldots, n$. Let $\hat{\boldsymbol{\Sigma}} := n^{-1}\vec{\boldsymbol{Z}}^{\top}\vec{\boldsymbol{Z}}$ be the empirical covariance matrix

(1) If the covariance matrix $\boldsymbol{\Sigma}$ has maximum eigenvalues $\phi_{\max} < +\infty$, then for all $\sigma > 0$

$$P\left[\left|\left|\left|\hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\right|\right|\right|_2 \geq \phi_{\max}\delta(n,p,\sigma)\right] \leq 2\exp(-n\sigma^2/2)$$

(2) If the matrix $\boldsymbol{\Sigma}$ has minimum eigenvalue $\phi_{\min} > 0$, then for all $\sigma > 0$

$$P\left[\left|\left|\left|\hat{\boldsymbol{\Sigma}}^{-1} - \boldsymbol{\Sigma}^{-1}\right|\right|\right|_2 \geq \frac{\delta(n,p,\sigma)}{\phi_{\min}}\right] \leq 2\exp(-n\sigma^2/2)$$

Proof. The proof follows from Lemma C.1 setting $\vec{\boldsymbol{Z}} = \vec{\boldsymbol{Q}}\sqrt{\boldsymbol{\Sigma}}$ where $\vec{\boldsymbol{Z}}$ and $\vec{\boldsymbol{Q}}$ are random matrices. See Wainwright (2009) Lemma 9.

High dimensional results

The following Lemmata are derived from results in Ravikumar, Wainwright, Raskutti, et al. (2011). They may be used to bound the deviation of the sample covariance matrix in the high-dimensional setting. The result below gives a rate in the case of sub-Gaussian tails; however, it is also possible to construct polynomial tail bounds where the variables moments are bounded.

***Lemma* C.3.** *Lemma 1 (Ravikumar, Wainwright, Raskutti, et al. 2011)*

Consider a zero-mean random vector $\vec{Z} \in \mathbb{R}^p$ with covariance $\boldsymbol{\Sigma}$ such that each $Z_i/\sqrt{\Sigma_{ii}}$ is sub-Gaussian with parameter ζ. Given n i.i.d samples let the associated sample covariance be denoted $\hat{\boldsymbol{\Sigma}}$. The absolute error then satisfies the bound

$$P[|\hat{\Sigma}_{i,j} - \Sigma_{i,j}| > \delta] \leq 1/f(n,\delta) \;, \tag{C.1}$$

where

$$f(n,\delta) = \frac{1}{4}\exp(a_0 n\delta^2) \quad \text{and} \quad a_0 = (128(1+4\zeta^2)^2)^{-1} \;,$$

for all $\delta \in (0, 8(1+4\zeta^2))$ and assuming $\max_i(\Sigma_{ii}) = 1$.

While the above tail bound provides constraints on the deviation of individual entries in the sample covariance, we are more generally interested in the error across the whole matrix. For example, this is required to bound Eq. 4.3.5. The following lemma provides a maximum error estimate which we can then use to bound the required Frobenius norm.

***Lemma* C.4.** *Lemma 8 (Ravikumar, Wainwright, Raskutti, et al. 2011)*

Let $\boldsymbol{W} = \hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}^0$. For any $\tau > 2$ and sample size n such that $\bar{\delta}_f(n, p^\tau) \leq 1/v_0$ we have

$$P[\|\boldsymbol{W}\|_\infty \geq \bar{\delta}_f(n,p^\tau)] \leq p^{2-\tau} \to 0 \;,$$

where $\bar{\delta}_f(n,p^\tau) := \arg\max\{\delta \mid f(n,\delta) \leq p^\tau\}$.

Proof. Consider the sub-Gaussian exponential tail conditions in Lemma C.3 where $\delta \in (0, v_0^{-1}]$ and $v_0^{-1} = 8(1+4\zeta^2)\max_i(\Sigma_{ii})$. Applying the union bound we obtain

$$P[\max_{i,j}|W_{i,j}| \geq \delta] \leq p^2/f(n,\delta)\ .$$

Setting $\delta = \bar{\delta}_f(n,p^\tau)$ gives $P[\max_{i,j}|W_{i,j}| \geq \bar{\delta}_f(n,p)] \leq p^2/f(n,\bar{\delta}_f(n,p^\tau)) = p^{2-\tau}$ as $f(n,\bar{\delta}_f(n,p^\tau)) = p^\tau$ through definition.

□

***Lemma* C.5.** *Sample Error in Frobenius Norm (HD)*

In the high-dimensional setting where it is possible $p \geq n$ the sampling term is bounded according to Ravikumar, Wainwright, Raskutti, et al. (2011) (Lemmas C.3 and C.4). Given the definition of the sampling error in Eq. 4.3.8, we can then use Lemma C.3 to bound the Frobenius norm by noting that:

$$\begin{aligned} P[\|\vec{\boldsymbol{W}}_k^{(n)}\|_F \geq \epsilon] &\leq p^2 P[|[\vec{W}_k^{(n)}]_{i,j}| \geq p^{-1}\epsilon] \\ &\leq p^2 4\exp\left\{-\frac{n\epsilon^2}{2a_k^2p^2}\right\}, \end{aligned}$$

where $a_k = 8(1+4\zeta_k^2)$ and $p^{-1}\epsilon \in (0, a_k)$.

C.2 Proof of Lemma 4.1 (Optimality Conditions)

Proof. Stationarity conditions for GFGL

In GFGL we have a set of conditions for each time-point which must be met jointly. Unlike non-fused estimators, we also have to consider the stationarity conditions due to a differenced term. The GFGL objective can then be re-written in terms of this difference, such that $\{\hat{\boldsymbol{\Gamma}}^{(t)}\}_{t\in[T]} = \arg\min_{\{\boldsymbol{\Gamma}^{(t)}\}\in\mathbb{R}^{p\times p}}\{\mathcal{G}(\hat{\boldsymbol{S}},\boldsymbol{\Gamma})\}$ where

$$\mathcal{G}(\hat{\boldsymbol{S}},\boldsymbol{\Gamma}) := \sum_{t=1}^{T}\left(-\log\det(\sum_{s\leq t}\boldsymbol{\Gamma}^{(s)}) + \mathrm{tr}(\hat{\boldsymbol{S}}^{(t)}\sum_{s\leq t}\boldsymbol{\Gamma}^{(s)})\right) + \lambda_1\sum_{t=1}^{T}\|\sum_{s\leq t}\boldsymbol{\Gamma}^{(s)}_{\setminus ii}\|_1 + \lambda_2\sum_{t=2}^{T}\|\boldsymbol{\Gamma}^{(t)}\|_F\ .$$

For each point $l \in [T]$ the derivative when taken with respect to $\boldsymbol{\Gamma}^{(l)}$ evaluated at the point $\{\hat{\boldsymbol{\Gamma}}^{(t)}\}_{t\in[T]}$ gives a zero matrix $\mathbf{0} \in \mathbb{R}^{p\times p}$. For the log-det term, we

have $F(\mathbf{\Gamma}) := -\log\det(\sum_{s\leq t}\mathbf{\Gamma}^{(s)})$, letting $\mathbf{\Theta}^{(t)} = \sum_{s\leq t}\mathbf{\Gamma}^{(s)}$,

$$\frac{\partial}{\partial \mathbf{\Gamma}^{(l)}}F(\mathbf{\Gamma}) = \begin{cases} -(\mathbf{\Theta}^{(t)})^{-1} & \text{if } t \geq l \\ 0 & \text{otherwise} \end{cases} .$$

This follows from the chain rule on $g(\mathbf{\Theta}^{(t)}) = -\log\det(\mathbf{\Theta}^{(t)})$ and $(\partial/\partial\mathbf{\Gamma}^{(l)})g(\mathbf{\Theta}^{(t)}) = (\partial/\partial\mathbf{\Theta}^{(t)})g(\mathbf{\Theta}^{(t)}) \times (\partial/\partial\mathbf{\Gamma}^{(l)})\mathbf{\Theta}^{(t)}$. The derivative of the first sum of log-det components is given as

$$\frac{\partial}{\partial \mathbf{\Gamma}^{(l)}}\sum_{t=1}^{T}\Big(-\log\det(\sum_{s\leq t}\mathbf{\Gamma}^{(s)})\Big) = \sum_{t=l}^{T} -(\sum_{s\leq t}\mathbf{\Gamma}^{(s)})^{-1} .$$

The trace term is much simpler, as the trace operation can be linearly separated i.e. $\mathrm{tr}(\boldsymbol{A}+\boldsymbol{B}) = \mathrm{tr}(\boldsymbol{A}) + \mathrm{tr}(\boldsymbol{B})$, such that

$$\frac{\partial}{\partial \mathbf{\Gamma}^{(l)}}\sum_{t=1}^{T}\Big(\mathrm{tr}(\hat{\boldsymbol{S}}^{(t)}\sum_{s\leq t}\mathbf{\Gamma}^{(s)})\Big) \;=\; \sum_{t=l}^{T}\frac{\partial}{\partial \mathbf{\Gamma}^{(l)}}\mathrm{tr}(\hat{\boldsymbol{S}}^{(t)}\sum_{s\leq t}\mathbf{\Gamma}^{(s)}) = \sum_{t=l}^{T}\hat{\boldsymbol{S}}^{(t)} .$$

Setting the derivative to zero we obtain:

$$\mathbf{0} = \sum_{t=l}^{T}\Big(-(\sum_{s\leq t}\hat{\mathbf{\Gamma}}^{(s)})^{-1} + \hat{\boldsymbol{S}}^{(t)}\Big) + \lambda_1\sum_{t=l}^{T}\hat{\boldsymbol{R}}_1^{(t)} + \lambda_2\hat{\boldsymbol{R}}_2^{(l)} .$$

We now need a way to link the observations (via $\hat{\boldsymbol{S}}^{(t)}$) to the optimality conditions of the GFGL problem. Recalling that $\vec{X}^{(t)} \sim \mathcal{N}_p(\mathbf{0}, \mathbf{\Sigma}^{(t)})$, we can then link $\mathbf{\Sigma}^{(t)}$ to $\mathbf{\Gamma}^{(l)}$ via $\mathbf{\Sigma}^{(t)} = (\sum_{s\leq t}\mathbf{\Gamma}^{(s)})^{-1}$. We now have a way to relate the estimated precision matrices $\hat{\mathbf{\Gamma}}^{(t)}$ and the corresponding ground-truth. In the following, we will consider the deviation of the empirical covariance from the ground-truth, with a noise term given as $\vec{\mathbf{\Psi}}^{(t)} := \hat{\boldsymbol{S}}^{(t)} - \mathbf{\Sigma}^{(t)}$. Substituting $\mathbf{\Delta}^{(t)}$ into the above stationarity conditions for GFGL we obtain;

$$\sum_{t=l}^{T}\Big((\sum_{s\leq t}\mathbf{\Gamma}^{(s)})^{-1} - (\sum_{s\leq t}\hat{\mathbf{\Gamma}}^{(s)})^{-1}\Big) - \sum_{t=l}^{T}\vec{\mathbf{\Psi}}^{(t)} + \lambda_1\sum_{t=l}^{T}\hat{\boldsymbol{R}}_1^{(t)} + \lambda_2\hat{\boldsymbol{R}}_2^{(l)} ,$$

or equivalently:

$$\sum_{t=l}^{T}\Big((\mathbf{\Theta}^{(t)})^{-1} - (\hat{\mathbf{\Theta}}^{(t)})^{-1}\Big) - \sum_{t=l}^{T}\vec{\mathbf{\Psi}}^{(t)} + \lambda_1\sum_{t=l}^{T}\hat{\boldsymbol{R}}_1^{(t)} + \lambda_2\hat{\boldsymbol{R}}_2^{(l)} ,$$

thus obtaining the result in Lemma 4.1. □

C.3 Algebraic manipulation

This appendix contains detail on the algebraic steps for some of the manipulation. These are not listed as formal lemmata, but are provided to help the reader with replicating the analysis. The notes below should be considered in the context of the main text.

Differencing stationarity conditions (Eq. 4.3.1)

$$\|\sum_{t=\hat{\tau}_k}^{\tau_k-1}\left((\boldsymbol{\Theta}^{(t)})^{-1}-(\hat{\boldsymbol{\Theta}}^{(t)})^{-1}\right)-\sum_{t=\hat{\tau}_k}^{\tau_k-1}\vec{\boldsymbol{\Psi}}^{(t)}+\lambda_1\sum_{t=\hat{\tau}}^{\tau_k-1}\hat{\boldsymbol{R}}_1^{(t)}\|_F\leq 2\lambda_2\ .$$

The algebra required for the above step is as follows: Let $\boldsymbol{a}^{(\tau)}=\gamma\boldsymbol{b}^{(\tau)}/\|\boldsymbol{b}^{(\tau)}\|$, $\boldsymbol{a}^{(\hat{\tau})}=\gamma\boldsymbol{b}^{(\hat{\tau})}/\|\boldsymbol{b}^{(\hat{\tau})}\|$, it follows $\|\boldsymbol{a}^{(\tau)}\|=\gamma$ and $\|\boldsymbol{a}^{(\tau)}\|+\|\boldsymbol{a}^{(\hat{\tau})}\|=2\gamma \implies \|\boldsymbol{a}^{(\tau)}\|^2+\|\boldsymbol{a}^{(\hat{\tau})}\|^2+2\|\boldsymbol{a}^{(\tau)}\|\|\boldsymbol{a}^{(\hat{\tau})}\|=4\gamma^2$. Consider $\|\boldsymbol{a}^{(\tau)}-\boldsymbol{a}^{(\hat{\tau})}\|^2=\|\boldsymbol{a}^{(\tau)}\|^2-2\langle\boldsymbol{a}^{(\tau)},\boldsymbol{a}^{(\hat{\tau})}\rangle+\|\boldsymbol{a}^{(\hat{\tau})}\|^2$. Substituting for $\|\boldsymbol{a}^{(\tau)}\|^2+\|\boldsymbol{a}^{(\hat{\tau})}\|^2$ we find $\|\boldsymbol{a}^{(\tau)}-\boldsymbol{a}^{(\hat{\tau})}\|^2=4\gamma^2-2(\|\boldsymbol{a}^{(\tau)}\|\|\boldsymbol{a}^{(\hat{\tau})}\|+\langle\boldsymbol{a}^{(\tau)},\boldsymbol{a}^{(\hat{\tau})}\rangle) \implies \|\boldsymbol{a}^{(\tau)}-\boldsymbol{a}^{(\hat{\tau})}\|\leq 2\gamma$. The above result follows from this setting $\boldsymbol{a}^{(\tau_k)}\equiv\sum_{t=\tau_k}^{T}(\boldsymbol{\Theta}^{(t)})^{-1}-(\hat{\boldsymbol{\Theta}}^{(t)})^{-1}-\vec{\boldsymbol{\Psi}}^{(t)}+\lambda_1\hat{\boldsymbol{R}}_1^{(t)}$ and $\gamma=\lambda_2$.

Block-wise condition (Eq. 4.3.3)

Here, it is desired to replace the time indexed inverse precision matrices $(\boldsymbol{\Theta}^{(t)})^{-1}$ with the block-covariance matrices indexed $\boldsymbol{\Sigma}^{(k)}$ and $\boldsymbol{\Sigma}^{(k+1)}$. We can lower bound the difference in precision matrices as the sum of a difference between true values before τ_k , i.e. $\boldsymbol{\Sigma}^{(k+1)}-\boldsymbol{\Sigma}^{(k)}$ and the difference between the next (k+1st) true block and estimated block, i.e. $\hat{\boldsymbol{\Sigma}}^{(k+1)}-\boldsymbol{\Sigma}^{(k+1)}$. Consider the triangle inequality

$$\|\boldsymbol{\Sigma}^{(k)}-\hat{\boldsymbol{\Sigma}}^{(k+1)}\|_F+\|\hat{\boldsymbol{\Sigma}}^{(k+1)}-\boldsymbol{\Sigma}^{(k+1)}\|_F\geq\|\boldsymbol{\Sigma}^{(k)}-\boldsymbol{\Sigma}^{(k+1)}\|_F\ .$$

In the setting where $\tau_k<\hat{\tau}_{k+1}$ and $\tau_{k-1}<\hat{\tau}_k$ (this occurs with probability one conditional on C_T) we have from the above

(C.2)

$$\|\sum_{t=\hat{\tau}_k}^{\tau_k-1}(\boldsymbol{\Sigma}^{(t)}-\hat{\boldsymbol{\Sigma}}^{(t)})\|_F=(\tau_k-\hat{\tau}_k)\|\boldsymbol{\Sigma}^{(k)}-\hat{\boldsymbol{\Sigma}}^{(k+1)}\|_F$$

$$\geq (\tau_k - \hat{\tau}_k)\big(\|\boldsymbol{\Sigma}^{(k)} - \boldsymbol{\Sigma}^{(k+1)}\|_F - \|\hat{\boldsymbol{\Sigma}}^{(k+1)} - \boldsymbol{\Sigma}^{(k+1)}\|_F\big) \ .$$

C.4 Proof of Lemma 4.1

Lemma. *In standard dimensions $p < \sqrt{d_{\min}}$ we have*

$$P[\|\vec{\boldsymbol{W}}_k^{(\bar{\tau}_k - \tau_k)}\|_F > \eta_{\min}/3] \leq 2\exp\left\{-\eta_{\min}\sqrt{d_{\min}}/24\sqrt{2}\phi_{\max}\right\} \ .$$

Proof. Consider that

$$\begin{aligned} P[\|\vec{\boldsymbol{W}}_k^{(n)}\|_F > \epsilon] &\leq P\left[\sqrt{r}\left|\left|\left|\vec{\boldsymbol{W}}_k^{(n)}\right|\right|\right|_2 > \epsilon\right] \\ &= P[\left|\left|\left|\vec{\boldsymbol{W}}_k^{(n)}\right|\right|\right|_2 > \epsilon r^{-1/2}] \ , \end{aligned}$$

where $r = \text{rank}(\vec{\boldsymbol{W}}_k^{(n)}) \leq p$. Furthermore, consider using Lemma C.2, with the specific setting of the tail conditions such that $\delta(n,p,\sigma) = 2(\sqrt{p/n}+\sigma) + (\sqrt{p/n}+\sigma)^2$ and $\sigma = \sqrt{p/n}$ thus $\delta(n,p,\sqrt{p/n}) \leq 8\sqrt{p/n}$ thus

$$(\text{ C.3}) \qquad P\left[\left|\left|\left|\vec{\boldsymbol{W}}_k^{(n)}\right|\right|\right|_2 > 8\phi_{\max}\sqrt{p/n}\right] \leq 2\exp(-p/2) \ .$$

Let $8\phi_{\max}\sqrt{p/n} = \epsilon p^{-1/2}$, then one obtains $p = \epsilon\sqrt{n}/8\phi_{\max}$, substituting $\epsilon = \eta_{\min}/3$ and $n = \bar{\tau}_k - \tau_k > d_{\min}/2$, thus we obtain

$$P[\|\vec{\boldsymbol{W}}_k^{(\bar{\tau}_k - \tau_k)}\|_F > \eta_{\min}/3] \leq 2\exp\left\{-\frac{\eta_{\min}\sqrt{d_{\min}}}{24\phi_{\max}}\right\} \ .$$

□

C.5 Proof of Lemma 4.2

Lemma. *The probability of the intersection of $A_{T,k}$ and $D_T^{(m)}$ can be bounded from above by considering the events*

$$\begin{aligned} E_k^{'} &:= \{(\hat{\tau}_{k+1} - \tau_k) \geq d_{\min}/2\} \ , \\ E_k^{''} &:= \{(\tau_k - \hat{\tau}_k) \geq d_{\min}/2\} \ . \end{aligned}$$

In particular, one can demonstrate that:

$$(\text{ C.4}) \quad P[A_{T,k} \cap D_T^{(m)}] \leq P[A_{T,k} \cap E_k^{'} \cap D_T^{(m)}] + \sum_{j=k+1}^{K} P[E_j^{''} \cap E_j^{'} \cap D_T^{(m)}] \ .$$

Proof. The strategy is to expand the probability in terms of exhaustive events (relating to the estimated changepoint positions), under a symmetry argument, we assume $\hat{\tau}_k < \tau_k$. Noting $P[E_k^{'} \cup E_{k+1}^{''}] = 1$, then expanding the original event, we find

$$\begin{aligned} P[A_{T,k} \cap D_T^{(m)}] &\leq P[A_{T,k} \cap D_T^{(m)} \cap E_k^{'}] + P[A_{T,k} \cap D_T^{(m)} \cap E_{k+1}^{''}] \\ &\leq P[A_{T,k} \cap D_T^{(m)} \cap E_k^{'}] + P[D_T^{(m)} \cap E_{k+1}^{''}] \, . \end{aligned}$$

Now consider the event $D_T^{(m)} \cap E_{k+1}^{''}$ corresponding to the second term. One can then expand the probability of this intersection over the events $E_{k+1}^{'}$ and $E_{k+2}^{''}$ relating to the next changepoint, i.e

$$P[D_T^{(m)} \cap E_k^{''}] \leq P[D_T^{(m)} \cap E_{k+1}^{''} \cap E_{k+1}^{'}] + P[D_T^{(m)} \cap E_{k+1}^{''} \cap E_{k+2}^{''}] \, .$$

Again we note that $P[D_T^{(m)} \cap E_k^{''} \cap E_{k+2}^{''}]$ is upper bounded by $P[D_T^{(m)} \cap E_{k+2}^{''}]$ such that $P[D_T^{(m)} \cap E_k^{''} \cap E_{k+2}^{''}] \leq P[D_T^{(m)} \cap E_{k+2}^{''}]$. Cascading this over all changepoints $j = k+1, \ldots, K$ we have

$$P[D_T^{(m)} \cap E_k^{''}] \leq \sum_{j=k+1}^{K} P[D_T^{(m)} \cap E_j^{''} \cap E_{j+1}^{'}] \, .$$

□

C.6 Proof of Lemma 4.3

Lemma. *The probability of $D_T^{(l)}$ is bounded by*

$$P[D_T^{(l)}] \leq 2^K \sum_{k=1}^{K-1} \sum_{l \geq k}^{K-1} P[E_l^{''} \cap E_l^{'}] + 2^K P[E_K^{'}] \, .$$

Proof. Recall the definitions of the different events:

$$E_k^{'} := \{(\hat{\tau}_{k+1} - \tau_k) \geq \frac{d_{\min}}{2}\} \quad \text{and} \quad E_k^{''} := \{(\tau_k - \hat{\tau}_k) \geq \frac{d_{\min}}{2}\} \, .$$

For each new changepoint in the model, there is an extra option for this (latest changepoint) to trigger the event

$$\{\exists k \in [K],\ \hat{\tau}_k \leq \tau_{k-1}\} \, . \tag{C.5}$$

In particular, the total number of combinations (of changepoints) which could trigger this event doubles on the addition of an extra changepoint. Lemma 4.3 considers the probability of each of the changepoints being estimated to the left of τ_{k-1}. To start, we note that the probability of $D_T^{(l)}$ is bounded by

$$P[D_T^{(l)}] \leq \sum_{k=1}^{K} 2^{k-1} P[\max\{l \in [K] \,|\, \hat{\tau}_l \leq \tau_{l-1}\} = k] \,. \tag{C.6}$$

The term $P[\max\{l \in [K] \,|\, \hat{\tau}_l \leq \tau_{l-1}\} = k]$ describes the probability that the last changepoint (such that $\hat{\tau}_l$ is to the left, i.e. before τ_{l-1}) is k. On increasing k by one (for $k \geq 2$), the number of combinations of left/right estimates for previous changepoints doubles. For example, consider the case for $k = 3$ such that the event $S_3 := \{\hat{\tau}_3 \leq \tau_2\}$ is triggered, see Figure C.1. The possible results for previous changepoints are then $S_2 := \{\hat{\tau}_2 \leq \tau_1\}$, it's compliment S_2^c, and the event $S_1 := \{\hat{\tau}_1 \leq 1\}$ or S_1^c. In total, there are 2^2 ways that the event S_3 can occur[6]. In general for the changepoint k there are 2^{k-1} combinations of events that allow S_k to be triggered. However, since these events are not mutually exclusive, this only provides an upper bound.

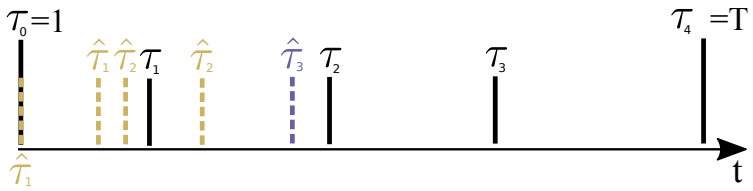


Figure C.1 – The gold changepoint estimates indicate examples of allowable positions for the changepoints $l < k = 3$ which satisfy $\{\hat{\tau}_l \leq \tau_{l-1}\}$. Note: for the case displayed $K = 3$ and $k = 3$ thus there are 4 combinations of changepoints (in gold) that permit the purple event $\max\{l \in [4] \,|\, \hat{\tau}_l \leq \tau_{l-1}\} = 3$.

Harchaoui et al. (2010) and Kolar et al. (2012) note that an event where the kth changepoint is the largest to satisfy $\{\hat{\tau}_l \leq \tau_{l-1}\}$, is a subset of events relating to later changepoints $l \geq k$. Correspondingly, we have

$$\{\max\{l \in [K] \,|\, \hat{\tau}_l \leq \tau_{l-1}\} = k\} \subseteq \cup_{l=k}^{K} \{\tau_l - \hat{\tau}_l \geq d_{\min}/2\} \cap \{\hat{\tau}_{l+1} - \tau_l \geq d_{\min}/2\} \,. \tag{C.7}$$

[6]Arguably, there are actually 3 combinations of changepoint event that can cause S_3 as $\hat{\tau}_1 > \hat{\tau}_0 = 1$ by definition. However, this does not effect the upper bound.

The union bound applied to (C.7) provides us with the bound:

$$P[\max\{l \in [K] \,|\, \hat{\tau}_l \leq \tau_{l-1}\} = k] \leq \sum_{l \geq k} P[\{\tau_l - \hat{\tau}_l \geq \frac{d_{\min}}{2}\} \cap \{\hat{\tau}_{l+1} - \tau_l \geq \frac{d_{\min}}{2}\}] \,,$$

and thus

$$P[D_T^{(l)}] \leq \sum_{k=1}^{K} 2^{k-1} \sum_{l \geq k}^{K} P[\{\tau_l - \hat{\tau}_l \geq \frac{d_{\min}}{2}\} \cap \{\hat{\tau}_{l+1} - \tau_l \geq \frac{d_{\min}}{2}\}] \,.$$

Since we want an upper bound, the largest factor (2^K) can be taken out the summation. The term $k = K$ contains the event $\{\hat{\tau}_{K+1} - \tau_K \geq d_{\min}/2\}$, this occurs with probability one as the last changepoint $\hat{\tau}_{K+1} = T + 1$. We can thus truncate the final term and obtain the bound:

$$\begin{aligned} P[D_T^{(l)}] \;\; \leq \;\; & 2^K \sum_{k=1}^{K-1} \sum_{l \geq k}^{K-1} P[\{\tau_l - \hat{\tau}_l \geq \frac{d_{\min}}{2}\} \cap \{\hat{\tau}_{l+1} - \tau_l \geq \frac{d_{\min}}{2}\}] \\ & + 2^K P[\{\tau_K - \hat{\tau}_K \geq \frac{d_{\min}}{2}\}] \,. \end{aligned}$$

The above can be written in a shortened form by relating it to the events $E_k^{'}, E_k^{''}$ defined in (4.3.13), such that

$$P[D_T^{(l)}] \leq 2^K \sum_{k=1}^{K-1} \sum_{l \geq k}^{K-1} P[E_l^{''} \cap E_l^{'}] + 2^K \mathbb{P}[E_K^{''}] \,.$$

□

Chapter 5

Locally Stationary Wavelet (LSW) Processes

In previous chapters we discussed how one may relax the requirement that statistical models are sampled identically across time. However, to enable estimation, we were required to assume that the models met various smoothness constraints, for example corresponding to continuous, piecewise, or grouped variation. While these models allow dynamics, they assume that draws at different time-points are independent. In reality, we see large correlations between observations that are taken nearby in time or space; for example, levels of crime may be similar in nearby districts (Bowers et al. 2003), or stock prices may be correlated to lagged prices nearby in time (Cont 2005). Such dependency between observations has not yet been encoded in our models.

In the following sections, we aim to extend some of the previously discussed regularised estimation ideas to a class of non-stationary models which can describe both the dynamic and dependent nature of time-series. To describe dependency across time we need a way of representing a signal that can account for variation across segments, as opposed to individual elements of the time-series. To this end, it is useful to discuss the representation of signals in terms of their projection onto a set of basis functions. For example, it is often considered to model the Fourier coefficients of a signal. Such a modelling

approach is commonly referred to as spectral analysis (Brillinger 1981; Priestly 1981).

In this section, we first discuss how Fourier analysis can enable us to describe identical, but dependent signals. However, to describe non-stationary processes we are required to develop time localised representations of the process. Two such approaches are introduced, namely, the *evolutionary Fourier process*, and the *locally-stationary wavelet (LSW) process*. In particular, the LSW model has received much recent attention in the literature due to both its statistical tractability and modelling flexibility. In this chapter, we discuss recent results for LSW processes and the estimation of model parameters. The proceeding chapters, several M-estimators are introduced to help enforce assumptions on the model parameters of LSW processes. Notably, regularised LSW model estimation is demonstrated in both one-dimensional time-series, and multi-dimensional image processing settings. Regularised estimation of the LSW spectrum is shown to enhance the interpretation of spectral estimates, while also increasing model robustness. The final chapter extends the LSW model to a multivariate setting. Importantly, this provides a connection with earlier chapters where one may apply regularised graph identification methods for spectral estimation.

5.1 Evolutionary Fourier Processes

In order to motivate and understand non-stationary time-series models, it is prudent to first consider the construction of stationary processes. Specifically, we will be concerned with second-order stationarity, sometimes known as weak-stationarity this requires that a stochastic process $\{X_t\}$ has mean and covariance properties that are shift invariant $E[X_t] = E[X_{t'}]$ and $\text{Cov}[X_t, X_{t+\tau}] = \text{Cov}[X_{t'}, X_{t'+\tau}]$ for all $t, t', \tau \in \mathbb{Z}$. Although not studied here, one should note that in addition to second-order stationarity it is possible to consider higher-order measures of stationarity. For example, Priestly (1981) considers stationarity of higher order moments, Brillinger (1981) considers stationarity in terms of the cumulants of a process.

5.1.1 Spectral Representation of Processes

A fundamental description of stationary processes can be considered via the *Cramér representation*:

***Proposition* 5.1.** *Cramér Representation of second order-stationary processes*

All zero-mean second order stationary processes $\{X_t\}$*, for* $t \in \mathbb{Z}$*, may be written in the form*

$$X_t = \int_{-\pi}^{\pi} A(\omega) \exp(i\omega t) dZ(\omega), \quad t \in \mathbb{Z} \,, \tag{5.1.1}$$

where $A(\omega)$ *is the amplitude of the process in the spectral domain (denoted by* frequency *ω), and* $dZ(\omega)$ *is an orthonormal increment process such that* $E[dZ(\omega_1)\overline{dZ(\omega_2)}] = E[|dZ(\omega_1)|^2]\delta(\omega_1 - \omega_2)$.

Let us now consider how the amplitude of the spectra $A(\omega)$ is related to the auto-covariance $c_X(\tau) := \text{Cov}[X_t, X_{t+\tau}]$. We find[1]:

$$\begin{aligned} c_X(\tau) &= \int_{-\pi}^{\pi} A(\omega_1)A(-\omega_2) \exp(it\omega_1 - i(t+\tau)\omega_2) E[dZ(\omega_1)\overline{dZ}(\omega_2)] \\ &= \int_{-\pi}^{\pi} |A(\omega)|^2 \exp(-i\tau\omega) E[|dZ(\omega)|^2] \,, \end{aligned}$$

where the second line comes from considering that $E[dZ(\omega_1)\overline{dZ}(\omega_2)] = \delta(\omega_1 - \omega_2)$. Taking the final expectation, we obtain

$$c_X(\tau) = 2\pi\sigma^2 \int_{-\pi}^{\pi} |A(\omega)|^2 \exp(-i\tau\omega) d\omega \tag{5.1.2}$$

where $\sigma^2 = E[\epsilon_\tau^2]$ is the variance of the integral of the increment process $\epsilon_\tau = (1/2\pi)\int_0^{2\pi} \exp(it\omega) dZ(\omega)$. It therefore follows that the second order properties of the process $c_X(\tau)$ are directly specified through the spectral amplitude. When $\tau = 0$ the variance of (5.1.2) can be written as

$$\text{Var}[X_t] = \int_{-\pi}^{\pi} dH(\omega) \,, \tag{5.1.3}$$

where $dH(\omega) = |A(\omega)|^2 d\mu(\omega)$ is referred to the *power-spectrum* of the process; as the process is stationary, this does not depend on time.

[1]For simplicity, assume the process has zero mean such that $\text{Cov}[X_t, X_{t+\tau}] = E[X_t X_{t+\tau}]$.

Given the wide array of signals which appear to behave in non-stationary manner, much work has been put into extending these models. However, while the Cramêr representation allows for us to describe any stationary process, there is not a unique way to generalise to non-stationary processes. In the Cramêr representation the exponential basis $\{\exp(i\omega t)\}$ is utilised. It turns out, that if one uses the exponential basis alongside an orthogonal increment process to represent $\{X_t\}$ or covariance $\{c_X(\tau)\}$, then the resultant process will be second-order stationary. To observe this, one can consider expanding the process $\{X_t\}$ in terms of an expansion of more general orthogonal functions $\{\phi_t(\cdot)\}$:

***Proposition* 5.2.** *General Orthogonal Expansion c.f. Priestly (1981)*

Let $\{X_t\}$ be a zero-mean, not necessarily stationary process and $\{\phi_t(\omega)\}$ be a family of functions that are quadratically integrable with respect to a measure $\mu(\omega)$ defined on the real line:

$$\int_{-\pi}^{\pi} |\phi_t(\omega)|^2 d\mu(\omega) < \infty \ , \quad \text{for all } t \ . \tag{5.1.4}$$

If for all t, t' the auto-covariance admits a representation of the form $\mathrm{Cov}[X_t, X_{t'}] = \int_{-\pi}^{\pi} \overline{\phi}_t(\omega)\phi_{t'}(\omega)d\mu(\omega)$, then the resulting process admits a representation of the form

$$X_t = \int_{-\pi}^{\pi} \phi_t(\omega) dZ(\omega) \ ,$$

where $\{Z(\omega)\}$ is an orthogonal process with $E[|dZ(\omega)|^2] = d\mu(\omega)$.

