On Variable Selection in High Dimensions, Segmentation and Multiscale Time Series

Rafal Baranowski

Department of Statistics

London School of Economics and Political Sciences

This dissertation is submitted for the degree of

Doctor of Philosophy

September 2016

Dedicated to Ewa.

Declaration

I certify that the thesis I have presented for examination for the PhD degree of the London School of Economics and Political Science is solely my own work other than where I have clearly indicated that it is the work of others (in which case the extent of any work carried out jointly by me and any other person is clearly identified in it). The copyright of this thesis rests with the author. Quotation from it is permitted, provided that full acknowledgement is made. This thesis may not be reproduced without my prior written consent. I warrant that this authorisation does not, to the best of my belief, infringe the rights of any third party. I declare that my thesis consists of about 65000 words.

Rafal Baranowski

September 2016

Statement of conjoint work

I confirm that Chapter 4 was jointly co-authored with Doctor Yining Chen and Professor Piotr Fryzlewicz and I contributed 60% of this work.

I confirm that Chapter 5 was jointly co-authored with Professor Piotr Fryzlewicz and I contributed 80% of this work.

Chapters 3 and 4 have been submitted to peer-reviewed statistical journals. We plan to submit Chapter 5 for publication soon.

Rafal Baranowski

September 2016

Acknowledgements

First and foremost, I would like to thank my supervisor, Professor Piotr Fryzlewicz, for his constant guidance, enthusiasm, encouragement and extremely professional feedback on my work. I am also grateful to Doctor Yining Chen, with whom I had the luck to work on a research paper which constitutes one of the chapters in this thesis.

I wish to express my gratitude to all the staff and students in the Department of Statistics at the London School of Economics for providing me with a great working environment. I am also deeply grateful for the financial support of the London School of Economics Research Studentship without which this thesis could not have been neither undertaken nor completed.

Lastly, I would like to thank my family, especially my parents, for all their love and for showing me by example that hard work always pays off.

Abstract

In this dissertation, we study the following three statistical problems.

First, we consider a high-dimensional data framework, where the number of covariates potentially affecting the response is large relatively to the sample size. In this setting, some of the covariates are observed to exhibit an impact on the response spuriously. Addressing this issue, we rank the covariates according to their impact on the response and use certain subsampling scheme to identify the covariates which non-spuriously appear at the top of the ranking. We study the conditions under which such set is unique and show that, with high probability, it can be recovered from the data by our procedure, for rankings based on measures commonly used in statistics. We illustrate its good practical performance in an extensive comparative simulation study and on microarray data.

Second, we propose a generic approach to the problem of detecting the unknown number of features in the time series of interest, such as changes in trend or jumps in the mean, occurring at the unknown locations in time. Those locations naturally imply the decomposition of the data into segments of homogeneity, the knowledge of which is useful in e.g. estimation of the mean of the series. We provide a precise description of the type of features we are interested in and, in two important scenarios, demonstrate that our methodology enjoys appealing theoretical properties. We show that the performance of our proposal matches or surpasses the state of the art in the scenarios tested and present its applications on three real datasets: oil price log-returns, temperature anomalies data

and the UK House Price Index

Finally, we introduce a class of univariate multiscale time series models and propose an estimation procedure to fit those models from the data. We demonstrate that our proposal, with a large probability, correctly identifies important timescales, under the framework in which the largest timescale in the model diverges with the sample size. A good empirical performance of the method is illustrated in an application to high-frequency financial returns for stocks listed on New York Stock Exchange.

For all proposed methods, we provide efficient and publicly-available computer implementations.

Table of contents

List of figures 13

List of tables 14

List of algorithms 17

1 Introduction 18

2 Literature review 22

2.1 High-dimensional variable selection 22

2.1.1 Variable selection via Penalised Likelihood minimisation 25

2.1.1.1 PLH with ℓ_0-norm type penalty 26

2.1.1.2 PLH with ℓ_1-norm and ℓ_2-norm type penalties 26

2.1.1.3 PLH with other types of penalties 28

2.1.2 Variable screening methods 28

2.1.3 Subsampling in variable selection 30

2.2 Data segmentation and multiple change-point detection 31

2.2.1 Canonical change-point detection problem 33

2.2.1.1 Multivariate optimisation 33

2.2.1.2 Binary Segmentation 34

2.2.1.3 Other approaches 38

2.2.2 Regression change-point models . . . 39
2.2.2.1 Methodology of Bai and Perron (1998) . . . 40
2.2.2.2 Trend filtering . . . 41
2.2.3 Other change-point detection problems . . . 42
2.2.3.1 Change in variance and/or mean . . . 42
2.2.3.2 Nonparametric change-point detection . . . 43
2.3 Multiscale time series models . . . 44
2.3.1 Multiscale time series models of Ferreira et al. (2006) . . . 45
2.3.2 Mixed Data Sampling Regression Models . . . 45

3 Ranking-Based Variable Selection 47
3.1 Introduction . . . 47
3.2 Motivating examples . . . 52
3.3 Methodology of Ranking-Based Variable Selection . . . 54
3.3.1 Notation . . . 54
3.3.2 Definition of a k-top-ranked and the top-ranked set . . . 54
3.3.3 Top-ranked set for a class of variable rankings . . . 55
3.3.4 Ranking-Based Variable Selection . . . 58
3.3.5 The Ranking-Based Variable Selection algorithm . . . 60
3.3.6 Relations to existing methodology . . . 61
3.3.6.1 Stability selection . . . 61
3.3.6.2 The bootstrapped rankings of Hall and Miller (2009a) . . . 61
3.3.6.3 Computational complexity of the related methods . . . 62
3.4 Theoretical results . . . 63
3.5 Iterative extension of RBVS . . . 65
3.6 Simulation study . . . 66
3.6.1 Simulation methods . . . 66

3.6.2 Choice of parameters of the RBVS algorithm . . . 68
3.6.3 Simulation models . . . 69
3.6.4 Comments on the results . . . 70
3.7 Data examples . . . 78
3.7.1 Prostate cancer data set . . . 78
3.7.2 Boston housing data set . . . 80
3.8 High-dimensional simulation study . . . 82
3.9 Computational aspects . . . 98
3.9.1 Details of the implementation of the RBVS algorithm . . . 98
3.9.2 Algorithmic differences between RBVS and StabSel . . . 99
3.9.3 Simulation code . . . 101
3.10 Proofs . . . 104
3.10.1 Proof of Proposition 3.3.1 . . . 104
3.10.2 Proof of Theorem 3.4.1 and discussion of some of its aspects. . . . 106

4 Narrowest-Over-Threshold change-point detection 112
4.1 Introduction . . . 112
4.2 Methodology . . . 119
4.2.1 Setup . . . 119
4.2.2 Main idea . . . 121
4.2.3 Log-likelihood ratios and contrast functions . . . 123
4.2.3.1 Scenario (S1) . . . 123
4.2.3.2 Scenario (S2) . . . 125
4.2.3.3 Scenario (S3) . . . 126
4.2.3.4 Scenario (S4) . . . 127
4.2.4 The NOT algorithm . . . 128
4.2.5 Theoretical properties of NOT . . . 129

4.3 Computational aspects . . . 131
4.3.1 Computing contrast functions in linear time . . . 131
4.3.2 The NOT solution path algorithm . . . 132
4.3.3 An illustrative example . . . 135
4.3.4 Parameter choice . . . 136
4.3.4.1 Choice of M . . . 136
4.3.4.2 Choice of the threshold ζ_T . . . 137
4.3.5 Computational complexity of the NOT and NOT solution path algorithms . . . 138
4.4 Simulation study . . . 140
4.4.1 Simulation methods . . . 140
4.4.2 Simulation models . . . 143
4.4.3 Results and discussion . . . 150
4.5 Real data analysis . . . 153
4.5.1 OPEC Reference Basket oil price . . . 153
4.5.2 Temperature anomalies . . . 156
4.5.3 UK House Price Index . . . 157
4.6 Proofs . . . 160
4.6.1 Some useful lemmas . . . 160
4.6.1.1 The piecewise constant case . . . 160
4.6.1.2 The piecewise linear continuous case . . . 163
4.6.2 Proof of Theorem 4.2.1 . . . 167
4.6.3 Proof of Theorem 4.2.2 . . . 175

5 Multiscale autoregression 180

5.1 Introduction . . . 180
5.2 Methodology and theory . . . 184

5.2.1 Notation . . . 184
5.2.2 Large deviations for the OLS estimator in AR(p) . . . 185
5.2.3 Estimation of the timescales with NOT . . . 186
5.2.4 AMAR algorithm and its theoretical properties. . . . 188
5.3 Practicalities and simulated examples . . . 191
5.3.1 Parameter choice and other practicalities . . . 191
5.3.1.1 Choice of the threshold ζ_T . . . 191
5.3.1.2 Choice of p . . . 192
5.3.1.3 Choice of M . . . 192
5.3.1.4 Computational complexity. . . . 192
5.3.2 Simulation study . . . 193
5.4 Application to high-frequency data from NYSE TAQ database . . . 196
5.4.1 Data preprocessing . . . 197
5.4.2 Rolling window analysis . . . 198
5.4.3 Results and discussion . . . 200
5.4.4 Simulated data with real volatility . . . 205
5.5 Large deviations for LSE estimators in stationary AR(p) models . . . 206
5.5.1 Some properties of the **B** matrix . . . 207
5.5.2 Two useful lemmas . . . 209
5.5.3 Proof of Theorem 5.2.1 . . . 210
5.6 Proof of Theorem 5.2.2 . . . 216

6 Conclusions **223**

References **227**

List of figures

3.1 Probabilities corresponding to the most frequently occurring subsets of covariates in Example 3.2.1 and 3.2.2. 53

3.2 Classification rates in the application to the prostate cancer data set. . . 79

3.3 Probabilities corresponding to the most frequently occurring subsets of covariates in the application to the Boston housing data set. 80

4.1 Motivating example 118

4.2 Examples of $\boldsymbol{\phi}_{s,e}^{b}$ and $\boldsymbol{\psi}_{s,e}^{b}$ 124

4.3 Illustrative example 136

4.4 Segmentation trees obtained with Algorithm 4.7 137

4.5 Execution times of Algorithm 4.7 140

4.6 Examples of data generated from simulation models studied in Section 4.4.2.144

4.7 Change-point analysis on the daily OPEC Reference Basket oil price . . . 154

4.8 Change-point analysis for the GISSTEMP data set 156

4.9 Change-point analysis for the monthly percentage changes in the UK House Price Index 158

5.1 Example of high-frequency trades data for Apple Inc. 183

5.2 Example of time series generate from AMAR model 193

List of tables

3.1 Computational complexity of Algorithm 3.3 and its competitors 63

3.2 Simulation results for **Model (A)** 73

3.3 Simulation results for **Model (B)** 74

3.4 Simulation results for **Model (C)** 75

3.5 Simulation results for **Model (D)** 76

3.6 Simulation results for **Model (E)** 77

3.7 Prediction errors in the Boston housing data example 82

3.8 Simulation results in the high-dimensional example for $m = 50$, $s = 5$ and $K = 0$ 84

3.9 Simulation results in the high-dimensional example for $m = 100$, $s = 5$ and $K = 0$ 85

3.10 Simulation results in the high-dimensional example for $m = \frac{n}{2}$, $s = 5$ and $K = 0$ 86

3.11 Simulation results in the high-dimensional example for $m = 50$, $s = 5$ and $K = 5$ 87

3.12 Simulation results in the high-dimensional example for $m = 100$, $s = 5$ and $K = 5$ 88

3.13 Simulation results in the high-dimensional example for $m = \frac{n}{2}$, $s = 5$ and $K = 5$ 89

3.14 Simulation results in the high-dimensional example for $m = 50$, $s = 10$ and $K = 0$ . . . 90
3.15 Simulation results in the high-dimensional example for $m = 100$, $s = 10$ and $K = 0$ . . . 91
3.16 Simulation results in the high-dimensional example for $m = \frac{n}{2}$, $s = 10$ and $K = 0$ . . . 92
3.17 Simulation results in the high-dimensional example for $m = 50$, $s = 10$ and $K = 5$ . . . 93
3.18 Simulation results in the high-dimensional example for $m = 100$, $s = 10$ and $K = 5$ . . . 94
3.19 Simulation results in the high-dimensional example for $m = \frac{n}{2}$, $s = 10$ and $K = 5$ . . . 95
3.20 Computation times in the high-dimensional example for $s = 5$ and $K = 0$ 96
3.21 Informal comparison of the RBVS and StabSel algorithms . . . 100

4.1 Intervals considered in Figure 4.3a . . . 135
4.2 Simularation results for $\varepsilon_t \sim \mathcal{N}(0, 1)$ . . . 146
4.3 Simularation results for $\varepsilon_t \sim \mathcal{N}(0, 2)$ . . . 147
4.4 Simularation results for $\varepsilon_t \sim \text{Laplace}\left(0, (\sqrt{2})^{-1}\right)$ . . . 148
4.5 Simularation results for $\varepsilon_t \sim (3/5)^{1/2} t_5$ . . . 149
4.6 Change-points dected in the log-returns of the daily oil price series . . . 154

5.1 Simulation results for the data following (5.1) with parameters given in Section 5.3.2 . . . 195
5.2 Ticker symbols and the industries for the companies analysed in Section 5.4. 197
5.3 Out-of sample performance of the forecasts obtained with AMAR for 5-minute returns with $p = 480$ . . . 201

5.4 Out-of sample performance of the forecasts obtained with AMAR for 5-minute returns with $p = 960$ 202

5.5 Out-of sample performance of the forecasts obtained with AMAR for 10-minute returns with $p = 240$ 203

5.6 Out-of sample performance of the forecasts obtained with AMAR for 10-minute returns with $p = 480$ 204

5.7 Simulation results for the data following (5.20) with parameters given in Section 5.4.4 206

List of Algorithms

2.1 Binary Segmentation 35

2.2 Wild Binary Segmentation 37

3.3 Ranking-Based Variable Selection 60

3.4 Iterative Ranking-Based Variable Selection 66

3.5 Top-ranked sets 99

4.6 Narrowest-Over-Threshold algorithm 128

4.7 NOT solution path 133

5.8 NOT algorithm for estimation of the time-scales in AMAR models 187

5.9 AMAR algorithm 188

5.10 AMAR train algorithm 199

Chapter 1

Introduction

Many questions that arise in modern statistics are inspired by high-dimensional data sets, that are nowadays ubiquitous in fields such as genomics, neuroscience, high-frequency finance or economics, to name but a few. For example, a substantial progress has been made over past 20 years in the high-dimensional regression which is now an essential tool in genomics (Bühlmann et al., 2014).

The core chapters of this thesis propose methodologies to tackle three statistical problems: variable selection in high-dimensional data, change-point detection, segmentation and nonparametric function estimation and multiscale modelling of univariate time series. In Chapter 2, we review the statical literature relevant to these problems. Each of the subsequent three chapters begin with an introductory section, where we give further motivations for our work. The remainder is structured as follows.

Chapter 3. Ranking-Based Variable Selection for high-dimensional data

In this chapter, we propose Ranking-Based Variable Selection (RBVS), a technique aiming to identify covariates affecting the response, being the variable of interest, in high-dimensional data. The RBVS algorithm uses certain subsampling scheme to identify the set of covariates which non-spuriously appears at the top of a chosen variable ranking. We study the conditions under which such set is unique

and show that it can be successfully recovered from the data by our procedure. Unlike the majority of the existing high-dimensional variable selection techniques, RBVS does not depend on any thresholding or regularity parameters. Moreover, RBVS does not require any model restrictions on the relationship between the response and covariates, it is therefore widely applicable, both in a parametric and non-parametric context. We illustrate its good practical performance in an extensive comparative simulation study and on real data. The RBVS algorithm is implemented in the publicly available R packages **rbvs** (Baranowski et al., 2015) and **rbvsGPU** Baranowski (2016).

Chapter 4. Narrowest-Over-Threshold detection of multiple change-points and change-point-like features

In this chapter, we propose a new, generic and flexible methodology for nonparametric function estimation, in which we first estimate the number and locations of any features that may be present in the function, and then estimate the function parametrically between each pair of neighbouring detected features. Examples of features handled by our methodology include change-points in the piecewise-constant signal model, kinks in the piecewise-linear signal model, and other similar irregularities, which we also refer to as generalised change-points. Our methodology works with only minor modifications across a range of generalised change-point scenarios, and we achieve such a high degree of generality by proposing and using a new multiple generalised change-point detection device, termed Narrowest-Over-Threshold (NOT). The key ingredient of NOT is its focus on the smallest local sections of the data on which the existence of a feature is suspected. Crucially, this adaptive localisation technique prevents NOT from considering subsamples containing two or more features, a key factor that ensures the general applicability of NOT. For selected scenarios, we show the consistency and near-optimality of

NOT in detecting the number and locations of generalised change-points, and discuss how to extend the proof to other settings. The NOT estimators are easy to implement and rapid to compute: the entire threshold-indexed solution path can be computed in close-to-linear time. Importantly, the NOT approach is easy to extend by the user to tailor to their own needs. There is no single competitor, but we show that the performance of NOT matches or surpasses the state of the art in the scenarios tested. Our methodology is implemented in the R package **not** (Baranowski et al., 2016b).

Chapter 5. Multiscale autoregression on adaptively detected timescales

Motivated by the notoriously difficult task of predicting high-frequency financial returns, in Chapter 5 we introduce Adaptive Multiscale Autoregressive (AMAR) time series models, where the quantity of interest is explicitly modeled as linearly dependent on its own past averages over unknown timescales. Combining the Ordinary Least Square method with the Narrowest-Over-Threshold approach described in Chapter 4, we propose an estimation procedure for identifying both the number and locations of the relevant timescales from the data. We demonstrate that this procedure consistently recovers the timescales under the framework in which both the number of the timescales and the largest timescale diverge with the sample size. In an application to data from the New York Stock Exchange Trades and Quotes Database, we show that our proposal offers relatively good performance in terms of the out-of-sample forecasting of high-frequency financial returns. The proposed methodology is implemented in the R package **amar** (Baranowski and Fryzlewicz, 2016a).

Chapter 6 summarises our contributions and points a number of directions for future research.

Statistical problems we deal with in Chapters 3, 5 and 4 are essentially different

from each other, as they are inspired by differently structured data. However, from the methodological and theoretical point of view, there is a common ground between those chapters, namely high-dimensionality. This is due to the fact that the complexity of the considered problems, measured either in terms of the number of covariates (Chapter 3) or in terms of the number of parameters in the corresponding models (Chapters 4 and 5), potentially grows with the sample size. Therefore all problems satisfy the general definition of high-dimensionality given in Fan and Lv (2010) and as such, should be regarded as high-dimensional.

Chapter 2

Literature review

In this chapter, we provide a review of the statistical literature related to the problems covered in this thesis: high-dimensional variable selection, multiple change-point detection and multiscale time series modelling.

2.1 High-dimensional variable selection

Suppose we observe $Y_1, \ldots, Y_n$, being n observations of the response, and that for each $i = 1, \ldots, n$ there are p predictors $X_{i1}, \ldots, X_{ip}$ which potentially influence Y. In this thesis, the variable selection problem is understood as the situation in which only a small number of predictors $\mathcal{S} \subset \{1, \ldots, p\}$ contribute to the response and our aim is to use the observed sample to identify those. When p large in comparison to the sample size n, the variable selection problem is said to be high-dimensional.

The high-dimensional variable selection problem has attracted a considerable attention across various scientific disciplines, such as genomics (Bickel et al., 2009; Bühlmann et al., 2014; Singh et al., 2002), economics (Korobilis, 2013; Scott and Varian, 2013), finance (Aït-Sahalia and Brandt, 2001; Stock and Watson, 2002; Tian et al., 2015), neuroimaging (Rosa et al., 2015; Schwartzman et al., 2009; Valdés-Sosa et al., 2005), or

machine learning (Du Jardin, 2010; Rakotomamonjy, 2003), to name a few. An extensive list of high-dimensional data analysis problems can be found in Donoho (2000) and Bühlmann et al. (2016). There exists a large body of statistical literature proposing methodologies designed to tackle high-dimensional data, an excellent overview of which can be found in Fan and Lv (2010), Bühlmann et al. (2011), Hastie et al. (2015) and Bühlmann et al. (2016). In this section, we briefly discuss different approaches to the high-dimensional variable selection problem, knowledge of which is vital in Chapter 3, where we propose our contribution to the problem.

On a broad level, we group the existing variable selection techniques according to the assumptions on the relationship between the response and the predictors, distinguishing three scenarios. The vast majority of the literature on the variable selection problem studies the following Linear Regression Model (LRM)

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j X_{ij} + \varepsilon_i, \quad , i = 1, \ldots, n \tag{2.1}$$

where $\beta_0, \beta_1, \ldots, \beta_p \in \mathbb{R}$ are the unknown regression coefficients and ε_i is the random error term, typically required to satisfy $\mathrm{E}\,\varepsilon_i = 0$, $\mathrm{E}\,\varepsilon_i^2 = \sigma^2$ and $\mathrm{E}\,\varepsilon_i\varepsilon_j \neq 0$ for $i \neq j$. The set of the variables that contribute to Y is then simply defined as

$$\mathcal{S} = \{1 \leq j \leq p : \beta_j \neq 0\}. \tag{2.2}$$

Let $\mathrm{E}\,Y_i|(X_{i1}, \ldots, X_{ip})$ denote the conditional expectation of Y given $X_{i1}, \ldots, X_{ip}$. Hereafter we assume that $\mathrm{E}\,|Y_i| < \infty$, which ensures that the conditional expectation exists. An important branch of the high-dimensional statistics literature assume that the relationship between the response and the predictors can be modelled with the

Generalised Linear Models (McCullagh and Nelder, 1989, GLMs) of the form

$$\mathrm{E}\, Y_i|(X_{i1},\ldots,X_{ip}) = g^{-1}(\sum_{j=1}^{p} \beta_0 + \beta_j X_{ij}), \tag{2.3}$$

where $\beta_0, \beta_1, \ldots, \beta_p \in \mathbb{R}$ are again unknown regression coefficients and $g : \mathbb{R} \mapsto \mathbb{R}$ is the (invertible) link function. In (2.3) the set of important variables is defined as (2.2).

Finally, the third scenario with respect to which we discuss the variable selection techniques is the nonparametric regression model, where

$$\mathrm{E}\, Y_i|(X_{i1},\ldots,X_{ip}) = f(X_{i1},\ldots,X_{ip}), \tag{2.4}$$

for an unknown, measurable function $f : \mathbb{R}^p \mapsto \mathbb{R}$. Here the important variables are defined as

$$\mathcal{S} = \{1 \leq j \leq p : \mathrm{E}\, Y_i|(X_{i1},\ldots,X_{ip}) \text{ functionally depends on } X_{ij}\}. \tag{2.5}$$

Regardless of the nature of the relationship between the response and the predictors, it is commonly assumed that the number of variables in $\mathcal{S}$ is small in comparison to p. In the context of LRM and GLMs, this condition is known as the *sparsity* assumption.

In the remainder of this section, we assume that $\beta_0 = 0$ in (2.1) and (2.3) and denote by $\mathbf{Z}_i = (Y_i, X_{i1}, \ldots, X_{ip})'$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)'$. For any $q \geq 1$, the ℓ_q-norm of any vector $\mathbf{v} \in \mathbb{R}^n$ is defined as $\|\mathbf{v}\|_q = \left(\sum_{j=1}^{n} |v_j|^q\right)^{1/q}$. Additionally, for $q = 0$ we set $\|\mathbf{v}\|_0 = \sum_{j=1}^{n} \mathbb{I}\,(v_i \neq 0)$, i.e. the number of non-zero coordinates of $\mathbf{v}$. Although $\|\mathbf{v}\|_0$ is not a properly defined norm, as it does not satisfy the absolute scalability condition, we refer to $\|\mathbf{v}\|_0$ as the ℓ_0-norm of $\mathbf{v}$, which is a common practice in the variable selection literature.

2.1.1 Variable selection via Penalised Likelihood minimisation

A substantial number of the variable selection techniques proposed in the context of (2.1) or (2.3), are derived from the solution of the following Penalised Likelihood (PLH) minimisation problem

$$\text{argmin}_{\boldsymbol{\beta}\in\mathbb{R}^p}\left(-2\ell\left(\boldsymbol{\beta}|\mathbf{Z}_1,\ldots,\mathbf{Z}_n\right)+\text{pen}_{\boldsymbol{\lambda}}\left(\boldsymbol{\beta}\right)\right), \tag{2.6}$$

where $\ell\left(\boldsymbol{\beta}|\mathbf{Z}_1,\ldots,\mathbf{Z}_n\right)$ is the log-likelihood of $\boldsymbol{\beta}$ given $\mathbf{Z}_1,\ldots,\mathbf{Z}_n$ and $\text{pen}_{\boldsymbol{\lambda}}$ is the penalty function depending on the tuning parameters vector $\boldsymbol{\lambda}=(\lambda_0,\ldots,\lambda_K)$ for some integer $K\geq 0$. Heuristically speaking, the aim of (2.6) is to find the estimates of $\boldsymbol{\beta}$ which guarantee that the resulting model fits the data well, but also satisfies some additional constrains on its complexity. Let $\hat{\boldsymbol{\beta}}=\left(\hat{\beta}_1,\ldots,\hat{\beta}_p\right)$ denote a solution of (2.6). Typically, the penalty function is designed such that (for an appropriately chosen $\boldsymbol{\lambda}$) a large number of $\hat{\beta}_j$'s are shrunk to zero. The resulting estimate of $\mathcal{S}$ is then defined as

$$\hat{\mathcal{S}}=\{1\leq j\leq p:\hat{\beta}_j\neq 0\}. \tag{2.7}$$

One of the most widely-studied examples of PLH is derived from the linear model (2.1) with the standard Gaussian i.i.d. noise. After omitting constants which does not change the minimum, (2.6) in the Guassian linear model simplifies to

$$\text{argmin}_{\boldsymbol{\beta}\in\mathbb{R}^p}\left(\sum_{i=1}^{n}\left(Y_i-\sum_{j=1}^{p}\beta_j X_{ij}\right)^2+\text{pen}_{\boldsymbol{\lambda}}\left(\boldsymbol{\beta}\right)\right). \tag{2.8}$$

2.1.1.1 PLH with ℓ_0-norm type penalty

Many classic model selection tools can be formulated as the PLH minimisation problem with the penalty function of the following form

$$\text{pen}_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \lambda_0 \|\boldsymbol{\beta}\|_0 , \tag{2.9}$$

where $\lambda_1 > 0$. For example, setting $\lambda_0 = 2$ recovers the Akaike's information criterion (Akaike, 1998); $\lambda_0 = \log(n)$ yields the Schwarz's information criterion (Schwarz, 1978). Some works propose modifications of the classic information criteria addressing probles arrising in the high-dimensional variable selection, see e.g. Bogdan et al. (2004) or Chen and Chen (2008). However, from the computational point of view, solving (2.6) with the penalty given by (2.9) requires an exhaustive search over all subsets of $\{1, \ldots, p\}$, hence it is not feasible if p is larger than a few dozens (Candes and Tao, 2007).

2.1.1.2 PLH with ℓ_1-norm and ℓ_2-norm type penalties

A popular class of penalties, which yield computationally tractable PLHs is of the form

$$\text{pen}_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \sum_{j=1}^{p} \lambda_j |\beta_j|, \tag{2.10}$$

where $\lambda_j > 0$ for all $j = 1, \ldots, p$. When $\lambda_j \equiv \lambda_0$ for some $\lambda_0 > 0$, (2.10) simplifies to

$$\text{pen}_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \lambda_0 \|\boldsymbol{\beta}\|_1 , \tag{2.11}$$

which recovers the penalty introduced by Alliney and Ruzinsky (1994) and Tibshirani (1996) in the context of (2.1). The latter author termed the resulting method Least absolute shrinkage and selection operator (Lasso). Owing to certain geometric properties of the ℓ_1-norm, PLH with the Lasso penalty leads to the solutions $\hat{\boldsymbol{\beta}}$ with certain

coefficients shrunk to be exactly zero. Lokhorst (1999); Park and Hastie (2007); Shevade and Keerthi (2003); Van de Geer (2008) study variable selection with the (2.11) penalty in the GLMs, for a disscusion of these and other developements of the Lasso methodology, see Tibshirani (2011) and Bühlmann et al. (2011).

Zou and Hastie (2005) observe that, in the situation when two or more important predictors are highly correlated, Lasso tends to select only one of them. In order to deal with this issue, Zou and Hastie (2005) consider the elastic net penalty of the following form

$$\text{pen}_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \lambda_0 \|\boldsymbol{\beta}\|_1 + \lambda_1 \|\boldsymbol{\beta}\|_2^2, \tag{2.12}$$

with $\lambda_0, \lambda_1 \geq 0$. When $\lambda_0 = 0$, (2.12) simplifies to the classic ridge penalty (Hoerl and Kennard, 1970).

Variants of (2.10) with λ_j possibly different for each $j = 1, \ldots, p$, have been also extensively studied in the literature. For example, Zou (2006) suggests to set $\lambda_j = \lambda_0 \left|\hat{\beta}_j^{OLS}\right|^{-\gamma}$, where $\hat{\beta}_j^{OLS}$ denotes the simple OLS estimate of β_j and $\gamma > 0$ is a tuning parameter, and shows that this approach yields certain optimality properties. Meinshausen and Bühlmann (2010) suggest to set the penalty parameters to $\lambda_j = \lambda_0 U_j^{-1}$, where U_j's are independently drawn from the uniform distribution. This, combined with their subsampling scheme discussed in Section 3.3.6, yields a method that successfully recovers $\mathcal{S}$ in certain settings in which the standard Lasso fails. Another interesting development can be found in Bogdan et al. (2015), who propose the SLOPE penalty defined as

$$\text{pen}_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \sum_{j=1}^{p} \lambda_j |\beta_{j:p}|,$$

where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p \geq 0$ and $\beta_{j:p}$'s are the coordinates of $\boldsymbol{\beta}$ ordered such that $|\beta_{1:p} \geq |\beta_{2:p}| \geq \ldots \geq |\beta_{p:p}|$. Bogdan et al. (2015) show that, for carefully selected λ_j's and

under orthogonal designs, SLOPE allows for variable selection with a control of the False Discovery Rate (Benjamini and Hochberg, 1995, FDR); Su et al. (2016) demonstrate that the SLOPE estimates of $\boldsymbol{\beta}$ achieve minimax rates in the ℓ_2-norm sense.