If the covariance of the process only depends on lag, i.e. $c_X(t-t') = c_X(\tau)$, then via Bochner's theorem there exists a unique positive measure $\mu(\omega) \in [0,1]$ such that $c_X(\tau) = \int \exp(-2\pi i\tau\omega)d\mu(\omega)$. The representation (5.1.4) is now based on the exponential family $\{\phi_t(\omega)\} = \{\exp(i\omega t)\}$ and thus a representation of the process in the form $X_t = \int_{-\pi}^{\pi} \exp(i\omega t)dZ(\omega)$ is guaranteed (Priestly 1981). As such, the traditional family of complex exponential functions as used in the Cramêr representation are not expressive enough to deal with non-stationarity.

5.1.2 Oscillatory Processes

The idea of *oscillatory processes* as introduced in (Priestly 1981), are an early example of work where different functional forms for $\{\phi_t(\omega)\}$ are considered. Specifically, the oscillatory process lets $A_t(\omega)$ vary slowly over time to enable dynamics in the spectra. Consider the resultant family of basis function $\phi_t(\omega) = A_t(\omega)e^{i\omega t}$. Now, for a given family of oscillatory functions $\{\phi_t(\omega)\}$ an *evolutionary power spectrum* can be defined at time t as

$$dH_t(\omega) = |A_t(\omega)|^2 d\mu(\omega) \ .$$

We note, that from Prop. 5.2 we have $\text{Var}[X_t] = \int_{-\pi}^{\pi} |A_t(\omega)|^2 d\mu(\omega)$. The integral of the evolutionary power-spectrum therefore has a similar definition to the stationary case (5.1.3) in that we find $\text{Var}[X_t] = \int_{-\pi}^{\pi} dH_t(\omega)$. However, crucially, we now note that while normally the power-spectrum is determined by the behaviour of the process over all time, it is now localised to the neighbourhood of the point t.

All the non-stationary models considered in the following chapters possess a similar notion of time-varying spectra (or decomposition of variance across frequencies/scale). The down side to allowing such evolution is that, as with dynamic graphs, it increases model complexity. Again, in order to obtain some form of consistency we need to constrain how fast or in what way the spectrum can change over time. In the oscillatory process model, Priestly (1981) suggests to use locally supported windows to estimate $A_t(\omega)$.

5.1.3 Locally Stationary Processes

One of the problems with such a localised estimation approach to estimating oscillatory processes, is that asymptotically (as $T \to \infty$) we wish to gather increasing information relating to $A_t(\omega)$. If we construct an estimator $\hat{A}_t(\omega)$ we desire consistency such that $\hat{A}_t(\omega) \to A_t(\omega)$. However, if we allow for arbitrary non-stationarity, the more recent samples closer to T do not necessarily tell us much or anything about those at the start of the process. This point as noted by R Dahlhaus (1997) motivates the construction of a different asymptotic concept.

Rather than having the temporal support of $A_1(\omega), \ldots, A_T(\omega)$ grow as T increases, R Dahlhaus (1997) suggests to instead estimate the structure of a

function on the rescaled interval $[0,1]$. As an example, consider the auto-regressive process $X_t = a_t X_{t-1} + \epsilon_t$, where $\epsilon_t \overset{iid}{\sim} \mathcal{N}(0,\sigma^2)$. We would normally aim for inference on a_t over $t \in \{1,\ldots,T\}$. However, an alternative asymptotic analysis could aim to estimate a related function $A(t/T)$ on the interval $t/T \in [0,1]$. The re-scaling here allows us to asymptotically assess the function $A(z)$ at ever finer intervals. Applying this re-scaled time concept to the specification of the family $\{\phi_t(\omega)\}$ gives rise to the notion of locally-stationary processes:

***Definition* 5.1.** *Locally Stationary Fourier (LSF) Processes (R Dahlhaus 1997)*

The sequence of stochastic processes $\{X_{t;T}\}$ for $t = 1,\ldots,T$ is referred to as locally-stationary *with respect to sequence of coefficients a_t and trend $\mu(\cdot)$ if there exists a representation*

$$X_{t;T} = \mu\left(\frac{t}{T}\right) + \int_{-\pi}^{\pi} a_{t;T}(\omega)\exp(i\omega t)dZ(\omega)\ ,$$

where $Z(\omega)$ is a stochastic process on $[-\pi,\pi]$ with $\bar{Z}(\omega) = Z(-\omega)$ and the following conditions hold:

- *The cumulants of kth order are bounded, such that*

$$\text{cum}\{dZ(\omega_1),\ldots,dZ(\omega_k)\} = \delta_{2\pi}(\sum_{j=1}^{k}\omega_j)g_k(\omega_1,\ldots,\omega_{k-1})d\omega_1\ldots d\omega_k$$

  is the cumulant of kth order, where $\delta_{2\pi}$ is the 2π periodic extension of the dirac-delta function. Dependency across frequencies is bounded as $g_1 = 0$, $g_2(\omega) = 1$, and $|g_k(\omega_1,\ldots,\omega_k)| \leq M_k$ is bounded by a constant for all k.
- *There exists a constant C and a 2π-periodic continuous (in z,ω) function $A(z,\omega) : [0,1]\times\mathbb{R} \mapsto \mathbb{C}$, where $A(z,-\omega) = \bar{A}(z,\omega)$ such that*

$$\sup_{t,\omega}\left|a_{t;T}(\omega) - A\left(\frac{t}{T},\omega\right)\right| \leq \frac{C}{T}\ , \quad \text{for all } T\ .$$

The locally stationary process, can be viewed as an extension of oscillatory processes, but with the dynamics of $a_{t;T}(\omega)$ constrained to be asymptotically close continuously smooth function $A(z,\omega)$ on the interval $z = t/T \in [0,1]$. The aim of inference when identifying such a locally-stationary model is the smooth function $A(z,\omega)$ rather than the sequence of transfer functions $a_{t;T}(\omega)$. However, even with the re-scaled transfer function, one is still required to place

assumptions on the smoothness of this. Typically, one assumes $A(z,\omega)$ is continuous in z, that it is Lipschitz smooth and is differentiable (R Dahlhaus 1997; G.P. Nason et al. 2000). Different smoothness constraints on $A(z,\omega)$ result in different classes of process. For example, in the next chapter we consider how one may construct piecewise constant locally-stationary processes.

5.1.4 Spectral estimation procedures

Estimation for locally-stationary Fourier processes may be performed in a variety of ways (see Remark 5.1). For now, let us assume the *spectral density* is specified via a parametric function $S_\theta \equiv |A(z,\omega)|^2$. In stationary processes, estimation is usually performed taking into account all the data together via a function referred to as the periodogram. However, to take into account that the spectrum can change, one needs to examine the structure locally throughout time. Such a requirement motivates the introduction of a periodogram with a localised taper function $h(\cdot)$:

***Definition* 5.2.** *Localised Fourier Periodogram*

Let $h(z) : \mathbb{R} \mapsto \mathbb{R}$ be a data-taper function with $h(z) = 0$ for $z \notin [0,1)$, for window width $2M$. Define the Fourier coefficients of the tapered data as:

$$d_M(z,\omega) := \sum_{s=0}^{2M-1} h(s/N) x_{[zT]-M+s+1;T} \exp(-i\omega s) \; . \tag{5.1.5}$$

The localised Fourier periodogram is then defined as

$$I_M(z,\omega) = \frac{1}{2\pi H_{2,M}(0)} |d_M(z,\omega)|^2 \; , \tag{5.1.6}$$

where $H_{m,M}(\omega) = \sum_{s=0}^{2M-1} h(s/2M)^m \exp(-i\omega s)$ is a normalising constant dependent on the taper shape.

If we consider estimation in the maximum-likelihood sense, one may consider minimising a loss based on the method of Whittle (1953):

$$\hat{\theta}_T := \arg\min_\theta \underbrace{\left\{ \frac{1}{4\pi} \frac{1}{K} \sum_{k=1}^{K} \int_{-\pi}^{\pi} \left(\log S_\theta(z_k,\omega) + \frac{I_M(z_k,\omega)}{S_\theta(z_j,\omega)} \right) d\omega \right\}}_{\mathcal{L}_T(\theta)} \; , \tag{5.1.7}$$

where K is the number of shifted localised periodograms $I_M(z_k, \omega)$ we use to construct the likelihood. The positions of the periodograms z_k are given as $z_k = \Delta(k-1) + M$ where Δ is the shift in time between segments. R Dahlhaus (1997) initially introduces such a localised Whittle estimator as a heuristic. However, on further examination he demonstrates that if the true f and parametric f_θ probability distributions of the process $X_{t;T}$ are zero-mean Gaussian then the asymptotic KL-divergence is given as

$$\lim_{T\to\infty} \frac{1}{T} E_f[\log(f/f_\theta)] = \underbrace{\frac{1}{4\pi}\int_0^1 \int_{-\pi}^{\pi} \left(\log S_\theta(z,\omega) + \frac{S(z,\omega)}{S_\theta(z,\omega)} \right) d\omega dz}_{\mathcal{L}(\theta)} + \text{const} \, ,$$

where $S(z,\omega)$ is the true spectral density (R. Dahlhaus 1996). Asymptotic identification of the spectrum can therefore be demonstrated by proving that the discretely constructed Whittle likelihood estimator converges to that of the true estimate $\theta_0 := \arg\min_\theta \mathcal{L}(\theta)$ such that $\hat{\theta}_T \to \theta_0$. For a proof of this convergence, see Thm. 3.2 R Dahlhaus (1997).

***Remark* 5.1.** *Asymptotics and alternate estimation procedures*

Asymptotic properties of the Fourier transform of stochastic process are well understood. In particular, the work of Brillinger (1972, 1974, 1981) provide results, that given certain mixing conditions, i.e. dependency in process is restricted for large lags, then the Fourier coefficients (c.f. Eq. 5.1.5) will be distributed in a Gaussian manner. As such, the second order properties of the process are asymptotically dominant, and the ML inspired Whittle likelihood provides an appropriate method for estimation. Alternative estimation methods include averaging across different tapers, or the use of moving windows (Nuttall et al. 1982; Walden 2000). More recently, the work of Cohen et al. (2010, 2011) examined multi-taper estimation for the Morlet wavelet spectra which effectively performs a localised Fourier transform of a process.

5.2 Introduction to Wavelet Bases

Thus far, our discussion of modifications to $\{\phi_t(\omega)\}$ has been restricted to the complex exponential family combined with a time-varying amplitude function. One of the main criticisms of such locally stationary Fourier models,

is that at the estimation stage one must specify the scale and shape of the taper function $h(\cdot)$ to suit the process $X_{t;T}$. Additionally, as observed in Eq. 5.1.5 the taper function modifies the contribution of the process equally at all frequencies. For example, the effective window (taper) width for large wavelength (low-frequency) structure is the same as that for high-frequency structure. This is not always desirable as estimates of the Fourier coefficients can be expected to have different levels of variance, i.e. typically high-frequency components are estimated at a faster rate.

One solution to such problems is to specify the process itself in terms of a set of localised basis functions known as *wavelets*. Rather than decomposing a function over a set of frequencies, the wavelet basis constitutes what is known as a *time-scale* representation. Such schemes can provide a natural extension of non-stationary Fourier process whereby the taper function is baked into the construction of the basis. In this section, we discuss how deterministic functions can be represented using a wavelet basis; the following section extends this discussion to the representation of stochastic processes.

Wavelet Decompositions

While there are many forms and shapes wavelets can take, the common principle is to build a set of basis functions around a locally supported function known as the *mother wavelet* $\psi(x)$. This function can then be shifted and scaled to create a family of functions. A set of *child wavelets* is defined as:

$$\left\{ \psi_{a,b}(x) := a^{-1/2} \psi\left(\frac{x-b}{a} \right) \mid (a,b) \in \mathbb{R}^+ \times \mathbb{R} \right\} ,$$

with shift b and scaling a. A linear combination of such wavelets can then be used to approximate a non-compactly supported function.

***Definition* 5.3.** *Continuous Wavelet Transform*

One can express the function $f(x)$ for $x \in \mathbb{R}$ as:

$$f(x) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_0^{\infty} \frac{1}{a^2} W_f(a,b) \psi_{a,b}(x) da db ,$$

with normalisation constant

$$C_\psi = \int_{\mathbb{R}} \frac{|\tilde{\psi}(\omega)|^2}{|\omega|} |d\omega < \infty ,$$

where $\tilde{\psi}(\omega)$ is the Fourier transform of $\psi(x)$.

The integration constraint in the above represents what is known as an *admissibility condition* for the wavelet $\psi(x)$. The *continuous wavelet transform (CWT)* $W_f(a, b)$ is the resultant function over 2-dimensional input (a, b) that specifies the contribution of each child wavelet $\psi_{a,b}(x)$ required to describe the function $f(x)$.

The CWT function defined with respect to arbritary (a, b) is very flexible in describing functions and is inherently redundant. This means, that in practice it is hard to specify the continuous function as one has to calculate an often intractable integral. To simplify this calculation, one may consider a transformation which is sampled at a set of values (a, b) that take specific values. In particular, we concern ourselves with two classes of wavelet transform:

Decimated: In this case the scaling and translation of the mother wavelet is such that the child wavelet is given as

$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}x - k) \ .$$

The form above is observed when one sets the values of $a = 2^j$ and $b = k \cdot 2^j$ where k, j are integers. Such a choice of a, b is known as *critical sampling*. Setting (a, b) in this way gives a unique invertible transformation whilst enabling $\mathrm{CWT}(j, k) : \mathbb{Z} \times \mathbb{Z} \mapsto \mathbb{R}$ to be a function of discrete inputs. The term critical sampling corresponds to the fact that any coarser decimation of the wavelet will result in a non-invertible transform; the function will not be able to be uniquely recovered from the wavelet coefficients.

Non-Decimated: In this scheme, the translational shift (k) is also scaled by the factor of 2^j, the child wavelets are defined as:

$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}(x - k)) \ .$$

Unlike the decimated wavelets, the non-decimated set of wavelets are not orthogonal; they form an over-complete and highly redundant basis. One advantage of this extra redundancy is that the wavelets can now be shifted at all scales by an amount specified at the finest scale of analysis. In contrast with the decimated transform the number of values that k can take is the same at all scale levels.

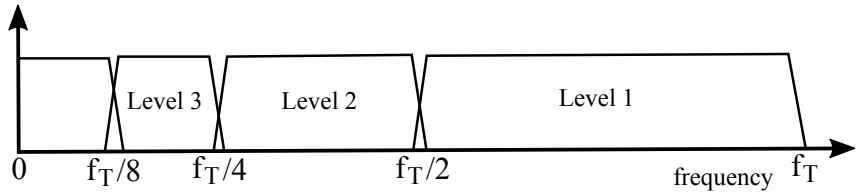


Figure 5.2.1 – Frequency coverage of wavelet decomposition with dyadic sampling of scales.

While the discretisation schemes above allow a more concise representation of a signal than the CWT; they still require an infinite number of coefficients to describe the signal, the integrals now become infinite summations, i.e. $f(x) = \sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} d_{j,k}\psi_{j,k}(x)$. If we consider compression of the wavelet in the time-domain is equivalent to stretching the spectrum and shifting it upwards (see Figure 5.2.1); then in order to cover the complete spectrum down to zero frequency we again require an infinite number of wavelets. However, in real life, we never directly observe a continuous underlying function but instead sample a function at a finite number of points.

Multi-resolution analysis

To use wavelets to represent such functions it is useful to consider the transform in the context of a scheme known as *multi-resolution analysis (MRA)*. As introduced by Mallat (1989), this framework suggests an interpretation of the wavelet transform where scaled basis functions are used to span nested sub-spaces. For example, consider the sequence of subspaces $\{V_j\}$ such that:

$$\{0\} \cdots \subset V_1 \subset V_0 \subset V_{-1} \cdots \subset V_{-j} \cdots \subset L^2(\mathbb{R}) \; .$$

If we consider a function which is restricted to a subspace V_0, then under shifting the MRA requires this function to remain in the space, i.e.

$$f \in V_0 \iff f(x-k) \in V_0 \; . \tag{5.2.1}$$

Additionally, and crucially, we now let there be a mapping between subspaces according to scaling, such that

$$f(x) \in V_0 \iff f(2^j x) \in V_j \; . \tag{5.2.2}$$

Similarly to the wavelet function, let us introduce a function $\phi_0(x)$ known as the scaling function and use shifted/scaled versions of these $\{\phi_{j,k}(x) := \phi_j(x-k)\}$ to construct an orthonormal basis in V_j. Combined with properties (5.2.1, 5.2.2) we obtain the relation $\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k)$. Let us again consider the wavelet functions $\{\psi_{j,k}\}$ and let these span respective subspaces W_j. It turns out (Mallat 1989), that there exists a function $\psi(x)$ such that W_j is the orthogonal complement of V_j in V_{j-1}. As such, the inner products $\{\langle \psi_{j,k}, f\rangle \,|\, k \in \mathbb{Z}\}$ contain all the information that is present in approximation to f at scale $j-1$, but lacking at coarser level j (Daubechies 1990).

If the function $\phi(x)$ has finite support, then one can represent this function as a *scaling equation* that takes the form of a discrete summation[2]:

$$\phi(x) = \sqrt{2}\sum_k h[k]\phi(2x-k) \; .$$

If we choose an appropriate wavelet such that $W_0 \subset V_1$, then we are required to find a set of coefficients $g[k]$ such that

$$\psi(x) = \sqrt{2}\sum_k g[k]\phi(2x-k) \; .$$

The coefficients $h[k]$ and $g[k]$ respectively correspond to low and high-pass filters; they maintain the relationship $g[k] = (-1)^k h[1-k]$. In signal processing, such filters are known as *quadrature mirror filters*. For further discussion on the existence and uniqueness of the coefficients which define these filters see Daubechies (1992).

Now that we have discussed how to specify the wavelet and scaling functions, we can work in this expanded basis to study the decomposition of a function. Consider the projection of a function f onto the sub-spaces is given as

$$\begin{aligned} \langle \phi_{j-1,k}, f\rangle &= \langle \phi_{j,k}, f\rangle + \langle \psi_{j,k}, f\rangle \\ &= \sum_k c_{j,k}\phi_{j,k} + \sum_k d_{j,k}\psi_{j,k} \; , \end{aligned} \tag{5.2.3}$$

[2]Note: the indexing for $h[k]$ for $k \in \mathbb{Z}$ refers to a discrete sequence $h[1], h[2] \ldots$ with local support. The notation is commonly used in signal processing and allows us to distinguish that h is a function of discrete integers.

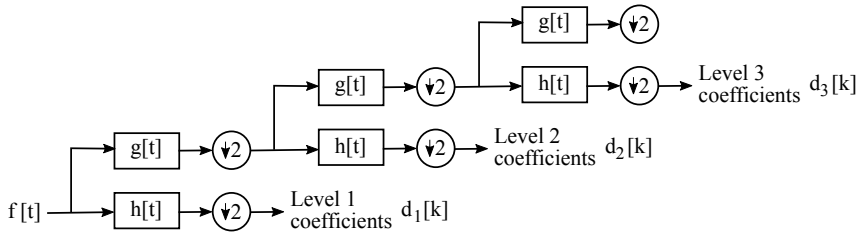


Figure 5.2.2 – Filter-bank representation of the discrete (decimated) wavelet transform as described in Definition 5.2.

where $c_{j,k}$ and $d_{j,k}$ are known respectively as *approximation* and *detail* coefficients. As a result we observe that the projection at a finer scale level can be calculated in terms of those at the coarser levels. Starting at coarsest level J and simply iteratively applying (5.2.3) leads to

$$f(x) = \langle \phi_{J,k}, f \rangle + \langle \psi_{J,k}, f \rangle + \ldots + \langle \psi_{1,k}, f \rangle$$

and the corresponding orthogonal discrete wavelet series decomposition

$$f(x) = \sum_k 2^{-J/2} c_{J,k} \phi(2^{-J} - k) + \sum_{j=1}^{J} \sum_k 2^{-j/2} d_{j,k} \psi(2^{-j} x - k) \, . \tag{5.2.4}$$

Due to the two-scale relation, multiplying (5.2.3) by $\phi_{j,k}$ and taking inner products gives:

$$\begin{aligned} c_{j,k} = \langle \phi_{j,k}, \langle \phi_{j-1,k}, f \rangle \rangle &= 2^{1/2} \sum_l h[l - 2k] c_{j-1,l} \\ d_{j,k} = \langle \psi_{j,k}, \langle \phi_{j-1,k}, f \rangle \rangle &= 2^{1/2} \sum_l g[l - 2k] c_{j-1,l} \, . \end{aligned} \tag{5.2.5}$$

What the MRA gives us, is a way to plug the wavelet decomposition so we do not have to consider an infinite set of scale levels. For example, in the above analysis we worked with a finite scale level J which defined the coarsest level of analysis. If we consider Eq. 5.2.4, one notes that detail coefficients are required for J scales, the rest of the function is represented via the approximation coefficient $c_{J,k}$.

Discrete (Decimated) Wavelet Transform

In practice, and in our applications, we will be dealing with discretely indexed functions or processes of finite size T. Assuming that $T = 2^J$ the discrete wavelet decomposition (5.2.4) permits an efficient algorithm for the computation of coefficients $\{d_{j,k}\}_{j=1}^{J}, \{c_{J,k}\}$. Such a procedure is defined below, and graphically illustrated in Figure 5.2.2.

Let $f[t]$ be a discrete signal and define the discrete convolution $y[t] = (f * g)[t] := \sum_{k=-\infty}^{\infty} f[k]g[t-k]$, let $(y \downarrow l) := y[lt]$ denote the sub-sampling operator. The discrete wavelet coefficients for scale j are then given by

$$\begin{aligned} d_{1,k} &= (f * h) \downarrow 2[k] \quad y_1[k] := (f * g) \downarrow 2[k] \\ d_{2,k} &= (y_1 * h) \downarrow 2[k] \quad y_j[k] := (y_{j-1} * g) \downarrow 2[k] \\ &\vdots \\ d_{j,k} &= (y_{j-1} * h) \downarrow 2[k] \,, \end{aligned}$$

where one notes that the number of coefficients used to describe a segment of data $f[t]$ for $t = 1, \ldots, T = 2^J$ at each scale decreases dyadically. The total number of coefficients depends on the size of the support of $h[\cdot]$ where $|\{d_{j,k} \forall j, k\}| = \lfloor T + |\text{supp(h)}| \rfloor / 2$.

***Remark* 5.2.** *Signal Extension*

In this work, we perform theoretical analysis on signals which maintain the constraint $T = 2^J$. However, in practice, one can easily imagine that a signal may not be exactly of this length. In such a case we need to specify a way to extend the signal over the set $t \in \{1, \ldots, 2^{\lceil \log_2 T \rceil}\}$. Throughout this thesis I typically use a zero-padding method whereby the signal is centered in the interval and then padded on either side with zero entries. In synthetic settings, the simulation can usually be set up to avoid the need for padding.

Discrete (Stationary) Wavelet Transform

The discrete wavelet transform with non-decimated wavelets is commonly referred to as the *stationary wavelet transform (SWT)*. This algorithmic transform is a simple modification to the DWT whereby the downsampling is removed and instead the filters are up-sampled at each scale. The filters applied

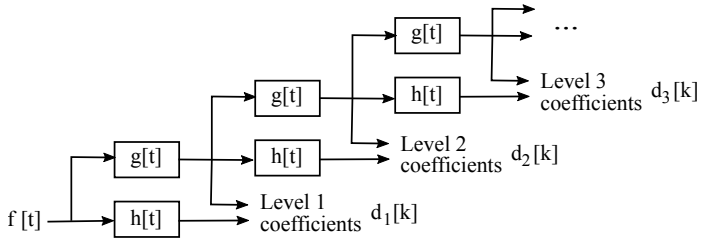


Figure 5.2.3 – Filter-bank representation of the stationary wavelet transform.

at each scale h_{j+1}, g_{j+1} are given by inserting zeros in the filter at the previous scale, such that

$$h_{j+1}[k] = h_j \uparrow 2[k] \quad , \quad g_{j+1}[k] = g_j \uparrow 2[k] \; .$$

Assuming a dyadic length $T = 2^J$, the number of coefficients at each level is now the same, i.e. $|\{d_{j,k}\}| = |\{d_{j',k}\}| = T$ for all j, j'. The SWT has a property known as *shift-invariance*, such that, on shifting a signal by lag τ the wavelet coefficients are now also shifted, i.e. $f[t] = f[t+\tau] \implies d^{SWT}_{j,k} = d^{SWT}_{j,k+\tau}$. The DWT does not possess this property as the number of coefficients at each scale is different, instead the values of the coefficients will change, i.e. $d^{DWT}_{j,k} \neq d^{DWT}_{j,k+\tau}$.

5.3 The Locally Stationary Wavelet Process

In the previous section we discussed how one can represent a deterministic function in terms of a set of wavelets. In analogy to the locally stationary processes in Sec 5.1.3 which utilised the Fourier basis, we can now use wavelets to specify a class of stochastic process. In particular, let us consider the set of discrete wavelets $\boldsymbol{\psi}_j \in \mathbb{R}^{N_j}$ which have dimension $N_j = (2^j - 1)(|\text{supp}(h)| - 1) + 1$, where $h[k]$ is an appropriate low-pass filter. The construction of the discrete wavelets can be achieved iteratively according to:

$$\begin{aligned} \psi_{1,t} &= \sum_{k\in\mathbb{Z}} g[t-2k]\delta_{0,k} = g[t] \\ \psi_{j+1,t} &= \sum_{k\in\mathbb{Z}} h[t-2k]\psi_{j,k} \quad \text{for } t = 0, \ldots, N_{j+1} - 1 \; . \end{aligned}$$

***Remark* 5.3.** *Discrete Wavelet Notation*

One should note, that in general the discrete wavelets $\{\boldsymbol{\psi}_j\}$ *are not just sampled versions of the standard continuous wavelet* $\{\psi_j(x)\}$[3]. *The above scheme creates a set of dilated wavelets, however, we also wish to shift these in order to build our set of basis functions. The integer shifted version of the wavelet are denoted as* $\psi_{j,t}[k] := \psi_{j,t-k}$ *for* $t-k \in \{0,\ldots,N_j-1\}$.

The class of processes introduced here are known as *locally stationary wavelet (LSW)* processes, and have attracted considerable research attention in recent years. While the original formulation is due to G.P. Nason et al. (2000), extensions to model random fields (Eckley et al. 2010; Nunes et al. 2014; S. Taylor et al. 2014) and multivariate processes (Park et al. 2014; Sanderson et al. 2010) have also been proposed.

***Definition* 5.4.** *1d-Locally Stationary Wavelet Process*

A locally-stationary wavelet process is a doubly indexed stochastic process $\{X_{t;T}\}_{t=0,\ldots,T-1}$. *It is defined over the discrete scales* $j = 1,\ldots,J$, *where* $T = 2^J \geq 1$, *and has the following mean-square representation:*

$$X_{t;T} \;=\; \sum_{j=1}^{J}\sum_{k=1}^{T} w_{j,k;T}\psi_{j,k}[t]\epsilon_{j,k}\;, \tag{5.3.1}$$

where $\epsilon_{j,k}$ *is a random orthonormal increment sequence, i.e* $\epsilon_{j,k} \perp \epsilon_{j',k'}$ *for all* $j \neq j'$ *and* $k \neq k'$, *and* $E[\epsilon_{j,k}] = 0$, $\mathrm{Cov}(\epsilon_{j,k},\epsilon_{j',k'}) = \delta_{j,j'}\delta_{k,k'}$ *for all* j,j',k,k'. *The basis functions used in the construction are given by* $\{\psi_{j,k-t}\}$, *for* $j = 1,\ldots,J$ *and* $k \in \mathbb{Z}$.

***Assumption* 5.1.** *Bounded-deviation from Lipschitz Spectrum*

The continuous function $W_j(z)$ *is connected with the discrete structure* $w_{j,k;T}$ *according to the bound:*

$$\sup_k |w_{j,k;T} - W_j(k/T)| \leq C_j/T\;, \tag{5.3.2}$$

and the sequence $\{C_j\}$ *is bounded such that* $\sum_{j=1}^{\infty} C_j < \infty$. *Additionally, the function* $W_j(z)$ *should be Lipschitz smooth and have finite power, such that:*

[3]For the Haar system, the discrete wavelets can be considered sampled versions of their continuous equivalents i.e. $\boldsymbol{\psi}_1 = (g[0],g[1]) = 2^{-1/2}(1,-1)$ and $\boldsymbol{\psi}_2 = (h[0]g[0],h[1]g[0],h[0]g[1],h[1]g[1]) = 2^{-1}(1,1,-1,-1)$.

$$\sum_{j=1}^{\infty} |W_j(z)|^2 < \infty \text{ uniformly in } z \in (0,1) \tag{5.3.3}$$

with Lipschitz constant L_j uniformly bounded in j and $\sum_{j=1}^{\infty} 2^j L_j < \infty$.

Again, it should be noted that in the spectral domain, the stochastic noise source is assumed to be independent, both across scale and time. *Note: the spectra is now defined with respect to the set of scales $j = 1, \ldots, J$ rather than frequencies.* The wavelet basis functions therefore act in place of the Fourier complex exponential basis. An important consequence of this is that the model has a very large ambient parameter space and may be considered non-parametric. For example, consider that coefficients $w_{j,k;T}$ must be specified both for each scale and each time point. As discussed, and similarly to the LSF process, this gives the LSW process the ability to describe both stationary and non-stationary processes. However, as when dealing with dynamic graphical models, we are also required to constrain time variation in the parameterisation. In the LSW model, this is canonically (G.P. Nason et al. 2000) achieved by requiring parameters be close to being Lipschitz smooth. Similarly to the LSF process (5.1), the LSW parameters $\{w^0_{j,k;T}\}$ are tied to an underlying continuous spectral modulation function $W_j(k/T) : [0,1] \mapsto \mathbb{R}$ for all scales $j \in \mathbb{N}$.

The set of functions $W_j(z)$ can be considered the equivalent of $A(z,\omega)$ in the LSF process (Definition 5.1). However, where the Fourier equivalent was defined with respect to an integral over continuous frequencies, we now have a discrete summation over an increasing number of scales; increasing in the sense that as $T \to \infty$, $J_T = \log_2 T \to \infty$. With the LSF process we defined (and parameterised) the spectral density as $S_\theta(z,\omega) := |A(z,\omega)|^2$, an analogous quantity in the LSW process is given by the below:

***Definition* 5.5.** *Evolutionaty Wavelet Spectrum (EWS)*

The evolutionary wavelet spectrum *is defined (with respect to the wavelets $\{\psi_{j,k}\}$) as $S_j(z) := |W_j(z)|^2$ for $j = 1, \ldots, J_T$, $z \in (0,1)$.*

In the asymptotic limit ($T \to \infty$) the EWS can be related to the discrete transfer function $\{w_{j,k;T}\}$, such that $S_j(z) = \lim_{T\to\infty} |w_{j,[zT];T}|^2$ for all $z \in$

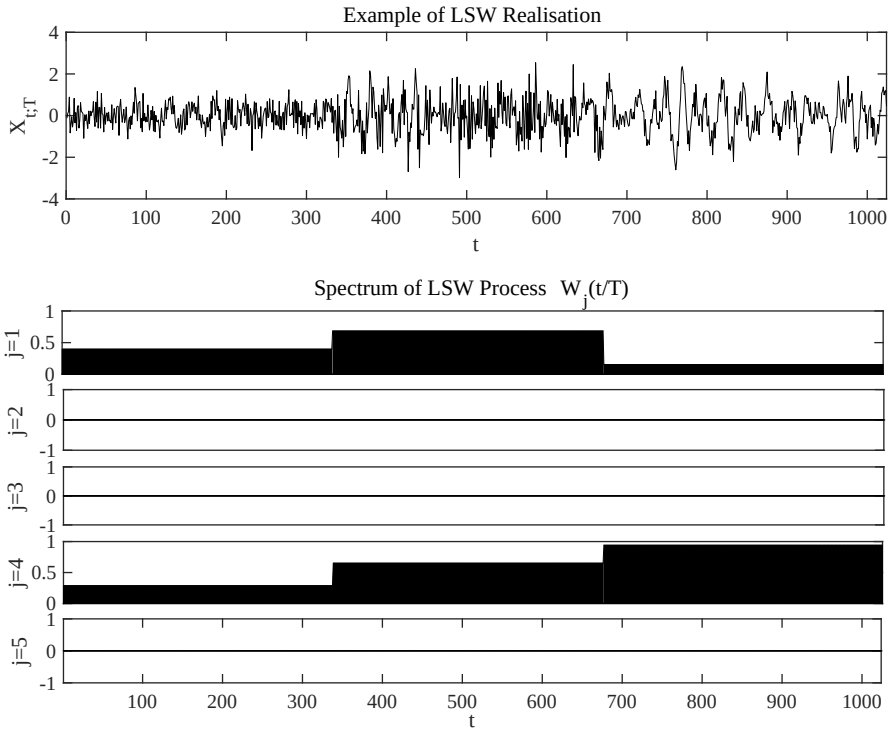


Figure 5.3.1 – Top: An example draw from a one-dimensional LSW process. Bottom: The corresponding transfer functions $W_j(k/T)$ that generated the process. In this case $T = 1024$ and $J = 10$, only $W_j(z)$ for scales $j = 1, \ldots, 5$ are plotted, the others are set to zero.

$(0, 1)$. In contrast to the spectral density, the EWS is defined at a discrete set of scales according to different scalings of the wavelet basis. The assumption (5.3.3) gives the LSW process a finite variance, and as $E[\epsilon_{j,k}] = 0$ the process also maintains zero mean $E[X_{t;T}] = 0$.

5.3.1 Properties of LSW Processes

One of the benefits of the LSW model construction is that it allows for a parsimonious representation of a stochastic processes auto-covariance function. In particular, the over-complete representation allows for the description of time-varying auto-covariances, and thus provides a powerful tool for non-stationary analysis. We here introduce some properties of the canonical LSW process as per Definition. 5.4.

Let the auto-covariance of a process be denoted

$$c_{X;T}(z,\tau) = \mathrm{Cov}[X_{[zT];T}, X_{[zT]+\tau;T}]$$

for $z \in (0,1)$ and shift $\tau \in [-zT, \ldots, (1-z)T] \subset \mathbb{Z}$. This quantity describes the covariance properties of the process about a time point zT, but with a finite length T. An asymptotic equivalent of the above is given by the *local auto-covariance (LACV)* and defined as:

$$c_{X;\infty}(z,\tau) := \sum_{j=1}^{\infty} S_j(z)\Psi_j[\tau] \,, \tag{5.3.4}$$

for $\tau \in \mathbb{Z}$ and $z \in (0,1)$ where

$$\Psi_j[\tau] := \sum_{k\in\mathbb{Z}} \psi_{j,k}[0]\psi_{j,k}[\tau] \,, \tag{5.3.5}$$

is known as the *autocorrelation wavelet*.