2.1.1.3 PLH with other types of penalties

Fan and Li (2001) propose yet another class of penalty functions, which are designed to return asymptotically unbiased, sparse and continuous in the data estimates of $\boldsymbol{\beta}$. They note that ℓ_q-based penalties in general do not satisfy some of those requirements and consider

$$\text{pen}_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \sum_{j=1}^{p} p_{\lambda}(|\beta_j|), \tag{2.13}$$

where p_{λ} is defined for $t \geq 0$ through the derivative

$$p'_{\lambda}(t) = \lambda_0 \left(\mathbb{I}(t \leq \lambda_0) + \frac{\max\{a\lambda_0 - t, 0\}}{(a-1)\lambda_0} \mathbb{I}(t > \lambda_0) \right),$$

and $a > 2$ is a tuning parameter. Fan and Li (2001) prove that the solutions of (2.8) with the penalty function specified this way satisfies all aforementioned requirements.

Zhang (2010) considers the penalty of form given by (2.13) with

$$p_{\lambda}(t) = \lambda_0 \int_0^t \max\{1 - x/(\lambda_0\gamma), 0\} dx \tag{2.14}$$

for some $\gamma > 0$. PLH estimates with this penalty are asymptotically unbiased and attain certain minimax convergence rates for the estimation of $\boldsymbol{\beta}$.

2.1.2 Variable screening methods

Fan and Lv (2008) point out two limitations of the variable selection achieved through

minimisation of PLH. First, theoretical conditions required for the consistency of the estimates given by (2.6) are unlikely to be satisfied when p is large in relation to n. Second, solving (2.6) for large values of p and n is typically computationally expensive. To tackle those issues in the context of (2.1), Fan and Lv (2008) introduce the concept of *variable screening*, defined as follows. Let $\hat{\omega}_j$ be a measure evaluating the impact of the j'th predictor X_{ij} on the response Y_i (e.g. Pearson correlation coefficient), calculated for the sample $Z_1, \ldots, Z_n$ and let $d_n < p$ be an integer, preferably smaller than n. Variable screening is then simply defined as the act of removing predictors which exhibit week relationship to the response. Denote by

$$\hat{\mathcal{S}}_{d_n} = \{1 \leq j \leq p : \hat{\omega}_j \text{ is among the first } d_n \text{ largest of all}\}, \tag{2.15}$$

i.e. the set of variables that survive after the screening. The variable screening based on the measure $\hat{\omega}_j$ is said to posses *the sure screening property* if

$$\mathbb{P}\left(\mathcal{S} \subset \hat{\mathcal{S}}_{d_n}\right) \underset{n}{\to} 1. \tag{2.16}$$

Fan and Lv (2008) show that in the linear model (2.1), the variable screening based on the Pearson correlation coefficient, termed as Sure Independence Screening (SIS), achieves the sure screening property, under certain restrictions on the correlation between predictors. If this is satisfied for d_n comparable to the sample size n, we can expect better estimation accuracy by applying the PLH estimation directly on the reduced set of predictors $\hat{\mathcal{S}}_{d_n}$. Importantly, the sample correlations are quick to compute, therefore SIS provides a solution to the aforementioned problems in (2.1).

Due to their simplicity and wide applicability, variable screening procedures attracted considerable attention in the statistical literature. Consequently, many well-known statistical measures have been shown to possess the sure screening property, e.g. Kendall's

τ correlation (Li et al., 2012a) or distance correlation (Li et al., 2012b). An excellent overview of these developments can be found in Liu et al. (2015).

2.1.3 Subsampling in variable selection

A particular difficulty in performing variable selection or variable screening in high-dimensional data using techniques described in the previous sections, is that their performance is sensitive to the choice of the tuning parameters parameters (Fan and Tang, 2013; Meinshausen and Bühlmann, 2010). A variant of cross-validation (CV), so called k-fold CV, where $1 \leq k \leq n$, is one of the most popular methods employed to choose the tuning parameters (Arlot et al., 2010; Friedman et al., 2001). In this method, the original sample is randomly divided into k subsamples of approximately equal length. Next, each subsample is used as the validation set, on which a number of models, corresponding to different tuning parameters and fitted on the remaining data, are evaluated and the parameter yielding the best model in terms of the average evaluation criterion is selected. Typically, k-fold CV is used in conjunction with a criterion that asses the predictive power of the model, e.g. the Mean Squared Error in LRM.

In Section 3.3.6, we discuss two alternatives to CV that use subsampling to perform variable selection taking different than prediction-oriented approach, namely, Stability Selection of Meinshausen and Bühlmann (2010) and the bootraped rankings of Hall and Miller (2009a).

2.2 Data segmentation and multiple change-point detection

Let Y_t, $t = 1, \ldots, T$, be a univariate time series. Informally, the multiple-change point detection problem is defined as the situation in which there exist the unknown *change-points* in time $1 \leq \tau_1 < \ldots < \tau_q < T$ such that for each $j = 0, 1, \ldots, q$, the distribution of Y_t exhibits certain degree of homogeneity across the segments between the change-points, i.e. for $t = \tau_j + 1, \ldots, \tau_{j+1}$, where $\tau_0 = 1$ and $\tau_{q+1} = T$ for notational convenience. The primary goal in this setting is to estimate both the number and the locations of the change-points. Moreover, as the estimated change-points imply the segmentation of the data into homogeneous blocks, it also often of interest to fit a model that describes the homogeneity in each segment.

There are numerous applications where the multiple change-point problem arises, e.g. in genomics (Olshen et al., 2004; Zhang and Siegmund, 2007), neuroscience (Ombao et al., 2001; Schröder and Ombao, 2015), finance (Andreou and Ghysels, 2002; Aue and Horváth, 2013; Ewing and Malik, 2013; Schröder and Fryzlewicz, 2013), oceanography (Killick et al., 2010, 2013), climatology (Beaulieu et al., 2012; Cahill et al., 2015; Ruggieri, 2013), hydrology (Wang et al., 2014) or acoustic sensing signals (Pickering, 2016). In this section, we review a selection of multiple change-point detection problems and methods, which serves as a starting point to the discussion of Chapter 4. For a more exhaustive presentation of various change-point problems, we refer the reader to the following books: Basseville et al. (1993); Brodsky and Darkhovsky (2013); Chen and Gupta (2011); Wu (2007).

A large body of the change-point detection literature studies problems that are

formulated using the following model

$$Y_t = f_t + \varepsilon_t, \quad t = 1, \ldots, T, \tag{2.17}$$

where f_t is a deterministic signal with the change-points in its structure at $1 \leq \tau_1 < \ldots < \tau_q < T$ and the random noise ε_t is a stationary time series exactly or approximately centred at zero. In Section 2.2.1, we discuss methods that deal with the case of f_t being piecewise-constant, which is the most widely studied example of (2.17). Section 2.2.2 concerns the more general scenario, in which f_t in each segment is a (possibly non-constant, e.g. linear) function of time or other non-stochastic covariates.

Another common way of formulating the multiple change-point detection problem is to assume that

$$Y_t \text{ for } t = \tau_j + 1, \ldots, \tau_{j+1} \text{ are i.i.d. } F_j\text{-distributed}, \tag{2.18}$$

where $j = 0, \ldots, q$ and each F_j is a cumulative distribution function of some distribution and the consecutive cdfs are different, i.e. $F_j \neq F_{j+1}$. For example, (2.18) can be used to model the situation in which both or either of $\mathrm{E}(Y_t)$ and $\mathrm{Var}(Y_t)$ are piecewise constant functions of time. Section 2.2.3 discusses this and other change-point detection problems in the context of (2.18). Naturally, models (2.17) and (2.18) overlap to certain extend, e.g. the piecewise constant case in (2.17) can be modelled using (2.18) assuming that F_j are of the form $F_j(x) = F(x - \theta_j)$, where θ_j are scalars satisfying $\theta_j \neq \theta_{j+1}$ and F is a known cumulative distribution function. In general, however, there are some important examples that can be modelled only with either (2.17) or (2.18), as e.g. in the case of piecewise-constant variance or piecewise-linear mean of Y_t.

We remark that in this thesis we focus on retrospective (*a posteriori* or *off-line*) change-point detection, which assumes that the entire sample $Y_1, \ldots, Y_T$ is available at

the time of the analysis. When the observations arrive one by one and the aim of the analysis is to detect changes in the most recent data, the problem is classified as *on-line* change-point detection. A survey of literature in this area can be found in Basseville et al. (1993); Kawahara and Sugiyama (2012); Lai (2001).

2.2.1 Canonical change-point detection problem

The scenario in which f_t in (2.17) is piecewise-constant, i.e.

$$f_t = \theta_j, \text{ for } t = \tau_j + 1, \ldots, \tau_{j+1}, \tag{2.19}$$

where $j = 0, \ldots, q$ and $\theta_0, \theta_1, \ldots, \theta_q \in \mathbb{R}$ satisfy $\theta_j \neq \theta_{j+1}$ for $j = 0, \ldots, q$, is one of the most widely studied examples of (2.17). We refer to (2.17) with the signal f_t given by (2.19) as to the *canonical change-point detection problem*.

Early literature on the canonical change-point detection problem largely focuses on the detection of a single change-point in the data (Davis, 1979; Hawkins, 1977; Sen and Srivastava, 1975; Worsley, 1986), and is typically stated as a hypothesis testing problem. Throughout this section, we focus on the case of multiple change-points in f_t.

2.2.1.1 Multivariate optimisation

When it is suspected that multiple change-points are present, the estimators of τ_j's are often formulated as the solutions of the following multivariate optimisation problem

$$\underset{\substack{1 \leq \tau_1 \leq \ldots \leq \tau_q < T \\ q \leq q_{max}}}{\operatorname{argmin}} \left(\text{Cost}(Y_1, \ldots, Y_T, \tau_1, \ldots, \tau_q) + \text{pen}(q, \tau_1, \ldots, \tau_q) \right) \tag{2.20}$$

where $\text{Cost}(Y_1, \ldots, Y_T, \tau_1, \ldots, \tau_q)$ is the cost function, $\text{pen}(q, \tau_1, \ldots, \tau_q)$ is the penalty function and q_{max} is the maximum number of change-points, that is typically assumed to be fixed (Chen and Gupta, 2011). For example, Yao (1988) considers the Schwarz's

Information Criterion (SIC, Schwarz (1978)) penalty $\text{pen}(q, \tau_1, \ldots, \tau_q) = (2q+1)\log(T)$ with the cost function

$$\text{Cost}(Y_1, \ldots, Y_T, \tau_1, \ldots, \tau_q) = -T \log \left(\sum_{j=0}^{q} \sum_{t=\tau_j+1}^{\tau_{j+1}} \left(Y_t - (\tau_{j+1} - \tau_j)^{-1} \sum_{l=\tau_j+1}^{\tau_{j+1}} Y_l \right)^2 \right) \tag{2.21}$$

which is derived from the log-likelihood function of $\theta_1, \ldots, \theta_{q+1}, \tau_1, \ldots, \tau_q$ given the data $Y_1, \ldots, Y_T$ under the assumption that the noise is i.i.d. Gaussian. Killick et al. (2012a) define

$$\text{Cost}(Y_1, \ldots, Y_T, \tau_1, \ldots, \tau_q) = -\sum_{j=0}^{q} \sup_{\theta_j} \log \ell(Y_{\tau_j+1}, \ldots, Y_{\tau_j}; \theta_j),$$

where $\ell(Y_{\tau_j+1}, \ldots, Y_{\tau_{j+1}}; \theta_{j+1})$ denotes the likelihood of θ_{j+1} given $Y_{\tau_j+1}, \ldots, Y_{\tau_{j+1}}$, and consider the linear penalty $\text{pen}(q, \tau_1, \ldots, \tau_q) = \lambda(q+1)$, where $\lambda > 0$ is a tuning parameter. An example of (2.20) with the penalty depending on both the number and the locations of the change-points can be found in Zhang and Siegmund (2007). For certain cost functions and penalties linear in the number of change-points, dynamic programming techniques (Bertsekas, 1995) can be used to compute (2.20) in $O(T)$ average time (Killick et al., 2012a; Maidstone et al., 2016; Rigaill, 2010). In general, however, solving (2.20) is computationally expensive with the typical computational complexity of the order of $O(q_{max}T^2)$ and not straightforward to implement.

2.2.1.2 Binary Segmentation

Binary Segmentation (BS, Vostrikova (1981)) is a generic approach that estimates the change-points sequentially, detecting just a single change-point at each stage of the procedure solving a one-dimensional optimisation problem, as opposed to solving a multivariate optimisation problem as in (2.20). Its main building block is a test statistic

Algorithm 2.1 Binary Segmentation

Input: Data vector $\mathbf{Y} = (Y_1, \ldots, Y_T)'$, threshold $\zeta_T > 0$, $\mathcal{S} = \emptyset$.
Output: Set of estimated change-points $\mathcal{S} \subset \{1, \ldots, T\}$.

procedure BS(s, e, ζ_T)
  if $e - s < 1$ **then** STOP
  else
    if $\max_{s_m \leq b \leq e_m} \mathcal{C}^b_{s_m, e_m}(\mathbf{Y}) \leq \zeta_T$ **then** STOP
    else
      $b^* := \operatorname{argmax}_{s_{m^*} \leq b \leq e_{m^*}} \mathcal{C}^b_{s_{m^*}, e_{m^*}}(\mathbf{Y})$
      $\mathcal{S} := \mathcal{S} \cup \{b^*\}$
      BS(s, b^*, ζ_T)
      BS$(b^* + 1, e, \zeta_T)$
    end if
  end if
end procedure

$\mathcal{C}^b_{s,e}(\mathbf{Y})$, often referred to as a *contrast function*, defined for any $1 \leq s \leq b \leq e \leq T$. The contrast function is constructed such that $\max_{s \leq b \leq e} \mathcal{C}^b_{s,e}(\mathbf{Y}) > \zeta_T$ for certain $\zeta_T > 0$ indicates presence of a change-point in $[s, e]$ located at $b^* = \operatorname{argmax}_{s \leq b \leq e} \mathcal{C}^b_{s,e}(\mathbf{Y})$. For example, Vostrikova (1981) considers the absolute value of the Cumulative Sum (CUSUM) statistic, defined as follows

$$\mathcal{C}^b_{s,e}(\mathbf{Y}) = \left| \sqrt{\frac{e-b}{l(b-s+1)}} \sum_{t=s}^{b} Y_t - \sqrt{\frac{b-s+1}{l(e-b)}} \sum_{t=b+1}^{e} Y_t \right|, \tag{2.22}$$

which can be derived from the Gaussian likelihood function (for details see Section 4.2.3). Algorithm 2.1 describes BS using pseudocode. The procedure is launched by the call BS(1,T,ζ_T) and at this initial stage the entire sample is searched for the most likely location of the change-point denoted by b^*. If the change-point is deemed significant, i.e. the corresponding maximum value of the contrast function exceeds the threshold ζ_T, b^* is added to the set of estimated change-points and a similar search is performed on the segments to the left and to the right of b^*, until no further change-points are detected.

The BS algorithm is easy to code and has a low computational complexity, typically

of the order of $O(T \log T)$. Those are among the reasons why it has many applications outside the canonical change-point detection problem, e.g. in the multiple detection of change-points in high-dimensional mean (Cho, 2016; Cho and Fryzlewicz, 2015; Wang and Samworth, 2016), variance (Inclan and Tiao, 1994), autocovariance (Cho and Fryzlewicz, 2012b), conditional variance (Fryzlewicz and Subba Rao, 2014), the frequency bands of autospectra and cross-coherences in multi-channel EEG data (Schröder and Ombao, 2015) or in a nonparametric setting (Matteson and James, 2014). However, as shown in Venkatraman (1992) and Fryzlewicz (2014), BS (with the CUSUM statistics employed as the contrast function) estimates the locations of the change-points in (2.19) at a sub-optimal rate and only under strong assumptions on the minimum spacing between the consecutive change-points. Those weaknesses of BS stem from the fact that maximising the CUSUM statistic is equivalent to finding a piecewise-constant function with a single change-point that fits the data $Y_s, \ldots, Y_e$ best in the least squares sense (see Section 4.2.3 for an explanation). Therefore, if at any stage of Algorithm 2.1 the $[s, e]$ interval contains multiple change-points, BS proceeds via fitting the wrong model, which can adversely impact its performance.

A number of attempts have been made in the literature to modify the BS procedure in order to address the issues mentioned above, see e.g. Olshen et al. (2004), Venkatraman and Olshen (2007) and Fryzlewicz (2014). Here we discuss the Wild Binary Segmentation (WBS) algorithm of Fryzlewicz (2014) in more detail, as we repeatedly refer to this method in Chapter 4. Algorithm 2.2 outlines the WBS procedure. At its initial stage, the contrast function is calculated over M randomly drawn intervals $[s_m, e_m]$, as opposed to calculating the contrast function just for $s = 1$ and $e = T$ as in the BS algorithm. Subsequently, the interval yielding the largest contrast is picked and, provided that the corresponding contrast exceeds the threshold, the b^* that maximises the contrast over that interval is added to the set of the estimated change-points. As in the standard BS,

Algorithm 2.2 Wild Binary Segmentation

Input: Data vector $\mathbf{Y} = (Y_1, \ldots, Y_T)'$, F_T^M being a set of M intervals, with start- and end- points drawn independently and uniformly with replacement from $\{1, \ldots, T\}$, $\mathcal{S} = \emptyset$.

Output: Set of estimated change-points $\mathcal{S} \subset \{1, \ldots, T\}$.

procedure WBS(s, e, ζ_T)
  if $e - s < 1$ **then** STOP
  else
    $\mathcal{M}_{s,e} := \left\{m : [s_m, e_m] \in F_T^M, [s_m, e_m] \subset [s, e]\right\}$
    if $\mathcal{M}_{s,e} = \emptyset$ **then** STOP
    else
      $m^* :\in \operatorname{argmax}_{m \in \mathcal{M}_{s,e}} \max_{s_m \le b \le e_m} \mathcal{C}^b_{s_m, e_m}(\mathbf{Y})$
      if $\max_{s_{m^*} \le b \le e_{m^*}} \mathcal{C}^b_{s_m, e_m}(\mathbf{Y}) \le \zeta_T$ **then** STOP
      else
        $b^* := \operatorname{argmax}_{s_{m^*} \le b \le e_{m^*}} \mathcal{C}^b_{s_{m^*}, e_{m^*}}(\mathbf{Y})$
        $\mathcal{S} := \mathcal{S} \cup \{b^*\}$
        WBS(s, b^*, ζ_T)
        WBS$(b^* + 1, e, \zeta_T)$
      end if
    end if
  end if
end procedure

similar procedure is then applied to the data to the left and to the right of b^*. Some of the intervals drawn contain exactly one change-point with high probability, provided that M is large enough, which is the reason why in the canonical change-point detection problem the WBS algorithm estimates the locations of the change-points at a near-optimal rate, as shown in Fryzlewicz (2014).

The WBS algorithm can be also applied to other change-point detection problems, e.g. in order to detect change-points in the second order structure of a time series (Korkas and Fryzlewicz, 2016). However, there is no guarantee that the intervals picked at each stage of the WBS procedure contain no more than one change-point, which is the reason why it fails to detect the change-points consistently in some settings outside the canonical change-point detection. An example of such setting is given in Section 4.1.

2.2.1.3 Other approaches

In this section, for the sake of completeness, we briefly mention two classes of change-point detection methods. The first class consists of methods which in the first step fit a piecewise-constant vector to the data and subsequently extract change-points from the obtained estimates, typically using certain post-processing techniques. A number of methods fall into this category, e.g. the Tail-Greedy Unbalanced Haar transform of Fryzlewicz (2016), trend filtering method of (Taylor and Tibshirani, 2014) with the post-processing procedure proposed by Lin et al. (2016) or the approach of Harchaoui and Lévy-Leduc (2012). Finally, Bayesian methods have been also studied in the context of the canonical change-point detection problem. A non-exhaustive list of early works includes Broemeling (1972, 1974); Chernoff and Zacks (1964), for an overview of more recent works that take a Bayesian perspective see Erdman and Emerson (2008); Jandhyala et al. (2013).

2.2.2 Regression change-point models

In this section, we discuss the problem of detecting change-points in (2.17) when the signal f_t follows

$$f_t = \theta_{1,j}x_{t1} + \theta_{2,j}x_{t2} + \ldots, \theta_{d,j}x_{td}, \text{ for } t = \tau_j + 1, \ldots, \tau_{j+1} \tag{2.23}$$

for each $j = 0, 1, \ldots, q$, where the predictors x_{tj} are non-stochastic, d is known, the vectors of parameters $\mathbf{\Theta}_j = (\theta_{1,j}, \ldots, \theta_{d,j})'$ are unknown and satisfy $\mathbf{\Theta}_j \neq \mathbf{\Theta}_{j+1}$ for $j = 0, \ldots, q-1$. An important example of (2.23) is the scenario in which f_t is a piecewise-polynomial function of time, i.e.

$$f_t = \theta_{1,j} + \theta_{2,j}t + \ldots, \theta_{d,j}t^{d-1}, \text{ for } t = \tau_j + 1, \ldots, \tau_{j+1}. \tag{2.24}$$

Naturally, for $d = 1$ (2.24) simplifies to the canonical change-point detection problem, therefore we focus our attention on the case of $d > 1$ in the discussion of this section. Furthermore, we distinguish a class of *continuous* piecewise-polynomial signals, i.e. f_t satisfying (2.24) for which definition (2.24) applied for all $t \in [1, T]$ yields a continuous function.

Early literature on change-point detection in model (2.17) with the signal given by (2.23) focus mainly on the problem of detecting a single change-point in the data, a survey of these developments can be found in Zacks (1982). The case of $q > 1$ has attracted considerably less attention in the early literature, however, some interesting contributions can be found. Gallant and Fuller (1973); Hudson (1966), e.g., propose methods for simultaneous estimation of τ_j's and $\theta_{k,j}$'s in the context of (2.24) using least squares methods and certain type of the Gauss-Newton method for finding solutions of the least squares criteria considered in the corresponding works.

In this section, we discuss two popular approaches for change-point detection in

(2.17) with the signal given by (2.23), that are formulated as multivariate optimisation problems. Examples of other approaches can be found in Leonardi and Bühlmann (2016), who combine the BS algorithm with Lasso to detect change-points in (2.23) in the case of large d or in Ruggieri (2013); Ruggieri and Antonellis (2016) who propose a Bayesian approach to estimate change-points in (2.24) when $d = 2$.

2.2.2.1 Methodology of Bai and Perron (1998)

Bai and Perron (1998) propose to estimate the change-points in a slightly more general version of model (2.17) with the signal given by (2.23) solving the following multivariate optimisation problem

$$\operatorname{argmin}_{\substack{1\leq\tau_1\leq\ldots\leq\tau_q<T \\ \forall_j(\tau_{j+1}-\tau_j)\geq h}} \left(\sum_{j=0}^{q} \inf_{(\theta_{1,j},\ldots,\theta_{d,j})'\in\mathbb{R}^d} \sum_{t=\tau_j+1}^{\tau_{j+1}} \left(Y_t - \theta_{1,j}x_{t1} - \theta_{2,j}x_{t2} - \ldots - \theta_{d,j}x_{td}\right)^2 \right), \tag{2.25}$$

where h is the user-specified lower bound on the minimum distance between the change-points and the number of change-points q is assumed to be known. As in the case of methods presented in Section 2.2.1.1, solutions of (2.25) can be found using dynamic programming techniques in $O(T^2)$ time (Bai and Perron, 2003).

Bai and Perron (1998) show that (2.25) yields consistent estimators of the change-points under week assumptions on x_{tj} and ε_t. They also propose a testing procedure to estimate q when it is unknown. However, in such case the computational complexity of their procedure is of the order $O(q_{max}T^2)$, where q_{max} denotes the maximum number of change-points imposed by the user, which is prohibitively slow for even moderately large sample sizes. We illustrate this point in the numerical examples of Section 4.4, where among other aspects, we compare empirical running times of various change-point detection techniques.

2.2.2.2 Trend filtering

Trend filtering Kim et al. (2009); Tibshirani (2014) is a technique whose main goal is to approximate the signal f_t in (2.17) using continuous piecewise-polynomials. For any d, the trend filtering estimate is defined as the solution of the following optimisation problem

$$\operatorname{argmin}_{\mathbf{f}\in\mathbb{R}^T}\left(\sum_{t=1}^{T}(Y_t-f_t)^2+\lambda\sum_{t=1}^{T}|(\mathbf{D}^d\mathbf{f})_t|\right), \tag{2.26}$$

where $\mathbf{f}=(f_1,\ldots,f_T)'$, $\mathbf{D}\in\mathbb{R}^{T\times T}$ denotes the discrete difference operator defined for any vector $\mathbf{v}=(v_1,\ldots,v_T)'\in\mathbb{R}^T$ as follows

$$(\mathbf{D}\mathbf{v})_t=\begin{cases}v_{t+1}-v_t, & \text{for } t=1,\ldots,T-1,\\ 0 & \text{for } t=T,\end{cases}$$

and $\lambda>0$ is a tuning parameter.

Trend filtering belongs to a wide class of ℓ_1-penalised methods discussed previously in Section 2.1.1 in the context of variable selection. The key property of the ℓ_1-type penalty term in (2.26) is that (for a carefully chosen λ) it leads to the solution $\hat{\mathbf{f}}=(\hat{f}_1,\ldots,\hat{f}_T)'$ such that $(\mathbf{D}^d\hat{\mathbf{f}})_t=0$ for the majority of times t, which essentially means that $\hat{f}_1,\ldots,\hat{f}_T$ is a piecewise-polynomial of degree at most $d-1$.

Tibshirani (2014) shows that the trend filtering estimates are closely related to the estimates obtained using the locally adaptive regression splines of Mammen and van de Geer (1997) and as such achieve the same minimax rate for the estimation of f_t. However, solutions of (2.26) are in general quicker to compute than the regression splines; for the discussion of the computational aspects of the trend filtering see Arnold and Tibshirani (2016).

When the true signal is assumed to follow (2.24), Lin et al. (2016) propose to use the

trend filtering estimates in order to detect the change-points in f_t, defining the estimators of τ_j's as τ such that $(\mathbf{D}^d\hat{\mathbf{f}})_\tau > 0$. We note, however, that this approach is not optimal for change-point detection in the case of $d = 1$, as shown in Brodsky and Darkhovsky (2013); Cho and Fryzlewicz (2011). Although we are not aware of similar results for $d > 1$, the empirical evidence of Section 4.4 suggest that change-point detection achieved through minimisation of (2.26) may not be optimal either.

2.2.3 Other change-point detection problems

In Sections 2.2.1 and 2.2.2, we focus on the problem of detecting change-points in various characteristics of $\mathrm{E}\, Y_t$. Below, we present a selective overview of approaches in the context of model (2.18), allowing for multiple change-points in other aspects of the distribution of Y_t than its mean.

2.2.3.1 Change in variance and/or mean

Arguably, one of the simplest departures from the piecewise-constancy in the mean of Y_t, is to allow its variance to change in a piecewise-constant manner. This problem is typically modelled using (2.18), by assuming that in the j'th segment, $j = 0, \ldots, q$, and for all $x \in \mathbb{R}$, the cdf is of the following form

$$F_x(t) = F\left(\frac{x - \theta_{1,j}}{\theta_{2,j}}\right), \tag{2.27}$$

where $F : \mathbb{R} \mapsto [0, 1]$ is a cdf of some distribution, e.g. standard Gaussian, typically assumed to be know, $\theta_{1,j} \in \mathbb{R}$, $\theta_{2,j} > 0$, and there is a change in at least one of the parameters in the consecutive segments, i.e. $(\theta_{1,j}, \theta_{2,j})' \neq (\theta_{1,j+1}, \theta_{2,j+1})'$ for all $j = 0, 1, \ldots, q - 1$.

A considerable number of early works study (2.27) under the assumption that there is at most one change-point in the data, with the mean of Y_t assumed to be constant,

i.e. $\theta_{1,j} \equiv \theta_{1,0}$, and known. In this setting, Hsu (1977) proposes a CUSUM-type test, assuming that the data are Gaussian and the location of the change-point is unknown; Hsu (1979) derives a test statistic for Gamma-distributed data with known location of the change; Hsieh (1984) proposes a non-parametric rank test. Menzefricke (1981) allows for different means in the segments and introduces a Bayesian procedure for detecting a single change-point in Gaussian data. Some authors, e.g. Chen and Gupta (1997); Horváth (1993) in the off-line and Hawkins and Zamba (2005) in the on-line change-point detection context, base their procedures on a likelihood-ratio type test, which is the approach we take in Chapter 4.

Many multiple change-point detection techniques that have been introduced in Section 2.2.1 in the context of the canonical change-point detection, can be also applied to estimate change-points in (2.18) with (2.27). For example, the methodology of Killick et al. (2012a) handles the case of (2.27), when the adequate likelihood function in (2.21) is specified; Inclan and Tiao (1994) introduces a slightly modified version of the Binary Segmentation procedure, with a contrast function derived from the likelihood-ratio test under the assumption that $\theta_{1,j} \equiv 0$ and the data are Gaussian; Schröder (2016) applies the Wild Binary Segmentation algorithm with a contrast function specifically designed to tackle the case of Gaussian data following (2.27) where changes may occur non-simultaneously, i.e. either $\theta_{1,j} \neq \theta_{1,j+1}$ and $\theta_{2,j} = \theta_{2,j+1}$ or $\theta_{1,j} = \theta_{1,j+1}$ and $\theta_{2,j} \neq \theta_{2,j+1}$ for some j.

2.2.3.2 Nonparametric change-point detection

A (possibly multiple) change-point detection problem defined by (2.18) is said to be nonparametric, when the cdfs $F_1, \ldots, F_{q+1}$ describing the distribution of Y_t in the segments between the change-points are unknown, excluding those examples in which F_j are known to belong to a parametric family of distributions.

Much of the literature on the nonparametric change-point detection problem deal with the case of a single change-point in the data and state it as a hypothesis testing problem. In this setting, Darkhovskh (1976); Pettitt (1979) consider a Mann-Whitney type test statistics, while Carlstein (1988); Ross and Adams (2012) introduce tests based on Cramer–Von Mises and Kolmogorov–Smirnov distances between empirical cdfs before and after a given change-point candidate. An example of a test that easily extends to the case of multivariate data can be found in Harchaoui et al. (2009), who proposes a kernel-based method.

The case of multiple change-points in the nonparametric setting has attracted considerably less attention in the statistical literature, however, we observe a growing interest in this problem in recent publications. For example, Matteson and James (2014) apply the Binary Segmentation algorithm using the energy statistic of Szekely and Rizzo (2005) as the contrast function, which allows for the detection of any type of distributional change in the multivariate data. In James and Matteson (2015), the authors consider a multivariate optimisation problem similar to (2.20), with the cost function based on a quicker to compute approximation of the energy statistic. Another example of a multivariate optimisation type approach is taken in Haynes et al. (2016a); Zou et al. (2014), with the cost function based on a functional of the joint non-parametric log-likelihood function of the data.

2.3 Multiscale time series models

Broadly speaking, a univariate time series X_t, $t = 1, \ldots, T$ is said to follow a multiscale model, when X_t is observed at a fine resolution, e.g. daily, but it depends on the observations of this or other time series recorded on a course scale, e.g. weekly, or vice-versa. An extensive list of data that fit this framework, as well as a review of the multiscale methodology can be found in Ferreira and Lee (2007), Ferreira et al. (2010)

and Nason (2010). This section discusses a selection of multiscale time series models, that are related to the Adaptive Multiscale Autoregressive time series models that we introduce in Chapter 5.

2.3.1 Multiscale time series models of Ferreira et al. (2006)

Ferreira et al. (2006) introduces a class of multiscale time seris models that consist of two main building blocks: Y_t, $t = 1, \ldots, Tm$, the fine level process, where $m > 1$ is known, and the coarse level aggregate process X_t defined as follows

$$X_t = m^{-1} \sum_{j=1}^{m} Y_{tm-j} + \varepsilon_t, \tag{2.28}$$

where the noise term $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$ form a i.i.d. sequence independent of the fine level process. To model the behaviour of the fine level process, Ferreira et al. (2006) recommends to choose a simple model, e.g. AR(1), and show that with such choice the resulting coarse level process given by (2.28) can emulate long memory process. To estimate (2.28) from the data, Ferreira et al. (2006) propose a Bayesian procedure based on Markov Chain Monte Carlo (Gilks, 2005).

2.3.2 Mixed Data Sampling Regression Models

Another attempt to model time series sampled at different frequncies can be found in Ghysels et al. (2004), who propose Mixed Data Sampling (MIDAS) regression model. Using the notation of the previous section, the MIDAS model is defined as follows

$$X_t = \beta_0 + \sum_{i=1}^{p} b_i(Y_{tm-i}; \boldsymbol{\beta}) + \varepsilon_t, \tag{2.29}$$

where $b_1(\cdot; \boldsymbol{\beta}), \ldots, b_p(\cdot; \boldsymbol{\beta})$ are given functions of the lagged observations recorded at a higher frequency and a low-dimensional vector of unknown parameters $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_q)'$

and ε_t is a random noise. Here the value of m indicates that for each recorded observation of X_t, m values of Y_i are sampled.

Data that are sampled at various frequencies are ubiquitous in finance and macroeconomics, which is perhaps one of the reasons why MIDAS models have found multiple applications in the corresponding literature, e.g. for forecasting of daily volatility (Ghysels and Valkanov, 2006), quarterly GDP growth by using monthly business cycle indicators (Bai et al., 2013; Clements and Galvão, 2009) or other daily financial data Andreou et al. (2013).

Depending on the specfication of $b_i(\cdot;\boldsymbol{\beta})$ in (2.29) are typically estimated using either Ordinary Least Squares of Nonlinear Least Squares, for details and examples see Ghysels et al. (2007). Here we mention one particular form of $b_i(\cdot;\boldsymbol{\beta})$ studied in Forsberg and Ghysels (2007), who consider

$$X_t = \beta_0 + \sum_{j=1}^{q} \beta_j \sum_{i=1}^{\tau_j} Y_{tm-i} + \varepsilon_t, \tag{2.30}$$

where $1 \leq \tau_1 < \ldots \leq \tau_q$ are known integers, as Chapter 5 introduces a model that assumes a similar structure of the conditional mean of the time series of interest. However, there are some important differences, e.g. $\tau_1, \ldots, \tau_q$ in our model are unknown. For more details, see Chapter 5.

Chapter 3

Ranking-Based Variable Selection for high-dimensional data

3.1 Introduction

Suppose Y is a response, covariates $X_1, \ldots, X_p$ constitute the set of random variables which potentially influence Y, and we observe $\mathbf{Z}_i = (Y_i, X_{i1}, \ldots, X_{ip})$, $i = 1, \ldots, n$, independent copies of $\mathbf{Z} = (Y, X_1, \ldots, X_p)$. In modern statistical applications, where p could be very large, even in tens or hundreds of thousands, it is often assumed that there are many variables having no impact on the response. It is then of interest to use the observed data to identify a subset of $X_1, \ldots, X_p$ which affects Y. The so-called variable selection or subset selection problem plays an important role in statistical modelling for the following reasons. First of all, the number of parameters in a model including all covariates can exceed the number of observations when $n < p$, which makes precise statistical inference not possible. Even when $n \geq p$, constructing a model with a small subset of initial covariates can boost the estimation and prediction accuracy. Second, parsimonious models are often more interpretable. Third, identifying the set of important variables can be the main goal of statistical analysis, which precedes further scientific

investigations.

Our aim is to identify a subset of $\{X_1, \ldots, X_p\}$ which contributes to Y, under scenario in which p is potentially much larger than n. To model this phenomenon, we work in a framework in which p diverges with n. Therefore, both p and the distribution of $\mathbf{Z}$ depend on n and we work with a triangular array, instead of a sequence. Our framework includes, for instance, high-dimensional linear and non-linear regression models. Our proposal, termed Ranking-Based Variable Selection (RBVS), can be in general applied to any technique which allows the ranking of covariates according to their impact on the response. Therefore, we do not impose any particular model structure on the relationship between Y and $X_1, \ldots, X_p$, however $\hat{\omega}_j = \hat{\omega}_j(\mathbf{Z}_1, \ldots, \mathbf{Z}_n)$, $j = 1, \ldots, p$, the measure used to assess the importance of covariates (either joint or marginal) may require some assumptions on the model. The main ingredient of the RBVS methodology is a variable ranking defined as follows.

Definition 3.1.1. The variable ranking $\mathbf{R}_n = (R_{n1}, \ldots, R_{np})$ based on $\hat{\omega}_1, \ldots, \hat{\omega}_p$ is a permutation of $\{1, \ldots, p\}$ satisfying $\hat{\omega}_{R_{n1}} \geq \ldots \geq \hat{\omega}_{R_{np}}$. Potential ties are broken at random.

A large number of measures can be used to construct variable rankings. In the linear model, the marginal correlation coefficient serves as an example of such a measure. It is the main component of the Sure Independence Screening (SIS, Fan and Lv (2008)). Hall and Miller (2009a) consider the generalized correlation coefficient, which can capture (possibly) non-linear dependence between Y and X_j's. Along the same lines, Fan et al. (2011) propose a procedure based on the magnitude of spline approximations of Y over each X_j, aiming to capture dependencies in non-parametric additive models. Fan and Song (2010) extend SIS to a class of GLMs, using estimates of the maximum marginal likelihood as the measure of association. Cho and Fryzlewicz (2012a) consider variable screening based on the tilted correlation, which accounts for high correlations between

the variables, when such are present. Li et al. (2012a) utilise the Kendall rank correlation coefficient, which can be applicable when Y is, for example, a monotonic function of the linear combination of $X_1, \ldots, X_p$. Several model-free variable ranking procedures have been also advocated in the literature. Li et al. (2012b) propose to rank the covariates according to their distance correlation (Székely and Rizzo, 2009) to the response. Zhu et al. (2011) propose to use the covariance between X_j and the cumulative distribution function of Y conditioning on X_j at point Y as the quantity estimated for screening purposes. He et al. (2013) suggest a ranking procedure relying on the marginal quantile utility; Shao and Zhang (2014) introduce a ranking based on the martingale difference correlation. An extensive overview of these and other measures that can be used for variable screening can be found in Liu et al. (2015). In this work we also consider variable rankings based on measures which originally have not been developed for this purpose, e.g. regression coefficients estimated via penalised likelihood minimisation procedures such as Lasso (Tibshirani, 1996), SCAD (Fan and Li, 2001) or MC+ (Zhang, 2010).

Variable rankings are used for the purpose of so-called variable screening (Fan and Lv, 2008). The main idea behind this concept is that truly important covariates are likely to be ranked ahead of the irrelevant ones, so variable selection can be performed on the set of the top-ranked variables. Variable screening procedures attained recently considerable attention due to their simplicity, wide applicability and computational gains they offer to practitioners. Hall and Miller (2009a) suggest that variable rankings can be used for the actual variable selection. They propose to construct bootstrap confidence intervals for the position of each variable in the ranking and select covariates for which the right end of the confidence interval is lower than some cutoff, e.g. $p/2$. This principle, as its authors admit, may lead to undesirable high rate of false positives, and the choice of the ideal cutoff might be very difficult in practice, which was the case in our real data study in Section 3.7. Hall and Miller (2009b) show that various types of the bootstrap are able

to estimate the distribution of the ranks consistently, however, do not prove that their procedure is able to recover the set of the important variables.

Another approach involving subsampling is taken by Meinshausen and Bühlmann (2010), who propose Stability Selection (StabSel), a general methodology aiming to improve any variable selection procedure. In the first stage of the StabSel algorithm, a chosen variable selection technique is applied to randomly picked subsamples of the data of size $\lfloor n/2 \rfloor$. Subsequently, the variables which are most likely to be selected by the initial procedure, i.e. their selection probabilities exceed a prespecified threshold, are taken as the final estimate of the set of the important variables. An appropriate choice of the threshold leads to finite sample control of the rate of false discoveries of a certain type. Shah and Samworth (2013) propose a variant of StabSel with a further improved error control.

Our proposed method also incorporates subsampling to boost existing variable selection techniques. Conceptually, it is different from StabSel. Informally speaking, RBVS sorts covariates from the most to the least important, while StabSel treats variables as either relevant or irrelevant and equally important in either of the categories. This has far-reaching consequences. First of all, RBVS is able to *simultaneously identify subsets of covariates* appearing to be important consistently over subsamples. The same is not computationally feasible for Stability Selection, which only analyses the *marginal* distribution of the initial variable selection procedure. The bootstrap ranking approach of Hall and Miller (2009a) relies on *marginal* confidence intervals, thus it can be also regarded as a "marginal" technique. Second, RBVS does not depend on any regularity parameters or a model complexity penalty, it only requires the parameters of the incorporated subsampling procedure (naturally, these are also required by the approaches of Hall and Miller (2009a) and Meinshausen and Bühlmann (2010)). RBVS can therefore be viewed as more automatic and data adaptive than both StabSel and the approach of

Hall and Miller (2009a), which is illustrated by the data example in Section 3.7.

The key idea behind RBVS stems from the following observation: although some subsets of $\{X_1, \ldots, X_p\}$ containing irrelevant covariates may appear to have a high influence over Y, the probability that they will exhibit this spurious relationship is very small. On the other hand, truly important covariates will typically consistently appear to be related to Y, both over the entire sample and over randomly chosen subsamples. This motivates the following procedure. In the first stage, we repeatedly assess the impact of each variable on the response, with the use of a randomly picked part of the data. For each random draw, we sort the covariates in decreasing order, according to their impact on Y, obtaining a ranking of variables. In the next step, we identify the sets of variables which appear in the top of the rankings frequently and we record the corresponding frequencies. Using these, we decide how many and which variables should be selected.

RBVS is a general and widely-applicable approach; it can be used with any measure of dependence between X_j and Y, either marginal or joint, both in a parametric and non-parametric context. We do not restrict Y and X_j's to be scalar, they can be e.g. multivariate, or be curves or graphs. RBVS focuses on variable selection; we provide empirical evidence that it outperforms prediction-based approaches when the predictive model is misspecified or non-identifiable. The covariates that are highly, but spuriously related to the response are less likely to exhibit relationship to Y consistently over the subsamples than the truly relevant ones, thus our approach is "reluctant" to select irrelevant variables. Finally, the RBVS algorithm is easily parallelizable and adjustable to available computational resources, making it useful in analysis of extremely high-dimensional data sets. Its R implementation is publicly available in the R package **rbvs** (Baranowski et al., 2015).

The rest of the chapter is organised as follows. Section 3.2 describes two examples which further motivate our proposal. In Section 3.3, we define the set of important

covariates for variable rankings and introduce the RBVS algorithm. Section 3.4 reports theoretical arguments showing that RBVS is a consistent statistical procedure. In Section 3.5, we propose an iterative extension of RBVS, which aims to boost its performance in the presence of strong dependencies between the covariates. The empirical performance of RBVS is illustrated in Section 3.6 and Section 3.7 on simulated and real data. Section 3.9 discusses various computational aspects of the proposed methodology. Finally, Section 3.10 contains the proofs of our theoretical results.

3.2 Motivating examples

To further motivate our methodology, we discuss the following examples.

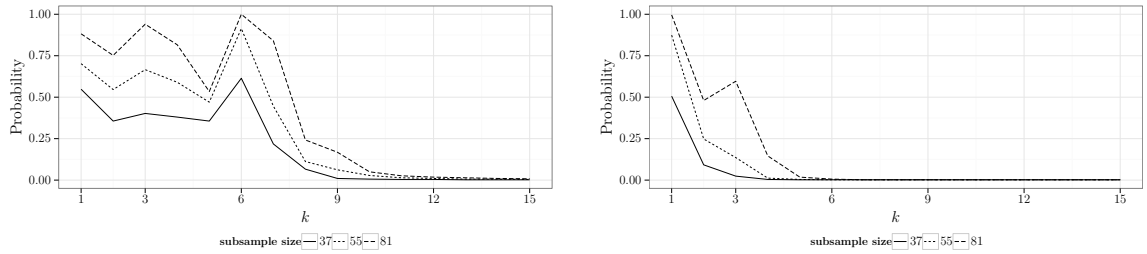
Example 3.2.1 (riboflavin production with *Bacillus subtils*, for details see Meinshausen and Bühlmann (2010))**.** The data set consists of the response variable being the logarithm of the riboflavin production rate and transformed expression levels of $p = 4088$ genes for $n = 111$ observations. The aim is to identify those genes whose mutation leads to a high concentration of riboflavin.

Example 3.2.2 (Fan and Lv (2008))**.** We consider a random sample generated from the linear model $Y_i = 5X_{i1}+5X_{i2}+5X_{i3}+\varepsilon_i$, $i = 1, \ldots, n$, where $(X_{i1}, \ldots, X_{ip}) \sim \mathcal{N}(0, \Sigma)$ and $\varepsilon_i \sim \mathcal{N}(0, 1)$ are independent, $\Sigma_{jk} = 0.75$ for $j \neq k$ and $\Sigma_{jk} = 1$ otherwise. The number of covariates $p = 4088$ and the sample size $n = 111$ are the same as in Example 3.2.1.

We consider the variable ranking defined in Definition 3.1.1, based on the sample marginal correlation coefficient in both examples. This choice is particularly reasonable in Example 3.2.2, where at the population level the Pearson correlation coefficient is the largest for X_1, X_2 and X_3 which are the only truly important ones. The linear model has been previously used to analyse the riboflavin data set (Meinshausen and Bühlmann,

2010), therefore the sample correlation may be useful in identifying important variables in Example 3.2.1 too.

Figure 3.1 demonstrates the "paths" generated by Algorithm 3.3 introduced in the next section. In both examples, the paths share common features, i.e. the estimated probability is large for the first few values of k and it declines afterwards. Interestingly, in Example 3.2.2 the curves reach levels very close to 0 shortly after $k = 3$, which is the number of the important covariates here. Crucially, the subset corresponding to $k = 3$ contains the three first covariates (X_{i1}, X_{i2}, X_{i3}), which are relevant in this example. This observation suggests that such paths as those presented in Figure 3.1 may be used to identify how many and which variables are important, hence it might be used for the purpose of variable selection.



(a) Example 3.2.1 (b) Example 3.2.2

Figure 3.1: Estimated probabilities corresponding to the k-element sets which appear to be the most highly correlated to the response, based on 500 subsamples. On the x-axis, k denotes the number of elements in a set. On the y-axis we have the estimated probability corresponding to the most frequently occurring subset of covariates of size k. The three different lines in each example correspond to a different subsample size used to generate paths details are given in Section 3.3).

3.3 Methodology of Ranking-Based Variable Selection

In this section, we introduce the Ranking-Based Variable Selection algorithm. The main purpose of RBVS is to find the set of *top-ranked* variables, which we formally define.

3.3.1 Notation

Hereafter, $|\mathcal{A}|$ stands for the number of elements in a set $\mathcal{A}$. For any $k = 0, \ldots, p$, we denote $\Omega_k = \{\mathcal{A} \subset \{1, \ldots, p\} : |\mathcal{A}| = k\}$. For every $\mathcal{A} \in \Omega_k$, $k = 1, \ldots, p$, we define the probability of its being ranked at the top by

$$\pi_n(\mathcal{A}) = \mathbb{P}\left(\{R_{n1}(\mathbf{Z}_1, \ldots, \mathbf{Z}_n), \ldots, R_{nk}(\mathbf{Z}_1, \ldots, \mathbf{Z}_n)\} = \mathcal{A}\right). \tag{3.1}$$

For $k = 0$, we set $\pi_n(\mathcal{A}) = \pi_n(\emptyset) = 1$. For any integer m satisfying $1 \leq m \leq n$, we define

$$\pi_{m,n}(\mathcal{A}) = \mathbb{P}\left(\{R_{n1}(\mathbf{Z}_1, \ldots, \mathbf{Z}_m), \ldots, R_{nk}(\mathbf{Z}_1, \ldots, \mathbf{Z}_m)\} = \mathcal{A}\right). \tag{3.2}$$

The random samples in our framework form a triangular array, hence we need to use a double subscript in the definition above.

3.3.2 Definition of a k-top-ranked and the top-ranked set

We define the set of the important variables in the context of variable rankings.

Definition 3.3.1. Any set $\mathcal{S} \in \Omega_k$, $k < p$, is said to be k-top-ranked when

$$\liminf_{n \to \infty} \pi_n(\mathcal{S}) > 0.$$

Definition 3.3.2. Set $\mathcal{A} \in \Omega_k$ is said to be top-ranked if it is k-top-ranked and a $k+1$-top-ranked set does not exist, i.e. $\liminf_{n\to\infty} \pi_n(\mathcal{A}) = 0$ for all $\mathcal{A} \in \Omega_{k+1}$. It is unique when the existence of another top-ranked set $\mathcal{A}' \in \Omega_{k'}$ implies $\mathcal{A} = \mathcal{A}'$ for all n sufficiently large.

Some remarks are in order. Firstly, Definition 3.3.1 formalises the statement that $\mathcal{A}$ appears at the top of the ranking with high probability. We use limit inferior in the definitions above as $\lim_{n\to\infty} \pi_n(\mathcal{A})$ in general does not exist. Furthermore, we consider $\liminf_{n\to\infty} \pi_n(\mathcal{S}) > 0$ in Definition 3.3.1, as in some scenarios it is strictly lower than 1. In Example 3.2.2, for instance, X_1, X_2, X_3 have equal impact on Y, hence $\liminf_{n\to\infty} \pi_n(\mathcal{A}) = 1/3$ for $\mathcal{A} = \{1\}, \{2\}, \{3\}$.

Secondly, although the top-ranked set is unique under our assumptions (see Section 3.3.3), this does not imply that other k-top-ranked sets are unique as well. In Example 3.2.2 again, we observe that $\{1\}, \{2\}, \{3\}$ are 1-top-ranked and $\{1,2\}$, $\{1,3\}$, $\{2,3\}$ are 2-top-ranked. However, the top-ranked set is unique and equal to $\{1,2,3\}$.

Finally, we have $\sum_{\mathcal{A}\in\Omega_k} \pi_n(\mathcal{A}) = 1$, hence $\max_{\mathcal{A}\in\Omega_k} \pi_n(\mathcal{A}) \geq \binom{p}{k}^{-1}$ for $k = 1, \ldots, p$. In particular, if p were bounded in n, the top-ranked set would not exist. Therefore, we restrict ourselves to the case of p diverging with n (allowing both $p \leq n$ and $p > n$). In Section 3.6 we show that RBVS works well for p both comparable to and much larger than n.

3.3.3 Top-ranked set for a class of variable rankings

The top ranked set defined in Definition 3.3.2 exists for a wide class of variable rankings, as we can learn from Proposition 3.3.1 below. Consider ω_j, $j = 1, \ldots, p$, a measure of the contribution of each X_j to the response, depending on the distribution of $\mathbf{Z} = (Y, X_1, \ldots, X_p)$ (thereby on n, as p changes with n). For ease of notation, assume $\omega_1 \geq \omega_2 \geq \ldots \geq \omega_p \geq 0$. Let $\hat{\omega}_j = \hat{\omega}_j(\mathbf{Z}_1, \ldots, \mathbf{Z}_n)$ be an estimator of ω_j. We make the

following assumptions.

(C1) For some $\vartheta \geq 0$ and any $c_\vartheta > 0$ we have $\sup_{j=1,\ldots,p} \mathbb{P}\left(|\hat{\omega}_j - \omega_j| \geq c_\vartheta n^{-\vartheta}\right) \leq C_\vartheta \exp\left(-n^\gamma\right)$, where constants $C_\vartheta, \gamma > 0$ do not depend on n.

(C2) There exists $l = 0, 1, \ldots, p$ such that for each $a > l$, there are many covariates which have the same impact on Y as X_a. More precisely, for each $a > l$ there exists $\mathcal{M}_a \subset \{l+1, \ldots, p\}$, such that $a \in \mathcal{M}_a$, the distribution of $\hat{\omega}_j, j \in \mathcal{M}_a$, is exchangeable and $|\mathcal{M}_a| \underset{n}{\to} \infty$.

(C3) Let s be the smallest non-negative integer l satisfying (C2). Assume that s is bounded in n. There exists $\eta \leq \vartheta$, where ϑ is as in (C1), and $c_\eta > 0$ such that $\min_{j=1,\ldots,s} \omega_j - \max_{j>s} \omega_j \geq c_\eta n^{-\eta}$ uniformly in n.

(C4) The number of covariates $p \leq C_1 \exp\left(n^{b_1}\right)$, where $0 < b_1 < \gamma$ and γ is as in (C1).

Condition (C1) holds for a wide range of measures. The sample correlation coefficient satisfies (C1) when the data follow a multivariate normal distribution (Kalisch and Bühlmann (2007), Lemma 1), or when $Y, X_1, \ldots, X_p$ are uniformly bounded (Delaigle and Hall (2012), proof of Theorem 1). Li et al. (2012a) in their Theorem 2 demonstrate that Kendall's τ meets (C1) under the Marginally symmetric condition and Multi-modal condition. Distance Correlation satisfies (C1) under regularity assumptions on the tails of distribution of X_j's and Y (Li et al. (2012b), Theorem 1). The Lasso estimates of the regression coefficients in the linear model meet (C1) if the covariates satisfy the sparse Riesz condition and the regression coefficients are sparse (Zhang and Huang (2008), Theorem 3).

In condition (C3), we assume that there is a gap between ω_s and ω_{s+1}, which means that ω_j separates the first s variables (which we believe to be relevant ones) from the remaining ones. The gap η is allowed to decrease slowly to zero. Furthermore, we assume s is bounded in n, which combined with diverging p implies that the number of truly

important covariates is very small (this can be seen as a variant of the so-called "sparsity" assumption, more on which later in this section). Conditions (C1) and (C3) together imply that the ranking based on on $\hat{\omega}_j$ has the sure independence screening property (Fan and Lv, 2008).

We note that Meinshausen and Bühlmann (2010) use the exchangeability assumption on the selection of noise variables. However, it concerns a variable selection procedure, while we impose restrictions on the measure $\hat{\omega}_j$. The main difference between their assumption and (C2) is that they require all covariates to be equally likely to be selected, while we allow for many groups within which each variable has the same impact on Y.

Condition (C2) can be linked to the sparsity assumption which requires that only a few covariates have a significant impact on the response. In our framework, these are $X_1, \ldots, X_s$. For the remaining covariates, the sparsity may require, for example, that the regression coefficients corresponding to them are zero. In (C2) each X_a, $a > s$, may contribute to Y, but, speaking heuristically, it is difficult to select X_a, $a > s$ with the largest contribution, as many covariates have the same impact on Y. We believe that this assumption is likely to be met at least approximately (in the sense that large groups of covariates exhibit similar small impact on the response), especially for large dimensions p. Condition (C4) restricts the maximum number of covariates, but it allows high-dimensional settings where the number of covariates grows exponentially with n.

Proposition 3.3.1. *Let $\mathbf{R}_n$ be a variable ranking based on $\hat{\omega}_j$, $j = 1, \ldots, p$, given in (3.1.1). Under conditions (C1)-(C4), the unique top ranked set defined in Definition 3.3.2 exists and equals $\mathcal{S} = \{1, \ldots, s\}$, where s is as in (C3).*

Proposition 3.3.1 can be applied to establish a link between the top-ranked set and the set of the important variables understood in a classic way. Consider the following linear regression model $Y = \sum_{j=1}^{p} \beta_j X_j + \varepsilon$, where β_j's are unknown regression coefficients, X_j's - random predictors and ε is an error term. In this model, the top-ranked set can coincide

with $\{k : \beta_k \neq 0\}$. To observe that, we consider the variable ranking given by (3.1.1) based on $\hat{\omega}_j = \widehat{\mathrm{Cor}}\,(Y, X_j)$, which satisfies (C1) when $(Y, X_1, \ldots, X_p)$ is e.g. Gaussian (Kalisch and Bühlmann, 2007). Condition (C2) is met when e.g. $\widehat{\mathrm{Cor}}\,(Y, X_j) = \rho$ for some $\rho \in (-1, 1)$ and all j such that $\beta_j = 0$, and $p \underset{n}{\to} \infty$. Imposing some restrictions on the correlations between the covariates, we also guarantee that (C3) holds. From Proposition 3.3.1, $\{k : \beta_k \neq 0\}$ is then the top-ranked set, provided that $p \underset{n}{\to} \infty$ no faster than in (C4).

3.3.4 Ranking-Based Variable Selection

Assume the top-ranked set $\mathcal{S}$ exists and is uniquely determined and denote by $s = |\mathcal{S}|$ its size. To construct an estimate of $\mathcal{S}$, we introduce the estimators of $\pi_{m,n}(\mathcal{A})$ defined by (3.2) using a variant of the m-out-of-n bootstrap (Bickel et al., 2012).

Definition 3.3.3. Let $B = 1, 2, \ldots$, $m = 1, \ldots, n$ and set $r = \lfloor n/m \rfloor$. For any $b = 1, \ldots, B$, let $I_{b1}, \ldots, I_{br}$ be mutually exclusive subsets of $\{1, \ldots, n\}$ of size m, drawn uniformly from $\{1, \ldots, n\}$ without replacement. Assume that the sets of subsamples are independently drawn for each b. For any $\mathcal{A} \in \Omega_k$, we estimate $\pi_{m,n}(\mathcal{A})$ by the fraction of subsamples in which $\mathcal{A}$ appeared at the top of the ranking, i.e. $\hat{\pi}_{m,n}(\mathcal{A}) = B^{-1} \sum_{b=1}^{B} r^{-1} \sum_{j=1}^{r} \mathbb{I}\left(\mathcal{A} = \mathbf{R}_{n,1:k}\left(\mathbf{Z}_i, i \in \mathrm{I}_{bj}\right)\right)$.

In general $\pi_{m,n}(\mathcal{A})$ can be different than $\pi_n(\mathcal{A})$, however, we later show that $\pi_{m,n}(\mathcal{A})$ and $\pi_n(\mathcal{A})$ are large for the same subsets, provided that m is not too small. This combined with some bounds on the estimation accuracy of $\hat{\pi}_{m,n}(\mathcal{A})$ will imply that $\hat{\pi}_{m,n}(\mathcal{A})$ can be used to find the top-ranked set from the data. In practice the number of elements in $\mathcal{S}$ is typically unknown, thus we need to consider subsets of any size in our estimation procedure. Under assumptions given in Section 3.4, $\pi_{m,n}(A)$ and $\pi_n(A)$ are large for the

same subsets, thus for n sufficiently large, $\mathcal{S}$ will be one of the following sets:

$$\mathcal{A}_{k,m} = \text{argmax}_{\mathcal{A}\in\Omega_k}\, \pi_{m,n}(\mathcal{A}),\ k = 0, 1, \ldots, p. \tag{3.3}$$

We define the sample counterparts of $\mathcal{A}_{k,m}$ as

$$\hat{\mathcal{A}}_{k,m} = \text{argmax}_{\mathcal{A}\in\Omega_k}\, \hat{\pi}_{m,n}(\mathcal{A}). \tag{3.4}$$

At this point we can better understand the importance of the parameter B introduced in Definition 3.3.3. Note that $\max_{\mathcal{A}\in\Omega_k} \hat{\pi}_{m,n}(\mathcal{A}) \geq (Br)^{-1}$. For moderate sample sizes, r may not be large, while we expect the majority of $\pi_{m,n}(\mathcal{A})$'s to be small, even smaller than $1/r$. In this situation, the bias of $\max_{\mathcal{A}\in\Omega_k} \hat{\pi}_{m,n}(\mathcal{A})$ with $B = 1$ is expected to be high and estimate of $\hat{\mathcal{A}}_{k,m}$ inaccurate. A moderate value of B brings $\hat{\mathcal{A}}_{k,m}$ closer to its population counterpart $\mathcal{A}_{k,m}$. The theoretical requirements on B are given in Section 3.4; our guidance for the choice of B in practice is provided in Section 3.6.2.

Under appropriate assumptions, $\hat{\mathcal{A}}_{s,m}$ equals $\mathcal{S}$ with high probability, as shown in Section 3.4. In practice, we do not know s and it should be estimated as well. One possibility is to apply hard thresholding rule and set $\hat{s}_\zeta = \min\left\{k :\ \hat{\pi}_{m,n}\left(\hat{\mathcal{A}}_{k+1,m}\right) \leq \zeta\right\}$, where $\zeta > 0$ is a prespecified threshold. This approach could be justified by the existence of the asymptotic gap between $\pi_{m,n}(\mathcal{A}_{s+1,m})$ and $\pi_{m,n}(\mathcal{A}_{s,m})$. However, the magnitude of this difference is typically unknown and can be rather small, which makes the choice of ζ difficult. As an alternative, we propose to estimate s by

$$\hat{s} = \text{argmin}_{k=0,\ldots,p-1} \frac{\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k+1,m})}{\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k,m})}, \tag{3.5}$$

which is the k where $\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k+1,m})$ declines the most drastically. This estimator has the advantage of not requiring any parameters. In Section 3.4, it is also shown to be

consistent.