The autocorrelation wavelets act to describe the difference between the signal projected onto a wavelet at one point, and another point at lag τ. This can be thought of a basis with which to describe second-order structure of the process. Additionally, and importantly, the finite and asymptotic autocovariance functions tend towards each other.

***Proposition* 5.3.** *Convergence of the LACV and auto-covariance (G.P. Nason et al. 2000)*

Letting $J = \log_2(T)$*, as* $T \to \infty$*, uniformly in* $\tau \in \mathbb{Z}$ *and* $z \in (0,1)$

$$|c_{X;\infty}(z,\tau) - c_{X,T}(z,\tau)| = \mathcal{O}(T^{-1}) \,.$$

The proof follows simply from substituting the process definition in for $c_{X;T}$ and then utilising the Lipschitz continuity of $S_j(z)$, a full treatment can be found in (G.P. Nason et al. 2000). While the asymptotic result (Prop. 5.3) enables the LSW model to describe the time-varying auto-covariance of a non-stationary processes, the non-orthogonality of the construction requires a careful treatment when it comes to estimation. We now discuss how one may invert Eq. 5.3.4 and how one may construct a sample based estimator of the LACV.

***Proposition* 5.4.** *Inversion of the LACV*

The spectrum can be related to the LACV via the inversion relation

$$S_j(z) = \sum_{l=1}^{\infty} A_{jl}^{-1} \sum_{\tau \in \mathbb{Z}} c_{X;\infty}(z, \tau) \Psi_l[\tau] \,, \tag{5.3.6}$$

where the linear operator $\boldsymbol{A}$ *is defined for* $j, l = 1, \ldots, \infty$ *as*

$$A_{jl} := \langle \Psi_j, \Psi_l \rangle = \sum_{\tau \in \mathbb{Z}} \Psi_j[\tau] \Psi_l[\tau] \,. \tag{5.3.7}$$

Additionally, the family $\{\Psi_j(\tau)\}_{j=1}^{\infty}$ *is linearly independent. Hence:*

(1) The EWS is uniquely defined given the corresponding LSW process.

(2) The operator A *is invertible, for each finite* J *the norm* $\|\boldsymbol{A}_J^{-1}\|$ *is bounded above by some* C_J*.*

Proof. The proof of the inversion relies on the invertibility of the $\boldsymbol{A}$ matrix, where the structure is formed by an inner product for each pair of scale levels $\{j, l\}$, i.e. a Gramm matrix. The proof of properties for $\boldsymbol{A}$ can be found in G.P. Nason et al. (2000). Result (1) is demonstrated via contradiction, where one can show that two LSW representations are necessarily the same for a given EWS. The second result (2) can be demonstrated by observing that $\boldsymbol{A}$ is a Gramm matrix, if the auto-correlation wavelets $\{\Psi_j\}$ are linearly independent this will be positive definite and thus invertible (it possesses all positive eigenvalues). An explicit demonstration that the matrix $\boldsymbol{A}$ has positive eigenvalues for the family of Shannon and Harr wavelet families can be found in (G.P. Nason et al. 2000). It is conjectured, but not demonstrated that such a property holds for all Daubechies' compactly supported wavelets.

If we consider Equation 5.3.6, in the setting where $\tau = 0$ then we find that $\Psi(0) = 1$, therefore the spectrum $S_j(z)$ constitutes a biased estimate of the variance $c_{X;\infty}(z, 0)$. The matrix $\boldsymbol{A}$ therefore plays a vital role, not only in demonstrating that the auto-covariance structure can be asymptotically identified, but also describing how variance at different scales in the spectra is translated to the time-domain. An example of the matrix $\boldsymbol{A}$, alongside the autocorrelation wavelets $\{\Psi_j[\tau]\}$ can be seen in Figure 5.3.2. Of particular

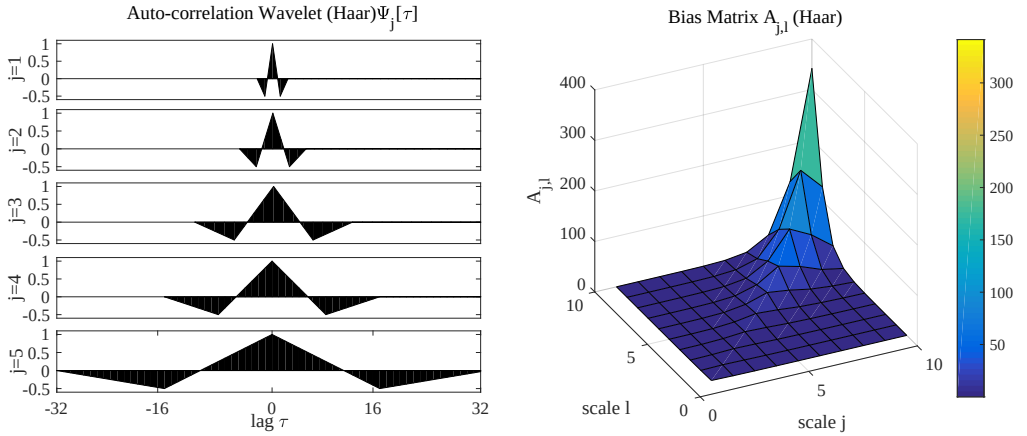


Figure 5.3.2 – Example of the auto-correlation wavelets $\Psi_j(\tau)$ (left) and corresponding matrix $\boldsymbol{A}$ (right) for the Haar family of wavelets.

interest in the remainder of this thesis, is how we can estimate the spectra $S_j(z)$ of a process from finite data.

5.4 Estimation of the Evolutionary Wavelet Spectrum

In the previous section, an inversion relation (Eq. 5.3.6) was introduced that allowed us to relate the wavelet spectrum to the local-autocovariance. In this final introductory section, we shall discuss how one may use this relation to estimate the EWS from data. This section aims to present a review of previous approaches from the literature for estimation. The focus here is on the canonical one-dimensional LSW process, while the following chapters introduce methods for estimating the wavelet spectra for 2-D fields, such as images, and in multivariate settings. It is worth noting that the literature has a strong focus on asymptotic analysis of estimators. Typically, an estimation scheme is proposed, and then a paper will consider it's asymptotic consistency properties. However, while these estimators are asymptotically consistent, they often fail to produce sensible estimates in practice. For example, we often obtain negative estimates for the EWS, a quantity which theoretically should always be greater than or equal to zero. Regularisation can help mitigate such issues,

however, discussion of this is deferred to later chapters. In this section we introduce canonical estimators for the spectra and some corresponding asymptotic consistency results.

Recall that the number of scales utilised in a LSW process increases according to $J = \log_2 T$. As a result, when dealing with real data we will only be able to measure structure at a finite number of scale levels. An important consequence of Theorem 5.4, is that for an increasing but finite number of scales $j = 1, \ldots, J$, the approximate spectrum $S_{j;T}(z) := \sum_{l=1}^{J} A_{jl}^{-1} \sum_{\tau} c_{X;T}(z, \tau)\Psi_l(\tau)$ tends to the true spectrum, i.e. $\lim_{T\to\infty} S_{j;T}(z) = S_j(z)$. We can thus asymptotically identify the spectrum, even though individual weights $\{w_{j,k;T}\}$ may not be uniquely identifiable (G.P. Nason et al. 2000).

Considering that the form of $\boldsymbol{A}$ is fixed in advance by the choice of wavelet basis function, we only need to estimate the quantity $\sum_{\tau} c_{X;T}(z, \tau)\Psi_l(\tau)$. Previously, it has been proposed (Fryzlewicz et al. 2006; G.P. Nason et al. 2000; Sachs and Schneider 1996) to estimate this quantity at each scale level $l = 1, \ldots, J$ via the wavelet periodogram.

***Definition* 5.6.** *Wavelet Periodogram*

Consider scales $j = 1, \ldots, J$ and positions $k = 1, \ldots, T$. The empirical wavelet coefficients of an LSW process are defined as:

$$d_j[k] = \sum_{t=1}^{T} x_{t;T}\psi_{j,k}[t] \,. \tag{5.4.1}$$

Analogous to the Fourier periodogram (5.1.6) the wavelet periodogram *is defined according to $I_j[k] = |d_j[k]|^2$.*

It is interesting to note the difference between the wavelet periodogram statistic and that of the local Fourier periodogram. While they both effectively square the transform coefficients, we note that the Fourier coefficients require the setting of a taper-function. In the wavelet case, this taper is baked into the wavelet basis functions due to $\{\psi_{j,k}[t]\}$ having compact support. Additionally, in the wavelet construction the width of this effective taper function is different for each scale level. Indeed this change in taper size is what defines the concept of scale in wavelet models.

5.4.1 De-biassing the wavelet periodogram

While in Fourier based models the periodogram can be used to directly estimate the spectrum, for example via the Whittle likelihood, in the LSW model things are somewhat complicated by the non-orthogonal choice of basis. Calculating the expectation of the wavelet periodogram reveals that used alone, it is both biased and inconsistent estimator for the spectrum. In the following, we will analyse the properties of the periodogram under sampling from the process. This is indicated when taking expectations, for example, $E_X[I_j[k]]$ refers to replacing the samples $\{x_{t;T}\}$ with their stochastic equivalents $\{X_{t;T}\}$.

***Proposition* 5.5.** *Bias of Raw Wavelet Periodogram (G.P. Nason et al. 2000)*
The bias can be demonstrated, assuming Gaussianity for $\epsilon_{j,k}$, as:

$$E_X[I_j[k]] = \sum_l A_{jl} S_l(k/T) + \mathcal{O}(2^j T^{-1}) \; . \tag{5.4.2}$$

For the vector constructed across scales, $I[k] := (I_1[k], \ldots, I_J[k])^\top$ we can invert the above result to obtain

$$E_X[\boldsymbol{A}_J^{-1} I[k]] = S[k/T] + \mathcal{O}(T^{-1}) \; .$$

The above result can be obtained by substituting the process definition into $I_j^X[k]$, the convergence rate of $\mathcal{O}(1/T)$ is a result of Lipschitz smoothness. One can now see that the periodogram is exactly the quantity we desired for the estimator motivated by the representation of the EWS in terms of the process auto-correlation (5.3.6). Specifically we note that

$$\lim_{T\to\infty} E_X[I_j[k]] = \sum_{\tau\in\mathbb{Z}} c_{X;\infty}(z,\tau)\Psi_l[\tau] \; ,$$

and the estimator

$$\bar{S}_j[k] = \sum_l A_{jl}^{-1} I_j[k] \; , \tag{5.4.3}$$

is unbiased such that $\lim_{T\to\infty} E_X[\bar{S}_j[k]] = S_j[k]$ for $j = 1, \ldots, J$. Whilst the estimator is shown to be unbiased, we now need to check for consistency.

***Proposition* 5.6.** *Variance of Raw Periodogram*

Assuming that $\epsilon_{j,k} \sim \mathcal{N}(0,1)$*, the variance of the raw periodogram is given as:*

$$\mathrm{Var}_X[I_j[k]] = 2\Big(\sum_{l=1}^{J} A_{jl} S_l(k/T)\Big)^2 + \mathcal{O}(2^j/T)\ , \tag{5.4.4}$$

for $j = 1, \ldots, J$ *and* $k = 1, \ldots, T$*.*

The result demonstrates that the periodogram by itself is inconsistent (a proof can be found in G.P. Nason et al. (2000)). In practice, and in the literature it is preferred to smooth the periodogram before de-biasing in an attempt to obtain consistency.

5.4.2 Periodogram Smoothing

We here detail several approaches to smoothing the wavelet periodogram as discussed in the literature. As noted by Fryzlewicz et al. (2006), the signal-noise ratio of the wavelet periodogram is always relatively low; asymptotically we obtain

$$E_X[I_j[k]]/\mathrm{Var}_X[I_j[k]]^{1/2} = 2^{-1/2}\ .$$

Not only is this signal-noise ratio low, but typically the periodogram sequences will also be correlated over nearby points in time due to non orthogonality of the non-decimated basis functions. The distance over which these correlations are significant depends on the ground-truth structure of the spectrum, and is thus generally not known in advance. An example of the raw wavelet periodogram, can be seen in Figure 5.4.1.

Sliding Window Smoothing

Perhaps the simplest form of smoothing is to utilise a sliding window approach whereby the spectrum is estimated at the centre of some localised period in time. In some sense, this is similar to the localised Fourier periodogram (5.1.6) which averages in the context of a data taper function.

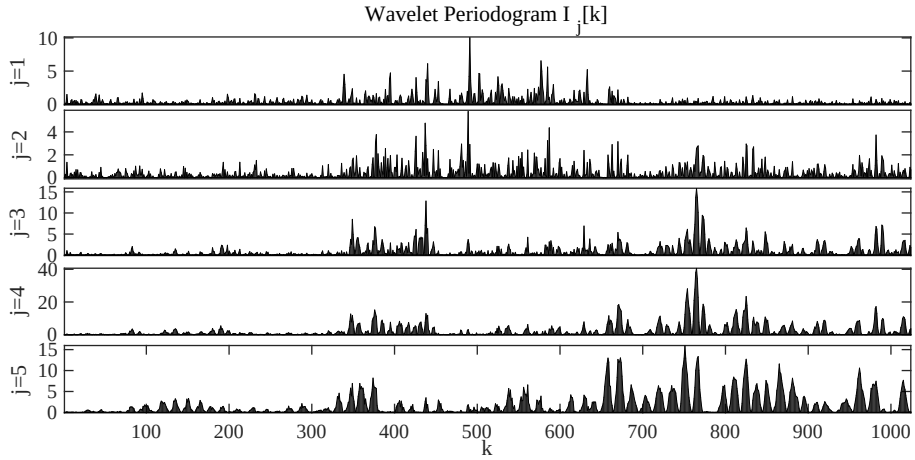


Figure 5.4.1 – Raw, unsmoothed wavelet periodogram, for the process realisation in Figure 5.3.1.

***Definition* 5.7.** *Kernel Smoothed Periodogram Estimator*

Let $k \in [t - M, t + M]$ be an interval of time on the discrete grid $t = 1, \ldots, T$. Let $h[k]$ be a weighting function for the data at point k. A smoothed periodogram estimator can be constructed such that:

$$\hat{S}_j(t/T) = \frac{1}{H} \sum_{k=t-M}^{t+M} \left(\sum_{l=1}^{J} A_{jl}^{-1} |h[k] d_{l,k}|^2 \right), \tag{5.4.5}$$

where $H = \sum_{k=t-M}^{t+M} h[k]^2$ is the integrated weighting function.

We note that the weighting function in the above plays a similar role to that in the local Fourier process models, for example via Eq. 5.1.6. However, unlike the Fourier equivalent, this is defined to be the same across all scale levels. In the particular case where

$$h[k] = \begin{cases} 1 & \text{if } k \in [t - M, t + M] \\ 0 & \text{otherwise} \end{cases},$$

the resulting estimator is equivalent to the *central moving average* estimator of Stevens (2013). It is also related to the smoothing technique employed in Park et al. (2014). In practice, the uniform kernel is preferred here, both for simplicity of analysis and because the wavelets are already localised in time.

Second Stage Wavelet Smoothing

A slightly more elaborate approach to periodogram smoothing is to smooth using a second-stage wavelet transform of the periodogram (G.P. Nason et al. 2000; Sachs, G.P. Nason, et al. 1997). Thresholding, or wavelet shrinkage methods can then be applied to this second-stage transform to act as a denoising step. Transforming back to the original periodogram results in a smoothed and asymptotically consistent estimator.

More precisely, one takes the raw periodogram and then performs a second stage of wavelet analysis, this time with a decimated set of wavelets $\{\bar{\psi}_{l,m}\}$. Taking the transform at scale l and position m we have the set of wavelet coefficients $\hat{v}_{l,m} = \sum_k d_{j,k}\bar{\psi}_{l,m}[k]$ for scales $2^l = o(T)$. Below the results of G.P. Nason et al. (2000) are quoted:

***Theorem* 5.1.** *Properties of Second-Stage DWT Coefficients (G.P. Nason et al. 2000)*

The DWT wavelet coefficients $\{\hat{v}_{l,m}\}$ of the periodogram from a Gaussian-LSW process at position $z = k/T$, with $2^l = o(T)$ obey uniformly in m,

$$E_X[\hat{v}_{l,m}] = \int_0^1 \sum_{i=1}^{J_T} A_{j,i}S_i(z)\bar{\psi}_{l,m}(z)dz = \mathcal{O}(2^{l/2}/T) \ ,$$

and

$$\mathrm{Var}_X[\hat{v}_{l,m}] = \frac{2}{T}\int_0^1 \left(\sum_{i=1}^{J_T} A_{j,i}S_i(z)\right)^2 \bar{\psi}^2_{l,m}(z) + \mathcal{O}(2^l T^{-2}) \ .$$

Furthermore, let $\hat{S}_j^{\bar{\psi}}(z)$ be the estimator obtained from inverse DWT of the coefficients $\hat{v}_{l,m}$ with the threshold $\lambda^2(l,m;j,T) = \mathrm{Var}_X(\hat{v}_{l,m})\log^2(T)$. For each fixed j, the estimate $\bar{S}$ obeys:

$$\int_0^1 E_X[\hat{S}_j^{\bar{\psi}}(z) - S_j(z)]^2 dz = \mathcal{O}(\log^2(T)/T^{2/3}) \ .$$

The proof of the consistency result above relies on results obtained throughout the 1990's relating to function denoising via wavelet shrinkage. In fact, the behaviour of such thresholded estimators is very closely related to the thresholding properties of estimators such as the lasso (c.f. 2.1.5). To avoid detracting from the main topic, i.e. smoothing the LSW periodogram, I have added some notes on denoising via wavelet shrinkage in Appendix 5.5.

Finally, while Theorem 5.1 holds for the case of the DWT smoothed periodogram, in practice, it is desirable to use a method that allows for time-invariance; that is, when we shift the data in time, the periodogram estimate should also shift. To this end, it is often suggested (G.P. Nason et al. 2000; Sachs and Schneider 1996) to apply a cycle spinning method, to perform translation-invariant denoising, see Coifman et al. (1995) for details of such a scheme. Briefly, the cycle spinning method works by shifting the data, in this case the periodogram at scale j a random number of positions (while maintaining ordering), smoothing the periodogram according to $\hat{v}_{l,m}$, and then performing the inverse DWT on these estimates. A modified second-stage transform known as the Harr-Fisz transform (Fryzlewicz et al. 2006) has also gained popularity for smoothing the non-decimated periodogram, we discuss this further in Chapter 6.

5.5 Summary

In this chapter, a set of *spectral* models and estimators for the representation of time-series were introduced. One key benefit of adopting a spectral approach to modelling time-series, is that often in the frequency/scale domain, the process can be considered to be in some sense sparse; that is only a subset of frequencies are required to describe the process. However, as discussed, the traditional Fourier representation is not appropriate for describing non-stationary processes. Instead, one can either allow the Fourier spectra to vary over time, c.f. Priestly's oscillatory processes, or adopt a localised wavelet like basis. The LSF and LSW processes (5.1, 5.4) form classes of stochastic processes respectively constructed over Fourier and wavelet basis functions. In both cases, a connection is made from the increasing set of time points $t = 1, \ldots, T$ to a continuous function over the restricted interval $z = t/T \in (0, 1)$. In this sense, they allow us to asymptotically represent, and recover the second order, autocovariance properties of a process, even if these are non-stationary. However, while the processes can now be non-stationary, they must maintain appropriate smoothness constraints to enable spectral identification. For example, in this chapter, we assumed that the underlying spectral transfer function $W(z)$ is Lipschitz smooth. Such assumptions place limits on the range of processes LSW models can represent, for example, they do not permit sharp jumps in

the wavelet spectra. However, in real dynamic systems, such sharp jumps may be present, and are important to detect; for instance, one may consider searching for structural breaks in financial time-series, or edges in textured images. In the following chapters, such smoothness assumptions are considered in the context of regularised estimators for the LSW spectra.

Appendix D

D.1 Wavelet Thresholding

Wavelet thresholding typically refers to the method of selecting certain active coefficients in a wavelet decomposition by thresholding the empirically obtained coefficients. Traditionally, such thresholding is performed in order to recover a function f_t from noisy measurements $\{x_t\}_{t=1}^T$. For example, we may observe the process $\{X_{t;T}\}$, i.e:

$$X_{t;T} = f(t/T) + Z_{t;f} \ , \tag{D.1}$$

where $Z_{t;f}$ is an independently sampled noise term, c.f. $Z_{t;f} \sim \mathcal{N}(0, \sigma^2)$. Let $v_{j,k}$ be the discrete wavelet coefficients of an observed sequence $\{x_t\}$. A somewhat ideal estimate of the function from the wavelet coefficients $v_{j,k}$ is defined via the *selective wavelet construction* as

$$\hat{f}_t = T_{\mathrm{SW}}(x_t, \delta) := \sum_{j,k \in \mathcal{S}_0} v_{j,k} \psi_{j,k} \ ,$$

where $\mathcal{S}_0$ is a finite list of (j,k) pairs. In reality, we do not know $\mathcal{S}_0$ and need to estimate this set, thresholding provides one way to achieve this. However, as when dealing with parameter selection in linear regression, there are many different thresholding functions available and several ways to set appropriate thresholds. In this section, I aim to give a brief review of wavelet thresholding methods proposed by Donoho, Johnstone, Neumann et al. throughout the 1990's (D. L. Donoho et al. 1995; D. Donoho 1995; D. Donoho et al. 1994, 1998; M. Neumann et al. 1995).

Remark. *Second-stage smoothing vs general functional recovery*

The discussion here is aimed at the general recovery of a function $f(\cdot)$ in the presence of noise $\{Z_{t;f}\}$. Traditionally, theory on denoising is developed assuming additive Gaussian noise. However, results for non-Gaussian smoothing also exist such as the work by M. Neumann et al. (1995) (these are covered briefly in Remark .4). With regards to LSW spectral estimation (as discussed in Sec. 5.4.2), we treat each scale level $I_{j,k}$ as the noisy realisations of the

spectral function $S(k/T)$. The setup here would mimic that of Eq. D.1 such that $I_{j,k} = S_j(k/T) + Z_{j,t;S}$.

For the purposes of this discussion, assume that a DWT of the process is performed alongside that of the true function, i.e. $\{Y_{j,k}\} = \mathrm{DWT}(\{X_{t;T}\})$ and $\{v_{j,k}\} = \mathrm{DWT}(\{f(t/T)\})$. At each scale level j of the DWT, we now assume that $Y_{j,k} = v_{j,k} + \epsilon Z_{j,k;v}$ where $Z_{j,k;v}$ is a noise term for time $k = 1, \ldots, 2^{J-j}$ and ϵ describes the scale of this noise. *Note: the exact distribution of $Z_{j,k;v}$ may be different from $Z_{j,k;f}$; the variables correspond respectively to noise sources in the time, and spectral domain.* In the following, let us temporarily drop the scale index j, and consider the noisy wavelet coefficients $\{Y_k\}_{k=1}^{N=2^{J-j}}$. If we assume a Gaussian noise source $Z_{k;v} \sim \mathcal{N}(0, \epsilon^2)$, then in the limit of large N we obtain $\lim_{N\to\infty} P[\{\max_t |Z_{t;v}| > \epsilon\sqrt{2\log(N)}\}] = 0$. Taking a threshold $\lambda = \epsilon\sqrt{2\log N}$, then it is unlikely that any contribution from the noise will breach this level. A particularly amazing fact, is that when using such a threshold in conjunction with hard/soft thresholding for the DWT the risk of the associated functional estimator is bounded to within a logarithmic factor of the oracle risk.

***Proposition* D.7.** *Theorem 1 (D. Donoho et al. 1994)*

Let $Y_k = v_k + \epsilon Z_{k;v}$ where $Z_{k;v} \sim N(0,1)$ and $\epsilon > 0$, defining the risk as $R(\hat{\boldsymbol{v}}, \boldsymbol{v}) := E[\|\hat{\boldsymbol{v}} - \boldsymbol{v}\|_2^2]$ for $\hat{\boldsymbol{v}} = \mathrm{soft}(\boldsymbol{y}; \epsilon\sqrt{2\log N})$ we obtain

$$R_{\mathrm{univ}}(\hat{\boldsymbol{v}}, \boldsymbol{v}) \leq (2\log N + 1)\left(\epsilon^2 + \sum_{k=1}^{N} \min(|v_k|^2, \epsilon^2)\right) .$$

While setting $\lambda = \epsilon\sqrt{2\log N}$ allows us to bound the risk, it is interesting to ask whether there is a some sense more optimal setting of λ. If we only have one observation of a random variable $\Gamma \sim \mathcal{N}(\mu, 1)$ then defining $\rho_{ST}(\lambda, \mu) := E[\{\mathrm{soft}(\Gamma, \lambda) - \mu\}^2]$ one can proceed by introducing the minimax quantities

$$\Lambda_N^* = \inf_{\lambda} \sup_{\mu} \frac{\rho_{ST}(\lambda, \mu)}{N^{-1} + \min(\mu^2, 1)} .$$

Selecting a threshold λ_N^* which is the largest λ attaining Λ_N^* enables the tighter bound:

***Proposition* D.8.** *Minimax Risk - Theorem 2 (D. Donoho et al. 1994)*

Defining the soft-thresholding estimator $\hat{\boldsymbol{v}}^* = \text{soft}(\boldsymbol{y}, \lambda_N^* \epsilon)$*, with known* ϵ *one can obtain the bound*

$$R(\hat{\boldsymbol{v}}^*, \boldsymbol{v}) \leq \Lambda_N^* \left(\epsilon^2 + \sum_{k=1}^{N} \min(v_k^2, \epsilon^2) \right) .$$

As one might expect, utilising this somehow optimised threshold we find both the resulting multiplier and threshold themselves are reduced in comparison to those of Prop. D.7. In particular, $\Lambda_N^* \leq 2\log N + 1$ and $\lambda_N^* \leq \sqrt{2\log N}$. However, asymptotically as $N \to \infty$, for any $\epsilon > 0$ one finds results broadly similar to Prop. D.7, i.e.

$$\Lambda_N^* \sim 2\log N , \quad \lambda_N^* \sim \sqrt{2\log N} .$$

With regards to recovering the function $f(t/T)$, we are interested, not in the recovery of the second-stage wavelet coefficients $v_{j,k}$, but the function itself. If we define $\hat{f}^*(t/T)$ as the inverse wavelet transform of the minimax thresholded coefficients $\hat{\boldsymbol{v}}^*$, then it is possible to translate results on the estimation of the coefficients $v_{j,k}$ to the function itself.

Corollary. *Universal Thresholding*

Following the above definitions, and Props. D.7, D.8, the risk $\|\hat{f}^* - f\|_2^2$ *can be bounded for all* f *and* $T = 2^{J+1}$ *according to*

$$R(\hat{f}^*, f) \leq \Lambda_T^* \left(\frac{\sigma^2}{T} + R_0(\text{SW}, f) \right) ,$$

where $R_0(\text{SW}, f) = \inf_{\mathcal{S}} R_{T,\mathcal{S}}(T_{SW}(x, \mathcal{S}), f)$ *is the* oracle *risk (it selects the best subset of coefficients for reconstruction). The asymptotic limit of the minimax threshold motivates the application of what is known as the universal threshold:*

$$\lambda^{\text{univ}} = \hat{\sigma}\sqrt{2\log T} .$$

Remark .4. *Application to periodogram smoothing*

With regards to estimation of the LSW spectrum, specific studies by H. Neumann (1996) and M. Neumann et al. (1995) establish error bounds over functions in Besov balls $\mathcal{F} = B_{p,q}^m$*. In the Gaussian case, setting* $\lambda = 2\log(T)/T^{1/2}$ *for all* j *leads to:*

$$\sup_{f \in \mathcal{F}} \left\{ E[\|\hat{f} - f\|_{L_2}^2] \right\} = \mathcal{O}\left(T^{-2m/(2m+1)} (\log(T))^2 \right) .$$

In the case of non-Gaussian time-series, let $\mathcal{J}_T = \{(j,k)\,|\,2^j \leq T^{1-\eta}\}$ *for some* $\eta > 0$*, and set the universal threshold as* $\lambda = \max_{j,k\in\mathcal{J}_n}\{\sigma_{j,k}\}\sqrt{2\log|\mathcal{J}_n|}$*. Then*

$$\sup_{f\in\mathcal{F}}\left\{E[\|\hat{f}-f\|_{L_2}^2]\right\} = \mathcal{O}\left((\log(T)/T)^{-2m/(2m+1)}\right) .$$

For more detail on the above results the reader is referred to Theorem 3.2a),b) in M. Neumann et al. (1995). The threshold given in Theorem 5.1, as suggested by G.P. Nason et al. (2000) satisfies the Gaussian case above, and uses the result with $m = 1$*,* $p = 1$*.*

It is interesting to note the relation between wavelet denoising and the lasso (Sec. 2.1.5), whereby, in the orthogonal design situation, the lasso becomes a thresholding operation. Indeed, the DWT is equivalent to this situation, where given the orthogonality of the wavelet basis the wavelet model is simply a linear regression model with orthogonal design. The λ in the lasso is therefore directly related to the thresholds discussed above. In the next chapter, this relation is highlighted in greater detail in the context of smoothing the LSW spectra.

Chapter 6

Regularised Estimation of LSW Spectra

At the end of the previous chapter we reviewed a set of methods for spectral estimation in LSW processes. The aim of this chapter, is to demonstrate how regularised estimation tools discussed in earlier chapters may be utilised for spectral estimation. As may already be clear, there are strong parallels between work on wavelet denoising (c.f. Appendix D.1) and that of sparse statistical estimation methods such as the lasso. Furthermore, as discussed in Chapter 2, there already exists a substantial framework for looking at the theoretical and empirical properties of regularised M-estimators. Such estimators provide additional modelling flexibility to statisticians when compared to standard wavelet thresholding techiques. For example, in the linear regression setting, methods such as the group-lasso, fused-lasso, and generalised lasso enable one to easily incorporate broad classes of prior knowledge into point estimation. In the context of spectral estimation, one can also make use of similar priors, for instance to promote sparsity at certain scales in the spectra, or grouping spectral changes.

In this chapter such sparsity constraints are utilised to restrict the variation of spectral estimates. In particular the Lipschitz smooth LSW definition is modified to allow for sudden jumps or changepoints in the spectra. Additionally, an extension of the LSW processes to describe 2-dimensional random

processes otherwise known as random fields is considered. The main contribution of this chapter is the development of a set of regularised estimators for the LSW spectrum. Their performance is then evaluated alongside existing methods such as moving average smoothing and second-stage wavelet denoising.

6.1 Piecewise-constant LSW processes

The original formulation of the LSW process (5.3.1) has several assumptions which limit its practicality. For example, the canonical formulation by G.P. Nason et al. (2000), has a requirement that the spectrum evolve in a Lipschitz continuous manner. However, many real-world processes may be expected to break such assumptions and the spectral structure of a time-series may adapt rapidly in certain situations, for example; stock prices may react very quickly to economic news, brain activity may change rapidly in structure at the start of a seizure, or when a person is asked to perform different tasks. Motivated by such requirements Fryzlewicz et al. (2006) extended the definition of the LSW process to accommodate changepoints and sharp discontinuities. Others to consider such piecewise constant spectral estimation are Killick et al. (2013), Van Bellegem et al. (2008) and Cho et al. (2012, 2015). Adapting to piecewise constant spectra requires modifications to the smoothness assumptions of the LSW model, which in turn motivate different classes of spectral estimators.

***Definition* 6.1.** *Piecewise-Constant LSW Process*

Let $\{X_{t;T}\}$ have with mean-square representation given by Eq. 5.3.1 (as per G.P. Nason et al. (2000)). However, instead of assuming Lipschitz continuity according to Ass. 5.1, we assume $W_j(z) \in \mathbb{R}$ is a piecewise constant function with a finite, but unknown number of jumps. Let C_j be the total magnitude of jumps in the spectrum $S_j(z) \equiv |W_j(z)|^2$ at scale level j. The total magnitude of jumps in the spectrum is thus controlled according to

$$\sum_{j=1}^{\infty} C_j 2^j < \infty \,,$$

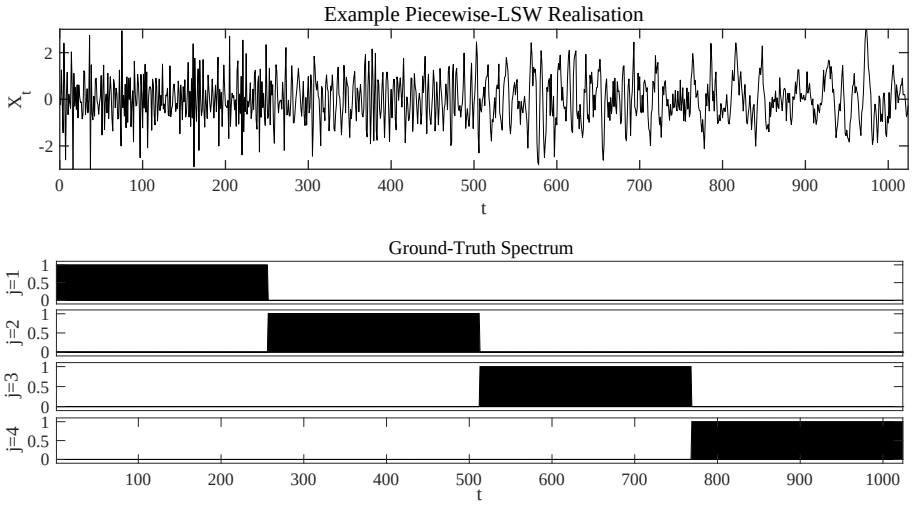


Figure 6.1.1 – Top: Example realisation from LSW process with spectra given below. Bottom: Depiction of the ground-truth spectra (for $j > 4$ $S_j(t/T) = 0$).

and $\sum_{j=1}^{\infty} S_j(z) < \infty$ *uniformly in* $z \in (0, 1]$. *If the jumps are bounded according to above, then the process* $\{X_{t;T}\}$ *is referred to as a piecewise-constant LSW process.*

Having defined the modification to the process in terms of piecewise constant spectra, we now need to modify the estimator to be able to track such changes. However, as noted by Fryzlewicz et al. (2006), a single element of the periodogram appears marginally distributed as a scaled χ^2_1 random variable, such that asymptotically:

$$I_{j,k} \sim E[I_{j,k}]Z_k^2 \ , \quad Z_k \sim \mathcal{N}(0, 1) \ ,$$

where the Z_k are correlated with each other. If our end-goal is estimation of the spectra, then the resulting problem of recovering $E[I_{j,k}] \approx \sum_l A_{j,l} S_l(z)$ is quite a different from the task of functional estimation in the presence of additive noise as is typically faced when performing denoising, for example via wavelet thresholding (Appendix D.1). In addition to the multiplicative noise structure, we note that the signal-to-noise ratio at each scale level is particularly low, as mentioned previously $E[I_{j,k}]/\sqrt{\mathrm{Var}[I_{j,k}]} = 2^{-1/2}$.