3.3.5 The Ranking-Based Variable Selection algorithm

The RBVS algorithm consists of the four main steps and, described by the pseudocode, it is defined in Algorithm 3.3. In Step 1 below, we draw subsamples from the data, using the subsampling scheme introduced in Definition 3.1.1. Subsequently in Step 2, for each subsample drawn we calculate the estimates of ω_j's based on the subsamples I_{bl}, and sort the sample measures $\{\hat{\omega}_j(\mathbf{Z}_i \in I_{bl})\}_{j=1}^{p}$ in non-increasing order to find $\mathbf{R}_n(\mathbf{Z}_i \in I_{bl})$ defined in Definition 3.1.1. Having computed the variable rankings, we proceed to Step 3, where for each $k = 1, \ldots, k_{max}$ we find $\hat{\mathcal{A}}_{k,m}$, the k-element set the most frequently occurring in the top of $\mathbf{R}_n(\mathbf{Z}_i \in I_{bl})$, $b = 1, \ldots, B$, $l = 1, \ldots, r$. Finally, in Step 4, probabilities $\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k,m})$ are used to find $\hat{s}$, the estimate of the size of the top-ranked set and $\hat{\mathcal{S}} = \hat{\mathcal{A}}_{\hat{s},m}$ is returned as the final estimate of $\mathcal{S}$. Note that in the algorithm we consider $\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k,m})$ for $k \leq k_{max}$. This decreases the computational burden involved in Step 4 (see Section 3.9.1).

Algorithm 3.3 Ranking-Based Variable Selection

Input: Random sample $\mathbf{Z}_i = (Y_i, X_{i1}, \ldots, X_{ip})$, $i = 1, \ldots, n$, subsample size m s.t. $1 \leq m \leq n$, positive integers k_{max}, B.

Output: The estimate of the set of important variables $\hat{\mathcal{S}}$.

procedure RBVS($\mathbf{Z}_1, \ldots, \mathbf{Z}_n, m, B, k_{max}$)

Step 1 Let $r = \lfloor n/m \rfloor$. For each $b = 1, \ldots, B$, draw uniformly without replacement m-element subsets $I_{b1}, \ldots, I_{br} \subset \{1, \ldots, n\}$.

Step 2 Calculate $\hat{\omega}_j(\mathbf{Z}_i \in I_{bl})$ and the corresponding variable ranking $\mathbf{R}_n(\mathbf{Z}_i \in I_{bl})$ for all $b = 1, \ldots, B$, $l = 1, \ldots, r$ and $j = 1, \ldots, p$.

Step 3 For $k = 1, \ldots, k_{max}$ find $\hat{\mathcal{A}}_{k,m}$ given by (3.4).

Step 4 Find $\hat{s} = \operatorname{argmin}_{k=0,\ldots,k_{max}-1} \frac{\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k+1,m})}{\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k,m})}$ and return $\hat{\mathcal{S}} = \hat{\mathcal{A}}_{\hat{s},m}$.

end procedure

3.3.6 Relations to existing methodology

In this section, we provide a brief overview of the differences between RBVS, StabSel and the bootstrap ranking approach of Hall and Miller (2009a).

3.3.6.1 Stability selection

Let us denote the selection probabilities $\pi_j = \mathbb{P}\left(j \in \hat{S}^\lambda\right)$, $j = 1, \ldots, p$, where $\hat{S}^\lambda$ is the set of variables selected by a chosen variable selection technique with its tuning parameter set to λ. The aim of StabSel is twofold: first, to select covariates that the initial procedure selects with a high probability, second, to bound the average number of false positives (denoted by $\mathrm{E}\,V$) below some prespecified level $\alpha > 0$. For this purpose, Meinshausen and Bühlmann estimate π_j's and select variables for which $\hat{\pi}_j > \pi$, where $\pi \in (2^{-1}, 1)$ is a prespecified threshold. To control $\mathrm{E}\,V$, one can set λ such that $|\hat{S}^\lambda| \leq q$, where $q \in \{1, \ldots, p\}$ depends on π and α and is adjusted to ensure $\mathrm{E}\,V \leq \alpha$. The exact formula for q and other possible ways of controlling $\mathrm{E}\,V$ are given in Meinshausen and Bühlmann (2010).

In contrast to StabSel, which needs a variable selection procedure, RBVS selects variables based on a variable ranking, which implies another difference. Namely, in our approach we consider joint probabilities $\pi_{m,n}(\mathcal{A})$, while in StabSel only marginal probabilities are used. The estimates of the joint probabilities can be used to determine the number of important covariates at the top of the variable ranking, not requiring the specification of any thresholding parameters, as we demonstrate in Section 3.4. Consequently, RBVS can be viewed as more automatic and "less marginal" than StabSel.

3.3.6.2 The bootstrapped rankings of Hall and Miller (2009a)

Let r_{nj} be the position of the jth covariate in the variable ranking $\mathbf{R}_n = (R_{n1}, \ldots, R_{np})$. Formally, $r_{nj} = l$ if and only if when $R_{nl} = j$. To identify important covariates based on

$\mathbf{R}_n$, Hall and Miller (2009a) compute $[r^-_{nj}, r^+_{nj}]$, two-sided, equal tiled, percentile-method bootstrap confidence intervals for r_{nj} at a significance level α. A variable is considered to be influential when r^+_{nj} is lower than some prespecified cutoff level c, for instance $c = p/2$. The number of variables selected by the procedure of Hall and Miller (2009a) depends therefore on α and c and "marginal" confidence intervals $[r^-_{nj}, r^+_{nj}]$. By contrast, RBVS is based on the joint probabilities $\hat{p}_{m,n}(\mathcal{A})$ and does not require the specification of any tuning parameters.

3.3.6.3 Computational complexity of the related methods

Let us denote by $c(n, p)$ the computational cost of evaluating $\hat{\omega}_j$ for all $j = 1, \ldots, p$ using n observations. Subsampling takes $O(Bn)$ operations. Finding all $\hat{\omega}_j$'s takes $c(m, p) \times Br$ manipulations. Next, evaluating rankings takes $O((p + k_{max} \log(k_{max}))Br)$ operations. Step 3 can be performed in $O(Brk^2_{max})$ basic operations. The final step requires $O(Brk_{max})$ operations, hence the computational complexity of Algorithm 3.3 is $c(m, p) \times Br + O(\max\{p, k^2_{max}\}Br)$. For our recommended choice of k_{max} and m see Section 3.6.2.

Table 3.1 summarises computational complexity of 3.3 and its competitors: SIS (Fan and Lv, 2008) and StabSel (Meinshausen and Bühlmann, 2010). For reference, we include the computational complexity of the k-fold cross-validation (k-fold CV), which is frequently used to find optimal parameters for e.g. Lasso, MC+ or SIS. The computational complexity of the method proposed by Hall and Miller (2009a) is comparable to StabSel, hence omitted in this comparison. In theory, SIS requires the least computational resources, especially in the case of $p >> n$. Simple k-fold cross-validation has the second lowest computational complexity. StabSel in the case of $n > \sqrt{p}$ is theoretically quicker than RBVS, however, the common factor $B \times c\,(n/2, p)$ typically dominates both $O(Bp)$ and $O(\max\{p, n^2\})$, therefore StabSel and RBVS usually take similar amount of

computational resources.

k-fold CV	SIS	StabSel	RBSS
$k \times c\left(\frac{(k-1)n}{k}, p\right)$	$O(np) + k \times c\left(\frac{(k-1)n}{k}, \frac{n}{\log(n)}\right)$	$B \times c\left(\frac{n}{2}, p\right) + O(Bp)$	$B \times c(\frac{n}{2}, p) + O(\max\{n^2, p\}B)$

Table 3.1: Computational complexity of Algorithm 3.3 and its competitors. The cost of the base learner in relation to the sample size n and the number of variables p is denoted by $c(n, p)$; B is the number of subsamples used in StabSel and RBVS. Parameters for SIS, StabSel, RBVS are set to the recommended values. For SIS, we assume that k-fold CV is used after the screening step.

3.4 Theoretical results

Under the theoretical framework below, we prove that Algorithm 3.3 recovers the top-ranked given by Definition 3.3.2 with probability tending to 1 when $n \to \infty$. As in Section 3.3, we consider a variable ranking based on measure $\hat{\omega}_j$, $j = 1, \ldots, p$, and, w.l.o.g., assume that its population counterpart satisfies $\omega_1 \geq \ldots \geq \omega_p \geq 0$. We make the following assumptions.

(A1) For some $\vartheta \geq 0$ and any $c > 0$ we have

$$\sup_{j=1,\ldots,p} \mathbb{P}\left(|\hat{\omega}_j(\mathbf{Z}_1, \ldots, \mathbf{Z}_m) - \omega_j| \geq c_\vartheta m^{-\vartheta}\right) \leq C_\vartheta \exp\left(-m^\gamma\right),$$

where constants $C_\vartheta, \gamma > 0$ and m are specified in (A5).

(A2) There exists $l = 0, 1, \ldots, p$ s.t. for each $a > l$ there exists $\mathcal{M}_a \subset \{l+1, \ldots, p\}$ s.t. the distribution of $\hat{\omega}_j(\mathbf{Z}_1, \ldots, \mathbf{Z}_m), j \in \mathcal{M}_a \cup \{a\}$, is exchangeable, where m is as in (A5).

(A3) Let s be the smallest non-negative integer l satisfying (A2). Assume that s is bounded in n. There exists $\eta \leq \vartheta$, where ϑ is as in (A1), and $c_\eta > 0$ s.t. $\min_{j=1,\ldots,s} \omega_j - \max_{j>s} \omega_j \geq c_\eta m^{-\eta}$ uniformly in n, where m is as in (A5).

(A4) There exist constants $C_1 > 0$ and $0 < b_1 < \gamma$ with γ as in (A1) s.t. $p \leq C_1 \exp\left(n^{b_1}\right)$.

(A5) The subsample size m goes to infinity at rate n^{b_2}, with $0 < b_2 < 1$ and $\gamma b_2 - b_1 > 0$, where γ is as in (A1) and b_1 as in (A4).

(A6) Subsets $\mathcal{M}_a$ defined in (A2) satisfy $\min_{a>s} |\mathcal{M}_a| \geq C_3 n^{b_3}$ with s as in (A3), $b_3 > 2(1 - b_2)$, b_2 from (A5) and C_3 not depending on n.

(A7) The number of random draws B is bounded in n, but $B \geq 3$ and $B^{\alpha-2/3} > \max_{k=1,\ldots,s} \binom{s}{k}$, for some $\alpha \in (2/3, 1)$ and s as in (A3).

(A8) The maximum subset size $k_{max} \leq C_4 n^{b_4}$ with a constant $C_4 > 0$ and b_4 satisfying $b_3 - b_4 > 2\alpha(1 - b_2)$, where b_2, b_3 and α are as in (A5), (A6) and (A7), respectively.

Assumptions (A4), (A2) and (A3) can be seen as natural extensions of (C1), (C2) and (C3) respectively, to the case when $\hat{\omega}_j$'s are evaluated with m out of n observations only. Similarly, both (A4) and (C4) limit the growth of the number of covariates, but p may be exponentially larger than n. Note that (A3) and (A4) are almost exactly the same as (C3) and (C4), but formally need to be repeated here, because they involve theoretically different constants; (A2) and (A6) combined together imply (C2).

Assumption (A5) establishes the required size of the subsample size m. It implies that both $n/m \underset{n}{\to} \infty$ and $m \underset{n}{\to} \infty$. Such condition is common in literature on bootstrap resampling and U-statistics, see for instance Bickel et al. (2012), Götze and Račkauskas (2001) or Hall and Miller (2009b). The lower bound on B given in (A7) is needed only in the case when some of relevant variables are equally important. Assumption (A6) imposes a lower bound on the number of covariates which have the same impact on Y. Combined with (A8), is is needed to justify that the sets of irrelevant covariates have sufficiently small empirical probabilities $\hat{\pi}_{m,n}(\mathcal{A})$. Note that (A6) imposes a lower bound on p ($p \geq C_3 n^{b_3}$).

Assumptions (A1) combined with (A3) and the technical conditions on the size of p imply that, with large probability, the set of covariates with the largest $\hat{\omega}_j$ coincides with the covariates with large ω_j at the population level. In practice, however, we do not know how many variables should be considered as important, nor do we know the threshold separating large $\hat{\omega}_j$'s from the small ones. Moreover, irrelevant covariates can spuriously exhibit large empirical impact on the response, especially when $p >> n$. The resampling based set probability estimation is necessary in order to discover variables which non-spuriously appear at the top of the analysed rankings. The following theorem establishes the consistency of the RBVS methodology; for the proof see Section 3.10.2.

Theorem 3.4.1. *Suppose assumptions (A1)-(A8) hold. Then* $\hat{\mathcal{S}} = \hat{\mathcal{A}}_{\hat{s},m}$, *where* $\hat{\mathcal{A}}_{\hat{s},m}$ *is given by* (3.4) *and* (3.5), *satisfies* $\mathbb{P}\left(\hat{\mathcal{S}} = \mathcal{S}\right) = 1 - o\left(n^{2\alpha b_2} \exp\left(-n^{(1-\alpha)b_2}\right)\right) \underset{n}{\to} 1$. $\hat{\mathcal{S}}$ *is therefore a consistent estimator of the top-ranked set* $\mathcal{S}$.

3.5 Iterative extension of RBVS

In the presence of strong dependence between covariates, measure $\hat{\omega}_j$ may fail to detect some important variables. For instance, a covariate may be jointly related but marginally unrelated to the response (see Fan and Lv (2008) or Barut (2013)). Under such a setting, the top-ranked set given in Definition 3.3.2 may contain just some of the important variables. To overcome this problem, we propose IRBVS, an iterative extension of Algorithm 3.3. Again using pseudocode, we describe IRBVS in Algorithm 3.4. In each iteration, IRBVS removes the linear effect on the response of the variables found at the previous iteration, it is therefore applicable when the relationship between Y and X_j's is at least approximately linear. It is possible to further extend our methodology; e.g. Barut (2013) demonstrates how to remove the impact of a given set of covariates on the response in Generalised Linear Models.

Algorithm 3.4 Iterative Ranking-Based Variable Selection

Input: Random sample $\mathbf{Z}_i = (Y_i, X_{i1}, \ldots, X_{ip})$, $i = 1, \ldots, n$, subsample size m s.t. $1 \leq m \leq n$, positive integers k_{max}, B.

Output: The estimate of the set of important variables $\hat{\mathcal{S}}$.

procedure IRBVS($\mathbf{Z}_1, \ldots, \mathbf{Z}_n, m, B, k_{max}$)

Initialise $\hat{\mathcal{S}} = \emptyset$.

repeat

Step 1 Find Y_i^*, X_{ij}^*, the residuals left after projecting Y, X_j onto the space spanned by the covariates with indices in $\hat{\mathcal{S}}$ and set $\mathbf{Z}_i^* = (Y_i^*, X_{ij}^*, j \in \{1, \ldots, p\} \setminus \hat{\mathcal{S}})$, $i = 1, \ldots, n$.

Step 2 Calculate $\hat{\mathcal{S}}^* = \text{RBVS}(\mathbf{Z}_1^*, \ldots, \mathbf{Z}_n^*, m, B, k_{max})$.

Step 3 Set $\hat{\mathcal{S}} := \hat{\mathcal{S}}^* \cup \hat{\mathcal{S}}$.

until $\hat{\mathcal{S}}^* \neq \emptyset$**; return** $\hat{\mathcal{S}}$.

end procedure

We note that iterative extensions of variable screening methodologies are frequently proposed in the literature, see for instance Fan and Lv (2008), Zhu et al. (2011) or Li et al. (2012a). A practical advantage of the IRBVS algorithm over its competitors is that, it does not require the specification of the number of variables added at each iteration or the total number of iterations. Moreover, IRBVS appears to offer better empirical performance than other iterative methods such as ISIS (Fan and Lv, 2008); see Section 3.6.

3.6 Simulation study

3.6.1 Simulation methods

We illustrate the performance of the RBVS and IRBVS algorithms on simulated data following models given in Section 3.6.3. In the first three models, which are linear, we apply RBVS with the absolute values of the following measures: Pearson correlation coefficient (PC) , the regression coefficients estimated via Lasso (Tibshirani, 1996), the regression coefficients estimated via MC+ algorithm (Zhang (2010)). Corresponding methods are termed, respectively, RBVS PC, RBVS Lasso and RBVS MC+. In **Model**

(E) Y is binary, in which example we consider rankings based on the distance correlation (DC, Li et al. (2012b)) aiming to capture any kind of dependence, leading to RBVS DC. Techniques using Algorithm 3.4 are termed IRBVS PC, IRBVS Lasso, IRBVS MC+ and IRBVS DC.

Recall that Lasso and MC+ estimators are defined as

$$\hat{\beta}_{pen} = \text{argmin}_{\beta} \left((n^{-1} \sum_{i=1}^{n} \left(Y_i - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 + \sum_{j=1}^{p} \text{pen}(|\beta_j|) \right),$$

where $\text{pen}(t) = \lambda t$ for Lasso, $\text{pen}(t) = \lambda \int_0^t \max\{0, (1 - x/(\gamma\lambda))\}\, dx$ for MC+ and $\lambda, \gamma > 0$ are tuning parameters. In StabSel, we set the tuning parameters such that $q \in \{1, \ldots, p\}$ among the estimated coefficients are non-zero, as per the recommendation of Meinshausen and Bühlmann (2010). To provide a fair comparison, we select λ for RBVS Lasso and RBVS MC+ in the exactly same way as for StabSel setting $q = \sqrt{(2\pi - 1)EVp}$, where $\pi = 0.6$ and $EV = 2.5$; $\gamma = 3$ for MC+ as in Breheny and Huang (2011). From our experience, the value of q has a limited impact on the performance of Algorithm 3.3, unless it is too small, i.e. smaller than the number of the important covariates. As in RBVS, StabSel is applied with PC, Lasso and MC+.

We also apply standard Lasso and MC+ algorithms. The theoretically optimal parameters for both methods did not perform well in our simulations, thus we use 10-fold cross-validation to choose λ. The final group of the techniques included in our comparison consists of SIS and its iterative extension ISIS ((Fan and Lv, 2008)). Standard ISIS procedure did not perform well in our experiments (it was selecting a very large number of false positives), therefore we apply a modified version of ISIS which involves certain randomisation mechanism (Saldana and Feng (2014)). We use implementations of the Lasso and MC+ algorithms from the R package **ncvreg** (Breheny and Huang, 2011). For SIS based methods we use the R package **SIS** (Saldana and Feng, 2014). When it is

relevant, we estimate the regression coefficients using OLS for the variables selected by each of the chosen variable selection techniques.

3.6.2 Choice of parameters of the RBVS algorithm

RBVS involves the choice of three parameters, namely B, m and k_{max}. The B parameter has been introduced to decrease the randomness of the method. Naturally, the larger the value of B, the less the algorithm depends on a particular random draw. Assumption (A7) requires B to be sufficiently large and bounded in n. However, the lower bound given in (A7) depends on unknown constants. From the proof of Theorem 3.4.1, we learn that if B is too small, the estimator $\hat{s}$ given by (3.5) may underestimate s, in the case when $X_1, \ldots, X_s$ have exactly the same impact on Y. If this is not the case, the lower bound on B in (A7) is too conservative. Our recommendation is to take a moderate value of B from 100 to 500.

The problem of the choice of the subsample size m is more challenging. In Section 3.4, we require $m \to \infty$ at an appropriate rate, which is, however, unknown. In the finite-sample case m cannot be too small, as it is unlikely that $\mathbf{R}_n$ based on a small sample could give a high priority to the important variables. On the other hand, when m is too large (i.e. close to n), subsamples largely overlap. In practical problems, we propose to choose $m = \lfloor n/2 \rfloor$ and our simulation studies (Section 3.8) confirm that this choice results in good finite-sample properties of the RBVS-based methods.

From our experience, the value of k_{max} has a negligible impact on the outcome of RBVS, as long it is not extremely small. In all simulations conducted, $\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k,m})$ given by (3.4) reaches and stays at the level of $(Br)^{-1}$ for some $k \leq n$, so we recommend $k_{max} = \min\{n, p\}$.

3.6.3 Simulation models

We study the following simulation models.

Model (A) Taken from Fan and Lv (2008): $Y_i = 5X_{i1} + 5X_{i2} + 5X_{i3} + \varepsilon_i$, where $(X_{i1}, \ldots, X_{ip})$ are i.i.d. observations from $\mathcal{N}(0, \Sigma)$ distribution and ε_i follow $\mathcal{N}(0, 1)$ distribution. The covariance matrix satisfies $\Sigma_{ii} = 1$, $i, 1, \ldots, p$, $\Sigma_{ij} = \rho$, $|\rho| < 1$ for $i \neq j$. This is a relatively easy setting, where all important X_j's are "visible" to any reasonable marginal approach as they are the most highly correlated to Y at the population level.

Model (B) Taken from Fan and Lv (2008):

$$Y_i = 5X_{i1} + 5X_{i2} + 5X_{i3} - 15\sqrt{\rho}X_{i4} + \varepsilon_i, \tag{3.6}$$

where $(X_{i1}, \ldots, X_{ip})$ are i.i.d. observations from $\mathcal{N}(0, \Sigma)$ and ε_i follow $\mathcal{N}(0, 1)$ distribution. The covariance Σ is as in **Model (A)**, except $\Sigma_{4,k} = \Sigma_{j,4} = \sqrt{\rho}$. The challenge of this model is that X_{i4} has a large contribution to Y but it is marginally unrelated to the response.

Model (C) Factor model with two factors, taken from Meinshausen and Bühlmann (2010):

$$Y_i = \beta_1 X_{i1} + \ldots + \beta_p X_{ip} + \varepsilon_i, \tag{3.7}$$

where $X_{ij} = f_{ij}\phi_i + h_{ij}\psi_i + \theta_{ij}$ and $f_{ij}, \phi_i, h_{ij}, \psi_i, \theta_{ij}$, non-zero β_j's are i.i.d. $\mathcal{N}(0, 1)$. The number of $\beta_j \neq 0$ is set to $s = 5, 10$, and their indices are drawn uniformly without replacement. In this model some of the non-zero regression coefficients are potentially very small, thus the corresponding covariates are difficult to detect.

Model (D) Taken from Hall and Miller (2009a):

$$Y_i = X_{i1} - X_{i2} + \varepsilon_i, \tag{3.8}$$

where $X_{i1} = X_{i3} + X_{i4}$, $X_{i2} = X_{i3} + X_{i5}$, and $X_{i3}, \ldots, X_{ip}, \varepsilon_i$ are i.i.d. $\mathcal{N}(0,1)$. Uncountably many combinations of the first 5 covariates have the same explanatory power, so the model is not identifiable.

Model (E) Logistic regression model taken from Hall and Xue (2014):

$$\log \frac{q_i}{1-q_i} = -2.5 + \sum_{j=1}^{3} \frac{4-j}{3} \left\{ X_{ij} + X_{i,j+3} + \sin(X_{ij}) + e^{X_{i,j+3}} \right\}, \tag{3.9}$$

where $Y_i \in \{0,1\}$ follows a Bernoulli distribution with $q_i = \mathbb{P}(Y_i = 1 | X_{i1}, \ldots, X_{ip})$ and $(X_{i1}, \ldots, X_{ip})$ i.i.d. $\mathcal{N}(0, \Sigma)$ with $\Sigma_{ii} = 1$, $\Sigma_{ij+3} = 0.85$ for $j = 1, 2, 3$, $\Sigma_{ij} = 0$ otherwise. The dependence between the response and the important covariates $X_{i1}, \ldots, X_{i6}$ is highly non-linear.

3.6.4 Comments on the results

Tables 3.2–3.6 below contain the results. In **Model (A)**, all methods but RBVS PC almost always successfully recover the set of the important variables. StabSel, ISIS based methods and RBVS significantly reduce the average FP. Interestingly, RBVS Lasso and methods using Algorithm 3.4 perform better for higher dimensions p, which can be concluded from the values of FP+FN. In general, all techniques but CV and SIS offer very good performance.

Prediction based approaches (Lasso and MC+ with cross-validation) perform poorly in **Model (B)** when p is large. In this case, both Lasso and MC+ frequently miss one covariate. Even when X_1, X_2, X_3 and X_4 are detected, those techniques include a lot of irrelevant covariates. RBVS PC cannot detect X_4 which is not marginally related to Y.

IRBVS and ISIS, deal with this difficulty well and mostly selects all relevant variables, however, IRBVS based methods achieve lower average rate of false positives.

MC+ offers the best estimates of β_j's in **Model (C)**, however, IRBVS based methods perform similarly in this aspect selecting many fewer false positives than either Lasso or MC+. StabSel and RBVS based techniques fail to detect some of the important variables, however, RBVS Lasso is better than StabSel Lasso when either p is large, or $s = 10$. Finally, IRBVS PC, IRBVS Lasso and IRBVS MC+ perform similarly, suggesting that in this scenario IRBVS is robust against the choice of measure used for variable ranking.

The approach based on the marginal correlation proves to be the most effective in variable selection, when correlations between covariates are extremely strong, as we can learn from Table 3.5. In **Model (D)**, either RBVS PC or IRBVS PC achieves the best error control when p is large.

In **Model (E)**, IRBVS again proves to be the most effective variable selection technique, even though the linear model is not correct here. Moreover, we observe again that StabSel performs very well for small p, but it is significantly outperformed by IRBVS when p is greater than 100. Finally, the choice of the measure of association between Y and X_j have little impact on the quality of variable selection, yet PC yields the lowest FP+FN.

Overall, variable selection techniques incorporating the RBVS algorithm perform well, especially when p is much larger than n. Its iterative extension, IRBVS, in many cases is able to detect variables overlooked by pure RBVS and other techniques. A particular practical advantage of the IRBVS algorithm is that, unlike other iterative variable selection techniques such as ISIS (Fan and Lv, 2008) or IRRCS (Li et al., 2012a), it is fully automatic.

The performance of IRBVS is relatively robust against the choice of the measure used in the procedure. Therefore we recommend to adjust this choice to the available

computational resources and the size of the data. For large data sets ($p > 10000$, $n > 500$), we recommend using IRBVS PC, which is extremely quick to compute with the R package **rbvs** and achieves either the best or close to the best FP+FN in each example. In the case of moderate data sizes, penalised likelihood methods typically offer slightly better performance.

	Oracle	CV		SIS		StabSel			RBVS			ISIS		IRBVS		
		Lasso	MC+	Lasso	MC+	PC	Lasso	MC+	PC	Lasso	MC+	Lasso	MC+	PC	Lasso	MC+
$n = 100$ $p = 100$ $\rho = 0$																
FP	.00	2.00	.01	9.88	.03	.00	.01	.00	.01	.02	.02	2.06	3.01	.04	.04	.03
FN	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
FP+FN	.00	2.00	.01	9.88	.03	.00	.01	**.00**	.01	.02	.02	2.06	3.01	.04	.04	.03
ℓ_2	.18	.36	**.18**	.59	**.18**	**.18**	**.18**	**.18**	**.18**	**.18**	**.18**	.22	.36	**.18**	**.18**	**.18**
time	.00	.15	.25	.16	.44	.07	.82	.81	.07	.82	.81	.89	1.34	.12	1.25	1.24
$n = 100$ $p = 100$ $\rho = 0.75$																
FP	.00	5.00	.00	4.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
FN	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
FP+FN	.00	5.00	**.00**	4.00	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**
ℓ_2	.21	.84	**.21**	.66	**.21**	**.21**	**.21**	**.21**	**.21**	**.21**	**.21**	**.21**	**.21**	**.21**	**.21**	**.21**
time	.00	.06	.05	.11	.07	.02	2.37	.63	.02	2.37	.63	.22	.12	.03	2.78	1.00
$n = 100$ $p = 1000$ $\rho = 0$																
FP	.00	9.58	.80	7.80	2.59	.00	.00	.00	.20	.20	.20	.00	.00	1.00	.60	.40
FN	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
FP+FN	.00	9.58	.80	7.80	2.59	**.00**	**.00**	**.00**	.20	.20	.20	**.00**	**.00**	1.00	.60	.40
ℓ_2	.12	.74	**.12**	.64	.45	**.12**	**.12**	**.12**	**.12**	.24	**.12**	**.12**	**.12**	.37	.25	.24
time	.00	.94	1.33	.17	.46	.21	3.07	3.65	.21	3.07	3.65	2.57	2.60	.54	4.63	5.63
$n = 100$ $p = 1000$ $\rho = 0.75$																
FP	.00	10.68	.00	5.33	.18	.66	.17	.00	.66	.33	.33	.66	.50	1.99	1.16	1.16
FN	.00	.00	.00	.00	.00	.50	.00	.00	.50	.00	.00	.00	.00	.00	.00	.00
FP+FN	.00	10.68	**.00**	5.33	.18	1.16	.17	**.00**	1.16	.33	.33	.66	.50	1.99	1.16	1.16
ℓ_2	.25	.95	**.25**	.59	**.25**	3.05	**.25**	**.25**	3.05	**.25**	**.25**	.34	.33	.85	.85	.85
time	.00	.42	.29	.12	.09	.12	15.19	6.12	.12	15.19	6.12	1.70	1.58	.43	18.88	9.17
$n = 100$ $p = 5000$ $\rho = 0$																
FP	.00	14.31	.00	7.32	1.33	.00	.00	.00	.33	.33	.33	.00	.00	.33	1.00	.33
FN	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
FP+FN	.00	14.31	.00	7.32	1.33	**.00**	**.00**	**.00**	.33	.33	.33	**.00**	**.00**	.33	1.00	.33
ℓ_2	.19	.67	**.19**	.53	.29	**.19**	**.19**	**.19**	**.19**	.29	.29	**.19**	**.19**	**.19**	.49	.29
time	.00	2.06	3.89	.22	.52	.68	14.57	20.83	.68	14.57	20.83	12.75	13.00	1.20	22.57	31.99
$n = 100$ $p = 5000$ $\rho = 0.75$																
FP	.00	15.02	.00	3.05	.00	.01	.00	.00	.01	.00	.01	.00	.00	.01	.01	.01
FN	.00	.00	.00	.00	.00	.00	.00	.00	.01	.00	.00	.00	.00	.00	.00	.00
FP+FN	.00	15.02	**.00**	3.05	.00	.01	.00	**.00**	.02	.00	.01	.00	.00	.01	.01	.01
ℓ_2	.13	.98	**.13**	.58	**.13**	**.13**	**.13**	**.13**	**.13**	**.13**	**.13**	**.13**	**.13**	**.13**	**.13**	**.13**
time	.00	1.58	1.32	.23	.12	.53	52.91	25.63	.53	52.91	25.63	7.55	7.76	1.01	59.73	33.70

Table 3.2: **Model (A)**: The average number of False Positives (FP) and False Negatives (FN), the median of $\ell_2 = \|\hat{\beta} - \beta\|$, calculated over 500 realisations. Also average computation times in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM. Both RBVS and IRBVS use $B = 500$ and $m = \frac{n}{2}$. Bold: the lowest or within 10% of the lowest value of FP+FN (or ℓ_2 respectively). Underlined: best among non-iterative or iterative methods with the same base learner.

		CV		SIS		StabSel			RBSS			ISIS		IRBSS		
	Oracle	Lasso	MC+	Lasso	MC+	PC	Lasso	MC+	PC	Lasso	MC+	Lasso	MC+	PC	Lasso	MC+
$n = 100$ $p = 100$ $\rho = 0.5$																
FP	.00	30.10	14.25	16.93	15.98	.25	.31	.12	.86	.88	.62	5.90	2.29	1.49	1.70	.77
FN	.00	.00	.69	.94	.94	1.91	1.43	1.43	1.37	2.11	1.85	.00	.00	.49	1.16	.73
FP+FN	.00	30.10	14.94	17.87	16.91	2.16	1.75	**1.56**	2.23	2.99	2.47	5.90	2.29	1.98	2.87	**1.49**
ℓ_2	.27	3.90	11.14	11.22	11.23	12.67	12.66	12.63	12.41	12.43	12.60	.66	.64	**.48**	.78	.60
time	.00	.32	.28	.13	.18	.02	.60	.48	.02	.60	.48	.67	.51	.06	1.45	1.51
$n = 100$ $p = 100$ $\rho = 0.75$																
FP	.00	39.87	21.13	16.10	12.18	.37	.49	.01	.17	.77	1.10	6.49	2.74	1.20	1.52	1.73
FN	.00	.37	1.00	1.49	1.49	2.58	1.38	1.38	1.72	2.46	1.74	.00	.00	.95	1.33	.49
FP+FN	.00	40.24	22.13	17.59	13.67	2.95	1.87	**1.39**	1.89	3.24	2.85	6.49	2.74	2.15	2.85	2.22
ℓ_2	.48	8.27	13.49	13.60	13.63	15.31	15.10	15.19	15.30	15.14	14.93	1.21	.96	**.79**	1.17	.88
time	.00	1.35	.21	.13	.11	.02	.59	.50	.02	.59	.50	.57	.42	.04	1.35	1.48
$n = 100$ $p = 1000$ $\rho = 0.5$																
FP	.00	71.12	19.31	16.76	16.27	.31	.31	.08	.46	.69	.61	.77	.23	1.08	.77	.62
FN	.00	1.00	1.00	1.00	1.00	1.46	1.00	1.00	1.01	1.00	1.00	.46	.46	.01	.00	.00
FP+FN	.00	72.12	20.31	17.76	17.27	1.77	1.31	1.08	1.47	1.69	1.61	1.23	.69	1.09	.77	**.62**
ℓ_2	.36	10.93	11.16	11.36	11.39	12.48	12.48	12.48	12.40	12.44	12.44	.46	**.39**	.46	**.37**	**.38**
time	.00	2.23	.57	.13	.17	.11	3.26	3.59	.11	3.26	3.59	1.82	1.94	.36	6.49	8.50
$n = 100$ $p = 1000$ $\rho = 0.75$																
FP	.00	71.38	18.94	17.50	16.50	.25	.25	.00	.50	.50	.25	.51	.25	1.49	.51	.25
FN	.00	1.00	1.00	1.00	1.00	1.74	1.00	1.00	1.00	1.00	1.00	.75	.75	.00	.00	.00
FP+FN	.00	72.38	19.94	18.50	17.50	1.98	1.25	1.00	1.50	1.50	1.25	1.26	1.00	1.49	.51	**.25**
ℓ_2	.36	13.25	13.53	13.80	13.80	15.12	15.05	15.12	15.00	15.05	15.11	.70	.54	.67	.53	**.36**
time	.00	4.59	.38	.15	.13	.11	3.26	4.02	.11	3.26	4.02	1.67	2.17	.35	6.35	8.63
$n = 100$ $p = 5000$ $\rho = 0.5$																
FP	.00	76.17	18.01	17.81	17.80	.00	.00	.00	.00	.33	.17	.50	.33	.67	.33	.17
FN	.00	1.00	1.00	1.17	1.17	1.83	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00	.00	.00
FP+FN	.00	77.17	19.01	18.97	18.97	1.83	1.00	1.00	1.00	1.33	1.17	1.50	1.33	.67	.33	**.17**
ℓ_2	.26	11.02	11.23	11.61	11.61	12.56	12.52	12.52	12.52	12.53	12.53	.39	.45	.39	**.26**	**.26**
time	.00	5.48	1.90	.18	.22	.55	12.46	13.13	.55	12.46	13.13	8.02	8.02	1.61	29.85	43.66
$n = 100$ $p = 5000$ $\rho = 0.75$																
FP	.00	71.95	14.01	17.93	15.93	.01	.00	.00	.01	.00	.00	.01	.00	.01	.01	.00
FN	.00	1.00	1.00	1.00	1.00	1.01	1.00	1.00	1.00	1.00	1.00	.01	.01	.00	.00	.00
FP+FN	.00	72.95	15.01	18.94	16.93	1.02	1.00	1.00	1.01	1.01	1.00	.01	.01	.01	.01	**.00**
ℓ_2	.22	13.27	13.53	14.02	14.04	14.95	14.95	14.95	14.95	14.95	14.95	**.22**	**.22**	**.22**	**.22**	**.22**
time	.00	8.20	1.42	.20	.16	.53	17.66	20.43	.53	17.66	20.43	8.01	11.81	1.54	35.22	48.21

Table 3.3: **Model (B)**: The average number of False Positives (FP) and False Negatives (FN), the median of $\ell_2 = \|\hat{\beta} - \beta\|$, calculated over 500 realisations. Also average computation times in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM. Both RBVS and IRBVS use $B = 500$ and $m = \frac{n}{2}$. Bold: the lowest or within 10% of the lowest value of FP+FN (or ℓ_2 respectively). Underlined: best among non-iterative or iterative methods with the same base learner.

	Oracle	CV		SIS		StabSel			RBVS			ISIS		IRBVS		
		Lasso	MC+	Lasso	MC+	PC	Lasso	MC+	PC	Lasso	MC+	Lasso	MC+	PC	Lasso	MC+
						$n = 100$	$p = 100$	$s = 5$								
FP	.00	4.93	.60	7.68	1.35	.12	.00	.00	.24	.29	.18	5.25	3.64	1.36	1.06	.54
FN	.00	.52	.65	1.42	1.59	2.65	1.01	.83	2.48	1.23	.71	.65	.70	.94	.65	.71
FP+FN	.00	5.45	1.25	9.10	2.94	2.77	1.01	**.83**	2.72	1.52	**.89**	5.89	4.34	2.30	1.71	1.25
ℓ_2	.07	.19	.11	.38	.31	.54	.14	**.10**	.52	.16	**.09**	.19	.20	.15	.14	.14
time	.00	.17	.29	.17	.47	.06	.78	.86	.06	.78	.86	1.58	1.83	.18	1.42	1.33
						$n = 100$	$p = 100$	$s = 10$								
FP	.00	8.29	1.36	9.66	3.38	.01	.00	.00	.28	.13	.00	3.44	1.58	1.53	.88	.56
FN	.00	.90	1.28	4.08	4.23	7.63	6.18	5.34	7.52	6.40	6.07	1.17	1.03	2.34	2.57	2.61
FP+FN	.00	9.19	**2.64**	13.74	7.60	7.64	6.18	5.34	7.80	6.53	6.07	4.61	**2.61**	3.86	3.44	3.17
ℓ_2	.13	.26	**.18**	.66	.63	1.76	1.28	1.28	1.76	1.28	1.28	**.19**	**.18**	.19	.21	.21
time	.00	.06	.12	.13	.19	.03	.59	.56	.03	.59	.56	.84	1.03	.15	1.69	1.61
						$n = 100$	$p = 1000$	$s = 5$								
FP	.00	19.75	2.46	10.45	2.52	.23	.15	.00	.31	.46	.23	.92	.30	1.38	.99	.68
FN	.00	.46	.62	2.23	2.23	2.54	1.31	.92	3.00	1.00	.69	.77	.77	.69	.69	.69
FP+FN	.00	20.21	3.07	12.69	4.75	2.77	1.46	**.92**	3.30	1.46	**.92**	1.69	1.08	2.08	1.69	1.38
ℓ_2	.11	.36	.23	.56	.47	.55	.26	.15	.62	.18	**.12**	.15	.15	.16	.18	.15
time	.00	1.06	1.38	.17	.47	.21	2.62	2.97	.21	2.62	2.97	7.50	7.56	.75	4.56	5.10
						$n = 100$	$p = 1000$	$s = 10$								
FP	.00	26.17	2.27	12.06	6.06	.21	.04	.00	.22	.09	.00	.74	.21	1.81	1.01	.68
FN	.00	1.18	1.10	5.30	5.37	7.53	5.72	5.74	7.70	4.70	2.62	2.79	2.62	1.66	1.23	1.14
FP+FN	.00	27.35	3.37	17.35	11.44	7.74	5.76	5.74	7.92	4.79	2.62	3.53	2.83	3.47	2.24	**1.82**
ℓ_2	.13	.48	.23	1.57	1.46	1.97	1.35	1.17	2.02	.93	.32	.27	.24	.24	.23	**.20**
time	.00	.40	.79	.17	.29	.16	2.00	2.51	.16	2.00	2.51	7.03	7.15	.95	6.32	6.25
						$n = 100$	$p = 5000$	$s = 5$								
FP	.00	36.09	3.70	13.02	8.29	.01	.00	.00	.00	.33	.00	.01	.00	.67	.34	.00
FN	.00	.01	.01	1.69	1.69	2.36	1.02	1.67	2.69	1.01	.03	.03	.03	.02	.01	.01
FP+FN	.00	36.10	3.71	14.71	9.98	2.38	1.02	1.67	2.70	1.34	.03	.04	.04	.69	.35	**.02**
ℓ_2	.07	.38	.09	.70	.60	.77	.31	.37	1.16	.31	**.07**	**.07**	**.07**	.12	**.07**	**.07**
time	.00	2.19	4.09	.22	.58	.76	13.07	13.07	.76	13.07	13.07	37.82	38.13	3.26	29.67	24.03
						$n = 100$	$p = 5000$	$s = 10$								
FP	.00	42.48	3.85	13.25	7.43	.30	.05	.00	.17	.08	.03	.28	.16	1.15	.53	.33
FN	.00	2.07	1.48	6.47	6.49	7.83	6.53	7.38	8.21	6.10	5.75	4.85	4.94	2.86	2.32	2.69
FP+FN	.00	44.56	5.33	19.72	13.92	8.13	6.58	7.38	8.38	6.17	5.78	5.12	5.10	4.00	**2.85**	**3.02**
ℓ_2	.13	.69	.29	1.73	1.63	1.85	1.32	1.60	1.99	1.13	.97	.58	.60	.28	**.25**	**.25**
time	.00	1.51	1.62	.16	.18	.35	3.04	3.09	.35	3.04	3.09	11.74	11.70	1.87	10.15	9.22

Table 3.4: **Model (C)**: The average number of False Positives (FP) and False Negatives (FN), the median of $\ell_2 = \|\hat{\beta} - \beta\|$, calculated over 500 realisations. Also average computation times in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM. Both RBVS and IRBVS use $B = 500$ and $m = \frac{n}{2}$. Bold: the lowest or within 10% of the lowest value of FP+FN (or ℓ_2 respectively). Underlined: best among non-iterative or iterative methods with the same base learner.

	CV		SIS		StabSel			RBVS			ISIS		IRBVS		
	Lasso	MC+	Lasso	MC+	PC	Lasso	MC+	PC	Lasso	MC+	Lasso	MC+	PC	Lasso	MC+
						$n = 100$	$p = 100$								
FP	8.37	1.40	10.62	3.54	.09	.13	.08	.59	.43	.40	3.32	2.93	1.22	.88	.71
FN	1.57	2.00	1.29	1.90	.30	1.84	1.80	.25	1.51	1.19	1.25	2.00	.25	1.51	1.19
FP+FN	9.93	3.40	11.91	5.45	**.39**	1.97	1.88	.84	1.94	1.58	4.58	4.93	1.47	2.39	1.89
time	.11	.46	.15	.21	.04	.18	.18	.04	.18	.18	.68	.65	.10	.34	.34
						$n = 200$	$p = 100$								
FP	8.05	1.25	12.52	2.32	.06	.17	.13	.52	.49	.49	9.87	7.12	1.23	.93	.73
FN	1.57	2.00	1.39	2.00	.01	1.79	1.60	.00	1.48	.71	1.27	2.00	.00	1.48	.71
FP+FN	9.62	3.25	13.92	4.32	**.07**	1.96	1.73	.52	1.97	1.20	11.14	9.12	1.23	2.41	1.44
time	1.01	2.21	.20	.59	.08	.53	.55	.08	.53	.55	2.02	2.21	.17	.97	1.01
						$n = 100$	$p = 1000$								
FP	19.73	2.71	14.86	11.55	.25	.16	.05	.36	.26	.17	.29	.36	.77	.54	.32
FN	1.69	2.00	1.17	1.54	.56	1.91	1.90	.79	1.76	1.32	1.40	2.00	.76	1.76	1.32
FP+FN	21.42	4.71	16.03	13.09	**.81**	2.07	1.94	1.15	2.01	1.49	1.69	2.36	1.52	2.29	1.65
time	.77	1.58	.22	.50	.13	.85	.96	.13	.85	.96	3.66	3.87	.29	1.62	1.85
						$n = 200$	$p = 1000$								
FP	17.15	2.30	27.07	19.37	.22	.22	.12	.48	.28	.14	1.12	.97	.91	.56	.31
FN	1.73	2.00	1.22	1.60	.01	1.85	1.73	.02	1.52	.78	1.24	2.00	.02	1.52	.78
FP+FN	18.88	4.30	28.29	20.98	**.23**	2.07	1.85	.50	1.79	.92	2.36	2.97	.92	2.08	1.09
time	1.55	3.30	.25	1.26	.35	1.67	1.56	.35	1.67	1.56	6.12	5.85	.72	2.89	2.89
						$n = 100$	$p = 5000$								
FP	23.84	4.61	15.72	14.21	.30	.10	.00	.26	.19	.10	.08	.14	.52	.38	.22
FN	1.80	2.00	1.18	1.38	.84	1.95	1.96	1.09	1.86	1.58	1.67	2.00	1.01	1.85	1.58
FP+FN	25.64	6.61	16.91	15.59	**1.15**	2.05	1.97	1.36	2.04	1.69	1.75	2.14	1.54	2.23	1.80
time	2.51	3.77	.29	.73	1.20	5.44	6.27	1.20	5.44	6.27	19.48	18.46	2.47	8.91	11.70
						$n = 200$	$p = 5000$								
FP	24.50	3.35	29.75	28.80	.23	.19	.10	.28	.19	.10	.16	.27	.56	.39	.18
FN	1.77	2.00	1.13	1.28	.04	1.90	1.80	.08	1.58	.88	1.21	2.00	.08	1.58	.88
FP+FN	26.27	5.35	30.89	30.08	**.27**	2.09	1.90	.36	1.78	.98	1.37	2.27	.64	1.98	1.06
time	4.16	7.66	.37	2.20	3.60	6.39	8.17	3.60	6.39	8.17	30.50	35.93	4.56	12.25	14.64

Table 3.5: **Model (D)**: The average number of False Positives (FP) and False Negatives (FN) calculated over 500. Also average computation times in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM. Both RBVS and IRBVS use $B = 500$ and $m = \frac{n}{2}$. Bold: methods with the lowest or within 10% of the lowest value of FP+FN. Underlined: best result among non-iterative or iterative methods with the same choice of the base learner.

	CV		SIS		StabSel				RBSS				ISIS		IRBVS			
	Lasso	MC+	Lasso	MC+	PC	DC	Lasso	MC+	PC	DC	Lasso	MC+	Lasso	MC+	PC	DC	Lasso	MC+
$n = 100 \quad p = 100$																		
FP	10.61	8.81	11.93	10.71	.06	.06	.04	.04	.39	.36	.41	.28	3.37	3.75	.98	.91	.80	.61
FN	1.36	1.93	1.26	1.74	2.35	2.37	2.49	3.15	2.31	2.29	2.02	2.20	1.55	1.87	1.45	1.56	1.72	2.03
FP+FN	11.97	10.75	13.18	12.45	**2.41**	**2.42**	**2.53**	3.19	2.70	2.66	**2.42**	**2.48**	4.92	5.62	**2.43**	**2.47**	**2.52**	**2.63**
time	.21	1.10	.15	.32	1.43	1.49	.21	.77	1.43	1.49	.21	.77	.66	.71	1.57	3.45	.58	.91
$n = 200 \quad p = 100$																		
FP	10.69	8.51	15.91	13.67	.05	.06	.02	.02	.19	.23	.07	.04	9.53	10.26	.91	.91	.57	.41
FN	.59	1.05	.53	.93	1.69	1.71	1.21	1.20	1.47	1.47	.76	.56	.51	.80	.43	.52	.57	.49
FP+FN	11.28	9.56	16.44	14.60	1.73	1.77	1.23	1.22	1.66	1.71	.83	**.61**	10.04	11.06	1.34	1.43	1.14	.90
time	1.01	2.36	.15	.62	.52	4.41	.82	.40	.52	4.41	.82	.40	1.07	1.66	.62	11.04	1.29	.80
$n = 100 \quad p = 1000$																		
FP	19.72	17.83	15.21	15.12	.29	.26	.15	.01	.37	.31	.21	.13	.25	.27	.77	.65	.40	.29
FN	1.93	2.51	1.85	2.13	2.68	2.62	2.89	4.03	2.92	2.78	2.66	3.33	2.54	2.76	2.09	2.10	2.49	3.19
FP+FN	21.65	20.34	17.06	17.25	**2.97**	**2.88**	3.05	4.04	3.30	3.09	**2.87**	3.47	**2.79**	3.03	**2.86**	**2.75**	**2.88**	3.48
time	.81	1.81	.15	.34	.08	5.68	.35	.45	.08	5.68	.35	.45	1.07	1.04	.22	11.58	.69	.86
$n = 200 \quad p = 1000$																		
FP	22.88	18.40	28.75	28.82	.16	.20	.16	.07	.36	.31	.34	.24	1.17	1.41	.73	.68	.59	.35
FN	.88	1.34	.84	1.06	1.75	1.80	1.48	1.45	1.74	1.75	1.39	1.06	1.21	1.35	.92	1.14	1.25	1.03
FP+FN	23.76	19.74	29.60	29.88	1.91	2.00	1.64	1.52	2.10	2.05	1.73	**1.30**	2.38	2.76	1.64	1.82	1.84	**1.38**
time	1.32	3.29	.18	.83	.11	22.49	.50	.51	.11	22.49	.50	.51	2.25	2.25	.31	45.06	1.05	1.03
$n = 100 \quad p = 5000$																		
FP	25.35	23.29	16.07	16.08	.28	.27	.10	.00	.21	.17	.05	.05	.08	.07	.51	.40	.22	.18
FN	2.39	2.97	2.34	2.51	3.11	2.98	3.47	4.73	3.54	3.35	3.47	3.82	3.11	3.32	2.66	2.49	3.22	3.64
FP+FN	27.74	26.26	18.41	18.59	3.39	3.26	3.57	4.73	3.74	3.51	3.52	3.87	3.19	3.39	**3.17**	**2.89**	3.44	3.82
time	2.58	4.75	.19	.42	.93	53.96	2.71	3.43	.93	53.96	2.71	3.43	8.45	7.82	1.90	110.12	5.31	7.42
$n = 200 \quad p = 5000$																		
FP	32.22	25.59	30.27	30.55	.20	.33	.18	.03	.24	.25	.21	.13	.23	.28	.49	.49	.36	.21
FN	1.15	1.64	1.25	1.36	1.96	1.95	1.85	1.97	2.04	1.98	1.96	1.76	1.80	1.90	1.27	1.52	1.81	1.67
FP+FN	33.38	27.23	31.53	31.91	2.16	2.28	2.03	2.00	2.28	2.23	2.17	**1.90**	2.04	2.18	**1.76**	2.01	2.17	**1.88**
time	3.74	7.95	.26	1.00	.56	115.84	2.76	3.35	.56	115.84	2.76	3.35	8.35	8.74	1.64	363.91	7.09	7.74

Table 3.6: **Model (E)**: The average number of False Positives (FP) and False Negatives (FN) calculated over 500 realisations. Also average computation times in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM. Both RBVS and IRBVS use $B = 500$ and $m = \frac{n}{2}$. Bold: the lowest or within 10% of the lowest value of FP+FN. Underlined: best among non-iterative or iterative methods with the same base learner.

3.7 Data examples

We present applications to two well-known datasets: the prostate cancer data and the Boston housing data.

3.7.1 Prostate cancer data set

We compare performance of RBVS against its two competitors, StabSel (Meinshausen and Bühlmann, 2010) and the approach of Hall and Miller (2009a) (HM). To provide a fair comparison, we apply these three methods with the same subsamples taken from the data described below, drawn as in Definition 3.3.3. Besides the number of subsamples and their size, we need to specify the threshold π and the bound for the expected number of false positives $\mathrm{E}\,V$ for StabSel, the significance level α and the cut-off level c for HM. We try several values for each pair of these parameters.

We analyse the Prostate cancer data (Singh et al., 2002) which is frequently used to evaluate the performance of various classification methods (Pochet et al. (2004), Fan and Fan (2008), Hall and Xue (2014)). It consists of expression levels of $p = 12600$ genes from 52 tumour and 50 normal prostate samples in the training set, and 9 tumour and 25 normal samples in the test set coming from an independent experiment. The response variable Y is binary (1 for tumour samples, 0 for normal samples) and X_j, the expression of the j'th gene, is a continuous variable. In this setting, we take the sample correlation coefficient to identify the covariates that affect the response, which was previously used in this and similar classification problems; see Fan and Lv (2008) and Hall and Xue (2014).

We use RBVS, HM and StabSel on the training set to identify the important genes. Still on the training set, we fit the logistic regression model, using the selected covariates only. Subsequently, we use the fitted model to classify samples in the test set. Finally, we record the number of correctly classified samples. The entire experiment is repeated

50 times, to minimise the impact of a particular random draw, and the medians are reported.

The median correct classification rate on the test set for the RBVS algorithm is 31 out of 34 and this is always achieved using from 3 to 6 genes only, both for subsamples of size $m = \left\lfloor \frac{n}{2} \right\rfloor = 51$ and $m = \left\lfloor \frac{3n}{4} \right\rfloor = 76$. For some random draws, RBVS selects exactly 4 genes, which result in the classification rate of 33. Figure 3.2 summarises the corresponding numbers for the StabSel and HM algorithms, with various tuning parameters of these methods. For $m = \left\lfloor \frac{n}{2} \right\rfloor$, there exists one pair of parameters that leads to a better error control for StabSel and HM (33 correctly classified samples), however, RBVS is always better when $m = 76$. The parameters which are the best in this example are much different from those recommended for StabSel and HM. Unlike its competitors, RBVS automatically selects an appropriate number of genes, being particularly effective in this example.

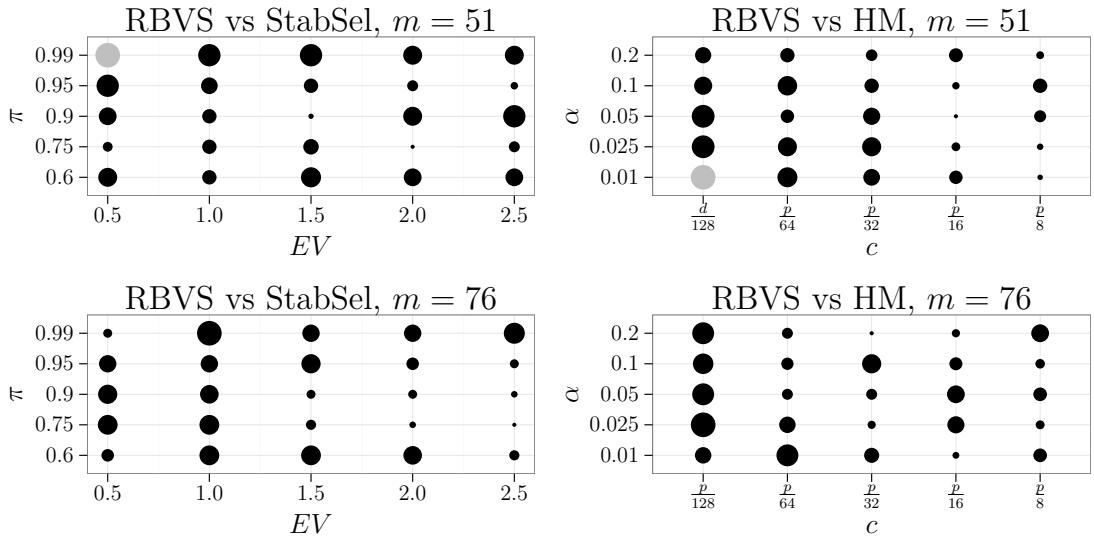


Figure 3.2: Prostate cancer data set: the median of the number of correctly classified samples on the test set, evaluated over 50 runs of the algorithms studied. The larger a circle, the better classification rate. Grey colour indicates the cases where the median classification rate is no worse than 31, the median classification rate achieved by RBVS PC. The number of subsamples $B = 500$.

3.7.2 Boston housing data set

We apply our methodology to the Boston housing data set (Harrison and Rubinfeld, 1978) which has been frequently adopted to illustrate performance of various variable selection and estimation techniques (see e.g. Radchenko and James (2010), Cho and Fryzlewicz (2012a) or Fan et al. (2014). We use Boston Housing data available in the R package **mlbench** (Leisch and Dimitriadou, 2010) containing 15 numerical covariates which may have influence over the median price recorded in $n = 506$ locations. As in Cho and Fryzlewicz (2012a), we additionally consider interaction terms between the explanatory variables so the final data set has $p = 120$ covariates.

Harrison and Rubinfeld (1978) used the linear model to analyse the price, thus we apply RBVS combined with the linear measures introduced Section 3.6.1.

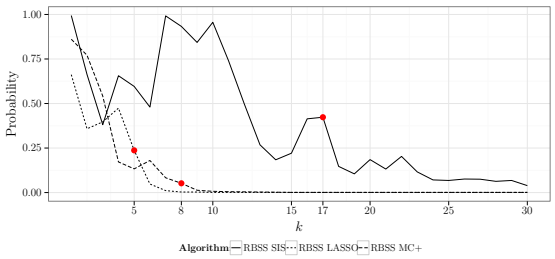


Figure 3.3: The Boston housing data: the estimated probabilities corresponding to the k-element subsets top-ranked the most frequently. The dots indicate the probability at $k = \hat{s}$, which is the number of elements selected according to the suggested approach. The subsample size $m = \frac{n}{2} = 253$ and $B = 250$.

Figure 3.3 shows a "RBVS path", i.e. probabilities corresponding to the k-lement subsets of covariates the most frequently occurring as the most influential ones (defined by (3.4)). The "probability path for RBVS PC declines much slower than those corresponding to RBVS Lasso and RBVS MC+. This results in a different numbers of selected variables; RBVS PC chooses 17 covariates, while RBVS MC+ 8 and RBVS Lasso MC+ just 5. We argue that in this example RBVS PC, as based on a marginal measure, includes some variables that are not useful in a predictive model. Intuitively, if two or more variables

were highly correlated to the response, then interactions formed of any two of those would be highly correlated to Y.

To investigate predictive usefulness of RBVS based methods, we split the data randomly, assembling approximately 50%, 25% and 25% observations to the train, validation and test sets, respectively. On the training set, we select variables and obtain OLS estimates of the regression coefficients (for Lasso and MC+ we consider all set candidates on their solution paths, for RBVS based methods we take the subsample size equal to $m = \left\{\frac{1}{8}, \frac{2}{8}, \ldots, \frac{7}{8}\right\} n_{train}$). Next, we evaluate the average prediction error on the validation set and choose the covariates minimising the error. Finally, we find the average prediction error, R squared coefficient (R^2) and adjusted R squared (R^2_{adj}) on the test set.

Table 3.7 reports the results averaged over 500 random splits of the data; PG in this summary corresponds to the linear model studied in Pace and Gilley (1997), Section 2.2. RBVS PC, RBVS Lasso and RBVS MC+ perform similar to PG in terms of prediction accuracy, which can be seen from the corresponding values of the test error and R^2. However, RBVS Lasso and RBVS MC+ choose on average only 9 variables and consequently perform best in terms of R^2_{adj}. Lasso and MC+ achieve the best test error; however, they select about 50 variables on average. By contrast, IRBVS Lasso and IRBVS MC+ choose no more than 27 covariates, yet they achieve similar prediction accuracy as Lasso and MC+ respectively. Both RBVS PC and IRBVS PC perform reasonably well in terms of prediction accuracy, however, they select more variables than the remaining RBVS and IRBVS based techniques. This is probably caused by the strong correlations between covariates, which is due to the way the data set has been produced.

				RBVS			IRBVS		
	PG	Lasso	MC+	SIS	Lasso	MC+	SIS	Lasso	MC+
test error	0.037	0.032	0.032	0.038	0.038	0.038	0.036	0.033	0.033
R^2	0.773	0.803	0.805	0.769	0.766	0.765	0.780	0.798	0.801
R^2_{adj}	0.735	0.638	0.609	0.708	0.748	0.747	0.571	0.739	0.745
no var	18.0	49.3	55.0	25.4	9.2	9.1	44.7	27.6	26.5

Table 3.7: Boston housing data : the test error, R squared, adjusted R squared and the number of selected variables, averaged over 500 test sets.

3.8 High-dimensional simulation study

The aim of the simulation study reported in this section is threefold. First, to provide an extensive comparison of the performance of RBVS and StabSel algorithms. Second, to investigate their utility in the "high-dimensional framework", where p is growing with n and the former is much larger than the latter. Third, to check how sensitive both approaches are to the choice of the subsample size m.