To deal with these issues Fryzlewicz et al. (2006) introduce a variance stabilising operation known as the Haar-Fisz transform. A summary of the

Haar-Fisz transform and it's properties when applied to piecewise constant χ^2 noise are given below:

***Definition* 6.2.** *Haar-Fisz Transform*

(1) Let[1] $H = \log_2 T$ *and* $s_{H,k} := Y_k^2 = I_{j,k}$ *for* $k = 1, \ldots, 2^J$.

(2) Starting at the course scales, for $h = H-1, H-2 \ldots, 0$ *construct* 2^h*-dimensional vectors* $\boldsymbol{s}_h, \boldsymbol{d}_h, \boldsymbol{f}_h$ *of the form*

$$s_{h,k} = (s_{h+1,2k} - s_{h+1,2k+1})/2$$
$$f_{h,k} = (s_{h+1,2k} - s_{h+1,2k+1})/2s_{h,k} \ ,$$

where $k = 1, \ldots, 2^h$.

(3) Now, going from fine to coarse scale modify the vectors $\boldsymbol{s}_{h+1}$

$$s_{h+1,2k} = s_{h,k} + f_{h,k}$$
$$s_{h+1,2k+1} = s_{h,k} - f_{h,k} \ ,$$

where $k = 1, \ldots, 2^h$.

(4) Define $U_n = \mathcal{F}\{Y_k^2\} := s_{H,k}$ *for* $k = 1, \ldots, 2^H - 1$ *as the Haar-Fisz transform of* Y_k^2 *and* $\mathcal{F}\{\}$ *as the Haar-Fisz operator.*

***Proposition* 6.1.** *Properties of the Haar-Fisz transform*

Let $Y_{t,T}^2 = \alpha(t/T)Z_{t;T}^2$ *for* $t = 1, \ldots, T$ *where* $\alpha(z)$ *is a piecewise constant function bounded from above, away from zero and with a finite number of jumps. Additionally, let* $Z_{t;T} \sim \mathcal{N}(0,1)$ *and be drawn i.i.d over time*[2].

Under these assumptions Fryzlewicz et al. (2006) demonstrate that the Haar-Fisz transform asymptotically achieves:

$$\mathcal{F}\{Y_{t;T}^2\} - \overline{Y^2} \approx (\mathcal{F}\{\alpha(t/T)\} - \alpha(t/T)) + (\mathcal{F}\{Z_{t;T}^2\} - \overline{Z^2}) \ , \tag{6.1.1}$$

where $\overline{Z^2}$ *is the sample mean over* $t = 1, \ldots, T$. *Additionally, the variance is stabilised, whereby* $\sigma^2 := \mathrm{Var}[\mathcal{F}Z_{t;T}^2] = \sum_{h=1}^{H}(2^{h-1}+1)^{-1} + 2^{-H}$. *The transformed noise is asymptotically normal*

$$\mathcal{F}^{(H_\eta)}\{Z_{t;T}^2\} \xrightarrow{d} \mathcal{N}(0, \sigma^2) \ ,$$

[1] I here distinguish between H and J which respectively refer to the maximum scale considered in the Haar-Fisz transform and the original LSW construction.

[2] Note: there is no correlation between time points in this model which contrasts with the case of the raw-periodogram $I_{j,k}$.

with no additional correlation introduced

$$\sigma^{-1}\mathrm{Cov}[\mathcal{F}^{(H_\eta)}Z^2_{t;T}, \mathcal{F}^{(H_\eta)}Z^2_{t';T}] \to 0 \ ,$$

for any t, t' *as* $H \to \infty$, $H_\eta = \lfloor (1-\eta)H \rfloor$ *and* $\eta \in (0,1)$.

Proof. For proof of the above the reader is referred to Fryzlewicz et al. (2006) (Proposition 6.1).

Unlike the log-transform, which is sometimes used to try and Gaussianise variables, the Haar-Fisz transform does not introduce a bias, while it roughly translates the multiplicative noise structure into an additive one (see Eq. 6.1.1). This is very attractive for spectral estimation as it allows the principled application of methods developed for smoothing assuming Gaussian additive noise, i.e. wavelet thresholding. In the next section, we will discuss the efficiency of regularised spectral estimation methods both with and without the Haar-Fisz transform.

6.2 Fused Lasso for Spectral Estimation

In Chapter 3, a regularised smoothing methodology for dynamic graphical models was demonstrated. In particular, one may recall that the M-estimators examined combined a smooth likelihood with a non-smooth regulariser. In the graphical model case, we implemented a method which imposed both smoothness and sparsity constraints jointly. In this section, we will investigate the application of a similar class of fused estimator, but where we only utilise the smoothing part of the regulariser. Specifically, we will apply the fused lasso estimator of R. Tibshirani et al. (2005) to the task of smoothing the LSW periodogram. Firstly, we apply the fused lasso, or piecewise constant trend filtering (R.J. Tibshirani 2014) to smooth the raw periodogram $I_{j,k}$, then secondly we apply such smoothing to the Haar-Fisz transformed periodogram $\mathcal{F}\{I_{j,k}\}$.

Given a piecewise LSW process $X_{t;T}$, consider its vectorised periodogram $\boldsymbol{I}_j := (|d_{j,1}|^2, \ldots, |d_{j,T}|^2)$ for $j = 1, \ldots, \log_2 T$. Furthermore, define the differencing matrix $\boldsymbol{D} \in \mathbb{R}^{(T-1)\times T}$ of the form

$$\boldsymbol{D} := \begin{pmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix} . \tag{6.2.1}$$

The fused-lasso proximity operator is defined as

$$\text{prox}_{FL}(\boldsymbol{\Gamma}; \lambda_1, \lambda_2) = \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^T} \|\boldsymbol{\Gamma} - \boldsymbol{\beta}\|_2^2 + \lambda_1\|\boldsymbol{\beta}\|_1 + \lambda_2\|\boldsymbol{D}\boldsymbol{\beta}\|_1 . \tag{6.2.2}$$

Applying this to the periodogram we obtain the fused-lasso (FL) periodogram estimator:

$$\hat{I}^{\text{FL}}_{j,k} := \text{prox}_{\text{FL}}(\boldsymbol{I}_j; \lambda_1, \lambda_2) . \tag{6.2.3}$$

Alternatively, a Haar-Fisz fused estimator (HF-F) may be defined as

$$\hat{I}^{\text{HF+F}}_{j,k} := \mathcal{F}^{-1}(\text{prox}_{\text{FL}}(\mathcal{F}\{\boldsymbol{I}_j\}; \lambda_1, \lambda_2)) . \tag{6.2.4}$$

Given a smoothed estimate of the periodogram, we can de-bias in the usual way, for example $\hat{S}^{\text{FL}}_j(k/T) = \sum_{l=1}^J A^{-1}_{l,j} \hat{I}^{\text{FL}}_{j,k}$.

6.2.1 Relation to Haar-Wavelet Denoising

As discussed in R.J. Tibshirani and J. Taylor (2011), the wavelet thresholding applied in the context of signal denoising (c.f. Appendix D.1) may be thought of as solving a lasso problem. Specifically, let us consider the function with additive noise

$$X_{t;T} = f(t/T) + Z_{t;f}$$

where $Z_{t;f}$ is an independently sampled noise term. Now, assume we obtain T observations of the process $\boldsymbol{x} = (x_1, \ldots, x_T)$, a point estimate of the wavelet coefficients $\boldsymbol{\beta}$ may be obtained through solving the lasso problem:

$$\hat{\boldsymbol{\beta}} := \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^T} \|\boldsymbol{x} - \boldsymbol{W}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1 , \tag{6.2.5}$$

where $\boldsymbol{W} \in \mathbb{R}^{T\times T}$ has an orthogonal set of wavelet basis as it's columns. The signal can then be approximated via $\hat{\boldsymbol{x}} = \boldsymbol{W}\hat{\boldsymbol{\beta}}$. In the context of signal approximation (6.2.5) can be transformed utilising orthogonality of $\boldsymbol{W}$ into the equivalent generalised lasso problem. For example, substituting $\boldsymbol{\theta} = \boldsymbol{W}\boldsymbol{\beta}$ we obtain:

$$\hat{\boldsymbol{f}} := \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^T} \|\boldsymbol{x} - \boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{W}^\top\boldsymbol{\theta}\|_1 \,, \tag{6.2.6}$$

where $\hat{\boldsymbol{f}} = (\hat{f}(1/T), \ldots, \hat{f}(1))$. The above representation demonstrates a clear link between wavelet thresholding schemes and the fused lasso scheme for signal denoising. When considering smoothing the empirical periodogram, one may compare smoothers of the form (6.2.3) with that of the generalised lasso problem in Eq. 6.2.6. In the setting where $\lambda_1 = 0$, such that there is no sparsity penalty, the difference between wavelet thresholding and fused estimation is due to the form of the transformation matrix $\boldsymbol{W}^\top$ vs $\boldsymbol{D}$. For the Haar family we obtain

$$\boldsymbol{W}^\top = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & \ddots \\ & 0 & \sqrt{2} & -\sqrt{2} \end{pmatrix} \quad \boldsymbol{D} := \begin{pmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix} ,$$

and note the clear similarity in the form of these transformations. The main difference between the transforms is that in the case of fused estimation the support of the rows in $\boldsymbol{D}$ overlaps. Such an overlap is reminiscent of that of the non-decimated wavelet transform, but only considering the finest scale levels. It suggests that the fused estimator may possess the translation invariant properties of the NDWT, but only requires a dictionary of $T-1$ rows, it only models the jumps in the data. In R.J. Tibshirani (2014) it is suggested that a trend filtering approach, a generalised form of using fused lasso for signal approximation, may outperform wavelet denoising in certain situations. In the following section the aim is to compare the empirical performance of fused lasso smoothing, and that of a second-stage Haar wavelet transform (with thresholding). It is noted that Fryzlewicz et al. (2006) utilise a similar second-stage Haar wavelet transform to perform smoothing on the Haar-Fisz transformed periodogram.

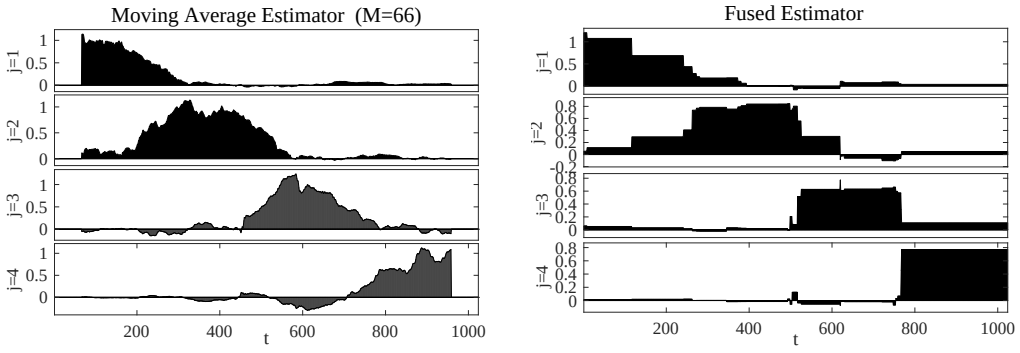


Figure 6.2.1 – Left: moving average estimator for the EWS (as defined in Def. 5.7) with half width $M = 66$. Right: Example of spectral estimation the regularised FL-LSW estimator with $\lambda_2 = 12$.

6.3 Synthetic Experiments

In order to assess the performance of the proposed estimators a piecewise constant spectrum is constructed. The energy of the spectra is distributed across the first 4 levels similarly to the example provided in Fryzlewicz et al. (2006). To quantify uncertainty in the spectral estimation $N = 100$ LSW processes are drawn from this spectrum, both at lengths of $T = 512$ and $T = 1024$. The models, as previously introduced are then fit to the individual processes for a wide range of tuning parameters. A simple moving window corresponding to the kernel estimator (Eq. 5.4.5) with a uniform kernel is also included. Varying the tuning parameters allows for one to select in some sense an optimum threshold for the simulated data and thus allowing fair comparison across methods. *Note: in real applications we do not have access to the ground-truth spectra, thus an alternative tuning mechanism must be found for setting the parameters λ_1, λ_2. This may utilise some form of cross-validation, if we have repeated draws of a stochastic process, or via some form of in-sample model complexity measure, c.f. BIC.*

For regularised methods which are applied directly to the raw wavelet periodogram; for example, the fused lasso spectral estimator $\hat{S}_j^{\mathrm{FL}}(k/T)$; tuning parameters are scaled such that $\lambda_F^{(j)} = \lambda_F^{(j)} 2^{j-1}$. This compensates for the natural increase in variance at coarser scale levels (due to the increased size of the wavelet support). For the smoothing methods that are applied to the

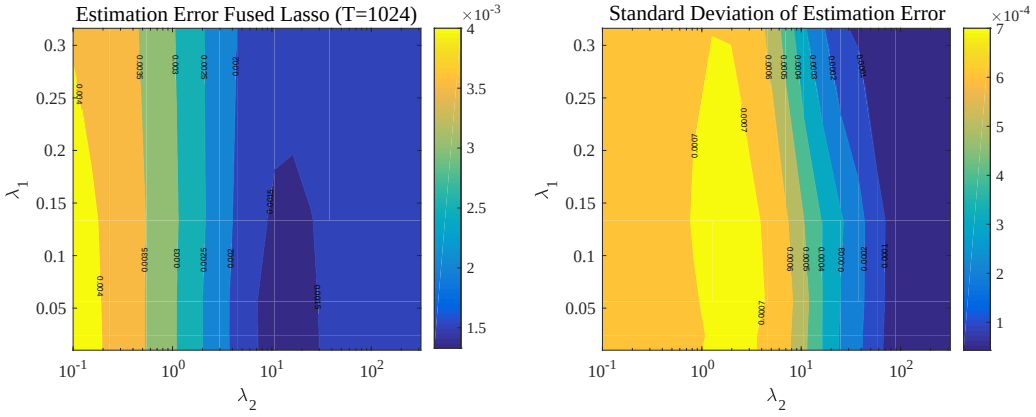


Figure 6.3.1 – Left: error surface for Fused Lasso (without Haar-Fisz transform) spectral estimator at $T = 1024$. Right: the associated standard deviation of error.

Haar-Fisz transformed periodogram, the thresholds are not increased for each scale as the transform has a variance stabilising effect. In these experiments the moving window smoother, the window width $2M + 1$ is kept constant for all scale levels. One could argue that such a smoothing window should also be scaled with the expected increase in variance for larger scales analogous to λ_F. However, in this case, the window is not scaled as when considering estimation at the boundaries (near $t = 1$ or $t = T$) the edge effects would vary for the different scales.

To track the estimation error of the different methods a very simple sum of squares criterion is utilised, of the form

$$\epsilon_T := \frac{1}{JT} \sum_{j=1}^{J} \sum_{k=1}^{T} \|\hat{S}_j(k/T) - S_j(k/T)\|_2^2 \, .$$

When interpreting results one should note that ϵ_T averages over all scales $j = 1, \ldots J = \log_2 T$; however, the true signal presents zero contribution from coarse scales ($j > 4$). The measure may therefore be slightly biased towards methods which can smooth the periodogram more effectively when the true spectrum is zero, i.e. $S_j(k/T) = 0$.

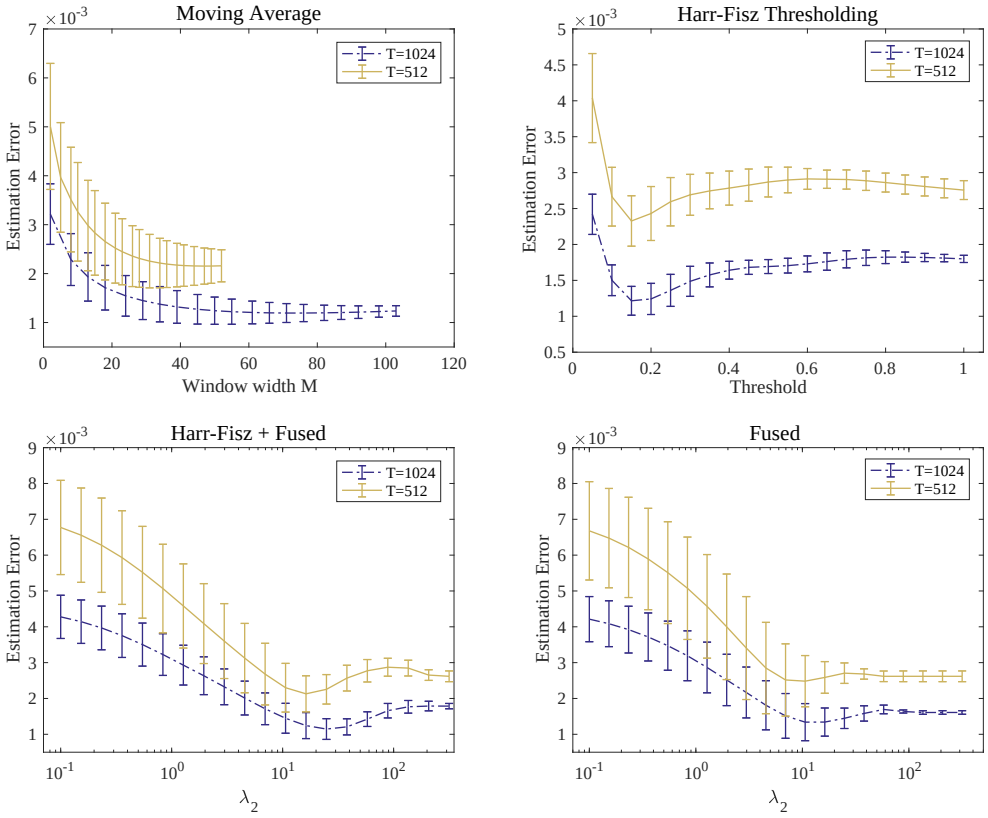


Figure 6.3.2 – Analysis of estimation error for two sample sizes, the statistics above are taken across $N = 100$ simulated piecewise LSW processes.

6.3.1 Results and Comparison of Methods

Results are summarised in Figs. 6.3.1, 6.3.2, 6.3.3 and Table 1. The first two figures demonstrate how performance varies as a function of tuning parameters, while the third figure and the table summarise the performance of the different methods with tuning parameters set at their optimal level (according to the minimisation of ϵ_T).

Let us first consider the results of the parameter sweep. When considering the fused estimator without Haar-Fisz (Figure 6.3.1), there is no clear benefit to implementing a sparsity inducing prior at the periodogram level. This is demonstrated through the parameter sweep where the minimum error is attained at $\lambda_1 = 0$. It also appears that the variance of the estimator increases for non-zero λ_1 suggesting the sparsity assumption is inappropriate when use

in conjunction with the raw periodogram. Consider that the error results (as given in the figures) are calculated in the de-biased reference frame, i.e. after we have linearly transformed the estimates with $\boldsymbol{A}_J^{-1}$. However, while the ground-truth spectrum is sparse across scales; it is not necessarily sparse in the frame of the periodogram (due to bias). In contrast with the sparsity parameter, minimising the error surface with respect to the λ_2 suggests that the smoothing part of the fused lasso estimator is required. The specific level of smoothness required will in general be dependent on the ground-truth $S_j(z)$. All cases in Figures 6.3.1, 6.3.2 demonstrate some minima is obtained for a non-zero smoothing parameter.

The results of Figure 6.3.2 and Table 1 lend evidence as to the benefit of utilising a variance stabilising transform (i.e. Haar-Fisz). Specifically, we note the Haar-Fisz (DWT and Fused) methods have reduced error variance and mean error at both $T = 512$ and $T = 1024$. The moving average estimator appears to perform well with regards to the error metric used, however, is slightly misleading as it is unable to recover the piecewise nature of the spectrum. This can be seen qualitatively in Figure 6.2.1 which displays a single realisation of estimates for the moving average and fused estimator (with $\lambda_1 = 0$). In addition to being unable to adapt to the piecewise smoothness, the moving window method has significant edge effects, in these experiments the moving average spectra is only estimated across the interval $t = M, \ldots, T - M$ to ensure the full window can be utilised. If the spectrum was smoothed right to the edge, with zero padding, we would expect the variance of the estimator at the edges to increase in-line with the number of periodogram observations in the smoothing window.

Furthermore, while the fused estimators, applied to both the Haar-Fisz, and raw periodogram appear to have similar qualitative form, i.e. piecewise constant; it is noted that both the mean and variance of the error when applied to the Haar-Fisz transformed periodogram is reduced (compare the bottom row of Figure 6.3.3). Perhaps the most interesting comparison, is between the performance of the second stage wavelet thresholding (Haar DWT + Thresholding) approach, and that of the fused estimator when applied to the Haar-Fisz periodogram (Top-Right, Bottom-Left of Figure 6.3.3). When viewing an individual example estimate of the Fused estimator we obtain structure

Table 1 – Summary of optimised estimation performance. Haar-Fisz (DWT) refers to using a second stage Haar-DWT with thresholding applied to the Haar-Fisz transformed periodogram, Haar-Fisz (Fused) is the smoothing method of Eq. 6.2.4.

Method	ϵ_{512} $(\times 10^{-3})$	Std(ϵ_{512})	ϵ_{1024}	Std(ϵ_{1024})
Moving Average	2.2	0.37	1.2	0.19
Haar-Fisz (DWT)	2.3	0.35	1.2	0.20
Haar-Fisz (Fused $\lambda_s = 0$)	2.1	0.50	1.1	0.29
Fused ($\lambda_s = 0$)	2.5	0.71	1.3	0.52
Fused (+ shrinkage)	2.5	0.71	1.3	0.53

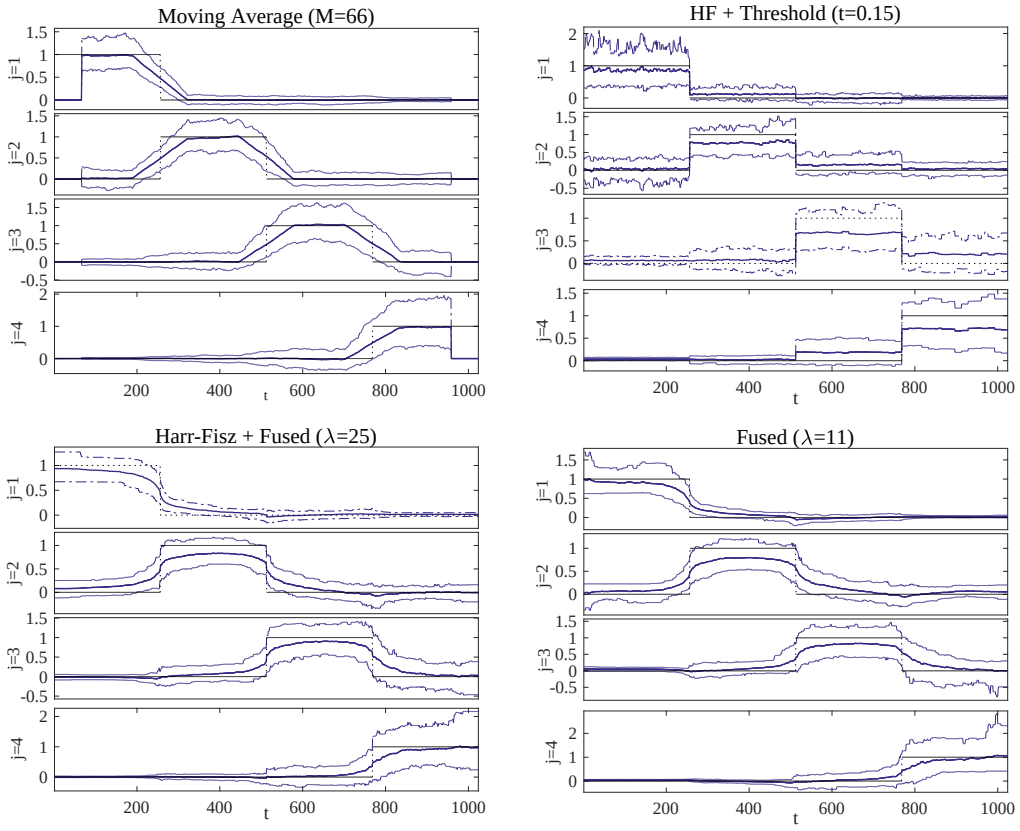


Figure 6.3.3 – Summary of estimation methods for piecewise constant LSW process with ground-truth structure as depicted in Figure 6.1.1. Tuning parameters are selected via error minimisation across the set of $N = 100$ experiments as described in Figure 6.3.2. Dashed lines indicate empirical 5th and 95th percentiles.

similar to that of (Figure 6.2.1). However, when looking at the average error over $N = 100$ realisations we do not see the clear jump in the spectra that is obtained in the case of DWT smoothing. As a conclusion, it appears that the Haar-DWT smoothing is more likely to recover the correct jump positions (changepoints), however, the actual spectral estimates appear to have more bias than that of the fused estimation scheme.

In summary, it has been demonstrate that the fused lasso estimator can be successfully used to identify piecewise constant spectra. It's performance is comparable with previously proposed second-stage DWT approaches such as those suggested by Fryzlewicz et al. (2006). Additionally, while simple moving window estimators result in viable strategies for recovering smoothly varying spectra, they are not suitable for settings where we expect a rapid change (or jump) in the spectral structure. The experiments here serve as empirical motivation to further investigate the theoretical properties of fused lasso type estimators for spectral estimation. In the next section, we consider an extension of the LSW process and regularised estimation to multiple dimensional fields such as images.

6.4 Piecewise Stationary Wavelet Fields

Many applications of wavelet models do not deal with time-series, but rather higher-dimensional ordered fields such as images or videos; the latter can be thought of a combination of temporal and spatial dimensions. While most of the work in this thesis focuses on the understanding dynamics in a single dimension, i.e. time, there is no reason it cannot be extended to higher orders. The original extension of the LSW framework and theory to 2-dimensional fields led to new applications in the analysis of textured images (Eckley et al. 2010). As in the one-dimensional setting, the well-principled design of the LSW process means that statistical properties such as asymptotic bias and variance of estimators are relatively well understood (c.f. Sec. 5.4, G. Nason (2013) and G.P. Nason et al. (2000)). As such, the estimated LSW spectrum can be analysed for the detection of non-stationarities which can aid in anomaly detection tasks (Nunes et al. 2014; S. Taylor et al. 2014). However, in a finite sample setting, one can often see inconsistencies between the estimates of the spectra and the modelling assumptions of the underlying

LSW process. In the previous section, a regularised smoothing approach enabled the incorporation of prior knowledge in terms of piecewise smoothness constraints. To this end, in the two dimensional setting one may regularise the estimator to try and enhance the finite sample performance of spectral estimation. It is here considered to perform such regularised smoothing to extract a piecewise constant and positive spectral estimate across a two-dimensional spectral field. Potential applications include performing texture segmentation or feature extraction for higher level image processing tasks such as classification or anomaly detection. An example of the method applied to a real image is demonstrated at the end of the chapter.

6.4.1 The Two Dimensional LSW-Process

In order to decompose a two dimensional field it is first required to extend the set of discrete wavelet functions so they can cover this extended space. Similarly to the univariate case, we here define a set of two-dimensional discrete wavelet filters $\{\psi^{(l)}_{j,\boldsymbol{k}}\}$; however, the positional index is now a vector such that $\boldsymbol{k} = (k_1, k_2)^\top \in \mathbb{Z}^2$. Specifically, the wavelets are a set of square matrices of size $N_j \times N_j$ and the size of the basis is given as $N_j = (2^j - 1)(N_h - 1) + 1$, where N_h is the number of non-zero elements in the associated low-pass mirror filter. In two dimensions wavelets may assume different orientations, this gives rise to an additional directional index $l \in \{h, v, d\}$ relating to the horizontal, vertical, and diagonal directions (see Figure 6.4.1). The elements of $\psi^{(l)}_{j,\boldsymbol{k}}$ are defined through the tensor products of the corresponding one-dimensional wavelets. In the horizontal direction h we have $\psi^{(h)}_{j,\boldsymbol{k}} = \phi_{j,k_1}\psi_{j,k_2}$, in the vertical direction $\psi^{(v)}_{j,\boldsymbol{k}} = \psi_{j,k_1}\phi_{j,k_2}$ and in the diagonal direction $\psi^{(d)}_{j,\boldsymbol{k}} = \psi_{j,k_1}\psi_{j,k_2}$, where $\phi_{j,k}$ are the associated father/scaling wavelets.

Again, as in the univariate case the LSW process is constructed with respect to the set of non-decimated wavelets such that there is an equivalent number of wavelet coefficients at each scale level $j = 1, \ldots, J$. To achieve this we simply translate the discrete wavelets $\{\psi^l_j\}$ over the space $\mathbb{Z}^2$, the non-decimated set of discrete wavelets are defined as $\psi^l_{j,\boldsymbol{u}}[\boldsymbol{r}] := \psi^l_{j,\boldsymbol{u}-\boldsymbol{r}}$ for all j, l, and $\boldsymbol{u}, \boldsymbol{r} \in \mathbb{Z}^2$.

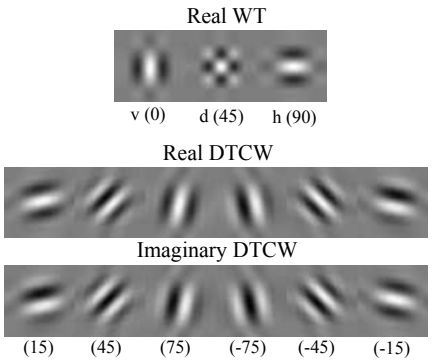


Figure 6.4.1 – Example of 2-dimensional wavelet filters. Top: the standard real-valued wavelets as discussed in this work. Bottom: the corresponding set of Dual-Tree Complex Wavelet (DTCW) filters within a complex LSW framework (see Remark 6.1).

***Definition* 6.3.** *2d LSW Process*

The 2d-locally stationary wavelet process is defined as a doubly indexed stochastic process over the field $\mathbb{R}^2$*. It is indexed by position* $\boldsymbol{r} = (r_1, r_2)$ *and image size* $\boldsymbol{R} = (R_1, R_2)$*. The process has a representation in the mean-square sense of:*

$$X_{\boldsymbol{r};\boldsymbol{R}} = \sum_{l} \sum_{j=1}^{\infty} \sum_{\boldsymbol{u} \in \mathbb{Z}^2} w^{(l)}_{j,\boldsymbol{u};\boldsymbol{R}} \psi^{(l)}_{j,\boldsymbol{u}}[\boldsymbol{r}] \epsilon^{(l)}_{j,\boldsymbol{u}} \ ,$$

where the summation over l *takes in three directions* $l = h, v, d$*. Again, the stochastic term* $\{\epsilon^{(l)}_{j,u}\}$ *encodes no dependency structure by itself and is an i.i.d zero-mean random variable, they are independent such that* $\epsilon^{(l)}_{j,u} \perp \epsilon^{(l')}_{j',u'}$ *for all* $j \neq j'$*,* $u \neq u'$ *and* $l \neq l'$*.*

As in the one-dimensional case, the transfer sequence $\{w^{(l)}_{j,\boldsymbol{u};\boldsymbol{R}}\}$ *is linked to an underlying spectrum confined to the interval* $(0, 1] \times (0, 1]$*. We require*

$$\sup_{\boldsymbol{u}} |w^{(l)}_{j,\boldsymbol{u};\boldsymbol{R}} - W^{(l)}_j(\boldsymbol{u}/\boldsymbol{R})| \leq C^{(l)}_j / T \ ,$$

where traditionally $W^{(l)}_j(\boldsymbol{u}/\boldsymbol{R})$ *may be assumed to be Lipschitz or otherwise continuous. However, we here allow this to vary sharply and posess discontinuities in a manner to the piecewise 1d model (Dfn. 6.1).*

***Remark* 6.1.** *Improved Resolution with the Dual-Tree Complex Wavelets*

Although not discussed here, the 2d-LSW model as presented may be expanded to use a set of Dual-Tree complex wavelets (DTCW) as originally introduced by Kingsbury (2001) and Selesnick et al. (2005). Such a family of directional wavelets allows a natural extension of the real valued wavelets and the construction of a complex valued LSW framework. The added flexibility of these models allows for an added degree of redundancy and therefore the can describe a greater range of textures. For example, in the 2d-case the DTCW LSW model results in enhanced directional resolution with six directional filters, as opposed to three in the real-valued framework presented here. For more information on such extensions the reader is directed to Nelson et al. (2016).

In previous works authors enforce smoothness on the process via a Lipschitz continuous transfer function $W_j^{(l)}(\boldsymbol{u}/\boldsymbol{R})$[3]. This function relates to the discrete amplitudes $w_{j,\boldsymbol{u}}^{(l)}$ such that the maximum deviation between the two decays asymptotically, i.e. $\sup_{\boldsymbol{u}} |w_{j,\boldsymbol{u};\boldsymbol{R}}^{(l)} - W_j^{(l)}(\boldsymbol{u}/\boldsymbol{R})| = \mathcal{O}(\min(\boldsymbol{R})^{-1})$. In the original LSW formulation (Eckley et al. 2010; G.P. Nason et al. 2000), this constraint is used to impose asymptotic smoothness on the transfer function. As sample size increases the authors then show a set of asymptotic properties that relate the auto-covariance (of the process) to the two dimensional Local Wavelet Spectrum (LWS)

$$S_j^{(l)}(\boldsymbol{u}/\boldsymbol{R}) = |W_j^{(l)}(\boldsymbol{u}/\boldsymbol{R})|^2$$

for $j = 1, \ldots, J_{\boldsymbol{R}} = \log_2(\min(R_1, R_2))$. Such results are analogous to their univariate counter-parts as discussed in Section 5.3.1; only subtle modifications are needed for controlling error with respect to $\min\{R_1, R_2\}$ as opposed to T.

In real images we should not realistically expect the spectral structure of images to vary in such a Lipschitz continuous manner, rather the spectral properties of the image may change abruptly; for instance where there exist two or more neighbouring regions of different materials or textures. As in the one-dimensional setting, an alternative is to construct piecewise-constant spectra such that variation is bounded through the size of the jumps. Such assumptions may be formalised along the lines of those in Definition 6.1.

[3]Note: In the position index $(\boldsymbol{u}/\boldsymbol{R})$ is taken to mean the pair of points $(u_1/R_1, u_2/R_2)$

The following section discusses the construction of regularised estimators for two dimensional piecewise-constant LSW processes. The estimation methods developed in this section are significantly different from previous works in that they estimate the spectrum jointly over $j = 1, \ldots, J$ instead of smoothing separately at each scale. While it may be possible to develop a formal theoretical framework for analysing such estimators, along the lines of Fryzlewicz et al. (2006), this is avoided at the current time and left as potential future work. As previously suggested, the point of this chapter and these early applications is to empirically motivate the development of a general regularised framework for spectral estimation.