The data are generated from the following linear model

$$Y_i = \beta_1 X_{i1} + \ldots, \beta_p X_{ip} + \varepsilon_i, \ldots, i = 1, \ldots, n, \tag{3.10}$$

where

- X_{ij}'s follow the factor model $X_{ij} = \sum_{l=1}^{K} f_{ijl}\varphi_{il} + \theta_{ij}$, with f_{ijl}, φ_{il}, θ_{ij}, ε_i i.i.d. $\mathcal{N}(0,1)$ and the number of factors equal either $K = 0$ (variables independent) or $K = 5$. We choose the factor model, as it provides a non-trivial dependence structure between the covariates and it is relatively easy and quick to simulate. The R package **rbvs** provides a C-implemented routine `gen.factor.model.design` which quickly generates the factor model design matrix.

- The number of non-zero $\beta_j's$ is set to $s = 5, 10$, their indices are drawn uniformly without replacement from $\{1, \ldots, p\}$. Their values are drawn independently and have same distribution as $\beta = \left(|Z| + \frac{\log(n)}{\sqrt{n}}\right) V$, where Z is a standard normal random variable and V is independent of Z with $\mathbb{P}(V = 1) = \mathbb{P}(V = -1) = \frac{1}{2}$. In

this setting, the impact of the important predictors is diminishing with n.

- The total number of variables $p = 100, 1000, 10000, 100000$.
- The sample size $n = 100, 200, \ldots, 1000$.
- The subsample size is set to $m = 50, 100, \frac{n}{2}$.

Due to a very large number of variables, we take the marginal correlation as a base learner for both StabSel and (I)RBVS, as it is least computationally demanding across measures studied in this chapter. All computations reported in this section are performed with the R package **rbvsGPU** (Baranowski, 2016), which provide a parallel implementation of RBVS PC and IRBVS PC, using to this end the CUDA framework (Luebke, 2008). The number of random splits is set to $B = 500\frac{m}{n}$, such that there always 500 subsamples used in total.

Unlike the RBVS algorithm, StabSel requires specification of the two tuning parameters. From our experience, the values recommended in Meinshausen and Bühlmann (2010) are fairly "optimal", we decided however to test robustness of the StabSel algorithm against the choice of its parameters. The bound on the error control is set to $EV = 2.5, 5$, while the thresholding probability $\pi = 0.55, 0.6, 0.75, 0.9$.

Tables 3.8–3.19 report results of this high-dimensional simulation study. Furthermore, Table 3.20 shows the average computation times in one of the simulation scenarios. The times for the other scenarios are similar, hence not reported here. We address each issue brought up in the introduction to this section in the comments below.

1. Comparison of StabSel to RBVS:
    - In the fixed m cases, RBVS typically outperforms StabSel. Moreover, for a moderate value of $m = 100$ and p fixed, the average number of false positives and false negatives decreases with n, which does not hold for StabSel.

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.57	2.38	3.03	3.53
300	1.50	2.27	3.00	3.47
400	1.41	2.33	2.98	3.48
500	1.53	2.32	2.98	3.46
600	1.47	2.29	2.95	3.46
700	1.56	2.34	2.96	3.46
800	1.44	2.27	2.97	3.50
900	1.61	2.34	2.98	3.44
1000	1.48	2.31	2.98	3.45

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**.35**	**.19**	**.41**	**.96**
300	**.16**	**.10**	**.45**	**1.06**
400	**.04**	**.12**	**.49**	**.98**
500	**.03**	**.15**	**.56**	**1.02**
600	**.06**	**.21**	**.62**	**1.18**
700	**.05**	**.26**	**.66**	**1.17**
800	**.04**	**.25**	**.73**	**1.12**
900	**.05**	**.32**	**.72**	**1.31**
1000	**.05**	**.27**	**.74**	**1.28**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	2.05	2.49	**2.93**	**3.40**
300	2.15	2.57	3.04	**3.46**
400	2.19	2.66	3.11	3.48
500	2.29	2.68	3.11	3.50
600	2.30	2.68	3.11	3.54
700	2.41	2.73	3.14	3.49
800	2.25	2.67	3.14	3.51
900	2.43	2.77	3.19	3.56
1000	2.30	2.70	3.09	3.47

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.91	2.43	**2.94**	**3.41**
300	2.01	2.52	3.05	3.48
400	2.07	2.63	3.11	3.50
500	2.22	2.62	3.10	3.52
600	2.23	2.64	3.12	3.56
700	2.33	2.70	3.16	3.52
800	2.16	2.63	3.15	3.54
900	2.35	2.74	3.18	3.59
1000	2.22	2.67	3.11	3.49

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	2.07	2.68	3.15	3.62
300	2.23	2.77	3.28	3.73
400	2.27	2.86	3.38	3.81
500	2.42	2.87	3.36	3.76
600	2.40	2.90	3.36	3.77
700	2.50	2.93	3.42	3.77
800	2.37	2.90	3.42	3.77
900	2.54	3.00	3.52	3.81
1000	2.42	2.91	3.34	3.73

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.72	**2.27**	**2.80**	**3.28**
300	1.85	2.35	**2.87**	**3.36**
400	1.92	2.48	**2.97**	**3.38**
500	2.05	2.49	**2.96**	**3.40**
600	2.05	2.48	2.98	**3.40**
700	2.15	2.56	3.02	**3.41**
800	2.02	2.49	3.02	**3.41**
900	2.20	2.59	3.03	3.45
1000	2.09	2.54	**2.98**	**3.38**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.68	**2.21**	**2.79**	**3.28**
300	1.83	2.31	**2.86**	**3.37**
400	1.88	2.47	**2.97**	**3.39**
500	2.02	2.47	**2.96**	**3.41**
600	2.02	2.47	2.98	**3.42**
700	2.14	2.54	3.03	**3.42**
800	1.99	2.47	3.02	**3.43**
900	2.17	2.57	3.03	3.46
1000	2.06	2.51	**2.97**	**3.39**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.76	2.46	**3.01**	**3.50**
300	1.95	2.58	3.15	3.62
400	2.02	2.68	3.23	3.68
500	2.17	2.70	3.24	3.64
600	2.17	2.73	3.23	3.68
700	2.28	2.78	3.29	3.67
800	2.14	2.73	3.28	3.68
900	2.32	2.83	3.36	3.71
1000	2.19	2.74	3.22	3.63

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.8: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 50$ and $B = 500\frac{m}{n}$, number of important variables $s = 5$ and number of factors $K = 0$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.53	1.82	2.19	2.79
300	1.04	1.40	1.87	2.60
400	.90	1.36	1.89	2.59
500	.85	1.31	1.86	2.55
600	.76	1.34	1.86	2.35
700	.83	1.33	1.90	2.32
800	.73	1.30	1.87	2.31
900	.76	1.32	1.88	2.39
1000	.68	1.30	1.85	2.39

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.49**	**.98**	**.66**	**.58**
300	**.60**	**.20**	**.11**	**.40**
400	**.32**	**.09**	**.10**	**.35**
500	**.18**	**.03**	**.09**	**.41**
600	**.12**	**.03**	**.09**	**.32**
700	**.06**	**.01**	**.11**	**.30**
800	**.02**	**.02**	**.15**	**.30**
900	**.01**	**.04**	**.16**	**.36**
1000	**.01**	**.04**	**.18**	**.41**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.24**	**1.59**	**2.10**	**2.34**
300	1.31	1.49	**1.84**	**2.21**
400	1.33	1.61	1.96	**2.33**
500	1.44	1.61	1.96	**2.28**
600	1.44	1.68	2.01	**2.34**
700	1.55	1.71	2.05	**2.31**
800	1.44	1.69	2.05	2.33
900	1.57	1.74	2.07	2.43
1000	1.50	1.72	2.06	2.41

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.15**	**1.64**	**2.12**	**2.56**
300	1.17	1.43	**1.82**	**2.48**
400	1.22	1.57	1.96	**2.56**
500	1.29	1.56	1.96	**2.54**
600	1.33	1.63	2.01	**2.35**
700	1.44	1.66	2.06	2.32
800	1.34	1.64	2.06	2.34
900	1.46	1.69	2.08	2.46
1000	1.40	1.70	2.07	2.42

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.19**	**1.61**	**2.06**	**2.63**
300	1.24	1.60	2.01	2.72
400	1.30	1.73	2.18	2.79
500	1.41	1.75	2.16	2.75
600	1.45	1.82	2.23	2.57
700	1.54	1.87	2.30	2.55
800	1.45	1.82	2.27	2.58
900	1.58	1.88	2.31	2.68
1000	1.51	1.87	2.25	2.63

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.13**	**1.74**	2.31	**2.51**
300	1.07	**1.33**	**1.72**	**2.14**
400	1.12	1.47	**1.84**	**2.24**
500	1.17	1.45	**1.83**	**2.19**
600	1.20	1.52	1.89	**2.23**
700	1.29	1.54	1.94	**2.23**
800	1.21	1.49	1.94	**2.23**
900	1.36	1.59	1.96	**2.36**
1000	1.26	1.57	1.96	**2.30**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.17**	1.92	2.40	**2.54**
300	**1.02**	**1.29**	**1.72**	**2.13**
400	1.07	1.43	**1.83**	**2.23**
500	1.12	1.42	**1.83**	**2.19**
600	1.15	1.50	1.88	**2.24**
700	1.25	1.53	1.94	**2.23**
800	1.18	1.48	1.95	**2.24**
900	1.33	1.56	1.96	**2.36**
1000	1.23	1.55	1.96	**2.31**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.21**	**1.69**	**2.10**	**2.31**
300	1.05	1.43	1.88	**2.30**
400	1.12	1.59	2.03	**2.43**
500	1.20	1.60	2.02	**2.38**
600	1.23	1.67	2.11	2.48
700	1.34	1.72	2.16	2.45
800	1.27	1.68	2.16	2.47
900	1.41	1.74	2.19	2.58
1000	1.32	1.74	2.15	2.55

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.9: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 100$ and $B = 500\frac{m}{n}$, number of important variables $s = 5$ and number of factors $K = 0$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.57	1.79	2.22	2.59
300	1.18	1.31	1.64	1.98
400	1.07	1.10	1.33	1.61
500	.95	1.00	1.13	1.40
600	.94	.88	1.03	1.15
700	.96	.77	.90	1.00
800	.85	.77	.84	.92
900	.73	.67	.74	.87
1000	.80	.62	.78	.84

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.54**	**.90**	**.64**	**.44**
300	1.35	**.80**	**.58**	**.33**
400	1.23	**.86**	**.53**	**.25**
500	1.26	**.87**	**.55**	**.27**
600	1.39	**.80**	**.51**	**.24**
700	1.32	.78	**.46**	**.23**
800	1.28	.82	**.41**	**.24**
900	1.19	.76	**.43**	**.22**
1000	1.21	.75	**.45**	**.29**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.23**	**1.58**	**2.10**	**2.35**
300	**.88**	**1.18**	**1.54**	**1.81**
400	**.75**	**.96**	**1.31**	**1.56**
500	**.62**	**.83**	1.18	**1.35**
600	**.49**	**.76**	1.08	1.19
700	**.47**	**.62**	.96	1.12
800	**.41**	**.60**	**.82**	1.02
900	**.35**	**.51**	.75	.96
1000	**.32**	**.44**	.82	.92

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.16**	**1.65**	**2.12**	**2.38**
300	**.81**	**1.22**	**1.57**	**1.87**
400	**.67**	**1.01**	1.33	1.62
500	**.58**	**.91**	1.20	1.41
600	**.48**	**.84**	1.13	1.22
700	**.47**	**.68**	.98	1.13
800	**.43**	**.68**	.88	1.05
900	**.34**	**.59**	.77	.99
1000	**.33**	**.52**	.86	.94

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.18**	**1.59**	**2.03**	**2.43**
300	**.83**	**1.19**	**1.47**	**1.90**
400	**.68**	**.94**	**1.26**	1.69
500	**.59**	**.87**	**1.09**	1.41
600	**.49**	**.78**	**.98**	**1.11**
700	**.49**	**.64**	**.86**	**1.00**
800	**.45**	**.61**	**.77**	**.89**
900	**.37**	**.51**	**.70**	**.85**
1000	**.35**	**.47**	**.74**	**.82**

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.15**	**1.77**	2.29	**2.52**
300	**.80**	1.33	1.78	2.06
400	**.66**	1.14	1.52	1.77
500	**.59**	1.05	1.40	1.61
600	**.50**	.97	1.34	1.48
700	**.49**	.82	1.17	1.36
800	**.45**	.82	1.13	1.29
900	**.36**	.71	1.00	1.22
1000	**.34**	.70	1.07	1.21

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.15**	1.91	2.41	**2.56**
300	**.80**	1.48	1.86	2.10
400	**.72**	1.33	1.62	1.81
500	**.66**	1.22	1.48	1.68
600	**.56**	1.13	1.46	1.52
700	**.55**	.94	1.28	1.42
800	**.51**	1.01	1.26	1.35
900	**.43**	.91	1.12	1.27
1000	**.42**	.89	1.16	1.29

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.20**	**1.72**	**2.12**	**2.33**
300	**.85**	**1.29**	**1.59**	**1.77**
400	**.76**	1.10	1.33	**1.54**
500	**.70**	**.99**	1.19	**1.30**
600	**.63**	.94	1.11	1.17
700	**.63**	**.77**	.99	1.07
800	**.55**	**.76**	.85	.98
900	**.49**	**.66**	.78	.92
1000	**.50**	.65	.86	.87

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.10: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = \frac{n}{2}$ and $B = 500\frac{m}{n}$, number of important variables $s = 5$ and number of factors $K = 0$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.77	2.45	3.20	3.70
300	1.66	2.44	3.17	3.68
400	1.62	2.38	3.18	3.66
500	1.63	2.39	3.15	3.63
600	1.50	2.31	3.16	3.61
700	1.61	2.38	3.12	3.72
800	1.54	2.35	3.15	3.67
900	1.54	2.37	3.09	3.81
1000	1.56	2.33	3.10	3.79

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**.28**	**.11**	**.09**	**.48**
300	**.12**	**.03**	**.04**	**.25**
400	**.04**	**.00**	**.05**	**.21**
500	**.02**	**.01**	**.03**	**.15**
600	**.01**	**.00**	**.03**	**.13**
700	**.00**	**.01**	**.04**	**.19**
800	**.00**	**.00**	**.05**	**.17**
900	**.00**	**.00**	**.01**	**.29**
1000	**.00**	**.00**	**.04**	**.15**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	2.21	2.62	**3.10**	**3.53**
300	2.29	2.66	3.18	**3.63**
400	2.34	2.74	3.20	**3.62**
500	2.39	2.71	3.21	**3.57**
600	2.37	2.75	3.27	**3.57**
700	2.43	2.83	3.26	**3.67**
800	2.37	2.84	3.31	**3.67**
900	2.41	2.87	3.31	**3.78**
1000	2.40	2.73	3.20	**3.72**

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	2.09	2.57	**3.09**	**3.54**
300	2.20	2.62	3.19	**3.65**
400	2.23	2.71	3.23	**3.64**
500	2.29	2.69	3.21	**3.59**
600	2.28	2.70	3.29	**3.58**
700	2.35	2.80	3.26	**3.68**
800	2.28	2.80	3.32	3.69
900	2.30	2.84	3.32	3.82
1000	2.35	2.71	3.20	**3.74**

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	2.24	2.77	3.33	3.76
300	2.40	2.92	3.43	3.89
400	2.47	2.99	3.46	3.91
500	2.47	2.91	3.46	3.84
600	2.50	3.01	3.59	3.86
700	2.58	3.04	3.49	3.94
800	2.53	3.05	3.56	3.92
900	2.56	3.11	3.60	4.04
1000	2.52	2.98	3.51	3.95

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.94	**2.43**	**2.98**	**3.42**
300	2.05	2.47	**3.07**	**3.52**
400	2.06	2.55	**3.11**	**3.51**
500	2.11	2.55	**3.09**	**3.47**
600	2.11	2.55	**3.15**	**3.48**
700	2.18	2.65	3.15	**3.56**
800	2.10	2.62	3.20	**3.58**
900	2.13	2.68	3.17	**3.68**
1000	2.20	2.56	**3.07**	**3.62**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.88	**2.39**	**2.97**	**3.42**
300	2.01	**2.43**	**3.06**	**3.52**
400	2.03	2.53	**3.10**	**3.52**
500	2.09	2.53	**3.07**	**3.48**
600	2.08	2.53	3.16	**3.49**
700	2.16	2.64	3.15	**3.57**
800	2.08	2.61	3.20	**3.58**
900	2.11	2.66	3.18	**3.70**
1000	2.18	2.53	**3.07**	**3.62**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	2.00	2.60	**3.18**	**3.66**
300	2.14	2.72	3.30	3.78
400	2.19	2.79	3.34	3.81
500	2.23	2.76	3.34	3.71
600	2.26	2.82	3.44	3.76
700	2.32	2.88	3.38	3.84
800	2.26	2.91	3.42	3.82
900	2.28	2.95	3.47	3.95
1000	2.32	2.80	3.34	3.84

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.11: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 50$ and $B = 500\frac{m}{n}$, number of important variables $s = 5$ and number of factors $K = 5$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.58	1.90	2.39	2.82
300	1.21	1.48	2.15	2.58
400	.97	1.48	2.03	2.52
500	.88	1.39	2.01	2.48
600	.90	1.30	2.01	2.50
700	.83	1.41	2.04	2.49
800	.83	1.42	1.97	2.54
900	.76	1.42	1.98	2.59
1000	.77	1.36	2.01	2.63

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.51**	**.88**	**.59**	**.31**
300	**.65**	**.23**	**.08**	**.01**
400	**.30**	**.05**	**.01**	**.00**
500	**.16**	**.02**	**.01**	**.00**
600	**.10**	**.00**	**.00**	**.00**
700	**.05**	**.00**	**.00**	**.00**
800	**.03**	**.00**	**.00**	**.00**
900	**.01**	**.00**	**.00**	**.00**
1000	**.02**	**.00**	**.00**	**.01**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.33**	**1.72**	**2.17**	**2.56**
300	1.45	1.61	**2.07**	**2.37**
400	1.44	1.70	2.05	**2.38**
500	1.50	1.70	2.09	**2.41**
600	1.53	1.69	2.10	2.51
700	1.54	1.78	2.20	**2.43**
800	1.54	1.83	2.15	**2.54**
900	1.55	1.86	2.15	2.61
1000	1.60	1.80	2.19	**2.62**

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.21**	**1.73**	**2.21**	**2.56**
300	1.33	1.56	**2.06**	**2.37**
400	1.30	1.65	2.04	**2.39**
500	1.36	1.65	2.09	**2.41**
600	1.41	1.66	2.10	2.52
700	1.46	1.75	2.20	**2.44**
800	1.43	1.82	2.15	2.55
900	1.43	1.83	2.17	2.62
1000	1.48	1.77	2.20	2.63

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.24**	**1.72**	**2.18**	**2.56**
300	1.40	1.71	2.25	2.59
400	1.42	1.82	2.28	2.62
500	1.48	1.81	2.30	2.64
600	1.54	1.87	2.31	2.75
700	1.58	1.96	2.41	2.67
800	1.57	1.97	2.40	2.77
900	1.57	2.03	2.41	2.88
1000	1.64	1.98	2.41	2.85

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.21**	**1.86**	2.41	**2.74**
300	1.21	**1.46**	**1.97**	**2.25**
400	1.21	1.52	**1.95**	**2.30**
500	1.24	1.53	**1.97**	**2.32**
600	1.29	1.53	**2.00**	**2.41**
700	1.31	1.65	2.06	**2.33**
800	1.27	1.69	2.02	**2.43**
900	1.32	1.70	2.05	**2.52**
1000	1.36	1.63	2.09	**2.54**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.23**	1.96	2.49	**2.75**
300	**1.16**	**1.41**	**1.96**	**2.24**
400	1.17	1.50	**1.95**	**2.30**
500	1.21	1.50	**1.97**	**2.33**
600	1.24	1.49	**2.00**	**2.42**
700	1.25	1.64	2.06	**2.34**
800	1.24	1.65	2.01	**2.44**
900	1.28	1.68	2.05	**2.53**
1000	1.32	1.61	2.09	**2.54**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.27**	**1.74**	**2.24**	**2.53**
300	1.21	1.57	**2.11**	**2.49**
400	1.21	1.67	2.15	**2.51**
500	1.27	1.68	2.19	2.54
600	1.35	1.70	2.19	2.65
700	1.36	1.79	2.31	2.57
800	1.36	1.85	2.26	2.67
900	1.36	1.89	2.29	2.74
1000	1.41	1.82	2.29	2.75

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.12: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 100$ and $B = 500\frac{m}{n}$, number of important variables $s = 5$ and number of factors $K = 5$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	1.59	1.88	2.38	2.79
300	1.37	1.41	1.83	2.12
400	1.10	1.17	1.45	1.70
500	.92	1.08	1.24	1.48
600	.91	.89	1.13	1.29
700	.82	.87	1.01	1.14
800	.80	.84	.88	1.10
900	.83	.75	.80	.93
1000	.72	.73	.86	.91

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.35**	**.85**	**.56**	**.30**
300	1.45	**.78**	**.58**	**.29**
400	1.44	**.78**	**.48**	**.24**
500	1.23	**.84**	**.52**	**.29**
600	1.29	**.81**	**.51**	**.24**
700	1.17	**.80**	**.50**	**.20**
800	1.23	**.83**	**.48**	**.25**
900	1.34	.82	**.46**	**.21**
1000	1.19	.79	**.51**	**.19**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.34**	**1.75**	**2.19**	**2.59**
300	**1.07**	**1.26**	**1.65**	**2.06**
400	**.81**	**1.05**	**1.37**	**1.69**
500	**.63**	**.94**	**1.23**	1.48
600	**.58**	**.74**	1.14	1.34
700	**.48**	**.75**	1.09	1.19
800	**.43**	**.67**	.89	1.17
900	**.42**	**.60**	.82	1.01
1000	**.37**	**.56**	.87	.99

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.23**	**1.75**	**2.22**	**2.59**
300	**1.00**	**1.29**	**1.71**	**2.08**
400	**.74**	**1.12**	**1.42**	**1.69**
500	**.59**	**.99**	1.27	1.50
600	**.55**	**.81**	1.17	1.37
700	**.46**	**.82**	1.14	1.23
800	**.39**	**.75**	.93	1.21
900	**.38**	**.65**	.86	1.02
1000	**.35**	**.63**	.90	1.01

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.25**	**1.73**	**2.19**	**2.56**
300	**1.01**	**1.26**	**1.68**	**1.95**
400	**.77**	**1.05**	**1.32**	**1.60**
500	**.62**	**.93**	**1.16**	**1.40**
600	**.58**	**.74**	**1.03**	**1.18**
700	**.47**	**.75**	**.99**	**1.10**
800	**.41**	**.68**	**.80**	**1.03**
900	**.40**	**.59**	**.76**	**.93**
1000	**.38**	**.58**	**.81**	**.85**

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.20**	**1.84**	2.40	**2.74**
300	**.97**	**1.35**	1.86	2.29
400	**.74**	1.22	1.63	1.85
500	**.59**	1.09	1.46	1.74
600	**.58**	.92	1.34	1.58
700	**.45**	.92	1.33	1.44
800	**.41**	.86	1.10	1.43
900	**.39**	.77	1.07	1.25
1000	**.37**	.76	1.07	1.26

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.23**	1.93	2.48	**2.78**
300	**1.01**	1.48	1.96	2.33
400	**.79**	1.32	1.72	1.92
500	**.65**	1.21	1.56	1.80
600	**.60**	1.05	1.43	1.65
700	**.50**	1.10	1.42	1.49
800	**.47**	.99	1.21	1.49
900	**.45**	.92	1.18	1.29
1000	**.42**	.93	1.17	1.31

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.27**	**1.79**	**2.23**	**2.56**
300	**1.02**	**1.32**	**1.69**	**2.03**
400	**.82**	**1.17**	**1.40**	**1.66**
500	**.69**	**1.05**	1.25	**1.44**
600	**.64**	**.88**	1.15	1.30
700	**.55**	**.87**	1.14	**1.13**
800	**.52**	**.80**	.91	1.13
900	**.51**	**.72**	.84	.96
1000	**.49**	**.70**	.90	.94

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.13: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = \frac{n}{2}$ and $B = 500\frac{m}{n}$, number of important variables $s = 5$ and number of factors $K = 5$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	6.46	7.50	8.32	8.93
300	6.27	7.48	8.33	8.88
400	6.39	7.44	8.31	8.81
500	6.31	7.38	8.18	8.82
600	6.35	7.41	8.31	8.85
700	6.29	7.47	8.22	8.85
800	6.34	7.43	8.17	8.82
900	6.41	7.46	8.24	8.87
1000	6.30	7.44	8.25	8.81

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.82**	**1.52**	**3.01**	**6.38**
300	**1.41**	**1.51**	**2.94**	**5.97**
400	**1.49**	**1.59**	**3.08**	**5.61**
500	**1.20**	**1.54**	**2.87**	**5.37**
600	**1.33**	**1.67**	**3.33**	**5.69**
700	**1.57**	**2.02**	**3.05**	**5.83**
800	**1.46**	**1.76**	**3.08**	**5.35**
900	**1.66**	**2.17**	**3.52**	**6.08**
1000	**1.29**	**1.91**	**3.04**	**5.36**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	6.97	**7.49**	**8.17**	**8.82**
300	6.96	7.60	8.35	8.92
400	7.14	7.69	8.32	8.84
500	6.98	7.57	8.21	8.83
600	7.06	7.72	8.37	8.92
700	7.18	7.73	8.39	8.89
800	7.15	7.75	8.35	8.85
900	7.23	7.73	8.41	9.02
1000	7.12	7.64	8.33	8.89

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	6.67	**7.39**	**8.17**	**8.84**
300	6.71	7.51	8.34	8.94
400	6.90	7.59	8.33	8.86
500	6.74	7.50	8.23	8.83
600	6.87	7.64	8.40	8.94
700	6.97	7.67	8.41	8.93
800	6.94	7.68	8.37	8.88
900	7.01	7.67	8.43	9.05
1000	6.89	7.58	8.33	8.91

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	6.92	7.81	8.54	9.11
300	6.99	7.96	8.71	9.24
400	7.25	8.10	8.76	9.17
500	7.09	7.96	8.61	9.14
600	7.21	8.08	8.86	9.25
700	7.34	8.13	8.79	9.25
800	7.26	8.11	8.74	9.20
900	7.39	8.20	8.78	9.37
1000	7.25	8.02	8.71	9.21

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**6.29**	**7.08**	**7.95**	**8.70**
300	6.39	**7.15**	**8.12**	**8.74**
400	6.53	**7.33**	**8.11**	**8.71**
500	6.40	**7.17**	**7.98**	**8.66**
600	6.53	**7.36**	**8.19**	**8.77**
700	6.56	**7.34**	**8.18**	**8.75**
800	6.61	**7.38**	**8.12**	**8.71**
900	6.68	**7.40**	**8.22**	**8.87**
1000	6.60	**7.31**	**8.12**	**8.75**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**6.13**	**7.00**	**7.92**	**8.70**
300	6.27	**7.08**	**8.09**	**8.76**
400	6.42	**7.27**	**8.11**	**8.72**
500	**6.30**	**7.11**	**7.98**	**8.67**
600	6.45	**7.33**	**8.19**	**8.80**
700	6.46	**7.30**	**8.18**	**8.77**
800	6.51	**7.32**	**8.12**	**8.75**
900	6.57	**7.34**	**8.22**	8.90
1000	6.51	**7.26**	**8.11**	**8.76**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**6.26**	**7.45**	**8.29**	8.98
300	6.46	7.62	8.53	9.08
400	6.70	7.73	8.54	9.05
500	6.50	7.63	8.42	9.02
600	6.69	7.77	8.61	9.13
700	6.73	7.84	8.62	9.13
800	6.80	7.81	8.53	9.10
900	6.87	7.84	8.61	9.24
1000	6.74	7.71	8.52	9.10

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.14: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 50$ and $B = 500\frac{m}{n}$, number of important variables $s = 10$ and number of factors $K = 0$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	4.69	5.79	6.75	7.61
300	4.21	5.42	6.53	7.38
400	3.97	5.31	6.37	7.31
500	3.77	5.30	6.40	7.22
600	3.85	5.36	6.37	7.24
700	3.95	5.35	6.42	7.24
800	4.01	5.31	6.40	7.24
900	4.01	5.37	6.41	7.24
1000	3.98	5.21	6.44	7.25

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**2.09**	**1.22**	**.93**	**1.38**
300	**.90**	**.46**	**.40**	**.70**
400	**.51**	**.17**	**.25**	**.75**
500	**.32**	**.07**	**.38**	**.84**
600	**.16**	**.15**	**.36**	**.86**
700	**.24**	**.18**	**.39**	**1.13**
800	**.11**	**.15**	**.47**	**1.09**
900	**.12**	**.16**	**.63**	**1.10**
1000	**.08**	**.13**	**.55**	**1.18**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	5.29	**5.31**	**6.05**	**6.91**
300	5.43	**5.32**	**6.05**	**6.79**
400	5.58	5.42	**6.02**	**6.88**
500	5.49	5.47	**6.11**	**6.85**
600	5.62	5.64	**6.15**	**6.79**
700	5.60	5.62	**6.20**	**6.89**
800	5.68	5.62	**6.35**	**6.89**
900	5.69	5.66	**6.34**	**6.98**
1000	5.65	5.60	**6.35**	**6.94**

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	4.73	**5.20**	**6.06**	**6.92**
300	4.93	**5.13**	**6.03**	**6.80**
400	5.02	**5.29**	**6.03**	**6.89**
500	4.94	5.31	**6.11**	**6.87**
600	5.14	5.52	**6.14**	**6.82**
700	5.17	5.50	**6.22**	**6.90**
800	5.25	5.48	**6.37**	**6.92**
900	5.28	5.54	**6.35**	**7.02**
1000	5.23	5.49	**6.36**	**6.96**

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.58**	**5.34**	**6.21**	**7.01**
300	4.84	5.54	**6.43**	**7.25**
400	5.04	5.65	6.41	7.32
500	5.00	5.74	6.54	7.26
600	5.21	5.93	6.60	7.34
700	5.24	5.90	6.73	7.34
800	5.34	5.95	6.85	7.35
900	5.39	5.98	6.79	7.49
1000	5.33	5.89	6.78	7.37

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.48**	**5.08**	**6.06**	**6.98**
300	4.60	**4.91**	**5.77**	**6.59**
400	4.71	**5.02**	**5.80**	**6.68**
500	4.64	**5.03**	**5.87**	**6.62**
600	4.84	**5.24**	**5.87**	**6.58**
700	4.91	**5.21**	**5.93**	**6.69**
800	4.95	**5.21**	**6.10**	**6.68**
900	5.02	**5.25**	**6.12**	**6.81**
1000	4.96	**5.20**	**6.08**	**6.74**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.22**	**5.07**	**6.11**	**6.97**
300	4.35	**4.82**	**5.74**	**6.58**
400	4.41	**4.91**	**5.78**	**6.68**
500	4.39	**4.93**	**5.82**	**6.64**
600	4.60	**5.13**	**5.85**	**6.60**
700	4.70	**5.14**	**5.93**	**6.72**
800	4.71	**5.14**	**6.08**	**6.72**
900	4.83	**5.19**	**6.12**	**6.85**
1000	4.74	**5.14**	**6.08**	**6.75**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.06**	**5.14**	**6.07**	**6.96**
300	4.22	**5.12**	**6.16**	**7.02**
400	4.38	5.31	**6.17**	**7.10**
500	4.37	5.34	**6.29**	**7.07**
600	4.60	5.60	**6.33**	**7.12**
700	4.71	5.57	6.42	**7.13**
800	4.78	5.57	6.61	**7.17**
900	4.85	5.62	6.55	7.29
1000	4.79	5.57	6.57	**7.20**

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.15: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 100$ and $B = 500\frac{m}{n}$, number of important variables $s = 10$ and number of factors $K = 0$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	4.66	5.84	6.78	7.54
300	3.52	4.52	5.46	6.29
400	2.80	3.62	4.51	5.46
500	2.38	3.19	3.98	4.72
600	2.20	2.77	3.44	4.17
700	2.13	2.58	3.21	3.81
800	1.99	2.35	3.04	3.53
900	1.81	2.14	2.81	3.27
1000	1.70	1.96	2.51	3.05

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**1.96**	**1.34**	**1.09**	**1.25**
300	**1.62**	**1.01**	**.63**	**.46**
400	**1.61**	**.97**	**.54**	**.35**
500	**1.65**	**.95**	**.51**	**.31**
600	**1.58**	**.90**	**.58**	**.26**
700	**1.61**	**.89**	**.51**	**.26**
800	**1.36**	**.90**	**.48**	**.26**
900	**1.44**	**.80**	**.54**	**.22**
1000	**1.39**	**.90**	**.56**	**.30**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	5.28	**5.31**	**6.05**	**6.93**
300	4.82	**4.22**	**4.94**	**5.63**
400	4.51	**3.46**	**4.06**	**4.88**
500	4.35	**3.02**	**3.64**	**4.29**
600	4.33	**2.70**	**3.15**	**3.82**
700	4.29	**2.48**	**2.90**	**3.54**
800	4.26	**2.29**	**2.86**	**3.26**
900	4.23	**2.09**	**2.60**	**3.02**
1000	4.21	**1.87**	**2.29**	**2.91**

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	4.70	**5.19**	**6.04**	**6.93**
300	4.00	**4.09**	**4.95**	**5.64**
400	3.52	**3.34**	**4.05**	**4.88**
500	3.11	**2.87**	**3.64**	**4.32**
600	2.96	**2.62**	**3.17**	**3.82**
700	2.87	**2.40**	**2.93**	**3.56**
800	2.76	**2.20**	**2.87**	**3.25**
900	2.61	**2.01**	**2.63**	**3.04**
1000	2.68	**1.76**	**2.30**	**2.93**

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.51**	**5.31**	**6.24**	**7.01**
300	3.64	**4.23**	**5.04**	**5.80**
400	2.99	**3.44**	**4.15**	**4.99**
500	2.54	**2.95**	**3.71**	**4.33**
600	2.36	**2.69**	**3.19**	**3.88**
700	**2.11**	**2.43**	**2.98**	**3.53**
800	2.04	**2.27**	**2.94**	**3.29**
900	**1.80**	**2.06**	**2.63**	**3.04**
1000	**1.69**	**1.83**	**2.36**	**2.88**

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.47**	**5.09**	**6.06**	**6.97**
300	3.78	**4.02**	**4.95**	**5.77**
400	3.29	**3.27**	**4.14**	**4.95**
500	2.96	**2.86**	**3.70**	**4.49**
600	2.83	**2.62**	**3.22**	**3.94**
700	2.72	**2.41**	**3.00**	**3.67**
800	2.63	**2.20**	**2.97**	**3.43**
900	2.48	**2.02**	**2.72**	**3.22**
1000	2.51	**1.75**	**2.43**	3.10

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.18**	**5.05**	**6.12**	**7.02**
300	**3.39**	**4.04**	**5.00**	**5.82**
400	2.81	**3.30**	**4.18**	**4.98**
500	2.41	**2.88**	**3.77**	**4.53**
600	2.25	**2.63**	**3.30**	**3.98**
700	**2.02**	**2.43**	**3.08**	**3.75**
800	**1.98**	**2.26**	**3.02**	**3.46**
900	**1.79**	**2.06**	**2.78**	**3.26**
1000	**1.67**	**1.84**	**2.48**	3.18

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.01**	**5.13**	**6.05**	**6.92**
300	**3.16**	**4.03**	**4.96**	**5.66**
400	**2.55**	**3.27**	**4.08**	**4.92**
500	**2.10**	**2.84**	**3.63**	**4.29**
600	**2.00**	**2.66**	**3.18**	**3.83**
700	**1.74**	**2.42**	**2.97**	**3.54**
800	**1.69**	**2.22**	**2.89**	**3.26**
900	**1.49**	**2.03**	**2.62**	**3.02**
1000	**1.37**	**1.78**	**2.32**	**2.88**

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.16: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = \frac{n}{2}$ and $B = 500\frac{m}{n}$, number of important variables $s = 10$ and number of factors $K = 0$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	7.23	8.05	8.77	9.35
300	7.04	8.02	8.74	9.27
400	7.02	7.90	8.68	9.26
500	6.83	7.88	8.62	9.12
600	6.96	7.94	8.69	9.11
700	7.14	7.90	8.63	9.11
800	6.96	7.97	8.63	9.14
900	7.01	7.94	8.68	9.20
1000	6.94	7.90	8.55	9.10

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**2.49**	**1.94**	**4.04**	**8.38**
300	**1.55**	**1.73**	**3.26**	**7.48**
400	**1.93**	**1.52**	**2.76**	**7.14**
500	**1.19**	**1.01**	**2.51**	**5.46**
600	**1.68**	**1.38**	**2.94**	**5.52**
700	**2.09**	**1.45**	**2.65**	**5.59**
800	**1.57**	**1.38**	**2.41**	**5.28**
900	**1.44**	**1.44**	**2.84**	**6.09**
1000	**1.66**	**1.41**	**2.13**	**5.01**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	7.37	**7.97**	**8.67**	**9.24**
300	7.41	8.17	8.79	9.33
400	7.48	8.15	8.81	9.39
500	7.44	8.09	8.74	9.26
600	7.55	8.17	8.83	9.24
700	7.66	8.25	8.83	9.31
800	7.51	8.20	8.80	9.31
900	7.67	8.26	8.89	9.37
1000	7.55	8.13	8.73	9.26

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**7.13**	**7.88**	**8.68**	**9.26**
300	7.24	8.09	8.80	9.33
400	7.26	8.08	8.83	9.39
500	7.25	8.02	8.75	9.27
600	7.39	8.09	8.85	9.27
700	7.50	8.16	8.85	9.33
800	7.29	8.12	8.82	9.33
900	7.47	8.20	8.90	9.39
1000	7.33	8.06	8.76	9.28

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	7.42	8.26	9.00	9.51
300	7.50	8.47	9.12	9.54
400	7.61	8.51	9.14	9.62
500	7.59	8.45	9.09	9.48
600	7.74	8.61	9.17	9.53
700	7.84	8.55	9.18	9.57
800	7.71	8.59	9.16	9.56
900	7.89	8.61	9.21	9.65
1000	7.75	8.51	9.10	9.51

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**6.75**	**7.63**	**8.50**	**9.13**
300	**6.92**	**7.83**	**8.59**	**9.23**
400	**6.91**	**7.76**	**8.61**	**9.25**
500	6.95	**7.77**	**8.53**	**9.10**
600	7.01	**7.78**	**8.64**	9.12
700	**7.11**	**7.86**	**8.60**	9.20
800	6.96	**7.82**	**8.60**	9.18
900	7.13	**7.90**	8.71	9.28
1000	6.97	**7.76**	**8.55**	9.15

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**6.63**	**7.56**	**8.48**	**9.13**
300	**6.83**	**7.76**	**8.58**	**9.23**
400	**6.84**	**7.71**	**8.62**	9.27
500	6.85	**7.71**	**8.53**	**9.11**
600	**6.95**	**7.74**	**8.65**	9.15
700	**7.05**	**7.84**	**8.61**	9.23
800	**6.91**	**7.78**	**8.59**	9.19
900	7.06	**7.87**	8.73	9.29
1000	**6.92**	**7.73**	8.55	9.16

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**6.83**	**7.94**	8.81	9.38
300	7.04	8.19	8.95	9.43
400	7.10	8.19	8.99	9.53
500	7.10	8.15	8.89	9.41
600	7.27	8.24	9.01	9.42
700	7.34	8.32	9.01	9.49
800	7.20	8.28	8.97	9.48
900	7.38	8.34	9.08	9.56
1000	7.20	8.22	8.91	9.44

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.17: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 50$ and $B = 500\frac{m}{n}$, number of important variables $s = 10$ and number of factors $K = 5$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	5.43	6.53	7.47	8.34
300	5.07	6.17	7.23	8.05
400	4.67	5.92	7.06	7.95
500	4.65	6.05	7.05	7.72
600	4.55	5.84	6.99	7.76
700	4.54	5.96	6.99	7.85
800	4.48	5.98	6.97	7.82
900	4.55	6.03	7.10	7.86
1000	4.56	5.89	7.03	7.71

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**2.13**	**1.57**	**1.13**	**1.89**
300	**1.01**	**.56**	**.30**	**.60**
400	**.56**	**.13**	**.21**	**.31**
500	**.44**	**.14**	**.23**	**.24**
600	**.31**	**.13**	**.16**	**.50**
700	**.28**	**.15**	**.16**	**.43**
800	**.25**	**.23**	**.20**	**.32**
900	**.12**	**.21**	**.24**	**.49**
1000	**.20**	**.19**	**.26**	**.15**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	5.64	**5.88**	**6.80**	**7.60**
300	5.84	**5.91**	**6.66**	**7.45**
400	5.93	5.95	**6.72**	**7.50**
500	5.93	6.11	**6.77**	**7.34**
600	5.96	5.99	**6.71**	**7.38**
700	5.95	6.13	**6.84**	**7.53**
800	5.93	6.16	**6.89**	**7.55**
900	6.03	6.21	**6.92**	**7.58**
1000	5.95	6.06	**6.87**	**7.43**

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**5.21**	**5.78**	**6.76**	**7.59**
300	5.38	**5.76**	**6.62**	**7.45**
400	5.50	**5.83**	**6.72**	**7.51**
500	5.49	**5.97**	**6.78**	**7.37**
600	5.50	5.87	**6.72**	**7.42**
700	5.55	6.02	**6.83**	**7.55**
800	5.54	6.06	**6.91**	**7.58**
900	5.67	6.08	**6.95**	**7.63**
1000	5.55	5.95	**6.88**	**7.47**

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**5.09**	**5.93**	**6.91**	**7.67**
300	5.42	**6.17**	**7.12**	**7.83**
400	5.58	6.20	7.19	**7.89**
500	5.59	6.39	7.20	7.79
600	5.64	6.35	7.24	7.84
700	5.70	6.47	7.27	7.99
800	5.72	6.45	7.35	8.03
900	5.85	6.55	7.41	8.01
1000	5.76	6.41	7.31	7.91

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.99**	**5.70**	**6.80**	**7.71**
300	**5.05**	**5.50**	**6.37**	**7.21**
400	5.16	**5.55**	**6.39**	**7.28**
500	5.16	**5.68**	**6.54**	**7.18**
600	5.17	**5.58**	**6.46**	**7.16**
700	5.25	**5.72**	**6.54**	**7.31**
800	5.24	**5.74**	**6.62**	**7.38**
900	5.36	**5.76**	**6.61**	**7.38**
1000	5.24	**5.66**	**6.58**	**7.23**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.72**	**5.66**	**6.80**	**7.72**
300	**4.80**	**5.41**	**6.34**	**7.20**
400	4.96	**5.42**	**6.38**	**7.29**
500	4.94	**5.58**	**6.52**	**7.20**
600	4.97	**5.49**	**6.43**	**7.18**
700	5.01	**5.65**	**6.54**	**7.33**
800	5.05	**5.68**	**6.62**	**7.40**
900	5.20	**5.71**	**6.63**	**7.41**
1000	5.06	**5.60**	**6.58**	**7.24**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.55**	**5.73**	**6.75**	**7.53**
300	**4.74**	**5.79**	**6.76**	**7.65**
400	4.97	**5.85**	**6.88**	**7.72**
500	4.97	**6.03**	**6.92**	**7.58**
600	5.02	5.92	**6.94**	**7.65**
700	5.13	6.11	7.05	**7.78**
800	5.14	6.14	7.11	7.83
900	5.29	6.21	7.14	**7.83**
1000	5.15	6.06	7.09	7.73

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.18: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = 100$ and $B = 500\frac{m}{n}$, number of important variables $s = 10$ and number of factors $K = 5$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	5.47	6.51	7.54	8.35
300	4.33	5.22	6.24	7.08
400	3.48	4.29	5.34	6.09
500	2.96	3.80	4.65	5.26
600	2.59	3.35	4.08	4.78
700	2.27	2.96	3.74	4.24
800	2.16	2.72	3.46	3.96
900	1.98	2.45	3.13	3.73
1000	1.83	2.32	2.90	3.48

(a) RBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	**2.21**	**1.48**	**1.11**	**1.83**
300	**1.95**	**1.13**	**.67**	**.45**
400	**1.59**	**1.00**	**.57**	**.30**
500	**1.73**	**.99**	**.56**	**.33**
600	**1.70**	**.97**	**.56**	**.32**
700	**1.70**	**.90**	**.49**	**.28**
800	**1.53**	**.91**	**.50**	**.28**
900	**1.52**	**.91**	**.50**	**.26**
1000	**1.46**	**.90**	**.51**	**.26**

(b) IRBVS PC

$n\backslash p$	10^2	10^3	10^4	10^5
200	5.65	**5.86**	**6.81**	**7.61**
300	5.02	**4.74**	**5.54**	**6.29**
400	4.71	**4.08**	**4.71**	**5.33**
500	4.49	**3.57**	**4.13**	**4.78**
600	4.38	**3.15**	**3.68**	**4.33**
700	4.29	**2.86**	**3.39**	**3.80**
800	4.23	**2.59**	**3.17**	**3.59**
900	4.29	**2.35**	**2.90**	**3.41**
1000	4.24	**2.17**	**2.71**	**3.21**

(c) StabSel PC $\pi = 0.55$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**5.22**	**5.74**	**6.78**	**7.60**
300	4.35	**4.63**	**5.52**	**6.29**
400	3.88	**3.99**	**4.72**	**5.33**
500	3.50	**3.44**	**4.14**	**4.79**
600	3.26	**2.97**	**3.67**	**4.33**
700	3.07	**2.74**	**3.39**	**3.80**
800	2.96	**2.52**	**3.18**	**3.60**
900	2.80	**2.26**	**2.92**	**3.42**
1000	2.71	**2.09**	**2.72**	**3.22**

(d) StabSel PC $\pi = 0.6$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**5.10**	**5.89**	**6.90**	**7.66**
300	**4.11**	**4.78**	**5.70**	**6.42**
400	3.50	**4.10**	**4.81**	**5.42**
500	3.05	**3.55**	**4.25**	**4.79**
600	2.70	**3.08**	**3.76**	**4.35**
700	2.49	**2.84**	**3.46**	**3.85**
800	2.30	**2.58**	**3.24**	**3.66**
900	2.11	**2.32**	**2.94**	**3.47**
1000	1.92	**2.14**	**2.73**	**3.22**

(e) StabSel PC $\pi = 0.75$ $EV = 2.5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.96**	**5.69**	**6.82**	**7.72**
300	**4.13**	**4.52**	**5.52**	**6.39**
400	3.65	**3.91**	**4.78**	**5.46**
500	3.32	**3.40**	**4.24**	**4.98**
600	3.09	**2.94**	**3.70**	**4.53**
700	2.93	**2.68**	**3.46**	**3.92**
800	2.80	**2.55**	**3.25**	**3.73**
900	2.69	**2.24**	**3.03**	**3.56**
1000	2.57	**2.09**	**2.82**	**3.34**

(f) StabSel PC $\pi = 0.55$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.70**	**5.70**	**6.87**	**7.72**
300	**3.78**	**4.51**	**5.60**	**6.39**
400	**3.29**	**3.89**	**4.84**	**5.49**
500	**2.89**	**3.43**	**4.26**	**5.05**
600	**2.55**	**2.95**	**3.75**	**4.56**
700	2.40	**2.73**	**3.51**	**3.97**
800	2.23	**2.59**	**3.31**	**3.77**
900	2.03	**2.28**	**3.08**	**3.61**
1000	1.89	**2.17**	2.92	**3.38**

(g) StabSel PC $\pi = 0.6$ $EV = 5$

$n\backslash p$	10^2	10^3	10^4	10^5
200	**4.53**	**5.71**	**6.77**	**7.57**
300	**3.54**	**4.56**	**5.52**	**6.31**
400	**3.04**	**3.91**	**4.72**	**5.33**
500	**2.67**	**3.39**	**4.13**	**4.77**
600	**2.28**	**2.96**	**3.65**	**4.32**
700	**2.08**	**2.67**	**3.39**	**3.77**
800	**1.96**	**2.54**	**3.19**	**3.62**
900	**1.75**	**2.23**	**2.94**	**3.43**
1000	**1.62**	**2.10**	**2.73**	**3.19**

(h) StabSel PC $\pi = 0.75$ $EV = 5$

Table 3.19: High-dimensional example: the average number of FP+FN (False Positives and False Negatives) calculated over 500 realisations with $m = \frac{n}{2}$ and $B = 500\frac{m}{n}$, number of important variables $s = 10$ and number of factors $K = 5$. Bold: result **better** than the corresponding value for RBVS PC.

$n\backslash p$	10^2	10^3	10^4	10^5
200	.04	.08	.41	2.55
300	.05	.10	.51	2.80
400	.05	.11	.50	3.13
500	.05	.13	.60	3.56
600	.06	.14	.69	6.81
700	.07	.16	.81	7.63
800	.07	.18	.91	8.12
900	.08	.21	1.05	9.39
1000	.09	.24	1.21	9.94

(a) RBVS PC, $m = 50$

$n\backslash p$	10^2	10^3	10^4	10^5
200	.14	.29	1.50	10.07
300	.14	.31	1.76	10.55
400	.14	.33	1.73	11.90
500	.16	.36	1.96	13.07
600	.16	.39	2.18	23.01
700	.17	.43	2.43	25.39
800	.18	.48	2.70	26.73
900	.20	.54	3.06	29.37
1000	.21	.60	3.52	30.36

(b) IRBVS PC, $m = 50$

$n\backslash p$	10^2	10^3	10^4	10^5
200	.05	.10	.55	3.54
300	.05	.11	.64	4.01
400	.05	.13	.63	4.55
500	.06	.14	.72	5.09
600	.06	.16	.83	8.39
700	.07	.18	.95	9.09
800	.08	.20	1.03	9.33
900	.08	.22	1.17	10.75
1000	.09	.25	1.35	11.32

(c) RBVS PC, $m = 100$

$n\backslash p$	10^2	10^3	10^4	10^5
200	.15	.32	1.78	12.79
300	.14	.31	1.90	13.59
400	.14	.32	1.81	15.30
500	.15	.34	1.98	16.95
600	.15	.37	2.25	24.75
700	.16	.41	2.49	26.01
800	.16	.45	2.60	26.48
900	.17	.49	2.98	29.63
1000	.18	.55	3.38	30.50

(d) IRBVS PC, $m = 100$

$n\backslash p$	10^2	10^3	10^4	10^5
200	.05	.10	.55	3.47
300	.05	.13	.82	5.15
400	.06	.16	.95	7.07
500	.07	.19	1.19	8.86
600	.08	.22	1.40	14.10
700	.08	.25	1.58	16.59
800	.09	.28	1.70	16.33
900	.10	.32	1.90	18.17
1000	.11	.36	2.30	20.30

(e) RBVS PC, $m = \frac{n}{2}$

$n\backslash p$	10^2	10^3	10^4	10^5
200	.15	.32	1.78	12.14
300	.17	.37	2.42	16.06
400	.17	.43	2.69	20.69
500	.19	.48	3.25	25.07
600	.20	.52	3.64	37.56
700	.21	.55	3.91	41.85
800	.22	.60	4.00	40.40
900	.24	.67	4.31	43.68
1000	.25	.74	5.30	47.96

(f) IRBVS PC, $m = \frac{n}{2}$

Table 3.20: High-dimensional example: average computation times achieved with the **rbvsGPU** package (Baranowski, 2016), calculated over 500 realisations with $B = 500\frac{m}{n}$, number of important variables $s = 5$ and number of factors $K = 0$. In a single run, computations are performed in parallel, using a Nvidia Quadro 4000 GPU and a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM.

- When the subsample size is set to $\frac{m}{2}$, there typically exists a set of parameters for StabSel such that it slightly outperforms RBVS. We have checked that RBVS in this setting selects slightly more false positives.
- Overall, performance of StabSel is sensitive to the choice of its parameter.
- "Optimal" parameters for StabSel in one example are not necessarily best in another case. For instance, in the $s = 5$, $K = 0$ and $m = \frac{n}{2}$ case $\pi = 0.75$ and $EV = 2.5$ results in the best error control, while for $s = 5$, $K = 0$ and $m = 50$ setting $EV = 5$ and $\pi = 0.6$ yields best $FP + FN$ rate.
- IRBVS almost uniformly outperforms both RBVS and StabSel, which demonstrates that the iterative extension of our methodology significantly improves its vanilla variant.

2. General comments on the impact of "high-dimensionality":

   - Perhaps a bit unexpectedly, performance of the IRBVS algorithm improves with dimensionality p growing. This phenomenon can be explained by the fact that a single irrelevant covariate is the less likely to appear at the top of the ranking, the more covariates with similar (spurious) impact on the response there are. We note that this surprising "blessing of dimensionality" has been observed in Fan et al. (2009).
   - IRBVS performs very well even for small/moderate values of n and m, even when p is very large.

3. Comments on the choice of the subsample size m:

   - For the IRBVS algorithm, $m = 100$ yields best $FP + FN$ in this example, often close to 0. On the other hand, choosing $\frac{m}{2}$ results in IRBVS occasionally picking some irrelevant covariates. We emphasise again, however, that IRBVS consistently outperforms RBVS nad StabSel.

- For the RBVS and StabSel algorithms, $m = \frac{n}{2}$ leads to best performance.
- The subsample size set to a small number ($m = 50$) results in a worse selection of the important variables.

3.9 Computational aspects

3.9.1 Details of the implementation of the RBVS algorithm

In this section, we provide a detailed description of our implementation of Algorithm 3.3, which is available in the R package **rbvs**. First we recall all necessary notation. By $\mathbf{Z}_i = (Y_i, X_{i1}, X_{ip})$, $i = 1, \dots n$ we denote a random sample we observe, where Y_i is a response and $\{X_{i1,}, \dots, X_{ip}\}$ is the set of the covariates. A chosen (empirical) measure of dependence between the response and $j'th$ covariates is denoted by $\hat{\omega}_j$, positive integer $m < n$ is a subsample size (parameter of our method), B is a positive integer (typically $B = 100, 500$).

The RBVS algorithm aims to identify the set of covariates which non-spuriously appears at the top of the variable ranking based on the empirical measure $\hat{\omega}_j$. It consists of four steps. Implementation of Step 1 is straightforward. It is worth noting that in Step 2 we do not actually need to evaluate complete rankings for each subsample, it is sufficient to find only a partial ranking, i.e. indices of the k_{max} top ranked variables, as only those are used in 3. The computational complexity of finding a full ranking is $O(p \log(p))$. For the partial ranking, it takes (on average) just $O(p + k_{max} \log(k_{max}))$ operations. The gain can be substantial when $p >> k_{max}$.

Let us recall that $\hat{\mathcal{A}}_{k,m} = \text{argmax}_{\mathcal{A} \in \Omega_k} \hat{\pi}_{m,n}(\mathcal{A})$, where Ω_k is the set of all k-element subsets of $\{1, \dots, p\}$. Despite the fact that the definition involves searching of the maximum empirical probability over a set the size of which grows extremely fast, finding $\hat{\mathcal{A}}_{k,m}$ is actually quick. This is because the number of the subsets which could have

appeared at the top of the ranking at least once is limited by the total number of evaluated rankings. In Step 3, we apply procedure outlined in Algorithm 3.4.

Algorithm 3.5 Top-ranked sets

Input: Variable rankings $(R_{l1}, \ldots, R_{lk_{max}})$, $l = 1, \ldots, Br$.
Output: Estimates $\hat{\mathcal{A}}_{k,m}$ and $\hat{\pi}_{m,n}\left(\hat{\mathcal{A}}_{k,m}\right)$ for $k = 1, \ldots, k_{max}$.
procedure kTopRankedSets($\{(R_{l1}, \ldots, R_{lk_{max}}\}_{l=1}^{Br}$)
for $k = 1, \ldots, k_{max}$ **do**
Step 1 for each l, insert R_{lk} into $S_{l,k-1}$ s.t. resulting sequence $S_{l,k}$ is in increasing order
Step 2 find S_k^* the most frequently occurring among $S_{1,k}, \ldots, S_{Br,k}$
Step 3 set $\hat{\mathcal{A}}_{k,m} = S_k^*$ and $\hat{\pi}_{m,n}\left(\hat{\mathcal{A}}_{k,m}\right) = \frac{no.\ l\ s.t.\ S_{l,k}=S_k^*}{Br}$
end for
end procedure

The computational complexity of Step 1 is of order $O(nBr)$ (for each k we use the fact that at the previous step $k-1$ elements are already in increasing order; we do not need to sort $R_{1,l}, \ldots, R_{k,l}$ from scratch). The second part is relatively quick – we need to find the most frequent element among k-element sequences. For each k, the computational complexity is $O(kBr)$. Therefore with $k_{max} = n$ in total the algorithm we use to find $\hat{\mathcal{A}}_{k,m}$ is of order $O(\max\{n^2, nrB\})$. Algorithm 3.5 can be easily run on multiple CPUs (which is supported by the **rbvs** package) or a GPU, which makes it feasible for extremely large data sets. In practice, Step 3 of the RBVS algorithm (Algorithm 3.3) takes much less computational time than Step 2. Moreover, the **rbvs** package provides optimised, C-implemented routines performing Algorithm 3.3, hence in particular Algorithm 3.5.

3.9.2 Algorithmic differences between RBVS and StabSel

RBVS and StabSel algorithms are based on the idea that to one can repeatedly apply a favourite variable ranking or, respectively, variable selection algorithm, and aggregate results to obtain "better" feature selection. In this section we describe the differences between the two approaches, focusing on the algorithmic side of the problem. We

	RBVS	StabSel
covariates are seen	from the most to the least important	as either relevant or irrelevant
no. model complexity parameters	0	2
no. other parameters	2	2
error control	no	yes
iterative extension	yes	no

Table 3.21: Informal comparison of the RBVS and StabSel algorithms.

demonstrate how to use rankings obtained in the RBVS algorithm to perform Stability Selection.

To ease presentation, we assume in this section that dependence between the response and covariates is measured with $\hat{\omega}_j$ equal to the absolute value of the regression coefficient estimated using a penalised likelihood method, e.g. Lasso. Moreover, we assume that the penalty is always selected such that there are exactly $k_{max} \in \{1, \ldots, p\}$ non-zero coefficients, where k_{max} is the same as in the RBVS algorithm (see Section 3.3.5).

The initial steps of the RBVS and StabSel algorithms are similar, i.e. we evaluate the measure of dependence over subsamples from the data. Let $I_l \subset \{1, \ldots, n\}$, $l = 1, \ldots, Br$ denote the indices of such randomly drawn subsamples of size $1 \leq m \leq n$, by $\hat{\omega}_{lj}$ denote the measure of dependence calculated over I_l. Subsequent steps, however, are different. In the RBVS algorithm, we sort $\hat{\omega}_{lj}$ and obtain variable rankings $(R_{l1}, \ldots, R_{lk_{max}})$. Next we find subsets which appear at the top of the rankings frequently and based on the frequencies we select the important variables. This way, we use full information on the magnitude of $\hat{\omega}_j$.

In Stability Selection, on the other hand, after calculating the measures of dependence, we find $S_l = \{j : \hat{\omega}_{lj} \neq 0\}$, therefore at this stage we completely ignore the information on the order of $\hat{\omega}_j's$. Subsequently, we find the proportions $\hat{\pi}_j$ equal to the number of sets S_l containing j. The covariates for which $\hat{p}_j$ exceeds prespecified threshold form the final set of the selected predictors.

Table 3.21 heuristically compares RBVS and StabSel methodologies.

3.9.3 Simulation code

The list below summarises how all methods consider in Section 3.6 have been executed.

- In **Model (A)**–**Model (E)**:
  - Lasso/MC+ CV: we used the cross-validation routine from the **ncvreg** (Breheny and Huang, 2011). The execution code is as follows.

```
cv.ncvreg(x, y, penalty="lasso")  # Lasso penalty
cv.ncvreg(x, y, penalty="MCP")   #MC+ penalty
```

  - (I)SIS Lasso/MC+: we used the R package **SIS** executing the following code.

```
SIS(x,y, penalty="lasso") # SIS Lasso
SIS(x,y, penalty="MCP") # SIS MC+
SIS(x,y, penalty="lasso", iter=TRUE, varISIS="aggr") # ISIS Lasso
SIS(x,y, penalty="MCP", varISIS="aggr") # ISIS MC+
```

    Note that we used an "aggresive" variant of the ISIS algorithm (for details see Saldana and Feng (2014)), which was basically much better than a standard ISIS in our simulations. Unfortunately, the **SIS** package does not allow for sample splitting in the standard SIS procedure.