6.4.2 Estimation for the 2d-LSW Spectrum

A natural estimator for the spectrum can be constructed around the *empirical periodogram* which is defined as $I^{(l)}_{j,\boldsymbol{r}} := |d^{(l)}_{j,\boldsymbol{r}}|^2$ where $d^{(l)}_{j,\boldsymbol{r}} = \sum_{\boldsymbol{k}\in[\boldsymbol{R}]} \psi^{(l)}_{j,\boldsymbol{r}}[\boldsymbol{k}] x_{\boldsymbol{r};\boldsymbol{R}}$ are the 2d-wavelet coefficients. It has previously been shown in the 1d-case (Prop. 5.5, G.P. Nason et al. (2000)), and 2d case (Eckley et al. 2010), that the empirical periodogram is by itself, both biased and inconsistent; more specifically

$$E[\hat{I}^{(l)}_{j,\boldsymbol{r}}] = \sum_{j,l} A_{(j,l),(j',l')} S^{(l')}_{j'}(\boldsymbol{r}/\boldsymbol{R}) + \mathcal{O}(\min(\boldsymbol{R})^{-1}) \; .$$

Similarly to the one-dimensional case, mixing between scales/directions is encoded by the matrix

$$\begin{aligned} A_{(j,l),(j',l')} &= \langle \Psi^{(l)}_j[\boldsymbol{k}], \Psi^{(l')}_{j'}[\boldsymbol{k}] \rangle \\ &= \sum_{\boldsymbol{k}} \Psi^{(l)}_j[\boldsymbol{k}] \Psi^{(l')}_{j'}[\boldsymbol{k}] \; , \end{aligned}$$

where $\Psi^{(l)}_j[\boldsymbol{\tau}] = \sum_{\boldsymbol{r}} \psi^{(l)}_{j,\boldsymbol{r}}[\boldsymbol{0}] \psi^{(l)}_{j,\boldsymbol{r}}[\boldsymbol{\tau}]$ is referred to as the auto-correlation wavelet (the summations over $\boldsymbol{r}, \boldsymbol{k}$ go over the support of the child wavelets $\psi^{(l)}_{j,\boldsymbol{r}} \psi^{(l)}_{j,\boldsymbol{r}}$). Such results are extensions of work in the 1D-case. However, rather than just mixing over scale, in the 2D-case we also observe a dispersion of power over direction l. Correspondingly, the biasing matrix $\boldsymbol{A}$ is now of size $JL \times JL$.

Inverting $\boldsymbol{A}$ now allows one to construct an unbiased estimator as $\hat{S}^{(l)}_{j,\boldsymbol{r}} = \sum_{j'l'} A^{-1}_{(j,l),(j',l')} I^{(l')}_{j',\boldsymbol{r}}$, although, as in the one-dimensional case this estimator is still not consistent, i.e. $\mathrm{Var}(\hat{S}^{(l)}_{j,\boldsymbol{r}}) \not\to 0$ as $\min(R_1, R_2) \to \infty$. To encourage

consistency one must perform some kind of smoothing over the image/samples. Typically the literature (Eckley et al. 2010; G.P. Nason et al. 2000) suggests to smooth $I^{(l)}_{j,\boldsymbol{r}}$ and then perform de-biasing, where smoothing is performed by either a second stage of wavelet transform and then thresholding, or adopting a moving average/kernel smoother. It is worth noting that all the pros/cons of smoothing methods in the one-dimensional case almost directly translate to the two-dimensional and higher order settings.

In particular, in this section we consider comparison to the de-biased kernel estimator given as:

$$\hat{K}^{(l)}_{j,\boldsymbol{r}} = \frac{1}{H} \sum_{\boldsymbol{k}=\boldsymbol{r}-\boldsymbol{M}}^{\boldsymbol{M}} \left(\sum_{j',l'} A^{-1}_{(j,l),(j',l')} |h[\boldsymbol{k}]|^2 I^{(l')}_{j',\boldsymbol{k}} \right) , \tag{6.4.1}$$

where $h[\boldsymbol{k}]$ is a kernel function defined over $\mathbb{Z}^2$ with bounded support $[\boldsymbol{r} - \boldsymbol{M}, \boldsymbol{r} + \boldsymbol{M}]$. This is a simple two-dimensional genearlisation of the kernel smoother in Eq. 5.4.5, in this case $\boldsymbol{M} = (M_1, M_2)$ and $H = \sum_{\boldsymbol{k}=\boldsymbol{r}-\boldsymbol{M}}^{\boldsymbol{M}} h[\boldsymbol{k}]^2$. For simplicity, in this work we use a box-car (uniform) kernel with width $M_1 = M_2 := M$.

While asymptotically, one may show that kernel smoothers akin to (6.4.1) can recover the (Lipschitz smooth) LWS spectrum in a consistent manner (G. Nason 2013; Park et al. 2014), the finite sample performance of such estimators is often lacking. This is particularly evident at coarser scale levels, where the wavelet periodogram is more correlated, and estimators possess higher variance. For an example of this in the one-dimensional case see Figure 6.2.1 where the kernel (moving window) estimator suggests a negative value for the spectra at $j = 4$. Such an estimate is inconsistent with the construction of the LSW process, this has consequences for model interpretation as the spectrum is required to be positive; it also makes it harder to use the spectrum for anomaly detection tasks as we cannot define a distribution with negative variance.

6.4.3 Regularised Least Squares Estimation

An alternative to kernel estimation is to construct a regularised estimator that can incorporate prior information about the LWS estimate. In this section, regularised estimation is implemented in a least-squares setting. However,

unlike in the previous section where the regulariser acted on the periodogram (or the Haar-Fisz periodogram), at each scale separately, the estimator in this case jointly estimates $\{S_j^{(l)}(\boldsymbol{u}/\boldsymbol{R}) \mid \forall(j,l)\}$.

***Definition* 6.4.** *Least-Squares Spectral Loss*

A quadratic loss function is defined over the whole set of scales $j = 1, \ldots, J_{\boldsymbol{R}}$ and directions $l = \{h, v, d\}$ as:

$$L(\hat{\mathcal{I}};\mathcal{U}) := \sum_{j,l} \|\hat{I}_{j,\boldsymbol{r}}^{(l)} - \sum_{j',l'} A_{(j,l),(j',l')} U_{j',\boldsymbol{r}}^{(l')}\|_2^2 \,, \tag{6.4.2}$$

where $\hat{\mathcal{I}} := \{\hat{I}_{j,\boldsymbol{r}}^{(l)} | \ \forall j,l,\boldsymbol{r}\}$ is an estimator of the biased spectrum. In the spectral estimation context, the set of parameters $\mathcal{U} := \{U_{j',\boldsymbol{r}}^{(l')} | \ \forall j,l,\boldsymbol{r}\}$ takes the place of the spectral estimate within the loss-function. If we consider that $\sum_{j',l'} A_{(j,l),(j',l')} U_{j',\boldsymbol{r}}^{(l')}$ is a biased estimator of the spectrum, the quantities $U_{j',\boldsymbol{r}}^{(l')}$ should represent the un-biased estimate. For convenience, a matrix version of the above can be constructed as

$$L(\hat{\boldsymbol{I}};\boldsymbol{U}) := \|\hat{\boldsymbol{I}} - \boldsymbol{A}_J \boldsymbol{U}\|_F^2 \tag{6.4.3}$$

where $\hat{\boldsymbol{I}} \in \mathbb{R}^{LJ \times R_1 R_2}$ is a reshaped estimate of the periodogram (the set $\hat{\mathcal{I}}$) and $\boldsymbol{U} \in \mathbb{R}^{R_1 R_2 \times LJ}$ is a matrix relating to the set $\mathcal{U}$.

A natural choice for the function $\hat{I}_{j,\boldsymbol{r}}^{(l)}$ would be to use the raw empirical periodogram $\hat{I}_{j,\boldsymbol{r}}^{(l)} := |d_{j,\boldsymbol{r}}^{(l)}|^2$. However, as a more general solution, we may opt to use the kernel estimator $\hat{I}_{j,\boldsymbol{r}}^{(l)} = \hat{K}_{j,\boldsymbol{r}}^{(l)}$ from Eq. 6.4.1. The beauty of the formulation in Eq. 6.4.2, is that it allows us to consider estimation of the LWS spectrum as a convex optimisation problem. For example, one can construct the estimator according to

$$\{\hat{S}_{j,\boldsymbol{r}}^{(l)}\} = \arg\min_{\boldsymbol{U}} \|\hat{\boldsymbol{I}} - \boldsymbol{A}_J \boldsymbol{U}\|_F^2 \,. \tag{6.4.4}$$

Note, in order to obtain the actual elements $\{\hat{S}_{j,\boldsymbol{r}}^{(l)}\}$ we must unpack the $R_1 R_2 \times LJ$ matrix $\boldsymbol{U}^*$that is the minimiser of $L(\hat{\boldsymbol{I}};\boldsymbol{U})$. In the following, such an unpacking operations are implicitly performed for statements like $\{\hat{S}_{j,\boldsymbol{r}}^{(l)}\} = \boldsymbol{U}^*$. If we now consider that$\hat{\boldsymbol{I}} = \{\hat{K}_{j,\boldsymbol{r}}^{(l)}\}$ as defined in (6.4.1), then the objective $L(\hat{\boldsymbol{I}};\boldsymbol{U})$ is minimised when $\hat{\boldsymbol{S}} = \boldsymbol{U}$ which implies $\hat{S}_{j,\boldsymbol{r}}^{(l)} = \hat{K}_{j,\boldsymbol{r}}^{(l)}$. As a result, if we impose no additional constraints on (6.4.4) then we recover the standard de-biased kernel estimator (6.4.1).

Table 2 – Regularised spectral estimators within the least-squares framework. Note that $\boldsymbol{I}$ refers to the use of the raw-wavelet periodogram $\boldsymbol{I}^{(l)}_{j,\boldsymbol{k}} = |d^{(l)}_{j,\boldsymbol{k}}|^2$.

Name - Abbreviation	Loss Fn.	Penalty
De-Biased Kernel - K	$\hat{K}^{(l)}_{j,\boldsymbol{r}}$	n/a
Positive Kernel - K(P)	$L(\hat{\boldsymbol{K}}; \boldsymbol{B})$	$l_{\mathbb{R}^+}$
TV-Positive - TV(P)	$L(\boldsymbol{I}; \boldsymbol{B})$	$l_{\mathbb{R}^+} + R$
TV-Kernel Positive - TVK(P)	$L(\hat{\boldsymbol{K}}; \boldsymbol{B})$	$l_{\mathbb{R}^+} + R$

In order to restrict solutions such that $\hat{\boldsymbol{S}} \geq 0$ one can simply introduce a positivity constraint on the loss via the addition of an indicator function, defined as

$$l_{\mathbb{R}^+}(\boldsymbol{U}) = \begin{cases} 0 & \text{if } U_{i,j} \geq 0 \, \forall i, j \\ \infty & \text{otherwise} \end{cases} .$$

In addition to the fact that the LWS should be positive, we may introduce further constraints that relate to how we want or expect the estimates to behave when applied to real images. Akin to the fused lasso in the one-dimensional setting, one can introduce a total-variation penalty which actively constrains the spectral variation across the whole image. Such a penalty can be introduced according to the function

$$R^{(l)}_j(\boldsymbol{U}) = \lambda \sum_{m=2}^{R_1} \sum_{n=2}^{R_2} \left| U^{(l)}_{j,(r_1^{(m)}, r_2^{(n)})} - U^{(l)}_{j,(r_1^{(m-1)}, r_2^{(n-1)})} \right| ,$$

or in matrix form as; $R(\boldsymbol{U}) = \lambda \| [\boldsymbol{D}_H \boldsymbol{U}; \boldsymbol{D}_V \boldsymbol{U}] \|_1$, where $\boldsymbol{D}_H, \boldsymbol{D}_V$ are differencing matrices operating respectively in the horizontal and vertical dimensions (the construction of these matrices follows that of Eq. 6.2.1). In the proceeding experiments, the effect of regularisation within the least-squares estimation scheme is examined. In particular, different combinations of regularisers are considered as listed in Table 2.

***Remark* 6.2.** *Effect of Regularisation*

Much like the kernel estimator in (6.4.1), the TV-estimators attempt to smooth the periodogram over space. However, unlike simply using $\hat{K}^{(l)}_{j,\boldsymbol{r}}$ *alone, the TV-constraint fuses estimates across the whole image resulting in a* global *estimator. Due to the* ℓ_1 *form of the norm, such estimators should promote piecewise structure in the spectra (akin to the fused lasso estimator in the previous section). Although, unlike in the previous section, estimation is performed here as a joint optimisation over all scales, it also enables the use of a local smoother (via the kernel) prior to being regularised. The work of Monti et al. (2014) considers the effect of several (Gaussian and boxcar) kernels for pre-smoothing in the context of dynamic GGM estimation. In that case, the shape of the kernel function had a significant effect on estimation performance (especially near changepoints). One may expect a similar situation in the spectral estimation setting. It is worth noting that the experiments below make no effort to optimise the shape of the kernel function and simply use the constant (boxcar) kernel.*

6.4.4 An ADMM Algorithm for Spectral Estimation

The extension of estimation to 2d-fields dramatically increases the number of parameters in the LSW model. In terms of the proposed regularised estimators, the number of parameters over which we need to optimise scales as $\mathcal{O}(\log_2(R)R^2)$. Furthermore, since the estimation is not separated over scales, all these parameters must be optimised jointly. The strategy suggested here is to tackle this optimisation task with an ADMM approach; this follows in a similar way to the graphical model estimation of Chapter. 3.

As previously introduced in Sections. 2.2.2 and 3.2, the ADMM method allows one to split up the optimisation problem across linearly separable portions of the objective. For instance, taking the K(P)-LWS objective, we may reformulate the optimisation problem in an explicitly constrained form

$$\hat{\boldsymbol{S}} := \arg\min_{\boldsymbol{U};\boldsymbol{V}=\boldsymbol{U}} \|\hat{\boldsymbol{K}} - \boldsymbol{A}\boldsymbol{U}\|_F^2 + l_{\mathbb{R}^+}(\boldsymbol{V}) \ ,$$

where $\boldsymbol{V}$ is as an auxiliary variable. In practice, and to ensure sufficient curvature, an augmentation term $\rho/2\|\boldsymbol{V} - \boldsymbol{U}\|_F^2$ is added to the Lagrangian.

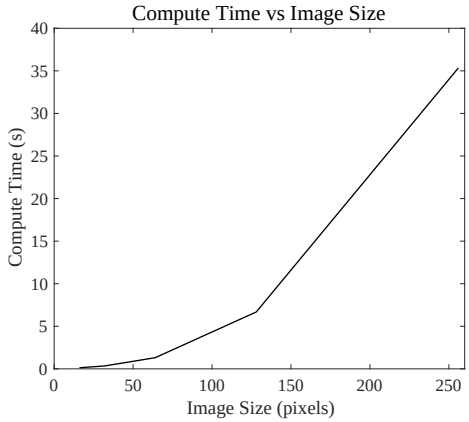


Figure 6.4.2 – Computational cost of spectral estimation via ADMM (Alg. 3).

In the general case, auxiliary variables are introduced for each term

$$\begin{aligned} \boldsymbol{V}_B = &\boldsymbol{V}_{\mathbb{R}^+} = \boldsymbol{U} \ , \\ &\boldsymbol{V}_{AB} = \boldsymbol{A}\boldsymbol{U} \ , \\ &\boldsymbol{V}_D = \boldsymbol{D}\boldsymbol{V}_B \ . \end{aligned} \tag{6.4.5}$$

These constraints can be written in matrix form $\boldsymbol{F}\boldsymbol{U} = \boldsymbol{G}\boldsymbol{V}$, where

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{I} \\ \boldsymbol{0} \\ \boldsymbol{I} \end{bmatrix} \ , \ \boldsymbol{G} = \begin{bmatrix} \boldsymbol{I} & & & \\ & \boldsymbol{I} & & \\ & -\boldsymbol{D} & \boldsymbol{I} & \\ & & & \boldsymbol{I} \end{bmatrix} \ .$$

It should be noted, that the sizes of the individual matrices follow from the equivalences in (6.4.5). The augmented Lagrangian (for K(P)-LWS) is now constructed as:

(6.4.6)

$$\mathcal{L}(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{P}_V) := \|\hat{\boldsymbol{K}} - \boldsymbol{Q}_{AB}\|_F^2 + l_{\mathbb{R}^+}(\boldsymbol{Q}_{\mathbb{R}^+}) + \lambda\|\boldsymbol{Q}_D\|_1 + \frac{\rho}{2}\|\boldsymbol{F}\boldsymbol{U} - \boldsymbol{G}\boldsymbol{V} - \boldsymbol{P}_V\|_F^2 \ ,$$

where $\rho^{-1}\boldsymbol{P}_V$ are Lagrange multipliers. This Lagrangian problem can now be solved through a series of updates, minimising $\mathcal{L}(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{P}_V)$ sequentially with respect to $\boldsymbol{U}$ and $\boldsymbol{V}$, then updating the dual $\boldsymbol{P}_V$ to keep track of the cumulative errors. This is a similar procedure to the ADMM procedure in Chapter 3 or Section 2.2.2.

One benefit of ADMM, particularly in our case, is that the updates required for $\arg\min_{\boldsymbol{U}} \mathcal{L}(\boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{V})$ are simply proximity operators. These can be calculated extremely quickly, projecting onto the positive real line for $l_{\mathbb{R}^+}$, and updating via the soft-thresholding operator for the TV-denoising methods. Algorithm 3 provides more details, in practice the experiments utilised a version of the SunSAL-TV algorithm (Iordache et al. 2012)[4]. While a standard ADMM scheme is guaranteed to converge if a solution exists for any $\rho > 0$, however this is not necessarily the case for the multi-block scheme utilised here (C. Chen et al. 2016; Lin et al. 2014). A full analysis of the convergence properties of this particular formulation is beyond the scope for this work. Rather, it should be noted in practice for a parameter of $\rho = 100$ no issues with convergence were found. Empirically, this allowed for reasonably rapid convergence $\sim$ 1minute for an $R = 256$ size image. As demonstrated by Figure 6.4.2, the algorithm appears to scale computationally as $\mathcal{O}(R^2)$.

Input: $\rho > 0, \bar{\boldsymbol{I}}, \boldsymbol{A}$
while *not converged* **do**
  Update primal variable:
  $\boldsymbol{U}^{(k+1)} \leftarrow \arg\min_{\boldsymbol{U}} \mathcal{L}(\boldsymbol{U}, \boldsymbol{V}^{(k)}, \boldsymbol{P}_V^{(k)})$
  Update auxilary variables:
  $\boldsymbol{V}^{(k+1)} \leftarrow \arg\min_{\boldsymbol{V}} \mathcal{L}(\boldsymbol{U}^{(k+1)}, \boldsymbol{V}, \boldsymbol{P}_V^{(k)})$
  Solve via proximity operators, i.e: $\boldsymbol{V}_{\mathbb{R}^+}^{(k+1)} \leftarrow \max(\boldsymbol{U}^{(k+1)} - \boldsymbol{P}_{V;\mathbb{R}^+}^{(k)}, 0)$
  and $\boldsymbol{V}_H^{(k+1)} \leftarrow \text{soft}(\boldsymbol{D}_H \boldsymbol{U} - \boldsymbol{P}_{V;H}, \lambda/\rho)$
  Update dual variable:
  $\boldsymbol{P}_V^{(k+1)} \leftarrow \boldsymbol{P}_V^{(k)} + (\boldsymbol{F}\boldsymbol{U}^{(k+1)} - \boldsymbol{G}\boldsymbol{V}^{(k+1)})$
end

Algorithm 3: 2D-LSW ADMM smoothing algorithm

6.5 Experiments

To test the recovery ability of the proposed estimators we can generate synthetic data-sets according to a simple piecewise constant texture model. The spectral structure as encoded through $S_{j,\boldsymbol{r}}^{(l)}$ is split into a set of blocks corresponding to regions in the plane with alternating values (see Figure 6.5.1). In

[4]The algorithm developed by Iordache et al. (2012) was originally developed for hyperspectral unmixing, but coincidently has the same optimisation structure as the proposed spectral regularisation scheme.

this experiment we consider recovery of the true structure $\{S_{j,b}^{(l)}\}$ from synthetic data

$$\{X_{\boldsymbol{r};\boldsymbol{R}}\} \sim \text{LSW}(\{w_{j,r}^{(l)} = (S_{j,\boldsymbol{r}}^{(l)})^{1/2}\})$$

where $\epsilon_{j,\boldsymbol{r}}^{(l)} \sim \mathcal{N}(0,1)$ over images of varying size. An example realisation of the process is given in Figure 6.5.1 (top-right). In these experiments true signal is restricted to scale $j = 1$ and is identically distributed across directions $l = \{h, v, d\}$.

Note that variance of the kernel smoothed periodogram when estimating structure at fine scales is naturally less than when measuring coarse structures due to the size of the wavelet basis functions. On the other hand, structure at larger scales induces longer range dependencies in the process and is more difficult to recover. The aim here is simply to give a high-level overview of the differences between the regularised and kernel estimators and as such all experiments have true structure isolated to scale $j = 1$. The motivation for this is computational, setting the true scale structure to $j = 1$ enables us to gain some insight on the estimation performance while keeping image sizes small $R \approx 128$ pixels.

6.5.1 Results on Synthetic Data

In order to examine statistical properties of the proposed estimators across the generative distribution, a cross-validation setup is utilised. This is slightly different to the approach of Section 6.3, in that performance is measured on a data-set completely independent from the training set. In order to select tuning parameters (h, λ) a parameter sweep is run across $N_{\text{train}} = 20$ images, from which $(\hat{h}, \hat{\lambda})$ are selected in order to minimise the error

$$\epsilon_{\text{test}}(h, \lambda) = \frac{1}{3JR^2} \sum_{j,l} \sum_{\boldsymbol{k} \in \boldsymbol{R}} \left| |w_{j,\boldsymbol{k}}^{(l)}|^2 - \hat{S}_{j,\boldsymbol{k}}^{(l)} \right| .$$

Figure 6.5.2 depicts error surfaces for the cross-validation experiment; and can be used to visualise the performance trade-off when introducing different levels of prior smoothness via λ and h. Clearly, estimation performance is enhanced by performing smoothing using either the kernel and/or regularisation. The distinctive kink in the cross-validation surface is typical for estimation across image sizes. As expected, due to the experimental setup (the blocks get larger

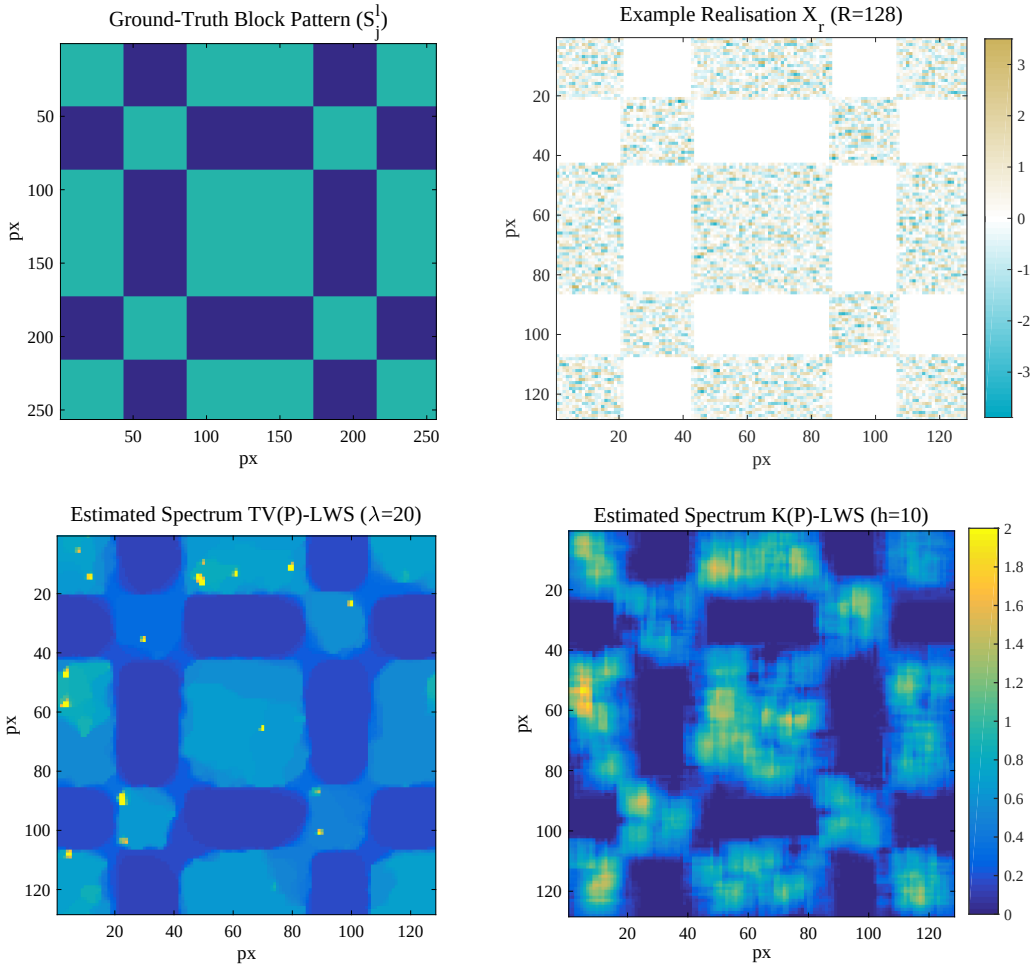


Figure 6.5.1 – Top-left: Ground-truth spectrum for piecewise constant LSW field; dark-blue $S_j^{(l)}(\boldsymbol{r}/\boldsymbol{R}) = 0$ and light-blue $S_j^{(l)}(\boldsymbol{r}/\boldsymbol{R}) = 1$. Top-right: Example of simulated process with image size $R = 128$ pixels. Bottom: Examples of recovered spectrum $\hat{S}^{(1)}_{1,\boldsymbol{b}}$ (at $R = 128$). Left: TV(P)-LWS with $\lambda = 20$, and Right: K(P)-LWS with $h = 10$.

in proportion to the image size), the optimal $(\hat{h}, \hat{\lambda})$ for each scale changes with the image size. It appears that a kernel on a similar scale to the small blocks is optimal, i.e. around $h = 10 \sim 20$ for $R = 128$. While optimal performance seems to require some kernel *and* some regularisation; the regularised estimate with $h = 1$ performs relatively well (see Figure 6.5.2). In practice, this means that TV-LWS resolves more clearly defined edge detail, for instance, as seen in Figure 6.5.1 (bottom-left vs bottom-right). Such a result mimics the smoothing

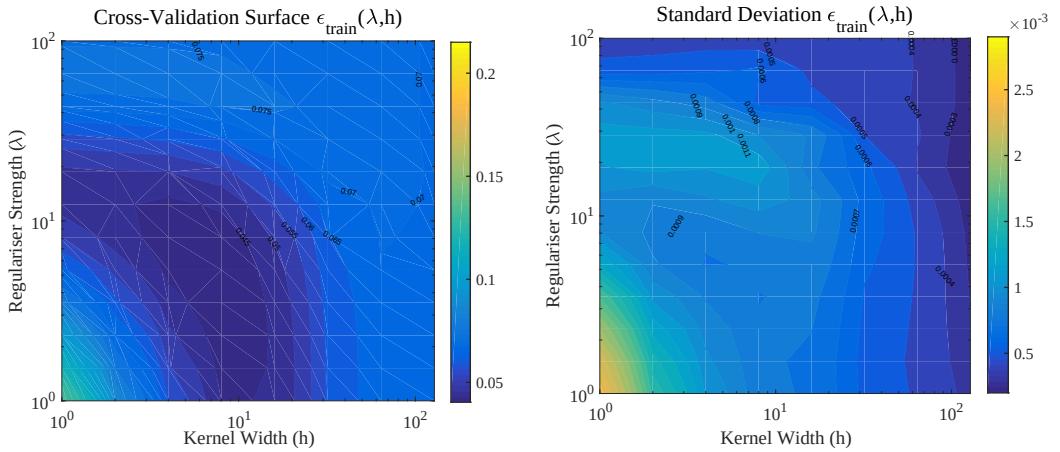


Figure 6.5.2 – Left: The mean cross-validation error surface ϵ_{train} . Right: the standard-deviation of the surface, taken over $N_{\text{train}} = 20$ synthetic images of size $R = 128$.

behaviour of the fused lasso in the one-dimensional case (Figure 6.2.1). Again, one observes that kernel estimation can provide consistent estimation in areas which the spectra is smooth but fails to track jumps in the spectrum.

Using the parameter sweep over $N_{\text{train}} = 20$ images to select an appropriate set of smoothing parameters $(\hat{h}, \hat{\lambda}) = \arg\min \epsilon_{\text{train}}(h, \lambda)$ the out-of sample performance can now be examined. Figure 6.5.3 summarises performance of the estimators on a test set of size $N_{\text{test}} = 100$ and enables us to compare the sample efficiency of different estimators. Clearly, the error for all smoothed estimators appears to decrease for increasing image size both at scale $j = 1$, and when averaged across all scales. This is in contrast to the raw-periodogram, which does not converge for scale $j = 1$ where all the true spectral structure is simulated. The hybrid method TVK(P)-LWS appears to perform best, converging faster at all scales. Again, this agrees with the cross-validation results which suggest a combination of kernel regularised smoothing is beneficial. Figure 6.5.3 demonstrates the favourable consistency rates for this estimator while also highlighting the inconsistency of the bias-corrected raw periodogram.

***Remark* 6.3.** *Kernel pre-smoothing is beneficial?*

The observation that the hybrid method performs well is interesting, as in this setting the ground-truth structure is piecewise constant, one might expect that the blurring effect of the kernel to hamper estimation performance. That

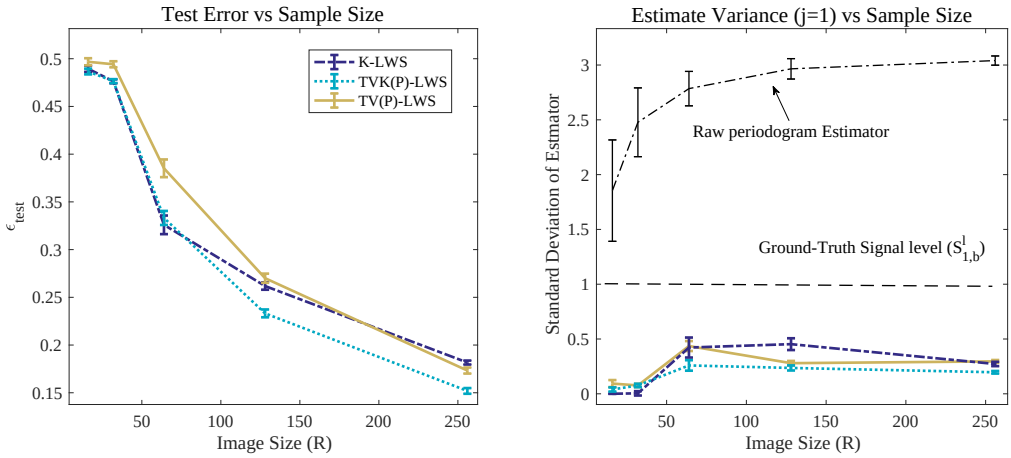


Figure 6.5.3 – Test performance of the various LWS estimators. Left: The error at scale level $j = 1$ as a function of image size. Right: standard-deviation of estimate taken over $N_{\text{test}} = 100$ simulations.

it actually improves performance suggests it may be beneficial to perform some smoothing of the spectrum before implementing the regularised smoothing. This provides an interesting question for future research: is localised pre-smoothing generally beneficial for regularised estimation, or is this due to the nature of the periodogram noise structure? In the case of the latter, pre-smoothing the periodogram helps decrease the variance of the input to the regularisation problem. Additionally, to some extent the distribution of the kernel periodogram estimator will be pulled towards a Gaussian. However, one should note that due to auto-correlation in the raw periodogram this will not simply be a χ^2 distribution with increased degrees of freedom. It may turn out that pre-smoothing is generally a useful strategy, as even in the dynamic GGM setting, it seems that using a kernel is beneficial to help stabilise the empirical covariance matrix prior to regularisation. In the dynamic GGM setting, one may compare the results of Monti et al. (2014) with the results in Chapter 3 and Gibberd and Nelson (2017).

6.5.2 Application to Real Images

The experiments presented so far have demonstrated the benefits of regularisation in a quantitative manner, i.e. providing more efficient and robust

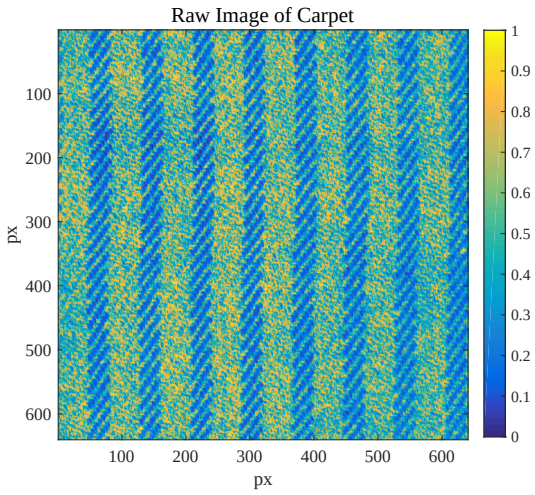


Figure 6.5.4 – Raw greyscale image of carpet mapped to interval $[0, 1]$.

recovery of the spectrum. However, many of the benefits of regularised estimators come from the improved interpretability of the estimates. To demonstrate some of these properties, it is of interest to consider how one may use the LSW process to describe a real image. In this case the model is applied to describe a texture taken from the Brodatz data set[5] with $R = 640$ (see Figure 6.5.4). While the LSW process doesn't necessarily assume a Gaussian stochastic component, it does require that the process is zero mean. Given this assumption, the image should first be pre-processed to remove any large-scale trends. In this case, the image is down-sampling to a size of $R_{\text{eff}} = 128 \times 128$ pixels, it is then standardized by *z-scoring* the data with respect to all pixels in the rescaled image.

A benefit of performing modelling with a Gaussian LSW process is that it gives the process a well defined likelihood. One may then in principle use this likelihood to perform anomaly detection, for example via a localised likelihood ratio test. In the example presented here such an application is not considered but it is worth noting that the pre-processing step of down-sampling is one way to try and pull the image distribution towards a Gaussian through local

[5]The figure can be found http://sipi.usc.edu/database/database.php?volume=textures and is originally found in the book Brodatz (1966)

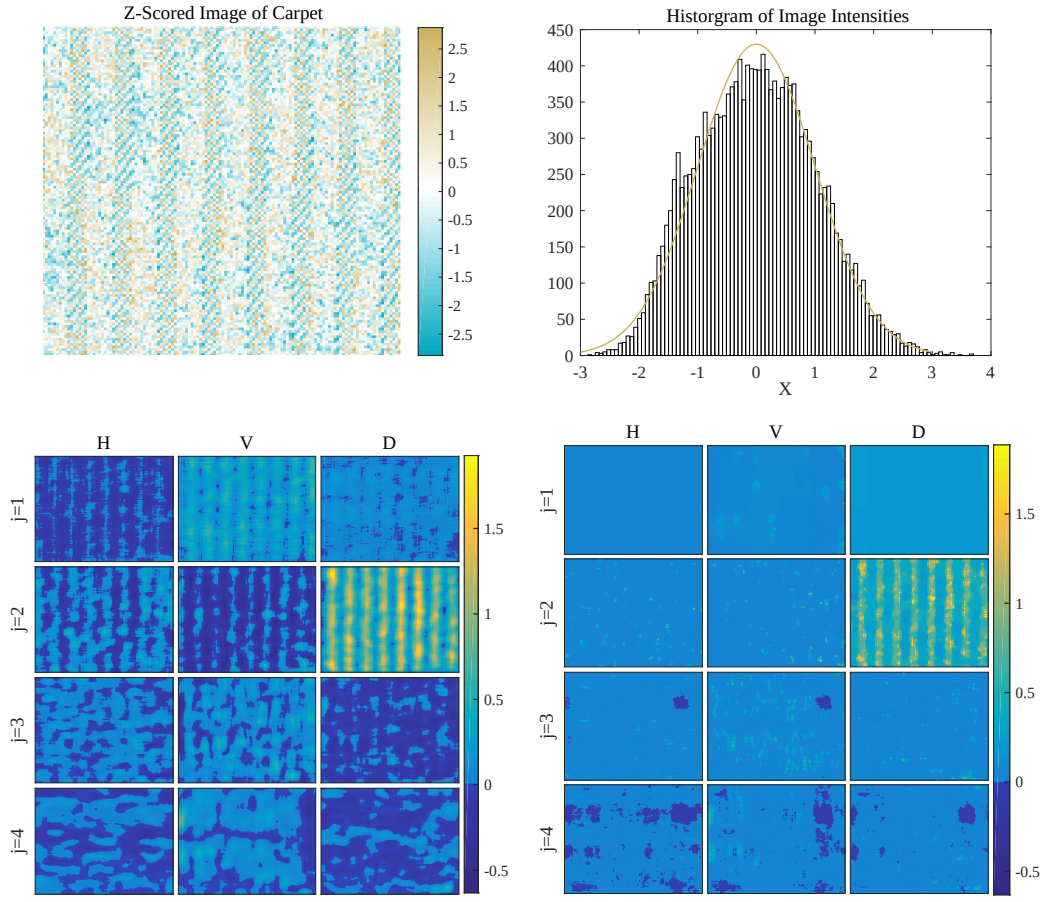


Figure 6.5.5 – Comparison of LWS estimation with regularised and kernel approach. Top-left: Original image. Top-right: Z-Scored image (normalised across both rows/columns). Bottom-left: Estimated LWS with KLWS $h = 16$. Bottom-right: Estimated LWS with TV(P)-LWS and $\lambda = 10$.

averaging. Figure 6.5.5 demonstrates the output of the analysis and how the pre-processing effectively produces a Gaussian intensity distribution.

When analysing a point estimate of the spectra, it is of interest to contrast the kernel based estimate (bottom-left) with that of the regulariser (bottom-right). Where, the kernel estimate often contains negative values for the spectra, an undesirable property as by definition the spectrum should be positive; the regularised case pretty much eradicates this issue[6]. More importantly, it

[6]The areas which are negative in the regularised case are actually very close to zero; this is due to numerical error in the optimisation procedure.

appears that the regularised estimator can appropriately deal with the negative regions of the spectrum while also maintaining key structural features. For instance, one clearly observes the diagonal banding in $j = 2$, $l = d$ is preserved in the regularised solution. Qualitatively, it would seem that the regularised model offers a more parsimonious description of the texture, the regularised spectra appears constant for large regions, whereas the kernel estimate has much more local variation.

6.6 Summary

Typically, wavelet decompositions may be used to break down an image into a set of features which can then be used for general image understanding tasks; for instance; image classification (Meher et al. 2007) or clustering (Bhattacharya et al. 2014). Enhanced estimation of the wavelet spectra may therefore aid in many of these applications. Specifically, if the wavelet spectra are considered as a feature for further analysis, i.e. as an input to a classifier, one would expect reducing variance of the features would correspondingly map to reduced variation in classification performance. It is worth remarking that while most of the work in this chapter is empirical in nature, it provides motivation to develop a more theoretically rigorous treatment of the estimators. A particularly interesting observation, is that in the one-dimensional case where we discussed using the fused lasso to estimate the spectra, it appeared that the ℓ_1 shrinkage did not have much beneficial effect, even though the LSW model could be considered sparse. Although not investigated further in this thesis, this result appears to be due to the bias imposed by the non-decimated wavelet framework, where the matrix $\boldsymbol{A}$ smears the spectrum across all scale levels. In future, one may wish to examine enforcing a sparsity constraint in the non-biased frame where such assumptions should enable more efficient estimation of the spectra. For example, in the context of the 2d-LSW estimators, we may enforce a sparsity assumption across scale levels by adding an additional penalty to Eq. 6.4.4. In the next section, we will discuss a final further extension to LSW models where we model multivariate time-series. This provides a direct link between the LSW framework and the dynamic graphical models of Chapters 3 and 4.

Chapter 7

Multivariate-LSW models

Classically, much of the literature in non-stationary time-series analysis (and particularly for LSW process) is focussed on analysing signals in the univariate setting (R Dahlhaus 1997; D. L. Donoho et al. 1995; G.P. Nason et al. 2000; Priestly 1981). However, many modelling applications require a multivariate treatment as data-streams cannot be analysed independently. For example, as discussed in the introduction (2) the analysis of brain activity via Electro-encephalography (EEG) is severely restricted if one is limited to analysing each signal independently. In applications such as this one is specifically interested in understanding how the behaviour in one data-stream behaves with respect to the others. Two popular measures of dependency between a set of variables are the correlation/covariance and partial-correlation/precision. In the GGM context, these measures are respectively related to dependency and conditional dependency between variables (see Section 2.3). In the context of time-series analysis, it is desirable to combine such measures of inter-stream dependence alongside auto-covariance (sequential dependence) structure of data-streams. In particular, given that spectral analysis allows for a decomposition of variance across scales it is possible to define a localised decomposition of such cross-stream-cross-time dependency across multiple scale levels. The correlation between streams localised to a specific scale is commonly referred to as *coherence* whereas the partial correlation is known as *partial coherence*[1].

[1]One should note that there are several specific forms of coherence defined in the literature. The form of coherence that we study here will be defined in the sequel.

Recently, several methods for analysing the coherence structure of multivariate non-stationary time-series have been proposed. Early efforts aimed to adapt locally stationary Fourier methods to the multivariate setting, for example R. Dahlhaus (2000) proposed an approximate Likelihood method for estimating the Fourier coherence. In Ombao and Van Bellegem (2008), a consistent method for estimating the non-stationary Fourier coherence is suggested based on time-localised linear filtering. An alternative approach is suggested in Ombao, Sachs, et al. (2005), where the signals are modelled by a set of basis selected from the smooth localised complex exponential (SLEX) family (a dyadic set of scaled SLEX basis functions). Estimation of the coherence is then performed by selecting a model which minimises the Kullback-Leibler distance between fitted models and the SLEX principle components of the time-series. Whilst this method can handle relatively large data-sets, the SLEX construction is limited to selecting basis from dyadically scaled blocks. More recently, Cohen et al. (2010) consider coherence in the context of a wavelet analysis for a bi-variate jointly stationary process. The work of Cohen et al. (2011) and Sanderson et al. (2010) further extends the analysis of wavelet coherence to classes of non-stationary bi-variate process. Additionally, the work of Cho et al. (2015) considers an extension of the multivariate LSW model proposed in Sanderson et al. (2010) to the general p-variate setting.

In this section, we introduce a multivariate LSW framework (similar to that of Cho et al. (2015), Park et al. (2014), and Sanderson et al. (2010)) which can describe both linear dependency between streams, and across time. Many authors assume the extension of coherence estimators to multivariate settings from the bi-variate setting is trivial. However, this is far from the case, the full multivariate setting requires us to deal with increased variance in estimators due to model complexity (this scales in order $\mathcal{O}(p^2)$). Without some form of model selection our models will tend to overfit the data. To this end, this chapter introduces a novel class of regularised (sparsity aware) estimators for the wavelet coherence. The empirical performance of these estimators is examined alongside demonstrating an application to the analysis of epileptic EEG data.

7.1 Extending the univariate LSW model

A simple extension of the univariate LSW model can be formed by introducing a transfer matrix in place of the weighting elements $w_{j,k}$. The construction we study here originally appeared in (Park et al. 2014)

***Definition* 7.1.** *Multivariate LSW Process*

The p-variate LSW process $\{\vec{X}_{t;T}\}$ *is defined as a collection of random vectors for* $t = 1, \ldots, T$*, such that*

$$\vec{X}_{t;T} = \sum_{j=1}^{\infty} \sum_{k} \boldsymbol{V}_j(k/T)\psi_{j,t}[k]\vec{\boldsymbol{\epsilon}}_{j,k} \,, \tag{7.1.1}$$

where $\boldsymbol{V}_j(k/T) \in \mathbb{R}^{p\times p}$ *is a transfer matrix, and random vector* $\boldsymbol{\epsilon}_{j,k}$ *defined at scales* $j = 1, \ldots, J_T = \log_2(T)$*. Specifically, the transfer matrix has a lower-diagonal form that encodes all dependency and contributions to the variance by wavelet at given scale/position. Furthermore, we assume that all the random vectors obey* $E[\boldsymbol{\epsilon}_{j,k}] = \mathbf{0}$ *and* $\mathrm{Cov}(\epsilon^{i}_{j,k}, \epsilon^{i'}_{j',k'}) = \delta_{j,j'}\delta_{i,i'}\delta_{k,k'}$[2]*.*

In a similar manner to the spectrum associated with the univariate LSW process we here construct the Local Wavelet Spectrum *(LWS) matrix as the outer product of the transfer functions*

$$\boldsymbol{S}_j(z) = \boldsymbol{V}_j(z)\boldsymbol{V}_j(z)^{\top} \,.$$

Additionally, the inverse LWS matrix (iLWS) is defined as $\boldsymbol{P}_{j;T}(z) = (\boldsymbol{S}_{j;T}(z))^{-1}$*. It is required that the LWS matrix is finite* $\sum_j S^{(q,r)}_{j;T}(u) < \infty$ *and each element is Lipschitz continuous with constants* $\sum_{j=1}^{\infty} 2^j L^{(q,r)}_j < \infty$*.*

We note, that the LWS matrix takes the form of a covariance matrix restricted to a specific scale level j. It is easy to then demonstrate the linear combination of LWS matrices

$$c^{(p,q)}(z,\tau) := \sum_{j=1}^{\infty} S^{(p,q)}_j(z)\Psi[\tau] \,,$$

[2]Note: this contrasts with the construction used in Sanderson et al. 2010 where structure may be split between the transfer matrix and the noise terms.

can asymptotically represent the auto-cross covariance of the process. *Note: The auto-correlation wavelet is still defined in the standard manner, i.e.* $\Psi[\tau] := \sum_j^{\infty} \psi_{jk}[0]\psi_{j,k}[\tau]$.

***Proposition* 7.1.** *Asymptotic Representation of Auto-Cross Covariance*

Asymptotically for $T \rightarrow \infty$ *one obtains*

$$|c^{(p,q)}(u,\tau) - \mathrm{cov}(X_{uT}^{(p)}, X_{[uT]+\tau}^{(q)})| = \mathcal{O}(T^{-1}) \ .$$

Proof. This is a simple consequence of Lipschitz continuity, it is almost identical to that of the 1d-case (c.f. G.P. Nason et al. (2000)) the full proof can be found in Park et al. (2014).

***Remark* 7.1.** *Exact vs approximate representation*

In the literature on LSW processes there appear to be two forms of model specification. One which utilises a discrete set of amplitudes $\{w_{j,k;T}\}$ *and links these to the continuous functions* $W_{j,k}(z)$*. For instance, the representation of G.P. Nason et al. (2000) has the condition* $\sup_k |w_{j,k;T} - W_j(k/T)| \leq C_j/T$*. A second parameterisation, such as that found in Park et al. (2014) and Sanderson et al. (2010) utilises the continuous functions i.e.* $W_j(k/T)$ *directly in the model representation. In this chapter we follow the latter representation, simply due to this being the one suggested in the literature. Asymptotically, it does not make much difference which representation we use, however, in a finite sample, it should be noted that the bounded deviation (original) representation is more flexible. As pointed out by R Dahlhaus (1997) the reason for using approximate representation is due to the observation that simple AR processes do not have a spectral representation which is exact in terms of* $W_j(k/T)$*.*

7.2 Estimation for Mv-LSW Spectra

Previous proposals (Park et al. 2014; Sanderson et al. 2010) for estimating the LWS matrix are based around the estimated wavelet periodogram $\boldsymbol{d}_{j,k} := \sum_{t=0}^{T-1} \boldsymbol{x}_t \psi_{jk}[t]$, where the smoothed periodogram matrix is given as:

$$\hat{\boldsymbol{I}}_{j,k} = \frac{1}{2M+1} \sum_{s=k-M}^{M} \boldsymbol{d}_{j,s} \boldsymbol{d}_{j,s}^{\top} \ . \tag{7.2.1}$$

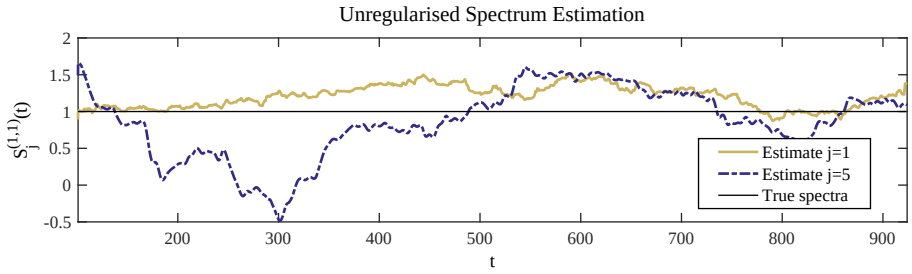


Figure 7.2.1 – Estimation of the on-diagonal spectrum, i.e. $\hat{S}_j^{(q,q)}$ using (7.2.2), for scales $j = 1$ and $j = 5$. Whilst the true spectra is equal to one in both cases, we see that the estimation error at the large scale lengths can cause negative variance estimates.

While this estimator is consistent, unfortunately it is also biased due to leakage in the periodogram between scale levels. To account for this, the estimator is corrected in the usual manner

$$\hat{\boldsymbol{S}}_{j,k} = \sum_{l=1}^{J} A_{jl}^{-1} \hat{\boldsymbol{I}}_{j,k} \ , \tag{7.2.2}$$

where $\boldsymbol{A}^{-1}$ is the usual inverse of the de-biasing matrix (see Eq. 5.3.7).

In the limit $M, T \to \infty$ Park et al. (2014) demonstrate that the estimator (7.2.2) is both bias free and consistent. However, in a finite sample setting where one considers that entries within A_{jl}^{-1} may be negative, there becomes a setting where $\hat{\boldsymbol{S}}_{j,k}^{(q,q)}$ can be negative. Figure 7.2.1 presents an example of this. Recalling, that $\boldsymbol{S}_j(k/T)$ appeared to be a covariance matrix, not only does such an estimate not make sense for a measure of variance, it contradicts our process construction (7.1.1). Furthermore, $\hat{\boldsymbol{S}}_{j,k}$ is not guaranteed to be positive-semi-definite. It is worth noting that Sanderson et al. (2010) also remark that the standard method of smoothing and then de-biasing the LWS matrix can result in estimators which are inconsistent with model specification. In the proceeding section, a regularised method is introduced in order to stabilise the estimation of the wavelet coherence estimates. These ensure that the model assumptions are met and providing an interpretable estimate of the LWS and iLWS matrices.

7.2.1 Modelling Spectra with Gaussian Graphical Models

If one assumes that the the wavelet periodogram is drawn from a multivariate Gaussian, $\boldsymbol{d}_{j,k} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{j,k})$ then maximum likelihood estimation of the covariance matrix is given by the empirical covariance estimator $\hat{\boldsymbol{\Sigma}}_{j,k} = \boldsymbol{d}_{j,k}(\boldsymbol{d}_{j,k})^{\top}$. Let us assume that we have $k = 1, \ldots, 2M+1$ observations drawn i.i.d from a Gaussian, such that $\boldsymbol{\Sigma}_{j,k} = \boldsymbol{\Sigma}_{j,l}$ for all $k, l \in \{0, \ldots, 2M+1\}$.

To estimate valid covariance matrices, it is proposed to first estimate a sparse precision matrix at each scale level $j = 1, \ldots, J$ and discrete time-step $k = 1, \ldots, T$. In particular, we construct a graphical lasso based estimator

$$\hat{\boldsymbol{\Theta}}_{j,k} := \arg\min_{\boldsymbol{U} \succeq 0} \big[-\log\det(\boldsymbol{U}) + \operatorname{tr}(\hat{\boldsymbol{S}}_{j,k}\boldsymbol{U}) + \lambda \|\boldsymbol{U}\|_1 \big] , \tag{7.2.3}$$

where $\|\boldsymbol{U}\|_1 = \sum_{p,q} |U_{p,q}|$. In the context of estimating the LWS and iLWS matrices, the above estimator has several advantages:

(1) The bias induced by $\lambda\|\boldsymbol{Z}\|_1$ imposes a sparsity structure on the resultant precision matrix estimate. If the assumption of a sparse ground-truth $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$ is valid, this can improve estimation performance reducing estimator variance.
(2) The estimated graphical models enable one to easily visualise key dependencies within a data-stream.
(3) The constrained problem forms a convex optimisation problem, enabling fast convergence to a global optima.

***Remark* 7.2.** *Suitability of the GGM Assumption*

If we assume Gaussian innovations $\vec{\boldsymbol{\epsilon}}_{j,k} \sim \mathcal{N}_p(\mathbf{0}, \boldsymbol{I})$, then a single set of wavelet coefficients $\boldsymbol{d}_{j,k} \in \mathbb{R}^p$ will also be distributed as a Gaussian. However, these will typically be dependent on those at another nearby point in scale/time, such that $\boldsymbol{d}_{j,k} \not\perp \boldsymbol{d}_{l,k'}$ if the points are relatively close in scale/time. If we use local kernel smoothing of the raw periodogram, as we do when we construct $\hat{\boldsymbol{S}}_{j,k}$ in (7.2.2), then we would expect to find ourselves in a non-i.i.d setting. As such, the empirical covariance does not provide a maximum-likelihood estimator of the quantity $\boldsymbol{I}_j(u) := \sum_l \boldsymbol{S}_l(u) A_{l,l}$. The estimator $\hat{\boldsymbol{I}}_{j,k}$ is not Wishart distributed as one would expect if $\boldsymbol{d}_{j,k}$ were independently drawn. As M increases (relative to the support of the wavelets) the dependency between samples

becomes less pronounced and in the limit of $M \to \infty$, $T \to \infty$ the wavelet coefficients will become less correlated. In this limit the method suggested which assumes $\boldsymbol{d}_{j,k} \perp \boldsymbol{d}_{j,k'}$ becomes appropriate, it may be possible to formalise this via some form of mixing assumption on the process. However, given that we want to improve the finite-sample performance of the estimators, it is not particularly useful to make such asymptotic arguments. Finally, although not applied in this work, it would be interesting to apply a further variance stabilistation step in the multivariate setting. For instance, one may apply the Haar-Fisz technique of Section 6.1.

There are many algorithms developed for solving problems of the form Eq. 7.2.3, for simplicity we adopt the approach of Friedman, Hastie, and R. Tibshirani (2008) whereby the dual of (7.2.3) is solved; instead of actively updating $\hat{\boldsymbol{\Theta}}_j$ we update the constrained covariance estimator $\hat{\boldsymbol{W}}_j = (\hat{\boldsymbol{\Theta}}_j)^{-1}$. The *graphical lasso (glasso)* algorithm (Friedman, Hastie, and R. Tibshirani 2008) iterates through updating columns/rows of this matrix to arrive at a global optima. One benefit of this method, is that it can take advantage of the sparsity structure within the target matrix, generally the sparser we assume the matrix (i.e. the larger we set λ) the faster a solution will be found. The process for computing the multi-scale graphical estimators we propose is outlined in Algorithm 4.

Input: $M, \lambda_j, \psi_{j,k}, \boldsymbol{X}_{t;T}$
Result: $\hat{\boldsymbol{W}}_{j,k},\ \hat{\boldsymbol{\Theta}}_{j,k}$
for $j, l = 1, \ldots, J,$ *and* $k = M, \ldots, T - M$ **do**
  $\boldsymbol{d}_{j,k} = \sum_{t=0}^{T-1} \boldsymbol{X}_t \psi_{jk}(t)$
  $\boldsymbol{I}_{j,k} = \frac{1}{2M+1} \sum_{m=-M}^{M} \boldsymbol{d}_{j,k+m} \boldsymbol{d}_{j,k+m}^{\top}$
  $A_{j,l} = \sum_{\tau} \Psi_j(\tau) \Psi_l(\tau)$
end
$\hat{\boldsymbol{S}}_{j,k} = \sum_{l=1}^{J} A_{jl}^{-1} \boldsymbol{I}_{j,k}$
for $j = 1, \ldots, J,\ k = M, \ldots, T - M$ **do**
  $(\hat{\boldsymbol{\Theta}}_j, \hat{\boldsymbol{W}}_{j,k}) = \text{glasso}(\hat{\boldsymbol{S}}_{j,k} + \lambda_j \boldsymbol{I}, \lambda_j)$
end

Algorithm 4: Algorithmic procedure for multi-scale graphical model estimation.

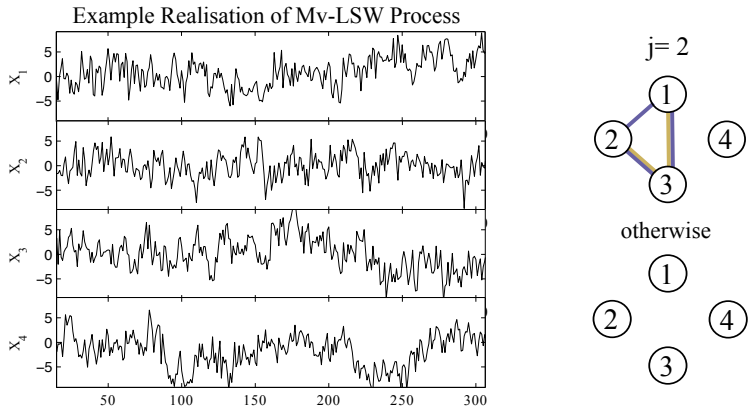


Figure 7.3.1 – Band limited coherence structure. Left: Example draw from a $p = 4$ dimensional MV-LSW process. Right: Graphical representation of noise structure limited in this case to band $j = 2$, purple and gold lines respectively represent non-zero coherence and partial coherence structure.

7.3 Synthetic Experiments

In this section, we consider synthetic experiments where we have full knowledge of the ground-truth structure. The performance of the basic smoothed estimator and the graphical estimator is compared. Specifically, we will assume that the ground-truth structure is in some sense sparse, i.e. that the process may be band limited, and dependency between data streams is limited.

In the LSW framework, data is generated according to Eq. 7.1.1, with the target of our estimation procedure being the inverse LWS matrix $\boldsymbol{\Theta}_j(z) \equiv (\boldsymbol{S}_j(z))^{-1} \equiv (\boldsymbol{V}_j(z)\boldsymbol{V}_j^{\top}(z))^{-1}$. We proceed by simulating a ground-truth iLWS matrix encoding a GGM at each scale with adjacency matrix $\boldsymbol{G} \sim$ ErdosRenyi(P, n) and then constructing:

$$\Theta_j^{0(i,l)} = \begin{cases} \sim Uniform[-\gamma, \gamma] & G_j^{(i,l)} = 1 \\ 0 & G_j^{(i,l)} = 0 \end{cases}, \text{ for } i \neq l \tag{7.3.1}$$

where γ acts to scale the iLWS off-diagonal elements, i.e. increasing the partial correlation between variables at a given scale. The transfer function $\boldsymbol{V}_j(z)$ can now be derived through LU decomposition. Realisations of the LWS process are then generated via Eq. 7.1.1, an example realisation can be seen in Figure 7.3.1.

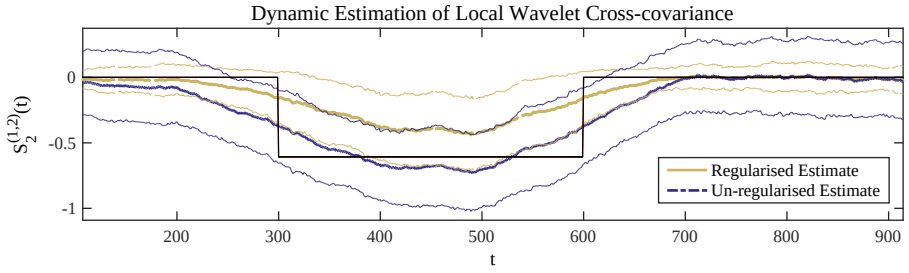


Figure 7.3.2 – Coherence estimation in the regularised and un-regularised mv-LSW framework. Dashed lines indicate the empirical standard deviation of the estimate over $N = 100$ realisations.

Estimator performance

To demonstrate the ability to track band limited structure we simulate precision matrices $\{\boldsymbol{\Theta}_j\}$ for $p = 4$ and $T = 1024$ $j = 1, \ldots, J = \log_2(T) = 10$. Off-diagonal structure, as generated by (7.3.1), is restricted to the specific band $j = 2$ as depicted in Figure 7.3.1.

The example in Figure 7.3.2 has piecewise constant ground-truth structure $\boldsymbol{S}_j(z)$, where we observe that both estimators are somewhat able to track changes. Although, as in previous chapters, the moving window smoothing is not capable of resolving jumps in the spectra. As discussed the Park estimator is unbiased, at least in the periods where the spectra is constant; however, we observe that it is relatively sensitive to the data, i.e. has large variance (see dashed lines). Our proposed MR-EGM estimator has biased estimation in the active period ($t = 300 - 600$), but reduced estimator variance. We note, that this is especially useful, when one considers that if the sparsity assumption is true then there will be a very large number of entries in the LWS matrix that are zero (c.f. the region $t = 600 - 900$ in Figure 7.3.2). In such cases, the imposed bias actually helps, as it reduces variance in the estimate around the true parameter.

In addition to the smoothing parameter M, the regularised spectral estimator introduces a sparsity tuning parameter λ. In the synthetic setting, we are in the fortunate situation where we know the ground-truth parameterisation, one can easily generate test and training sets by simply simulating more realisations from the LSW process in Eq. (7.1.1). If one desired, both

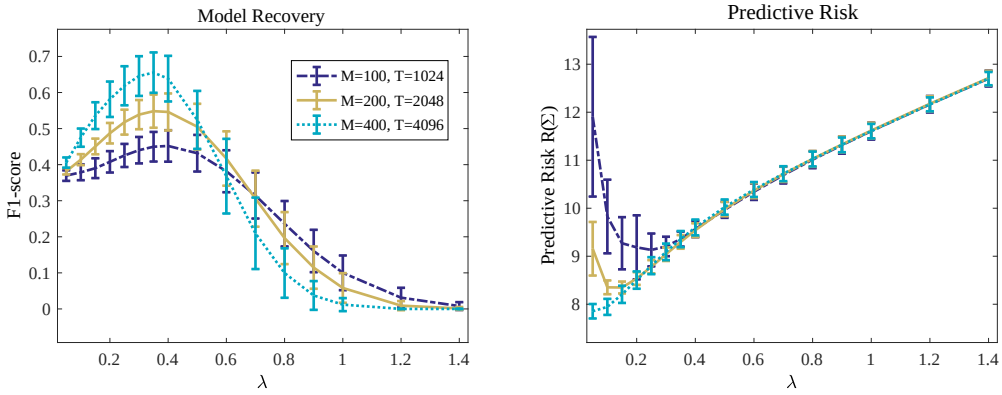


Figure 7.3.3 – Coherence estimation in the regularised and un-regularised mv-LSW framework. Error bars indicate empirical variance of measures over $N = 30$ realisations.

smoothing and regularisation parameters (M, λ) could be selected according to cross-validation based on an appropriate risk function. Such a procedure is attempted in a later section where we apply the graphical estimator to a real data-set.

For model consistency (i.e. selecting the correct sparsity pattern) one may consider maximising a balance of

$$\text{Precision} = |\hat{E}_j \cap E_j|/|\hat{E}_j| \ ,$$

and

$$\text{Recall} = |\hat{E} \cap E_j|/|E_j| \ ,$$

where $\hat{E}_j$, E_j denote respectively the estimated and ground truth edge sets at the jth level. Again, as in Chapter 3 we may consider the F_1-score, defined as

$$F_1 = 2(\text{precision} \times \text{recall})/(\text{precision} + \text{recall}) \ .$$

Alternatively, for predictive risk (approximating the true distribution) one can consider minimizing $R(\hat{\boldsymbol{\Theta}}_j) = \text{tr}(\hat{\boldsymbol{\Theta}}_j \boldsymbol{S}_j) + \log\det(\boldsymbol{\Theta}_j)$, the reader is referred to Zhou et al. (2010) for analysis of such risks in the non-wavelet setting (a similar risk analysis of the IFGL estimator is studied in Chapter 3).

To examine the effect of regularisation on the estimates we construct a set of experiments for different combinations of the parameters M, T, and λ. In these experiments, the dimensionality is fixed at $p = 10$ with $s = 10$ true

edges simulated from Eq. 7.3.1. For each length $T = \{1024, 2048, 4096\}$ a set of $N = 30$ observed processes is constructed according to the multivariate-LSW model. The smoothing window is scaled in proportion to the process length such that $M = \{100, 200, 400\}$. We then apply the regularised spectral estimator (7.2.3) to the set of processes and track both model-recovery (F_1 score) and the predictive risk.

The results as presented in Figure 7.3.3 are as one might expect, and demonstrate the benefit of regularisation when performing estimation in finite sample settings. With sparse models which corresponding to a large λ, we observe that estimator variance is reduced at all scales; this can be noted by looking at the empirical variance of the predictive risk measure. More strikingly, we note that in small sample environments, where M is small the un-regularised predictive risk is very large. Adding regularisation and imposing sparsity clearly helps reduce this, we note the distinctive kink in the predictive risk figure. Increasing the width of the smoothing window reduces the need for regularsation, for instance, by $M = 400$ the predictive risk is minimised for a setting of $\lambda \approx 0$.

In terms of model recovery, i.e. the recovery of the dependency pattern, we see that sparsity plays a key role at all smoothing window sizes. With a small window ($M = 100$), there is hardly any benefit to regularising in terms of model-recovery; the impact is felt more strongly in terms of predictive risk. However, when we get to larger window sizes, the situation is reversed; we do not need regulariser to reduce the risk, but it helps greatly in model recovery. Crucially, we note that the non-regularised estimator which is sparsity agnostic ($\lambda = 0$), cannot pick up model structure by itself, even when the smoothing window is very large.

The above experiments clearly demonstrate that regularised estimation of the spectra is beneficial in certain situtations. In the asymptotic setting, where $M \to \infty$ it appears that the regularised estimates offer an improvement in terms of model interpretability, i.e. the F_1- score seems to improve with increasing data. In the finite/small sample setting, the regulariser helps to reduce the predictive risk and reduces variance in the estimator. In the next section, we will discuss the application of such regularised coherence estimators to the analysis of electro-encethelographic (EEG) data.

7.4 Epileptic Electro-Encetheolograph (EEG) analysis

In the previous sections an overview of how regularised estimation methods may be adapted to the setting of LSW spectrum estimation. In this section it is considered how these methods can be applied in a real scientific application, namely that of understanding epilepsy seizures. More specifically, the goal here is to examine the similarity between seizure episodes for an individual suffering from epilepsy. The approach taken here, is to attempt to estimate a generative statistical model of the seizure process by associating this with an mv-LSW process. For the purposes of this study, we do not attempt to study the general population (of epilepsy patients), rather the population we are interested in representing is that of the electrophysiological behaviour for a specific patient.

7.4.1 Relation to previous work

Traditional clinical analysis of EEG data is limited to univariate analysis and the human interpretation of traces/spectra (Hunyadi 2014). In a more academic setting, studies have been developed for multivariate analysis; with several looking at the evolution of cross-correlation structure (Müller et al. 2005; Schindler et al. 2007). Often this has the aim of predicting or classifying epileptic activity (Hunyadi 2014). To date, I am not aware of any other studies that have considered the dynamics of epilepsy in the multivariate locally stationary wavelet setting proposed. The closest proposition to the method proposed here appears to be the work of Conlon et al. (2009), who consider cross-correlation structure between wavelet coefficients, i.e. at different scale levels. The work here differs from this approach in that a) bias is accounted for in the estimation of wavelet cross-covariance structure, and b) the robustness of estimation is considered through sparse de-noising.

In order to fit a generative model for the development of seizures, we need to make the assumption that the evolution of dynamics for each seizure are drawn from some general (patient specific) distribution. The aim of our modeling work is thus to estimate and gain some insight on the structure of this

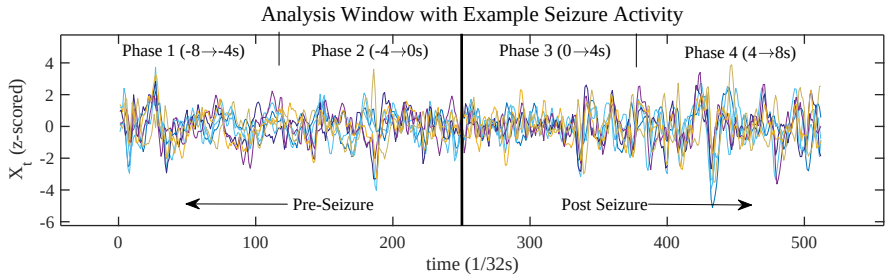


Figure 7.4.1 – Epileptic EEG sensor readings.

hypothesised distribution. Due to variation in seizure length, the assumption that all seizures are comparable seems suspect from the start. However, even though the length of the seizures is varied, one may postulate that high-level electrophysiological behaviour may share some similarities. The approach presented here will focus on investigating these similarities by modeling the dependency between electrodes as a function of both time and frequency.

Data and Pre-processing

In this study data from the PhyisoNet (Goldberger et al. 2000) database is examined which reports electroencephalography (EEG) traces for anonymous pediatric subjects at the Children's Hospital Boston. While the dataset contains seizures for a number of patients, the analysis presented here is restricted to a single female subject labelled *chb01_*, see Figure 7.4.1 and Table 1 for more details. The data is collected with 23 electrodes located according to the international 10-20 system. A graphical representation of electrode placement can be seen in Figure 7.4.6. For the selected subject seizures vary in length with an average duration of 63s.

As depicted in Figure 7.4.1, the experiment focusses on the early onset of seizures where we study the period $t_{\text{pre}} = 8\text{s}$ before a seizure, to $t_{\text{end}} = 8\text{s}$ after a seizure starts. The data is originally recorded at 256Hz, before being subsampled to 32Hz in preparation for analysis, the resulting data-set is of size $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)^\top \in \mathbb{R}^{p \times T}$, where $p = 23$ is the number of electrodes, and $T = 512$ is the number of samples. The analysis presented here is limited to the $N_s = 7$ seizures recorded when the subject was 11 years old. Before applying the wavelet analysis described below, the data is z-scored such that

Table 1 – Relevant data-sets and periods for study as taken from the Children's Hospital Boston Massacusets Institute of Technology (CHB-MIT) dataset (www.physionet.org).

seizure	file	start	end
1	chb01_03	2996	3036
2	chb01_04	1467	1494
3	chb01_15	1732	1772
4	chb01_16	1015	1066
5	chb01_18	1720	1810
6	chb01_21	327	420
7	chb01_26	1862	1963

when taken across the whole analysis window, it is centered such that it has mean zero, and variance one.

7.4.2 Cross-validation

Most estimation procedures require a number of tuning-parameters to be selected. In this case, we will apply the regularised estimator developed in the previous sections. For the purpose of this section, we will refer to (7.2.3) as the *multi-resolution exploratory graphical model (MR-EGM)* estimator (c.f. Gibberd and Nelson (2015a)). This estimator has two tuning parameters, the size of the smoothing window M, and the regualriser strength λ. These parameters must be set such that they are relevant for a given data-set. In the previous section, the behaviour of the the estimator was discussed in a synthetic setting where one could repeatedly draw samples from a known ground-truth distribution. These could then be used to compare the estimated and known distributions. However, in practice we don't know a-priori what distribution the data is drawn from, instead we can only train parameters on the limited samples (in our case the number of seizures N_s) that we have from the assumed generative model.

The approach proposed here, is to attempt to assess the predictive risk on a holdout test dataset (corresponding to one seizure), and train on the remaining $N_s - 1$ seizures. Rotating the test data-set across the seizures in a leave-one-out cross-validation fashion should then give us some idea of how well the model generalises from the trained data. Such a procedure mimics that performed

in Chapter 3, in that case we were trying to generalise across different days of network activity whereas we here try to generalise across seizures.

To construct a risk function one can consider extending the idealised setting where we know the ground-truth distribution. For the multivariate Gaussian distribution, the predictive risk for a pair of ground truth $\mathbf{\Sigma}$ and estimated $\hat{\boldsymbol{S}}$ covariance matrices is given as (Zhou et al. 2010):

$$R(\hat{\boldsymbol{S}}) = \operatorname{tr}((\hat{\boldsymbol{S}})^{-1}\mathbf{\Sigma}) + \log\det(\hat{\boldsymbol{S}}) \ .$$

Consider $f_{\boldsymbol{S}}$ to be the density for $\mathcal{N}(\mathbf{0}, \hat{\boldsymbol{S}})$ and the data is drawn under the ground truth structure $Z \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, then one notes that up to a constant $R(\hat{\boldsymbol{S}}) = -2E_0[\log(f_{\hat{\boldsymbol{S}}}(Z))]$. The likelihood and the risk are thus related via the density function. The risk is proportional to the negative expected log-likelihood (given the estimate) drawing under the true model. *Note: the risk function we use here is directly related to the Gaussian model, this again results in us making an assumption about the properties of the data generating process.*

Constructing a measure of risk

In practice we do not have access to $\mathbf{\Sigma}$, it is therefore suggested to estimate this from the test data as

$$\boldsymbol{S}_{j,t}^{(i_{\text{test}})} = \sum_{l=1}^{J} A_{jl}^{-1} \boldsymbol{I}_{l,t} \ ,$$

where $\boldsymbol{I}_{j,t}$ is the raw-periodogram matrix. We then define the leave one out cross-validation risk for each time point $t$and scale j as:

$$R_{\text{loo}}(\hat{\boldsymbol{S}}_{j,t}) := \frac{1}{N_s} \sum_{i_{\text{test}}=1}^{N_s} \sum_{i \neq \text{i}_{\text{test}}} \left[\operatorname{tr}((\hat{\boldsymbol{S}}_{j,t}^{(i)})^{-1} \boldsymbol{S}_{j,t}^{(i_{\text{test}})}) + \log\det(\hat{\boldsymbol{S}}_{j,t}^{(i)}) \right] . \tag{7.4.1}$$

Throughout the rest of the section, let us denote the un-regularised but smoothed coherence estimator (7.2.2) as $\hat{\boldsymbol{S}}^{\text{K}}(M)$; and the MR-EGM covariance estimator (7.2.3) as $\hat{\boldsymbol{S}}^{\text{MR}}(M, \lambda) := \hat{\mathbf{\Theta}}^{-1}(M, \lambda)$. Substituting these estimates into (7.4.1) in place of $\hat{\boldsymbol{S}}_{j,t}$ enables us to measure the leave-one-out cross-validated risk for both methods.

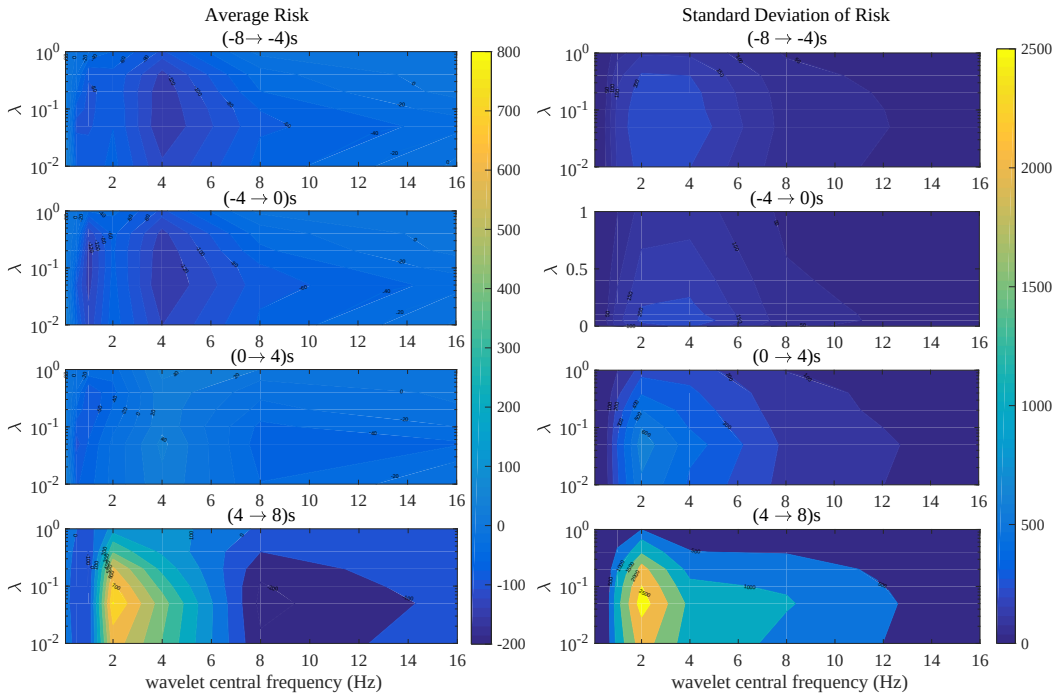


Figure 7.4.2 – Left: averaged risk throughout different periods leading up to, and throughout seizure activity. Right: empirical standard deviation of the risk.

Experiments

To examine the effect of regularisation on the risk (7.4.1) a series of experiments were performed over a range of λ. For simplicity, and the fact the smoothing parameter M is common to both methods only the effect of changing the regularisation parameter λ on the cross-validated risk is reported in the experiments. Since $R_{\text{loo}}(\hat{\boldsymbol{S}}_{j,t})$ reports a value for every scale j and timepoint t it will be highly variable. In order to gain some level of insight, we must therefore average the risk not only across seizures, but also across portions of the time.

Figure 7.4.2 presents the risk level $R_{\text{loo}}(\hat{\boldsymbol{S}}_{j,t})$ averaged across different segments of seizure activity. Specifically, the risk surfaces $R_{\text{loo}}(j, \lambda)$ are averaged over time in the phases leading up to, and throughout a seizure (-ve values indicate the time in seconds prior to seizure, +ve values indicate post seizure

periods). The unregularised estimate corresponding to $\hat{\boldsymbol{S}}^{\mathrm{K}}$ is found at $\lambda = 0$ (in the figures this appears as 10^{-2} due to the log-scaling).

The results are interesting, but also quite hard to interpret. For example, while it seems introducing a non-zero λ can help reduce the cross-validated risk in the period leading up to a seizure, this does not hold when the seizure actually starts occurring. For example, in the pre-seizure activity, where it appears that a regulariser of $\lambda \approx 0.1$ appears to minimise the risk; but a similar value during the seizure results in a very large risk measure. Additionally, and unlike in the synthetic data, the standard deviation of the risk is very large across many parts of the risk surface. Furthermore, while the regulariser helps reduce risk in the pre-seizure period, it does not help mitigate variance in the risk surface. The reasoning for this is unknown, but one should take into account the rather extreme demands we are placing on the cross-validation scheme given that we only have $N_s = 7$ seizures worth of data. With regards to the variation of optimal λ over time; it is quite reasonable that the optimal strength of regulariser change as seizure activity progresses. Indeed, it appears that the marginal variance of the signal increases throughout a seizure, which would intuitively suggest a large λ is required during these periods.

7.4.3 Epileptic brain Dynamics

In this section, the aim is to enable some level of interpretation of seizure activity in the context of the estimated MR-EGM model parameters. Our aim is to highlight and understand clusters of activity that may represent seizure activity. If such clusters exist, this may explain to some extent why the generalisation performance of the model decreases in the seizure activity phase. Crucially, in a clinical setting this may help diagnose and characterise complex seizure activity.

Our suggested multi-resolution methods results in a very high dimensional feature space. In a standard univariate analysis, the one-dimensional NDWT results in $\mathcal{O}(T \log_2(T))$ parameters. However, in the multivariate setting, the number of model parameters is compounded by the requirement to model coherence structure; in the mv-LSW framework the number of parameters scales as $\mathcal{O}(p^2 T \log_2(T))$. In order to analyse the estimated coherence structure $\{\hat{\boldsymbol{S}}_{j,t}\}_{j=1}^{J}$ as a function of time we turn to a principle component based

approach. Specifically, a feature matrix is constructed by vectorising the coherence at each time-point. Let

$$\boldsymbol{y}_t = (\hat{S}_{j,t}^{(1,1)}, \dots \hat{S}_{j,t}^{(p,p)} | \, \forall \, j = 1, \dots, J)^\top \in \mathbb{R}^{p_f} \, ,$$

represent one data point in the resultant $p_f = p^2 J$ dimensional mv-LSW feature space. These can then be concatenated into a matrix $\boldsymbol{Y} = (\boldsymbol{y}_1, \dots, \boldsymbol{y}_T)^\top$.

***Remark* 7.3.** *A note on PCA*

A popular tool for visualising multivariate feature vectors is principle component analysis (PCA). *Mathematically, PCA is defined as an orthogonal linear transformation of feature-vectors such that the variance of the feature vector is maximally described by the first principle component, i.e. for the first principle component we have* $\boldsymbol{w}_1 = \arg\max_{\boldsymbol{w}} \left[(\boldsymbol{w}^\top \boldsymbol{Y}^\top \boldsymbol{Y} \boldsymbol{w}) / (\boldsymbol{w}^\top \boldsymbol{w}) \right]$. *Since* $(\boldsymbol{Y}^\top \boldsymbol{Y}) \in \mathbb{R}^{p_f \times p_f}$ *is symmetric the first k-th principle components correspond to the eigenvectors associated with the largest k eigenvalues. Projecting features onto these eigenvectors allows us to visualise a high-dimensional features in a lower dimensional sub-space. In our case we will examine the brain dynamics within the highly redundant feature space provided by the MR-EGM and mv-LSW frameworks. Finally, rather than analyse seizures separately, it is of interest to select principle components that are relevant across all seizures. This may then enable the clustering of seizures, and provide valuable insight to clinicians. In order to jointly analyse the seizures, prior to performing PCA the feature vectors across all* N_s *seizure episodes are concatenated.*

To start with, it is interesting to see what the PCA analysis produces in the signal space, i.e. using the signal directly as a feature. The results of such an experiment are given in Figure 7.4.3, where the raw observations $\boldsymbol{X}_i \in \mathbb{R}^{p \times T}$ for $i = 1, \dots, N_s$ are projected into the first two principle components. There is no clear separation between seizures, but it is possible to distinguish seizure and non-seizure activity simply in terms of the variance; the black dots represent pre-seizure activity.

We will now compare this with the mv-LSW based approach where the dynamics of seizures may now be linked to the dynamics of the estimated wavelet coherence structure. In this setting the number of features is very high ($P_f = 4761$). However, due to the large redundancy in the wavelet representation the amount of variance explained in the first two components is

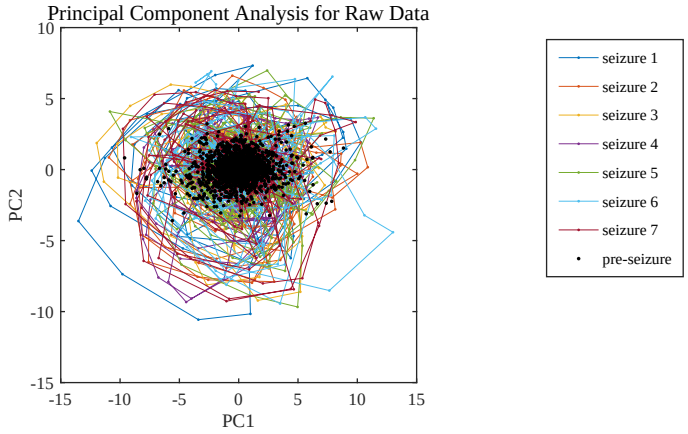


Figure 7.4.3 – Raw signals projected into the two dominant principle components.

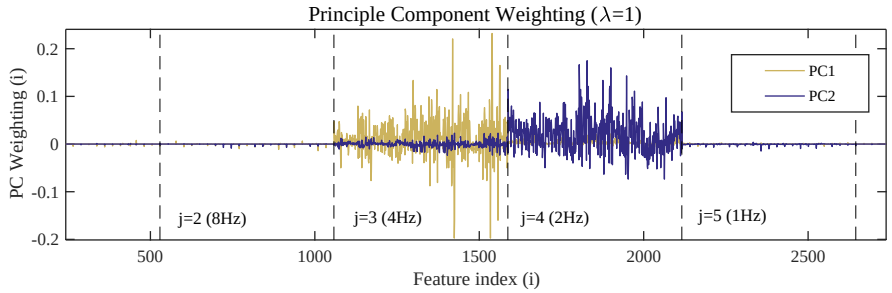


Figure 7.4.4 – Principle components of the estimated spectra after concatenation across multiple seizures.

still relatively high. To investigate the role of regularsation in the parameter estimates we perform estimation with different regulariser parameters $\lambda = \{0, 0.3, 1\}$. Table 2 presents the results of PCA applied to the resulting estimates. Clearly, as the level of sparsity is increased, we observe a corresponding increase in the proportion of variance that can be explained by the first two principle components. The effect of this can also be seen by visualising the resulting principle component vectors. For example, Figure 7.4.4 plots the loading of the principle components $\boldsymbol{w}_1, \boldsymbol{w}_2$ with the regularised estimator. As demonstrated by the sparsity column in Table 2, the proportion of zeros in the principle components increases as a function of regulariser strength.

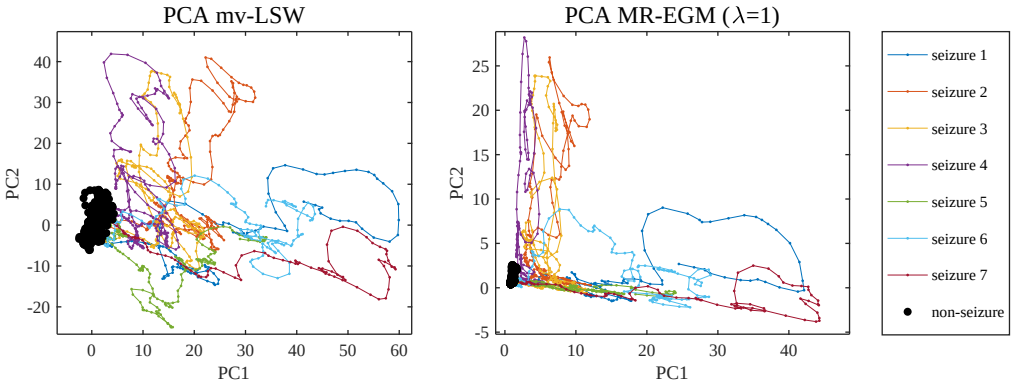


Figure 7.4.5 – Projecting the seizure activity onto the identified principle components (which represent the estimated spectral structure) reveals that seizures follow certain pathways through the spectral feature space.

Table 2 – Comparison of principle components of the MR-EGM model as a function of regularisation strengths. Sparsity is the proportion of zero elements in the principle components.

λ	sparsity	Var(PC1)	Var(PC2)
0	0	31.9%	15.5%
0.3	0.23	41.3%	14.2%
1	0.51	45.2%	13.9%

The benefit of principle component analysis is that it enables us to visualise the key dynamics of the seizure process in two dimensions. Such a visualization of the MR-EGM feature space is provided in Figure 7.4.5, where we compare dynamics in both regularised and non-regularised estimates. Clearly, there is a considerable difference in terms of how the seizure activity aligns itself with the estimated eigen-vectors. While in the non-regularised case, we can easily distinguish seizure and non-seizure activity, clustering between seizure activity is not so obvious. Now contrast this with the regularised case where $\lambda = 1$. Not only is seizure and pre-seizrure activity well separated, but seizure activity appears to fall into clusters which follow the principle components.

Such a separation might suggest that the subjects seizure activity fall into two distinct behavioral classes. To examine this behaviour, we can take the estimated principle components and project them back into the form of covariance matrices. Rather than interpreting the eigenvectors as a vector (c.f.

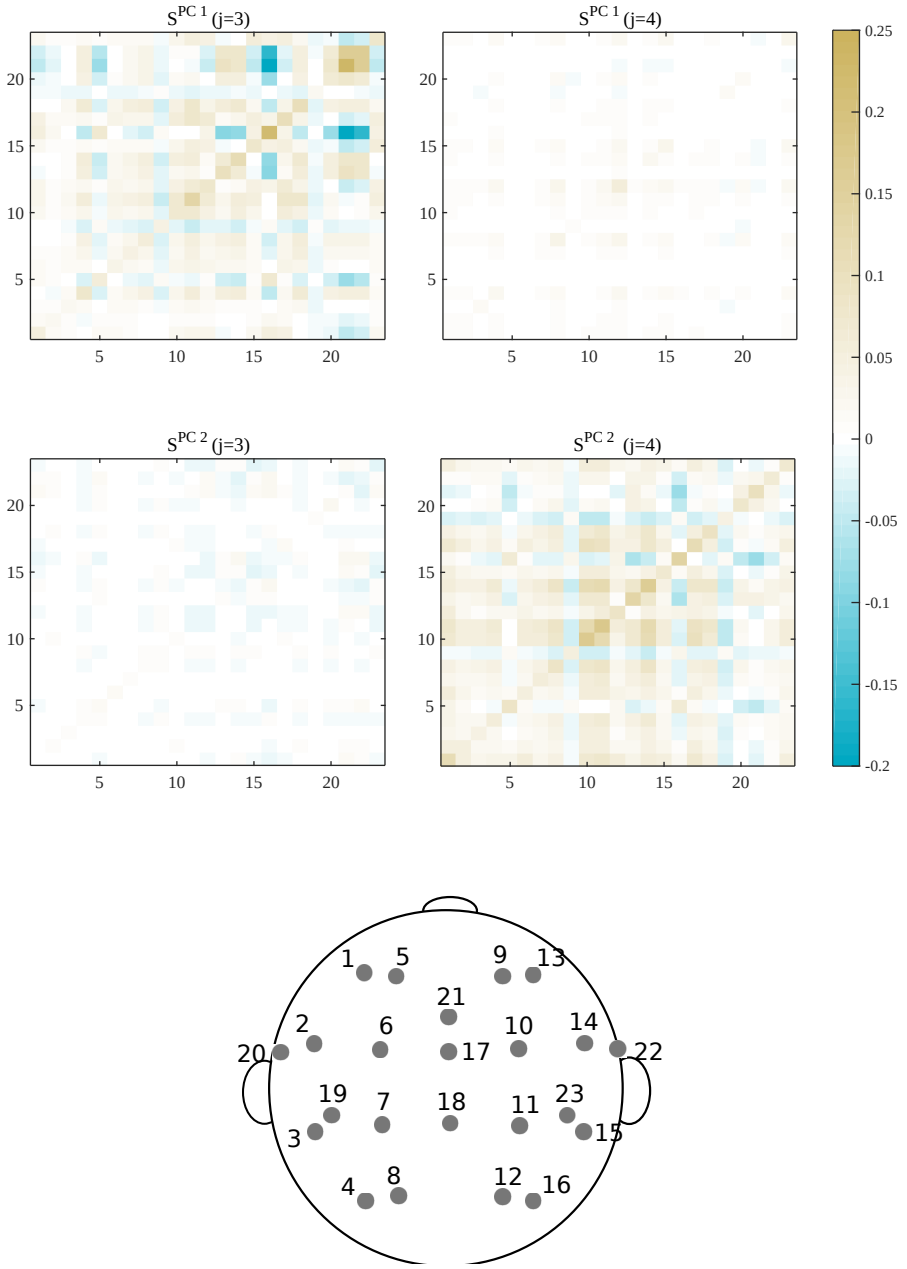


Figure 7.4.6 – Heatmap of estimate wavelet cross-spectrum corresponding to scales as identified in Figure 7.4.4.

Figure 7.4.4), we can now interpret these as covariance matrices restricted to different scale levels. For example, the first $p^2 = 529$ components of the principle component can be matched to the covariance matrix at the scale level

$j = 1$. Notationally, I will refer to such matrices as the *principle covariance* $\hat{\boldsymbol{S}}_j^{PC_1}$ for scale level j. From inspection of the principle components (Figure 7.4.4), we clearly see that most of the structure is contained within the scale levels corresponding to $j = 3$ and $j = 4$. Investigating the principle covariance matrices for these two components we notice very obvious differences in the structures (see Figure 7.4.6). As the eigenvectors describing $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are orthogonal, there is not much overlap in terms of the covariance structure. However, it is interesting to note that the structure appears to be split primarily by frequency band; while $\hat{\boldsymbol{S}}^{PC_1}$ is mainly confined to the $j = 3$ band corresponding to 4Hz (using the Haar wavelet), the other principle covariance matrix is dominant within the band $j = 4$ at 2Hz. The actual covariance structures seem to suggest that some seizures (1,6, and 7) may be characterised by $\hat{\boldsymbol{S}}_3^{PC_1}$, indicating a lack of activity in the nodes corresponding to the bottom left of the scalp (nodes 2,3,4,19), and strongly anti-correlated activity between nodes 16 and 21,22 (see Figure 7.4.6). The rest of the seizures are more aligned with $\hat{\boldsymbol{S}}_4^{PC_2}$ which indicates strong correlated activity in the mid-right area of the scalp (nodes 10, 11, and 12).

7.4.4 Discussion

In this section the MR-EGM models of the previous section (Gibberd and Nelson 2015a) were used to model time-frequency dependency structure within epileptic EEG data. Our attempts to understand the generalisability of the mv-LSW model show that controlling complexity via a sparsity constraint can reduce some measure of predictive risk. In particular, the model appears to capture structure in the 4Hz band corresponding to Theta waves in the period prior to seizure, however, this predictability is lost in the seizure phases. Previous studies (Conlon et al. 2009) demonstrated that activity in the Theta band was significantly reduced during some seizures, it is possible a mix of this reduction and potential clustering of seizure activity may account for this reduction in generalisation ability. On the other hand, such a method of assessing risk and setting regulariser parameters should not be taken too seriously when performed over such few seizures.

Further to the consideration of predictive risk, it was demonstrated how a principle component analysis of the estimated parameters may shed light

on seizure activity. In particular, encouraging sparsity appeared to help capture key components in the highly redundant mv-LSW parameter space. For example, within the subject analysed, two coherence matrices are identified that can characterise epileptic activity. Without further clinical data as to the location of the epileptic region in this particular subject, it is hard to further interpret these results more generally.

It is important to stress that this study was performed on a single subject, and future work may look at studying a wider range of seizures and/or across multiple patients. It would be particularly interesting if one could identify characteristic dependency patterns that generalise to large numbers of patients or seizures. One might also wish to expand the range of frequencies analysed; other studies suggest that higher frequency activity in the 60Hz band is a prominent feature of seizures (Conlon et al. 2009). Additionally, it may be of interest to assess the performance of other wavelet families, or implement alternative piecewise smoothing methods to attempt to detect different phases in epileptic activity. This latter point may consider extending the graph estimation algorithms of Chapters 3, 4 to the multivariate LSW framework.

7.5 Summary

This chapter bridges the gap between the first and latter half of the thesis by introducing a multivariate extension of the LSW process. In the multivariate setting, the LSW model not only needs to describe auto-correlation, but also cross-correlation structures. As such, it requires far more parameters than the standard one-dimensional construction. Estimation of these parameters with finite data proves a challenge, and regularisation can help in these settings. Specifically, the chapter demonstrates how the graphical lasso (as previously used to identify GGM) can be employed to estimate the cross-spectra of a multivariate LSW process.

Synthetic experiments demonstrate the considerable benefit of such regularisation where the underlying coherence structure is sparse. Not only does this help improve interpretability of the coherence estimates (we recover the correct sparsity structure), but it also improves the performance of the estimator in recovering the true distribution. Following these experiments, the

regularised estimator is applied to the analysis of real-world EEG data in order to characterise epileptic seizures. In this case, regularisation helps identify a set of sparse principle components which enables seizure activity to be split into two classes (in the single patient under analysis).

Chapter 8

Conclusion and Future Work

Throughout this thesis several methods for the analysis of multivariate time-series have been developed; however, there is still much more one could do to extend these approaches. In this chapter, I will take time to gather ideas from across different parts of the thesis before offering some concluding remarks. Some of the directions highlighted here are in fact already under consideration and are loosely ordered in terms of difficulty, easiest first.

8.1 Joint Changepoint and Graph estimation in High-Dimensions

Chapter 4 introduced some theoretical analysis for the GFGL class of estimators and primarily considered the standard dimensional setting where $T \to \infty$ and p is fixed. The original aim of this theoretical work was to obtain a high-dimensional result where both p, T increase, in such a case, one may have $p > T$, even asymptotically. A pathway to obtaining such results is discussed at the end of Chapter 4. In particular, it is suggested that obtaining a probabilistic bound on $\lambda_1 \geq 2R^*(\nabla L_{\hat{B}}(\boldsymbol{\Theta}_0; X_{1;T}))$ will be crucial to demonstrate high-dimensional consistency. In fact, it should not be too hard to derive such a bound, but considering the detail in calculating rates etc, I prefer to leave these results as future work instead of including them in the thesis.

On a more applied note, it is worth noting that there have been several requests for software to implement the GFGL/IFGL estimators. The majority of interest is from researchers considering fMRI or financial data-sets. However, while all the code developed in this thesis is available on request, it is not necessarily production ready or appropriately packaged for third parties. Given, the interest in the dynamic graphical estimators, I plan to make available a software package to make these methods more easily accessible. Preliminary work in this direction can be found via the python GraphTime package. The package is available via Github: https://github.com/GlooperLabs/GraphTime, or through the Python package manager via *"pip install Graphtime"*. 3.

8.2 A General M-Estimation Framework for LSW Spectra

One of the initial directions of interest in considering the LSW/LSF family of models was to see whether regularised estimation could provide a useful tool for statistical estimation. The results in this thesis on the wavelet modelling side are very much empirical in nature; however, this leaves lots of room for further theoretical analysis. In particular, when considering sparsity aware estimators, one naturally asks, how well can our estimators recover this sparsity. To answer such questions, and to fully understand the power of sparse estimators, one needs to encode sparsity assumptions in the model construction. For example, in the LSW framework, we often assume that signals are "band-limited" meaning that contribution to the variance of the process is limited to a subset of available frequencies/scales.

On an empirical level, the analysis of Section 7.3 can be thought of in this setting, whereby only a few elements in the coherence are non-zero. The fact that the graphical lasso based estimator can identify this model structure is promising, and as demonstrated by these experiments introducing sparsity can help form a robust estimate of the joint distribution. For example, in Fig. 7.3.3, a sparsity aware estimator obtains performance levels similar to of the dense model, but with using only half the amount of data. For a theoretical analysis of such estimators, one may consider adding strict sparsity assumptions in the process definition (Def. 7.1). The challenge would then be to assess how these

assumptions propagate into the resultant Fourier/wavelet coefficients. From this one could construct an appropriate likelihood for the spectrum which would form a basis for M-estimation.

8.3 Non-Gaussian, Non-Stationary Processes

The models and estimators presented in this thesis are primarily based on the parametric Gaussian distribution. However, there are many situations where such assumptions are restrictive. For example, the analysis of network traffic in Sec. 3.4.2 was limited in time-resolution due to the Gaussian assumption breaking down for small bin sizes. While, the wavelet extensions enable us to study and model dependencies at different length scales, we would also like to generalise our model to be flexible with regards to the distributional assumptions.

One way of achieving this is by parameterising the marginal and dependency structure separately, through what is known as a copula. As a result of Sklar's theorem (Choroś et al. 2010), we may write the joint density f in terms of its univariate marginals $F_1, \ldots, F_p$ and densities $f_1, \ldots, f_p$ as:

$$f(x_1, \ldots, x_p) = c(F_1(x_1), \ldots, F_p(x_p)) \prod_{i=1}^{p} f_i(x_i) \ ,$$

where $c = (u_1, \ldots, u_p) = \partial C(u_1, \ldots, u_p)/\partial u_1 \ldots \partial u_p$ is the density of the p-dimensional copula $C(u_1, \ldots, u_p; \boldsymbol{\theta})$. The above construction implies that we can decompose the log-likelihood as the sum of a dependent and marginal terms

$$L = \underbrace{L_c}_{\text{dependence}} + \underbrace{\sum_{i=1}^{p} L_i}_{\text{marginals}} \ .$$

Such use of copulas has been studied in a graphical context by the *non-paranomal* family of models promoted in J. Lafferty et al. (2012). If we can adopt GFGL within a copula setting, we may be able to generalise our model, to handle other types of data, for example multivariate count data (E. Yang et al. 2013).

Finally, in the context of wavelet or Fourier based models, it is interesting to note that while the process itself may not be Gaussian, the resulting Fourier/wavelet coefficients are, at least asymptotically (Brillinger 1981; Stevens 2013). Such an argument can be used to motivate a Gaussian likelihood based model for the wavelet coefficients, even in the setting where the process itself may be non-Gaussian. However, it remains as further work to formalise these kind of results, and it would be interesting to consider the robustness of the Gaussian assumption in finite sample settings.

8.4 Concluding Remarks

The contributions of this thesis are two-fold; firstly, to introduce and provide estimation mechanisms for dynamic graphical models, and secondly, to extend M-estimation concepts to a class of locally-stationary wavelet processes.

With regards to dynamic graph estimation, two novel classes of estimator (IFGL and GFGL) are proposed; furthermore, computational methods are developed to allow for the practical implementation of these estimators. The developed methods are demonstrated on two important applications, namely, the detecting changepoints in high-dimensional genetic time-course data, and the analysis of computer network traffic. Such applications demonstrate how accounting for dynamics can generate enhanced insight, particularly at the exploratory stage of analysis. Not only are the proposed dynamic graph estimators amenable to computation, but they also allow for some level of theoretical statistical analysis. To this end, under certain conditions on the generating process, it is demonstrated that GFGL can asymptotically recover changepoint positions.

The second half of the thesis focusses on the application of regularised smoothing methodologies to the spectra of classes of; 1) univariate, 2) multidimensional, and 3) multivariate LSW models. In all cases, the regularised methods perform well, often outperforming traditional estimators. The M-estimation framework enables one to easily incorporate sparsity or smoothness priors when identifying the spectra. Not only can the proposed estimators produce more sensible estimates (c.f. avoiding negative values in the spectra), but can also lead to enhanced interpretation. For example, in the multivariate setting, analysis assuming a sparse coherence structure enabled both the

better description (improved explained variance) and greater interpretation (enhanced cluster separation). However, while this thesis demonstrates that spectral M-estimation is empirically useful, it does not develop a theoretically rigorous analysis of such estimators; rather, such treatments provide an exciting opportunity for further research.

In summary, I hope this thesis provides a useful and balanced resource, for not only those who are interested in theoretically describing non-stationary processes, but also pragmatically understanding what is going on behind time-series data.

Bibliography

Ahmed, A. and E.P. Xing (2009). "Recovering time-varying networks of dependencies in social and biological studies". In: *Proceedings of the National Academy of Sciences of the United States of America* 106.29, pp. 11878–83. ISSN: 1091-6490. DOI: 10.1073/pnas.0901910106 (cit. on pp. 79, 81, 84, 86–88).

Akaike, H. (1973). In: *In 2nd International Symposium on Information Theory*, pp. 267–81 (cit. on p. 39).

Alaíz, C.M., A. Barbero, and J.R. Dorronsoro (2013). "Group Fused Lasso". In: *Artificial Neural Networks and Machine Learning – ICANN 2013*. Springer Berlin Heidelberg, pp. 66–73. DOI: 10.1007/978-3-642-40728-4_9 (cit. on p. 94).

Angelosante, D. and G.B. Giannakis (2011). "Sparse graphical modeling of piecewise-stationary time series". In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. DOI: 10.1109/icassp.2011.5946893 (cit. on pp. 85, 103).

Arbeitman, M.N. et al. (2002). "Gene expression during the life cycle of Drosophila melanogaster". In: *Science* 297.5590, pp. 2270–5. DOI: 10.1126/science.1072152 (cit. on p. 105).

Attrill, H. et al. (2016). "FlyBase: establishing a Gene Group resource for Drosophila melanogaster". In: *Nucleic Acids Research* 44(D1). DOI: 10.1093/nar/gkv1046 (cit. on p. 105).

Banerjee, O., L. El Ghaoui, and A. d'Aspremont (2008). "Model Selection Through Sparse Maximum Likelihood Estimation for Multivariate Gaussian or Binary Data". In: *Journal of Machine Learning Research* 9, pp. 485–516 (cit. on pp. 43, 61, 79, 81, 99).

Bauschke, H.H. and P.L. Combettes (2008). "A dykstra-like algorithm for two monotone operators". In: *Pacific Journal of Optimization* 4.3, pp. 383–391. ISSN: 13489151 (cit. on p. 95).

Beck, A. and M. Teboulle (2009). "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems". In: *SIAM Journal on Imaging Sciences* 2.1, pp. 183–202. DOI: 10.1137/080716542 (cit. on p. 50).

Bhattacharya, P., A. Biswas, and S.P. Maity (2014). "Wavelets-Based Clustering Techniques for Efficient Color Image Segmentation". In: *Proceedings of the Second International Conference on Advanced Computing, Networking and Informatics (ICACNI-2014)*, pp. 237–244. DOI: 10.1007/978-3-319-07353-8_28 (cit. on p. 224).

Bleakley, K. and J.P Vert (2011). "The group fused Lasso for multiple change-point detection". In: *arXiv preprint* (cit. on pp. 94, 104, 117, 118).

Bowers, K.J. and S.D. Johnson (2003). "Measuring the Geographical Displacement and Diffusion of Benefit Effects of Crime Prevention Activity". In: *Journal of Quantitative Criminology* 19.3, pp. 275–301 (cit. on p. 163).

Box, G.E.P. and N.R Draper (1986). *Empirical Model-building and Response Surface*. New York, NY, USA: John Wiley & Sons, Inc. ISBN: 0-471-81033-9 (cit. on p. 58).

Boyd, S., N. Parikh, and E. Chu (2011). "Distributed optimization and statistical learning via the alternating direction method of multipliers". In: *Foundations and Trends in Machine Learning* 3.1, pp. 1–122. DOI: 10.1561/2200000016 (cit. on pp. 48, 51, 52, 96).

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press. DOI: 10.1017/cbo9780511804441 (cit. on pp. 31, 48).

Brillinger, D.R. (1972). "The spectral analysis of stationary interval functions". In: *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Theory of Statistics*, pp. 483–513 (cit. on p. 170).

— (1974). "Fourier analysis of stationary processes". In: *Proceedings of the IEEE* 62.12, pp. 1628–1643. ISSN: 0018-9219. DOI: 10.1109/PROC.1974.9682 (cit. on p. 170).

— (1981). *Time Series: Data Analysis and Theory*. Philadelphia: SIAM. ISBN: 9780898715019 (cit. on pp. 164, 170, 252).

Brodatz, P. (1966). *Textures: A Photographic Album for Artists and Designers*. Dover Publications (cit. on p. 222).

Bühlmann, P. and S. van de Geer (2011). *Statistics for High-Dimensional Data*. Springer Series in Statistics. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 481–538. DOI: 10.1007/978-3-642-20192-9 (cit. on pp. 46, 48, 58, 71, 73).

Bunea, F., A. Tsybakov, and M. Wegkamp (2007). "Sparsity oracle inequalities for the Lasso". In: *Electronic Journal of Statistics* 1, pp. 169–194. DOI: 10.1214/07-EJS008 (cit. on p. 69).

Cai, T., W. Liu, and X. Luo (2011). "A Constrained l1 Minimization Approach to Sparse Precision Matrix Estimation". In: *Journal of the American Statistical Association* 106.494, pp. 594–607. DOI: 10.1198/jasa.2011.tm10155 (cit. on p. 59).

Candes, E.J. and T. Tao (2005). "Decoding by linear programming". In: *IEEE Transactions on Information Theory* 40698.December, pp. 1–22. DOI: 10.1109/tit.2005.858979 (cit. on pp. 42, 69).

Chen, C. et al. (2016). "The direct extension of ADMM for multi-block convex minimization problems is not necessarily convergent". In: *Mathematical Programming* 155.1-2, pp. 57–79. DOI: 10.1007/s10107-014-0826-5 (cit. on p. 217).

Chen, J. and Z. Chen (2008). "Extended Bayesian information criteria for model selection with large model spaces". In: *Biometrika* 95.3, pp. 759–771. DOI: 10.1093/biomet/asn034 (cit. on pp. 39, 40).

Cho, H. and P. Fryzlewicz (2012). "Multiscale and multilevel technique for consistent segmentation of nonstationary time series". In: *Statistica Sinica*. DOI: 10.5705/ss.2009.280 (cit. on p. 196).

— (2015). "Multiple-change-point detection for high dimensional time series via sparsified binary segmentation". In: *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 77.2, pp. 475–507. DOI: 10.1111/rssb.12079 (cit. on pp. 196, 226).

Chorośś, B., R. Ibragimov, and E. Permiakova (2010). "Copula Estimation". In: *Copula Theory and Its Applications*. Springer Berlin Heidelberg, pp. 77–91. DOI: 10.1007/978-3-642-12465-5_3 (cit. on p. 251).

Cohen, E.A.K. and A.T. Walden (2010). "A Statistical Study of Temporally Smoothed Wavelet Coherence". In: *IEEE Transactions on Signal Processing* 58.6, pp. 2964–2973. ISSN: 1053-587X. DOI: 10.1109/TSP.2010.2043139 (cit. on pp. 170, 226).

— (2011). "Wavelet Coherence for Certain Nonstationary Bivariate Processes". In: *IEEE Transactions on Signal Processing* 59.6, pp. 2522–2531. DOI: 10.1109/TSP.2011.2123893 (cit. on pp. 170, 226).

Coifman, R.R. and D.L. Donoho (1995). "Translation-Invariant De-Noising". In: *Wavelets and Statistics*. Springer New York, pp. 125–150. DOI: 10.1007/978-1-4612-2544-7_9 (cit. on p. 189).

Combettes, P.L. and J.C. Pesquet (2011). "Proximal Splitting Methods in Signal Processing". In: *Springer Optimization and Its Applications*. Springer New York, pp. 185–212. DOI: 10.1007/978-1-4419-9569-8_10 (cit. on pp. 52, 94, 95).

Conlon, T., H.J. Ruskin, and M. Crane (2009). "Seizure characterisation using frequency-dependent multivariate dynamics". In: *Computers in Biology and Medicine* 39, pp. 760–767. DOI: 10.1016/j.compbiomed.2009.06.003 (cit. on pp. 236, 246, 247).

Cont, R. (2005). "Long range dependence in financial markets". In: *Fractals in Engineering: New Trends in Theory and Applications*, pp. 159–179. DOI: 10.1007/1-84628-048-6_11 (cit. on p. 163).

Dahlhaus, R. (1996). "On the Kullback-Leibler information divergence of locally stationary processes". In: *Stochastic Processes and their Applications* 62.1, pp. 139–168. DOI: 10.1016/0304-4149(95)00090-9 (cit. on p. 170).

Dahlhaus, R (1997). "Fitting time series models to nonstationary processes". In: *The Annals of Statistics* 25.1, pp. 1–37. DOI: 10.1214/aos/1034276620 (cit. on pp. 167–170, 225, 228).

Dahlhaus, R. (2000). "A Likelihood Approximation for Locally Stationary Processes". In: *The Annals of Statistics* 28.6, pp. 1762–1794 (cit. on p. 226).

Danaher, P., P. Wang, and D.M. Witten (2013). "The joint graphical lasso for inverse covariance estimation across multiple classes". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 76.2, pp. 373–397. DOI: 10.1111/rssb.12033 (cit. on pp. 81, 84, 87, 88, 92, 127).

Daubechies, I. (1990). "The wavelet transform, time-frequency localization and signal analysis". In: *IEEE Transactions on Information Theory* 36.5, pp. 961–1005. DOI: 10.1109/18.57199 (cit. on p. 174).

— (1992). *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics. DOI: 10.1137/1.9781611970104 (cit. on p. 174).

Davidson, K.R. and S.J. Szarek (2001). "Local operator theory, random matrices, and Banach spaces". In: *Handbook of Banach Spaces*. Elsevier, pp. 317–366. DOI: 10.1016/s1874-5849(01)80010-3 (cit. on p. 153).

Deng, W. and W. Yin (2012). *On the Global Linear Convergence of Alternating Direction Methods*. Tech. rep., pp. 1–24 (cit. on p. 52).

Donoho, D. L. and I.M. Johnstone (1995). "Adapting to unknown smoothness via wavelet shrinkage". In: *Journal of the American Statistical Association* 90.432, pp. 1200–1224. DOI: 10.1080/01621459.1995.10476626 (cit. on pp. 43, 191, 225).

Donoho, D.L. (1995). "De-noising by Soft-thresholding". In: *IEEE Transactions on Information Theory* 41.3, pp. 613–627. ISSN: 0018-9448. DOI: 10.1109/18.382009 (cit. on pp. 43, 191).

— (2006). "Compressed sensing". In: *IEEE Transactions on Information Theory* 52.4, pp. 1289–1306. DOI: 10.1109/TIT.2006.871582 (cit. on p. 69).

Donoho, D.L. and I.M. Johnstone (1994). "Ideal Spatial Adaptation by Wavelet Shrinkage". In: *Biometrika* 81.3, p. 425. DOI: 10.2307/2337118 (cit. on pp. 43, 191, 192).

— (1998). "Minimax estimation via wavelet shrinkage". In: *The Annals of Statistics* 26.3, pp. 879–921. DOI: 10.1214/aos/1024691081 (cit. on p. 191).

Drton, M. and M.D. Perlman (2004). "Model selection for Gaussian concentration graphs". In: *Biometrika* 91.3, pp. 591–602. DOI: 10.1093/biomet/91.3.591 (cit. on p. 60).

Eckley, I.A., G.P. Nason, and R.L. Treloar (2010). "Locally stationary wavelet fields with application to the modelling and analysis of image texture". In: *Journal of the Royal Statistical Society. Series C: Applied Statistics* 59.4, pp. 595–616. DOI: 10.1111/j.1467-9876.2009.00721.x (cit. on pp. 178, 207, 210–212).

Eckstein, J. and D.P. Bertsekas (1992). "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators". In: *Mathematical Programming*. DOI: 10.1007/bf01581204 (cit. on p. 95).

Evangelou, M. and N.M. Adams (2016). "Predictability of NetFlow data". In: *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*. DOI: 10.1109/isi.2016.7745445 (cit. on pp. 109, 110).

Fogla, P. et al. (2006). "Polymorphic Blending Attacks". In: (cit. on p. 24).

Forslund, K., I. Pekkari, and E.L. Sonnhammer (2011). "Domain architecture conservation in orthologs". In: *BMC Bioinformatics* 12.1, p. 326. DOI: 10.1186/1471-2105-12-326 (cit. on p. 105).

Foygel, R. and M. Drton (2010). "Extended Bayesian Information Criteria for Gaussian Graphical Models". In: *Advances in Neural Information Processing Systems* (cit. on p. 60).

Friedberg, I. et al. (2015). "Combating advanced persistent threats: From network event correlation to incident detection". In: *Computers & Security* 48, pp. 35–57. DOI: 10.1016/j.cose.2014.09.006 (cit. on p. 24).

Friedman, J., T. Hastie, H. Hoefling, et al. (2007). "Pathwise coordinate optimization". In: *Annals of Applied Statistics*. DOI: 10.1214/07-aoas131 (cit. on p. 94).

Friedman, J., T. Hastie, and R. Tibshirani (2008). "Sparse inverse covariance estimation with the graphical lasso". In: *Biostatistics* 9.3, pp. 432–41. DOI: 10.1093/biostatistics/kxm045 (cit. on pp. 43, 61, 79, 81, 89, 126, 231).

Fryzlewicz, P. and G.P. Nason (2006). "Haar-Fisz estimation of evolutionary wavelet spectra". In: *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 68.4, pp. 611–634. DOI: 10.1111/j.1467-9868.2006.00558.x (cit. on pp. 184, 186, 189, 196–199, 201, 202, 207, 211).

Geer, S. van de and P. Bühlmann (2009). "On the conditions used to prove oracle results for the Lasso". In: *Electronic Journal of Statistics* 3, pp. 1360–1392. DOI: 10.1214/09-EJS506. eprint: 0910.0722 (cit. on pp. 70, 73).

Gibberd, A.J., M. Evangelou, and J.D.B. Nelson (2016). "The Time-Varying Dependency Patterns of NetFlow Statistics". In: *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*. DOI: 10.1109/icdmw.2016.0048 (cit. on pp. 80, 108).

Gibberd, A.J. and J.D.B. Nelson (2014a). "High dimensional changepoint detection with a dynamic graphical lasso". In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2684–2688. DOI: 10.1109/ICASSP.2014.6854087 (cit. on pp. 80, 83, 84, 87, 88).

— (2014b). "High dimensional changepoint detection with a dynamic graphical lasso". In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. DOI: 10.1109/icassp.2014.6854087 (cit. on p. 81).

— (2015a). "Estimating Multiresolution Dependency Graphs within the Stationary Wavelet Framework". In: *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. DOI: 10.1109/globalsip.2015.7418255 (cit. on pp. 238, 246).

— (2015b). "Regularized Estimation of Piecewise Constant Gaussian Graphical Models: The Group-Fused Graphical Lasso". In: *In review* (cit. on p. 89).

— (2016). "Estimating Dynamic Graphical Models from Multivariate Time-Series Data: Recent Methods and Results". In: *Lecture Notes in Computer Science*. Springer International Publishing, pp. 111–128. DOI: 10.1007/978-3-319-44412-3_8 (cit. on p. 80).

— (2017). "Regularized Estimation of Piecewise Constant Gaussian Graphical Models: The Group-Fused Graphical Lasso". In: *Journal of Computational and Graphical Statistics*. DOI: 10.1080/10618600.2017.1302340 (cit. on pp. 80, 85, 88, 119, 221).

Glowinski, R. and P. Le Tallec (1989). *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*. Society for Industrial and Applied Mathematics. DOI: 10.1137/1.9781611970838 (cit. on pp. 52, 95).

Gockenbach, M.S. (2016). *Linear Inverse Problems and Tikhonov Regularization* (cit. on p. 38).

Goldberger, A.L., L.A.N. Amaral, and L. Glass (2000). "Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals". In: *Circulation*. DOI: 10.1161/01.cir.101.23.e215 (cit. on p. 237).

Hallac, D. et al. (2017). "Network Inference via the Time-Varying Graphical Lasso". In: *Arxiv preprint arXiv:1703.01958* (cit. on p. 119).

Harchaoui, Z. and C. Lévy-Leduc (2010). "Multiple Change-Point Estimation With a Total Variation Penalty". In: *Journal of the American Statistical Association* 105.492, pp. 1480–1493. DOI: 10.1198/jasa.2010.tm09181 (cit. on pp. 125, 127, 128, 132, 136, 138, 141, 148, 160).

Haslbeck, J.M.B. and L.J. Waldorp (2016). "mgm: Structure Estimation for time-varying Mixed Graphical Models in high-dimensional Data". In: *arXiv:1510.06871*. arXiv: arXiv:1510.06871v2 (cit. on p. 87).

Hoerl, A.E. and R.W. Kennard (1970). "Ridge regression: Biased estimation for nonorthogonal problems". In: *Technometrics* 12.1, pp. 55–67 (cit. on p. 36).

Hunyadi, B. (2014). "Learning from structured EEG and fMRI data supporting the diagnosis of epilepsy". PhD thesis (cit. on p. 236).

Iglesias, F. and T. Zseby (2014). "Analysis of network traffic features for anomaly detection". In: *Machine Learning* 101.1-3, pp. 59–84. DOI: 10.1007/s10994-014-5473-9 (cit. on p. 24).

Iordache, M.S., J.M. Bioucas-Dias, and A. Plaza (2012). "Total variation spatial regularization for sparse hyperspectral unmixing". In: *IEEE Transactions on Geoscience and Remote Sensing* 50.11, pp. 4484–4502. DOI: 10.1109/TGRS.2012.2191590 (cit. on p. 217).

Jordan, M.I. (2004). "Graphical Models". In: *Statistical Science* 19.1, pp. 140–155. DOI: 10.1214/088342304000000026 (cit. on p. 54).

Killick, R., I.A. Eckley, and P. Jonathan (2013). "A wavelet-based approach for detecting changes in second order structure within nonstationary time series". In: *Electronic Journal of Statistics* 7.1, pp. 1167–1183. DOI: 10.1214/13-EJS799 (cit. on p. 196).

Kingsbury, N.G. (2001). "Complex wavelets for shift invariant analysis and filtering of signals". In: *Journal of Applied and Computational Harmonic Analysis* 10.3, pp. 234–253. DOI: 10.1006/acha.2000.0343 (cit. on p. 210).

Kolar, M. and E.P. Xing (2011). "On Time Varying Undirected Graphs". In: *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)* 15, pp. 407–415 (cit. on p. 82).

— (2012). "Estimating networks with jumps". In: *Electronic Journal of Statistics* 6, pp. 2069–2106. DOI: 10.1214/12-ejs739 (cit. on pp. 86–88, 125, 127, 128, 132, 133, 135, 136, 138, 139, 148, 160).

Koller, D. et al. (2007). "Graphical Models in a Nutshell". In: *Introduction to Statistical Relational Learning*. Ed. by L. Getoor and B. Taskar. MIT Press (cit. on p. 54).

Lafferty, J., H. Liu, and L. Wasserman (2012). "Sparse Nonparametric Graphical Models". In: *Statistical Science* 27.4, pp. 519–537. DOI: 10.1214/12-sts391 (cit. on pp. 59, 61, 87, 110, 251).

Lam, C. and J. Fan (2009). "Sparsistency and rates of convergence in large covariance matrix estimation". In: *The Annals of Statistics* 37.6B, pp. 4254–4278. DOI: 10.1214/09-aos720 (cit. on pp. 71, 72, 126).

Lauritzen, S.L. (1996). *Graphical Models*. Oxford (cit. on p. 57).

Lèbre, S. et al. (2010). "Statistical inference of the time-varying structure of gene-regulation networks". In: *BMC systems biology* 4, p. 130. DOI: 10.1186/1752-0509-4-130 (cit. on p. 105).

Lee, J.D. and T.J. Hastie (2015). "Learning the Structure of Mixed Graphical Models". In: *Journal of Computational and Graphical Statistics* 24.1, pp. 230–253. DOI: 10.1080/10618600.2014.900500 (cit. on p. 87).

Lin, T., S. Ma, and S. Zhang (2014). "On the Global Linear Convergence of the ADMM with Multi-Block Variables". In: *IEEE Transactions on Signal Processing* 62.7. DOI: 10.1137/140971178 (cit. on pp. 52, 217).

Lindquist, M.A. (2008). "The Statistical Analysis of fMRI Data". In: *Statistical Science* 23.4, pp. 439–464. DOI: 10.1214/09-STS282 (cit. on p. 21).

Liu, H., K. Roeder, and L. Wasserman (2010). "Stability Approach to Regularization Selection (StARS) for High Dimensional Graphical Models". In: *Advances in neural information processing systems* 24.2, pp. 1432–1440 (cit. on pp. 63, 93, 97).

Liu, J., L. Yuan, and J. Ye (2010). "An efficient algorithm for a class of fused lasso problems". In: *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, p. 323. DOI: 10.1145/1835804.1835847 (cit. on p. 94).

Mallat, S.G. (1989). "A theory for multiresolution signal decomposition: the wavelet representation". In: *IEEE Transactions on Pattern Analysis and*

Machine Intelligence 11.7, pp. 674–693. DOI: 10.1109/34.192463 (cit. on pp. 173, 174).

Meher, S.K., B.U. Shankar, and A. Ghosh (2007). "Wavelet-feature-based classifiers for multispectral remote-sensing images". In: *IEEE Transactions on Geoscience and Remote Sensing* 45.6, pp. 1881–1886. DOI: 10.1109/TGRS.2007.895836 (cit. on p. 224).

Meinshausen, N. (2008). "A note on the Lasso for Gaussian graphical model selection". In: *Statistics & Probability Letters* 78, pp. 880–884. DOI: 10.1016/j.spl.2007.09.014 (cit. on pp. 38, 74).

Meinshausen, N. and P. Bühlmann (2006). "High-dimensional graphs and variable selection with the lasso". In: *The Annals of Statistics*. DOI: 10.1214/009053606000000281 (cit. on pp. 61, 70, 126).

Monti, R.P. et al. (2014). "Estimating time-varying brain connectivity networks from functional MRI time series." In: *NeuroImage*. DOI: 10.1016/j.neuroimage.2014.07.033 (cit. on pp. 22, 83, 84, 87, 88, 92, 215, 221).

Müller, M. et al. (2005). "Detection and characterization of changes of the correlation structure in multivariate time series". In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 71, pp. 1–16. DOI: 10.1103/physreve.71.046116 (cit. on p. 236).

Nason, G. (2013). "A test for second-order stationarity and approximate confidence intervals for localized autocovariances for locally stationary time series". In: *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 75.5, pp. 879–904. ISSN: 13697412. DOI: 10.1111/rssb.12015 (cit. on pp. 207, 212).

Nason, G.P., R. von Sachs, and G. Kroisandt (2000). "Wavelet processes and adaptive estimation of the evolutionary wavelet spectrum". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 62.2, pp. 271–292. DOI: 10.1111/1467-9868.00231 (cit. on pp. 169, 178, 179, 181, 182, 184–186, 188, 189, 194, 196, 207, 210–212, 225, 228).

Negahban, S.N. et al. (2012). "A Unified Framework for High-Dimensional Analysis of M-Estimators with Decomposable Regularizers". In: *Statistical Science* 27.4, pp. 538–557. DOI: 10.1214/12-sts400 (cit. on pp. 38, 64, 65, 68, 69, 71, 125, 127, 146, 148, 151).

Nelson, J.D.B. and A.J. Gibberd (2016). "Introducing the locally stationary dual-tree complex wavelet model". In: *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3583–3587. DOI: 10.1109/ICIP.2016.7533027 (cit. on p. 210).

Nesterov, Y. (2005). "Smooth minimization of non-smooth functions". In: *Mathematical programming* 152, pp. 127–152. DOI: 10.1007/s10107-004-0552-5 (cit. on p. 49).

— (2007). "Gradient methods for minimizing composite objective function". In: *ECORE* 76.2007076, p. 2007. DOI: 10.1007/s10107-012-0629-5 (cit. on pp. 48, 50).

Neumann, H. (1996). "Spectral Density Estimation Via Nonlinear Wavelet Methods for Stationary Non-Gaussian Time Series". In: *Journal of Time Series Analysis* 17.6, pp. 601–633 (cit. on p. 193).

Neumann, M.H. and R. von Sachs (1995). "Wavelet Thresholding: Beyond the Gaussian I.I.D. Situation". In: *Wavelets and Statistics', Lecture Notes in Statistics* 103 (cit. on pp. 191, 193, 194).

Nunes, M.A., S.L. Taylor, and I.A. Eckley (2014). "A Multiscale Test of Spatial Stationarity for Textured Images in R". In: *The R Journal* (cit. on pp. 178, 207).

Nuttall, A.H. and G.C. Carter (1982). "Spectral estimation using combined time and lag weighting". In: *Proceedings of the IEEE* 70.9, pp. 1115–1125. ISSN: 0018-9219. DOI: 10.1109/PROC.1982.12435 (cit. on p. 170).

Ombao, H., R. von Sachs, and W. Guo (2005). "SLEX Analysis of Multivariate Nonstationary Time Series". In: *Journal of the American Statistical Association* 100.470, pp. 519–531. DOI: 10.1198/016214504000001448 (cit. on p. 226).

Ombao, H. and S. Van Bellegem (2008). "Evolutionary coherence of nonstationary signals". In: *IEEE Transactions on Signal Processing* 56.6, pp. 2259–2266. DOI: 10.1109/TSP.2007.914341 (cit. on p. 226).

Parikh, N. and S. Boyd (2013). "Proximal algorithms". In: *Foundations and Trends in optimization* 1.3, pp. 123–231. DOI: 10.1561/2400000003 (cit. on p. 49).

Park, T., I.A. Eckley, and H.C. Ombao (2014). "Estimating Time-Evolving Partial Coherence Between Signals via Multivariate Locally Stationary

Wavelet Processes". In: *IEEE Transactions on Signal Processing*, pp. 5240–5250. DOI: 10.1109/tsp.2014.2343937 (cit. on pp. 178, 187, 212, 226–229).

Patcha, A. and J. Park (2007). "An overview of anomaly detection techniques: Existing solutions and latest technological trends". In: *Computer Networks* 51.12, pp. 3448–3470. DOI: 10.1016/j.comnet.2007.02.001 (cit. on pp. 24, 108).

Priestly, M.B. (1981). *Spectral Analysis and Time Series* (cit. on pp. 164, 166, 167, 225).

Raskutti, G., M.J. Wainwright, and B. Yu (2010). "Restricted eigenvalue properties for correlated Gaussian designs". In: *The Journal of Machine Learning Research* 11, pp. 2241–2259 (cit. on p. 70).

— (2011). "Minimax rates of estimation for high-dimensional linear regression over lq-Balls". In: *IEEE Transactions on Information Theory* 57.10, pp. 6976–6994. ISSN: 00189448. DOI: 10.1109/TIT.2011.2165799 (cit. on p. 70).

Ravikumar, P., M.J. Wainwright, and J.D. Lafferty (2010). "High-dimensional Ising model selection using 1-regularized logistic regression". In: *The Annals of Statistics* 38.3, pp. 1287–1319. DOI: 10.1214/09-aos691 (cit. on pp. 59, 71, 72, 126, 146).

Ravikumar, P., M.J. Wainwright, G. Raskutti, et al. (2011). "High-dimensional covariance estimation by minimizing l1-penalized log-determinant divergence". In: *Electronic Journal of Statistics* 5.January 2010, pp. 935–980. DOI: 10.1214/11-ejs631 (cit. on pp. 71, 72, 74, 75, 103, 126, 147, 154, 155).

Ringberg, H. et al. (2007). "Sensitivity of PCA for traffic anomaly detection". In: *Proceedings of the 2007 ACM SIGMETRICS international conference on Measurement and modeling of computer systems - SIGMETRICS '07*. ACM Press. DOI: 10.1145/1254882.1254895 (cit. on p. 24).

Rockafellar, R.T. (1970). *Convex Analysis*. Princeton Math. Series (cit. on p. 49).

Rothman, A.J. et al. (2008). "Sparse permutation invariant covariance estimation". In: *Electronic Journal of Statistics* 2, pp. 494–515. DOI: 10.1214/08-ejs176 (cit. on pp. 71, 72).

Roy, S., Y. Atchadé, and G. Michailidis (2016). "Change point estimation in high dimensional Markov random-field models". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. DOI: 10.1111/rssb.12205 (cit. on p. 127).

Sachs, R. von, G.P. Nason, and G. Kroisandt (1997). *Adaptive estimation of the evolutionary wavelet spectrum*. Tech. rep. (cit. on p. 188).

Sachs, R. von and K. Schneider (1996). "Wavelet Smoothing of Evolutionary Spectra by Nonlinear Thresholding". In: *Applied and Computational Harmonic Analysis* 3.3, pp. 268–282. DOI: 10.1006/acha.1996.0021 (cit. on pp. 184, 189).

Saegusa, T. and A. Shojaie (2016). "Joint estimation of precision matrices in heterogeneous populations". In: *Electronic Journal of Statistics* 10.1, pp. 1341–1392. ISSN: 19357524. DOI: 10.1214/16-EJS1137 (cit. on pp. 71, 127, 146, 147).

Sanderson, J., P. Fryzlewicz, and M.W. Jones (2010). "Estimating linear dependence between nonstationary time series using the locally stationary wavelet model". In: *Biometrika* 97, pp. 435–446. DOI: 10.1093/biomet/asq007 (cit. on pp. 178, 226–229).

Scherrer, A. et al. (2007). "Non-gaussian and long memory statistical characterizations for internet traffic with anomalies". In: *IEEE Transactions on Dependable and Secure Computing* 4.1, pp. 56–70. DOI: 10.1109/tdsc.2007.12 (cit. on pp. 108, 110).

Schindler, K. et al. (2007). "Assessing seizure dynamics by analysing the correlation structure of multichannel intracranial EEG". In: *Brain* 130, pp. 65–77. DOI: 10.1093/brain/awl304 (cit. on p. 236).

Schwarz, G. (1978). "Estimating the dimension of a model". In: *The Annals of Statistics* 6.2, pp. 461–464. DOI: 10.1214/aos/1176344136 (cit. on p. 39).

Selesnick, I.W., R.G. Baraniuk, and N.G. Kingsbury (2005). "The Dual-Tree Complex Wavelet Transform". In: *IEEE Signal Processing Magazine* 22.6, pp. 123–151. DOI: 10.1109/msp.2005.1550194 (cit. on p. 210).

Soh, D. and S. Tatikonda (2014). "Testing Unfaithful Gaussian Graphical Models". In: *Advances in Neural Information Processing Systems* (cit. on p. 58).

Stevens, K.N. (2013). "New Methods for Spectral Estimation with Confidence Intervals for Locally Stationary Wavelet Processes". PhD thesis. University of Bristol (cit. on pp. 187, 252).

Swinburne, R. (1997). *Simplicity as Evidence of Truth*. Milwaukee: Marquette University Press (cit. on pp. 31, 32).

Taylor, S.L., I.A. Eckley, and M.A. Nunes (2014). "A Test of Stationarity for Textured Images". In: *Technometrics* 56.3, pp. 291–301. DOI: 10.1080/00401706.2013.823890 (cit. on pp. 178, 207).

Tibshirani, R. (1996). "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* (cit. on p. 42).

Tibshirani, R.J. (2014). "Adaptive piecewise polynomial estimation via trend filtering". In: *The Annals of Statistics* 42.1, pp. 285–323. DOI: 10.1214/13-AOS1189 (cit. on pp. 43, 199, 201).

Tibshirani, R.J. and J. Taylor (2011). "The solution path of the generalized lasso". In: *The Annals of Statistics* 39.3, pp. 1335–1371. DOI: 10.1214/11-AOS878 (cit. on pp. 42, 200).

Tibshirani, R. et al. (2005). "Sparsity and smoothness via the fused lasso". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.1, pp. 91–108 (cit. on pp. 42, 199).

Tseng, P. and S. Yun (2009). "A coordinate gradient descent method for nonsmooth separable minimization". In: *Mathematical Programming*. DOI: 10.1007/s10107-007-0170-0 (cit. on p. 89).

Van Bellegem, S. and R. Von Sachs (2008). "Locally adaptive estimation of evolutionary wavelet spectra". In: *The Annals of Statistics* 36.4, pp. 1879–1924. DOI: 10.1214/07-AOS524 (cit. on p. 196).

Van Den Berg, E. et al. (2008). *Group Sparsity via Linear-Time Projection*. Tech. rep. (cit. on pp. 123, 124).

Wainwright, M.J. (2009). "Sharp Thresholds for High-Dimensional and Noisy Sparsity Recovery Using l1-Constrained Quadratic Programming (Lasso)". In: *IEEE Transactions on Information Theory* 55.5, pp. 2183–2202. DOI: 10.1109/tit.2009.2016018 (cit. on pp. 73, 153, 154).

Walden, A.T. (2000). "A unified view of multitaper multivariate spectral estimation". In: *Biometrika* 87.4, pp. 767–788. DOI: 10.1093/biomet/87.4.767 (cit. on p. 170).

Wang, H. (2012). "Bayesian Graphical Lasso Models and Efficient Posterior Computation". In: *Bayesian Analysis* 7.4, pp. 867–886. DOI: 10.1214/12-ba729 (cit. on p. 61).

Wang, X. et al. (2015). "Solving Multiple-Block Separable Convex Minimization Problems Using Two-Block Alternating Direction Method of Multipliers". In: *Pacific Journal of Optimization* 11, pp. 645–667. arXiv: 1308.5294 (cit. on p. 52).

Whittle, P. (1953). "Estimation and information in stationary time series". In: *Arkiv for Matematik* 2.5, pp. 423–434. DOI: 10.1007/BF02590998 (cit. on p. 169).

Wright, S. and R.D. Nowak (2009). "Sparse reconstruction by separable approximation". In: *IEEE Transactions on Signal Processing* 57.7, pp. 2479–2493. DOI: 10.1109/tsp.2009.2016892 (cit. on p. 50).

Xu, P., H. Xu, and P.J. Ramadge (2013). "Detecting stimulus driven changes in functional brain connectivity". In: (cit. on p. 22).

Yang, E. et al. (2013). "On poisson graphical models". In: *Advances in Neural Information Processing Systems (NIPS)* (cit. on pp. 59, 110, 251).

Yang, S. et al. (2015). "Fused Multiple Graphical Lasso". In: *SIAM Journal on Optimization* 25.2, pp. 916–943. DOI: 10.1137/130936397 (cit. on pp. 84, 87–89).

Yuan, M. and Y. Lin (2006). "Model selection and estimation in regression with grouped variables". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68.1, pp. 49–67. DOI: 10.1111/j.1467-9868.2005.00532.x (cit. on pp. 42, 94, 117, 118).

Yuan, X. (2011). "Alternating Direction Method for Covariance Selection Models". In: *Journal of Scientific Computing* 51.2, pp. 261–273. DOI: 10.1007/s10915-011-9507-1 (cit. on pp. 61, 91).

Zeki, S. et al. (2014). "The experience of mathematical beauty and its neural correlates." In: *Frontiers in human neuroscience* 8, p. 68. DOI: 10.3389/fnhum.2014.00068 (cit. on p. 21).

Zhang, B., J. Geng, and L. Lai (2015). "Multiple change-points estimation in linear regression models via sparse group Lasso". In: *IEEE Transactions on Signal Processing* 63.9, pp. 2209–2224. DOI: 10.1109/TSP.2015.2411220 (cit. on p. 127).

Zhang, C. and J. Huang (2008). "The sparsity and bias of the Lasso selection in high-dimensional linear regression". In: *The Annals of Statistics* 36.4, pp. 1567–1594. ISSN: 0090-5364. DOI: 10.1214/07-AOS520 (cit. on p. 70).

Zhao, P. and B. Yu (2006). "On model selection consistency of Lasso". In: *The Journal of Machine Learning Research* 7, pp. 2541–2563 (cit. on p. 70).

Zhao, T. et al. (2012). "The huge Package for High-dimensional Undirected Graph Estimation in R." In: *Journal of machine learning research* 13, pp. 1059–1062 (cit. on p. 62).

Zhou, S., J. Lafferty, and L. Wasserman (2010). "Time varying undirected graphs". In: *Machine Learning* 80.2-3, pp. 295–319. DOI: 10.1007/s10994-010-5180-0 (cit. on pp. 61, 79, 82, 88, 99, 112, 234, 239).

Zou, H. and T. Hastie (2005). "Regularization and variable selection via the elastic net". In: *Journal of the Royal Statistical Society: Series B* 67.2, pp. 301–320. DOI: 10.1111/j.1467-9868.2005.00503.x (cit. on pp. 42, 46).

Zou, H., T. Hastie, and R. Tibshirani (2006). "Sparse Principal Component Analysis". In: *Journal of Computational and Graphical Statistics* 15.2, pp. 265–286. DOI: 10.1198/106186006X113430 (cit. on p. 115).