  - (I)RBVS PC/Lasso/MC+: these routines, implemented in the **rbvs** package, were executed as follows.

```
p <- ncol(x)
thr <- 0.6 # StabSel thresholding probability
EV <- 2.5 # StabSel bound
d <- ceiling(sqrt(p*EV*(2*thr-1)))
# RBVS PC
rbvs(x,y,measure="pc", iter=FALSE, k.max=p)
# RBVS Lasso
```

```
rbvs(x,y,measure="lasso", iter=FALSE, k.max=p, nonzero=p)
# RBVS MC+
rbvs(x,y,measure="mcplus", iter=FALSE, k.max=p, nonzero=p)
# IRBVS PC
rbvs(x,y,measure="pc", iter=TRUE, k.max=p, nonzero=p)
# IRBVS Lasso
rbvs(x,y,measure="lasso", iter=TRUE, k.max=p, nonzero=p)
# IRBVS MC+
rbvs(x,y,measure="mcplus", iter=TRUE, k.max=p, nonzero=p)
```

Note that we set the *nonzero* parameter as in the StabSel algorithm. This means that the tuning parameter λ for MC+ and Lasso algorithms is chosen such that *nonzero* out of p coefficients are different than zero.

– StabSel PC/Lasso/MC+: for these method we used the **rbvs** package. The example code below demonstrates how we do this with the marginal correlation used as a base learner.

```
p <- ncol(x)
thr <- 0.6 # StabSel thresholding probability
EV <- 2.5 # StabSel bound
d <- ceiling(sqrt(p*EV*(2*thr-1)))
# RBVS PC
rbvs.object <- rbvs(x,y,measure="pc", iter=FALSE, k.max=p)
pb <- rep(0, p) # vector with StabSel probabilities
for(j in 1:ncol(ranks)) pb[ranks[,j]] <- pb[ranks[,j]] + 1
active.stabsel <- which(pb>thr)
```

In Section 3.9.2 we discuss this approach in a greater detailed.

- In **Model (E)**: in this example we used the logistic regression model for Lasso/MC+. The code can be found below.

– Lasso/MC+ CV:

```
cv.ncvreg(x, y, penalty="lasso", family="binomial")  # Lasso penalty
cv.ncvreg(x, y, penalty="MCP", family="binomial")   #MC+ penalty
```

– (I)SIS Lasso/MC+:

```
SIS(x,y, penalty="lasso", family="binomial") # SIS Lasso
SIS(x,y, penalty="MCP", family="binomial") # SIS MC+
# ISIS Lasso
SIS(x,y, penalty="lasso", family="binomial", iter=TRUE,
varISIS="aggr")
SIS(x,y, penalty="MCP", family="binomial", varISIS="aggr") # ISIS MC+
```

– (I)RBVS PC/Lasso/MC+:

```
p <- ncol(x)
thr <- 0.6 # StabSel thresholding probability
EV <- 2.5 # StabSel bound
d <- ceiling(sqrt(p*EV*(2*thr-1)))
rbvs(x,y,measure="pc", iter=FALSE, k.max=p)   # RBVS PC
rbvs(x,y,measure="dc", iter=FALSE, k.max=p)   # RBVS DC
rbvs(x,y,measure="lasso", family="binomial" iter=FALSE,
 k.max=p, nonzero=p)   # RBVS Lasso
rbvs(x,y,measure="mcplus", family="binomial", iter=FALSE,
 k.max=p, nonzero=p)   # RBVS MC+
rbvs(x,y,measure="pc", iter=TRUE, k.max=p, nonzero=p)   # IRBVS PC
rbvs(x,y,measure="dc", iter=TRUE, k.max=p)   # IRBVS DC
rbvs(x,y,measure="lasso",  family="binomial", iter=TRUE,
 k.max=p, nonzero=p)   # IRBVS Lasso
rbvs(x,y,measure="mcplus",  family="binomial", iter=TRUE,
 k.max=p, nonzero=p)   # IRBVS MC+
```

– StabSel PC/Lasso/MC+: we used the **rbvs** package.

3.10 Proofs

3.10.1 Proof of Proposition 3.3.1

Proof. First, we show that $\pi_n(\mathcal{S})$ tends to 1. Indeed, using (C3) we have

$$\pi_n(\mathcal{S}) = \mathbb{P}\left(\min_{j=1,\dots,s} \hat{\omega}_j > \min_{j=s+1,\dots,p} \hat{\omega}_j\right) \geq \mathbb{P}\left(\max_{j=1,\dots,p} |\hat{\omega}_j - \omega_j| < \epsilon\right),$$

where $\epsilon = \frac{c_\eta n^{-\eta}}{2}$. Application of Bonferroni's inequality yields that

$$\mathbb{P}\left(\max_{j=1,\dots,p} |\hat{\omega}_j - \omega_j| < \epsilon\right) \geq 1 - p \sup_{j=1,\dots,p} \mathbb{P}\left(|\hat{\omega}_j - \omega_j| \geq \epsilon\right).$$

The last term is of order $1 - O\left(\exp\left(-n^{\gamma - b_1}\right)\right)$, which tends to 1 as n approaches infinity. This proves that $\mathcal{S}$ is a s-top-ranked set.

Consider any $\mathcal{A} \in \Omega_{k+1}$. We will prove that $\pi_n(\mathcal{A}) \underset{n}{\to} 0$. Denote by

$$\mathcal{E} = \{\min_{j=1,\dots,s} \hat{\omega}_j > \min_{j=s+1,\dots,p} \hat{\omega}_j\}.$$

When $\mathcal{S} \not\subset \mathcal{A}$, we have

$$\mathbb{P}\left(\min_{j\in\mathcal{A}} \hat{\omega}_j > \max_{j\notin\mathcal{A}} \hat{\omega}_j\right) \leq \mathbb{P}\left(\mathcal{E}^c\right) = 1 - \pi_n(\mathcal{S}) \underset{n}{\to} 0.$$

Now consider the case $\mathcal{S} \subset \mathcal{A}$ in which $\mathcal{A} \setminus \mathcal{S}$ has only one element denoted by a. On event $\mathcal{E}$ we have

$$\mathbb{P}\left(\left\{\min_{j\in\mathcal{A}} \hat{\omega}_j > \max_{j\notin\mathcal{A}} \hat{\omega}_j\right\} \cap \mathcal{E}\right) = \mathbb{P}\left(\left\{\hat{\omega}_a > \max_{j\notin\mathcal{A}} \hat{\omega}_j\right\} \cap \mathcal{E}\right),$$

hence it is sufficient to bound $\mathbb{P}\left(\hat{\omega}_a > \max_{j \notin \mathcal{A}} \hat{\omega}_j\right)$. Let us observe that

$$\mathbb{P}\left(\hat{\omega}_a > \max_{j \notin \mathcal{A}} \hat{\omega}_j\right) \leq \mathbb{P}\left(\hat{\omega}_a > \max_{j \in \mathcal{M}_a \setminus \{a\}} \hat{\omega}_j\right).$$

Using (C2), we have

$$\mathbb{P}\left(\hat{\omega}_a > \max_{j \in \mathcal{M}_a \setminus \{a\}} \hat{\omega}_j\right) \leq \frac{1}{|\mathcal{M}_a|}.$$

Eventually,

$$\pi_n(\mathcal{A}) \leq \mathbb{P}\left(\hat{\omega}_a > \max_{j \notin \mathcal{A}} \hat{\omega}_j\right) + \mathbb{P}\left(\mathcal{E}^c\right) \underset{n}{\to} 0.$$

To conclude that $\mathcal{S}$ is unique, we make the following observations.

1. Using (C2) and arguments analogical to those above, we can prove that $\pi_n(\mathcal{A}) \underset{n}{\to} 0$ for any $\mathcal{A} \subset \{1, \ldots, p\}$ such that $p > |\mathcal{A}| > s$.

2. On the other hand, $\pi_n(\mathcal{A}) \underset{n}{\to} 0$ for any $\mathcal{A} \subset \{1, \ldots, p\}$ such that $|\mathcal{A}| \leq s$ and there exists $a \in \mathcal{A}$, $a \notin \mathcal{S}$. This follows from $\pi_n(\mathcal{S}) \underset{n}{\to} 1$.

3. For any $k = 0, \ldots, s$, there exists k-top-ranked set. Indeed, let us take $k \leq s$ and consider $\mathcal{A}_n^* = \operatorname{argmax}_{\mathcal{A} \subset \{1,\ldots,s\}, |\mathcal{A}|=k} \pi_n(\mathcal{A})$. We have $\sum_{\mathcal{A} \subset \{1,\ldots,s\}, |\mathcal{A}|=k} \pi_n(\mathcal{A}) \underset{n}{\to} 1$, hence $\liminf_{n \to \infty} p(\mathcal{A}^*) > 0$, as s is bounded in n.

4. $\mathcal{S}$ is the only s-top-ranked set, as $\pi_n(\mathcal{S}) \underset{n}{\to} 1$ and $\sum_{\mathcal{A} \in \Omega_s} \pi_n(\mathcal{A}) = 1$.

□

3.10.2 Proof of Theorem 3.4.1 and discussion of some of its aspects.

One of the challenges in proving Theorem 3.4.1 is that $\hat{\pi}_{m,n}(\mathcal{A})$'s are not consistent estimators for $\pi_{m,n}(\mathcal{A})$'s when p is growing with n. This is due to the fact that the total number of subsets $\mathcal{A} \subset \{1,\ldots,p\}$ whose empirical probability $\hat{\pi}(\mathcal{A})_{m,n}$ is greater than zero cannot be greater than the number of subsamples used in the RBVS algorithm (i.e. rB). Despite this difficulty, $\max_{\mathcal{A}\in\Omega_k} \hat{\pi}_{m,n}(\mathcal{A})$ has a desirable property, i.e. it is small (with a large probability) when its population counterpart $\max_{\mathcal{A}\in\Omega_k} \pi_{m,n}(\mathcal{A})$ is small. On the other hand, the probability $\hat{\pi}_{m,n}(\mathcal{S})$ is significantly larger than the corresponding estimate for any other set, provided that $\pi_{m,n}(\mathcal{S})$ and n are big enough. Therefore $\mathcal{S}$ appears at the top of the ranking consistently over subsamples provided that $\pi_{m,n}(\mathcal{S})$ is significantly larger than 0.

The proof of Theorem 3.4.1 below begins by showing that the unknown probabilities $\pi_{m,n}(\mathcal{A})$ for $\mathcal{A} \subset \{1,\ldots,p\}$ such that $|\mathcal{A}| > s$, are uniformly small, and $\pi_{m,n}(\mathcal{S})$ converges to 1 at an exponential rate. Consequently, the size of the gap between $\pi_{m,n}(\mathcal{S})$ and $\pi_{m,n}(\mathcal{A})$ is 'large enough' when n is large. Precisely, we take n such that $\frac{\pi_{m,n}(\mathcal{S})}{\pi_{m,n}(\mathcal{A})} > (Br)^{2\alpha-\frac{1}{3}}$ and show that the estimators $\hat{\pi}_{m,n}(\mathcal{A})$ with large probability exhibit similar behaviour. To this end, we use the following Lemma's.

Lemma 3.10.1 (Proposition 2.4, Arcones and Giné (1993)). *Let $W_1,\ldots,W_B$ be binomial r.v. with the probability of success π and r trials. For any $1 > t > \pi$, we have*

$$\mathbb{P}\left(\frac{1}{B}\sum_{i=1}^{B} W_i \geq rt\right) \leq \left(\frac{\pi}{t}\right)^{rt}\left(\frac{1-\pi}{1-t}\right)^{r(1-t)},$$

for $0 < t < \pi$,

$$\mathbb{P}\left(\frac{1}{B}\sum_{i=1}^{B} W_i \leq rt\right) \leq \left(\frac{\pi}{t}\right)^{rt}\left(\frac{1-\pi}{1-t}\right)^{r(1-t)}.$$

Lemma 3.10.2. *Let $a_1, \ldots, a_l$ be non-negative numbers s.t. $\sum_{i=1}^{l} a_i \leq 1$ and $\max a_i \leq t$ for some $\frac{1}{l} \leq t \leq 1$. There exist $N \in \mathbb{N}$, $N \leq \frac{3}{t}$, and mutually exclusive sets $I_1, \ldots, I_N \subset \{1, \ldots, l\}$ s.t. $\sum_{i \in I_j} a_i \leq t$ and $\bigcup_{j=1}^{N} I_j = \{1, \ldots, l\}$.*

Proof. Note that it is sufficient to consider $\frac{1}{l} \leq t < 1$, as for $t = 1$ the statement of the lemma is obvious. For $l = 2$ partition $I_1 = \{1\}$ and $I_2 = \{2\}$ conditions stated in the lemma. Assume the statement holds for any positive integer l.

Consider $a_1, \ldots, a_{l+1}$ and, without loss of generality, assume that $a_1 \geq a_2 \geq \ldots \geq a_{l+1}$. Take $I_1 = \{1, \ldots, k\}$ where k is such that $\sum_{i=1}^{k} a_i \leq t$ and $\sum_{i=1}^{k+1} a_i > t$. Suppose $\frac{1}{2} < t < 1$ and take $I_2 = \{k+1\}$, $I_3 = \{k+2, \ldots, l+1\}$ (if $k < l$). It follows directly that $\sum_{i \in I_j} a_i \leq t, \quad i = 1, 2$ and $\sum_{i \in I_3} a_i \leq 1 - t < t$, so partition I_1, I_2, I_3 satisfies conditions of the lemma for $\frac{1}{2} < t < 1$.

Suppose now $\frac{1}{l+1} \leq t \leq \frac{1}{2}$ and define

$$\tilde{a}_i = \frac{a_{i+k}}{\sum_{u=k+1}^{l+1} a_u}, \; i = 1, \ldots, l-k+1.$$

We have $\sum_{i=1}^{l-k+1} \tilde{a}_i = 1$ and

$$\max_i \tilde{a}_i \leq \frac{t}{\sum_{u=k+1}^{l+1} a_u} \leq \frac{t}{1-t} \leq 1.$$

Using the induction assumption, we find a partition $\tilde{I}_1, \ldots, \tilde{I}_N$ of $\{1, \ldots, l-k+1\}$ and $N \leq \frac{3 \sum_{u=k+1}^{l+1} a_u}{t}$ such that

$$\sum_{i \in \tilde{I}_j} \tilde{a}_i \leq \frac{t}{\sum_{u=k+1}^{l+1} a_u},$$

for all $j = 1, \ldots, N$, which implies $\sum_{i \in \tilde{I}_j} a_{i+k} \leq t$. Moreover, we have

$$N + 1 \leq \frac{3(1 - \sum_{u=1}^{k} a_u)}{t} + 1 \leq \frac{3}{t} + \frac{t - \sum_{u=1}^{k+1} a_u}{t} \leq \frac{3}{t}.$$

Define $I_{j+1} = \tilde{I}_j + k$, $j = 1, \ldots, N$. Partition $I_1, \ldots, I_{N+1}$ satisfies conditions stated in the lemma. □

Lemma 3.10.3. *Let be $\Omega \subset \Omega_k$ for some $k = 1, \ldots, p-1$, $m \leq n$ and t_1, t_2 satisfying $\max_{\mathcal{A} \in \Omega} \pi_{m,n}(\mathcal{A}) \leq t_2 < t_1 < 1$. Then*

$$\mathbb{P}\left(\max_{\mathcal{A} \in \Omega} \hat{\pi}_{m,n}(\mathcal{A}) \geq t_1\right) \leq \frac{3}{t_2}\left[\left(\frac{t_2}{t_1}\right)^{t_1}\left(\frac{1-t_2}{1-t_1}\right)^{1-t_1}\right]^r,$$

where $\pi_{m,n}(\mathcal{A})$, $\hat{\pi}_{m,n}(\mathcal{A})$ are defined by (3.3) and (3.4), respectively.

Proof. Denote by $\mathcal{A}^1, \ldots, \mathcal{A}^l$ the elements of Ω. Applying Lemma 3.10.2 we find a partition $I_1, \ldots, I_N$ such that $\max_{j=1,\ldots,N} \sum_{i \in I_j} \pi_{m,n}(\mathcal{A}^j) \leq t_2$ and $N \leq \frac{3}{t_2}$. We have

$$\mathbb{P}\left(\max_{i=1,\ldots,l} \hat{\pi}_{m,n}(\mathcal{A}^i) \geq t_1\right) \leq N \max_{j=1,\ldots,N} \mathbb{P}\left(\sum_{i \in I_j} \hat{\pi}_{m,n}(\mathcal{A}^i) \geq t_1\right).$$

Note that $rB \sum_{i \in I_j} \hat{\pi}_{m,n}(\mathcal{A}_j)$ is a sum of B binomial random variables with the probability of success $p_j^* = \sum_{i \in I_j} \pi_{m,n}(\mathcal{A}^i)$ and r trials. We conclude from Lemma 3.10.1 that

$$\mathbb{P}\left(\sum_{i \in I_j} \hat{\pi}_{m,n}(\mathcal{A}^i) \geq t_1\right) \leq \left[\left(\frac{p_j^*}{t_1}\right)^{t_1}\left(\frac{1-p_j^*}{1-t_1}\right)^{1-t_1}\right]^r \leq \left[\left(\frac{t_2}{t_1}\right)^{t_1}\left(\frac{1-t_2}{1-t_1}\right)^{1-t_1}\right]^r,$$

which combined with $N \leq \frac{3}{t_2}$ finishes the proof. □

Now we are in position to prove Theorem 3.4.1.

Proof of Theorem 3.4.1. To ease notation, define $\hat{\omega}_{j,m} = \hat{\omega}_j(\mathbf{Z}_1, \ldots, \mathbf{Z}_m)$, $\delta = \pi_{m,n}(\mathcal{S})$ and $\theta = \max_{\mathcal{A} \not\subset \{1,\ldots,s\}, |\mathcal{A}| \leq k_{max}} \pi_{m,n}(\mathcal{A})$, where $\pi_{m,n}(\cdot)$ is given by (3.2). We start from showing that δ and θ are separated from each other for sufficiently large n.

Let us take $\epsilon = \frac{c_\eta m^{-\eta}}{2}$. Using (A1) and (A3) combined with a simple Bonferroni's

inequality we get

$$\delta \geq \mathbb{P}\left(\max_{j=1,\ldots,p} |\hat{\omega}_{j,m} - \omega_j| < \epsilon\right) \geq 1 - C_\epsilon p \exp(-m^\gamma).$$

Using (A4) and (A5) applied to this, we get $\delta = 1 + O(\exp(-n^{\gamma b_2 - b_1}))$, which tends to one with $n \to \infty$.

For any $\mathcal{A} \in \Omega_k$ $(k \leq k_{max})$ containing at least one $a \in \mathcal{A} \setminus \mathcal{S}$ we have

$$p_{n,m}(\mathcal{A}) \leq \mathbb{P}\left(\min_{j\in\mathcal{A}} \hat{\omega}_{j,m} \geq \max_{j\notin\mathcal{A}} \hat{\omega}_{j,m}\right) \leq \mathbb{P}\left(\hat{\omega}_{a,m} \geq \max_{j\in\mathcal{M}_a\setminus\mathcal{A}} \hat{\omega}_{j,m}\right) \leq \frac{1}{||M_a| - k_{max}|},$$

where $\mathcal{M}_a$ is as in (A2). Using (A6) and (A8) we conclude that $\theta = O(n^{b_4 - b_3})$. From (A8), this tends to zero faster than $(\frac{1}{Br})^{2\alpha} = O(n^{-2\alpha(1-b_2)})$, therefore for sufficiently large n we have

$$\theta \leq \left(\frac{1}{Br}\right)^{2\alpha} < \left(\frac{1}{Br}\right)^{\frac{1}{3}} \leq c_2 < c_1 < \delta,$$

where c_2, c_1 are constants not depending on n.

Take $t_1 = \left(\frac{1}{Br}\right)^\alpha$, $t_2 = t_1^2$ and define events $\mathcal{E}_k = \{\max_{\mathcal{A}\in\Omega_k, \mathcal{A}\notin\mathcal{S}} \hat{\pi}_{m,n}(\mathcal{A}) < t_1\}$, $k = 1, \ldots, k_{max}$, and $\mathcal{B} = \{\hat{\pi}_{m,n}(\mathcal{S}) > c_2\}$. We will demonstrate that $\hat{\mathcal{A}}_{\hat{s},m} = \mathcal{S}$ on the event $\mathcal{E} = \mathcal{B} \cap \bigcap_{k=1}^{k_{max}} \mathcal{E}_k$ and $\mathbb{P}(\mathcal{E}) \underset{n}{\to} 1$. To prove the latter assertion, we use Lemma 3.10.1 and bound

$$\mathbb{P}(\mathcal{B}^c) \leq \left[\left(\frac{\delta}{c_2}\right)^{c_2} \left(\frac{1-\delta}{1-c_2}\right)^{1-c_2}\right]^r \leq \exp\left(-C_{c_1,c_2} r\right), \tag{3.11}$$

where $C_{c_1c_2} = -\log\left(\left(\frac{c_1}{c_2}\right)^{c_2} \left(\frac{1-c_1}{1-c_2}\right)^{1-c_2}\right)$ is strictly positive. By Lemma 3.10.3

$$\mathbb{P}(\mathcal{E}_k^c) \leq \frac{3}{t_2}\left[\left(\frac{t_2}{t_1}\right)^{t_1} \left(\frac{1-t_2}{1-t_1}\right)^{1-t_1}\right]^r = \frac{3}{t_1^2}\left[\left(\frac{t_1}{1+t_1}\right)^{t_1} (1+t_1)\right]^r. \tag{3.12}$$

Now we take the logarithm of $\left(\frac{t_1}{1+t_1}\right)^{t_1}(1+t_1)$. After simple algebra we get

$$t_1 \log\left(\frac{t_1}{1+t_1}\right) + \log(1+t_1) = t_1 \log\left(1 - \frac{1}{1+t_1}\right) + \log(1+t_1),$$

which can be bounded using (A7) and $\log(1+x) \leq \frac{2x}{2+x}$ for $x \in (-1,0)$ and $\log(1+x) \leq \frac{x}{2}\frac{2+x}{1+x}$ for $x \geq 0$ (Topsøe, 2004) which together yield

$$t_1 \log\left(1 - \frac{1}{1+t_1}\right) + \log(1+t_1) \leq -t_1 \frac{(2 - t_1 - 2t_1^2)}{2(1+t_1)(1+2t_1)} \leq -\frac{t_1}{6}.$$

This applied to (3.12) yields

$$\mathbb{P}(\mathcal{E}_k^c) \leq \frac{3}{t_1^2} \exp\left(\frac{-rt_1}{6}\right) = C_{1,\alpha,B} r^{2\alpha} \exp\left(-C_{2,\alpha,B} r^{(1-\alpha)}\right), \tag{3.13}$$

with positive constants $C_{1,\alpha,B} = 3B^{2\alpha}$, $C_{2,\alpha,B} = \frac{1}{6B^\alpha}$. From (3.11), (3.13) and (A5), (A8) we have

$$\mathbb{P}(\mathcal{E}) \geq 1 - C_{1,\alpha,B} r^{2\alpha} k_{max} \exp\left(-C_{2,\alpha,B} r^{(1-\alpha)}\right) - \exp\left(-C_{c_1,c_2} r\right),$$

therefore $\mathbb{P}(\mathcal{E}) \underset{n}{\to} 1$.

The remaining arguments used in the proof are valid on $\mathcal{E}$. First, from the $c_2 > t_1$ we conclude that $\hat{\mathcal{A}}_{s,m} = \mathcal{S}$, where $\hat{\mathcal{A}}_{s,m}$ is given by (3.4), hence showing $\hat{s} = s$ proves $\hat{\mathcal{S}} = \mathcal{S}$. Denote $T_k = \frac{\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k+1,m})}{\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k,m})}$, then $\hat{s} = \operatorname{argmin}_{k=0,1,\ldots,k_{max}} T_k$. For $k < s$ we have $\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k+1,m}) \geq \frac{\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{s,m})}{\binom{s}{k+1}}$, hence

$$T_k \geq \frac{c_2}{\binom{s}{k+1}}, \quad k = 0, \ldots, s-1.$$

Directly from the definition of $\mathcal{E}_s$ and $\mathcal{B}$ we bound $T_s \leq \frac{t_1}{c_2}$, and, $\hat{\pi}_{m,n}(\hat{\mathcal{A}}_{k+1,m}) \geq \frac{1}{Br}$ for

any k,

$$T_k \geq \frac{\frac{1}{Br}}{t_1}, \quad k = s+1, \ldots, k_{max}.$$

To prove $T_k > T_s$ for $k = 0, \ldots, s-1$, it is sufficient to demonstrate that $\frac{c_2}{\binom{s}{k+1}} > \frac{t_1}{c_2}$. The latter follows from

$$\frac{c_2^2}{\binom{s}{k+1}}(Br)^{\alpha} \geq \frac{(Br)^{\alpha-\frac{2}{3}}}{\binom{s}{k+1}} \geq \frac{B^{\alpha-\frac{2}{3}}}{\max_{k=1,\ldots,s}\binom{s}{k}} > 1,$$

as $c_2 \geq \left(\frac{1}{Br}\right)^{1/3}$, $t_1 = \left(\frac{1}{Br}\right)^{\alpha}$ and $B^{\alpha-\frac{2}{3}} > \max_{k=1,\ldots,s}\binom{s}{k}$ by (A7). Similarly, to observe that $T_s < T_k$ for $k = s+1, \ldots, k_{max}$, we need to show $\frac{t_1}{c_2} < \frac{1}{t_1 Br}$. Since $\alpha > \frac{1}{3}$, this can be concluded from

$$\frac{Br}{c_2(Br)^{2\alpha}} \leq (Br)^{2\left(\frac{1}{3}-\alpha\right)} < 1.$$

Therefore T_k is necessarily minimised at $k = s$ on the set $\mathcal{E}$ meaning that $\hat{s} = s$, which finishes the proof. □

Chapter 4

Narrowest-Over-Threshold detection of multiple change-points and change-point-like features

4.1 Introduction

This chapter considers the canonical univariate statistical model

$$Y_t = f_t + \varepsilon_t, \quad t = 1, \ldots, T, \tag{4.1}$$

where the deterministic and unknown signal f_t is believed to display some regularity across the index t, and the stochastic noise ε_t is exactly or approximately centred at zero. Despite the simplicity of model (4.1), inferring information about f_t remains a task of fundamental importance in modern applied statistics and data science. We now mention a selection of applications in which the task of interest reduces to estimating or making inference on f_t or its functionals. In the analysis of DNA copy number data in genomics, f_t is usually modelled as piecewise-constant and the typical task is to

estimate change-points in f_t (Olshen et al., 2004). In mass spectrometry, it is often of interest to detect peaks in f_t (Antoniadis et al., 2010). In applied financial econometrics, a key task is to identify current trends in financial markets, which can be translated into searching for significant recent changes in some characteristics of f_t, e.g. the local slope (Schröder and Fryzlewicz, 2013). In climatology, detecting changes in the trends of temperature data (Cahill et al., 2015) can also be formalised as estimating changes in the slope of f_t while modelling it as piecewise-linear. In astrophysics, detecting Gamma-Ray Bursts (GRB; Kolaczyk, 1997) typically requires delicate statistical work resulting in the separation of f_t into slowly-varying background and faster-changing GRB's.

Depending on the nature and complexity of the statistical task involving f_t, a wider or narrower range of tools are at the statistician's disposal, and we provide a non-exhaustive list of the main approaches below. When the task is the simple estimation of f_t, linear methods such as kernel smoothing (Wand and Jones, 1994), spline smoothing (De Boor, 2001) or local polynomial regression (Fan and Gijbels, 1996; Simonoff, 2012) typically provide a useful reference point, and robust smoothing techniques (such as median filtering; Koch, 1996) may be of interest if the distribution of ε_t is heavy-tailed. On the other hand, when the interest is in more interpretable estimation, for example in the detection of "features" in f_t such as jumps or kinks, then more involved, non-linear techniques are usually required. If f_t is modelled as piecewise-constant and it is of interest to detect its change-points, several techniques are available, and we only mention a selection of older and more recent approaches. When ε_t is assumed to be Gaussian, both non-penalised and penalised least squares approaches were first considered by Yao and Au (1989). For specific choices of penalty functions, see e.g. Yao (1988) and Lavielle (2005). The Gaussianity assumption on the noise ε_t is relaxed to exponential family distributions in Lee (1997), Hawkins (2001) and Frick et al. (2014). In particular, Frick et al. (2014) also provide confidence intervals for the location of the estimated change-points. Note

that often this penalty-type approach requires a computational cost of at least $O(T^2)$, with the exception of the estimator proposed by Killick et al. (2012a), which achieves a linear computational cost (thus called the "Pruned Exact Linear Time", or PELT), but requires further assumption that change-points are separated by time intervals drawn independently from some probability distribution, a scenario in which considerations of statistical consistency are not generally possible. A nonparametric version of PELT is investigated by Haynes et al. (2016a). Another general approach is based on the idea of Binary Segmentation (BS; Vostrikova, 1981), which can be viewed as a greedy approach with a limited computational cost. Its popular variants include the circular binary segmentation (CBS; Olshen et al., 2004) and the Wild Binary Segmentation (WBS; Fryzlewicz, 2014). A more complete review in terms of up-to-date publications, software and applications can be found in the online repository *changepoint.info* maintained by Killick et al. (2012b). More general change-point problems, in which f_t is modelled as piecewise-parametric (not necessarily piecewise-constant) between "knots", the number and locations of which are unknown and need to be estimated, have attracted less interest in the literature and overwhelmingly focus on linear trend detection. Among them, we mention the approach based on least squares principle and Wald-type tests by Bai and Perron (1998), and trend filtering (Lin et al., 2016; Tibshirani, 2014).

The aim of this work is to propose a new, generic approach to the problem of detecting an unknown number of "features" occurring at unknown locations in f_t. By a feature, we mean a characteristic of f_t, occurring at a location t_0, that is detectable by considering a sufficiently large subsample of data Y_t around t_0. Examples include: change-points in f_t when it is modelled as piecewise-constant, change-points in the first derivative when f_t is modelled as piecewise-linear and continuous, and discontinuities in f_t when it is modelled as piecewise-linear but without the continuity constraint. We will provide a precise description of the type of features we are interested in later on.

Moving beyond f_t only, our approach will also permit the detection of similar features present in some distributional aspects of ε_t, for example in its variance. Since all types of features we consider describe changes in a parametric description of f_t, we use the terms "feature detection" and "change-point detection" interchangeably throughout the chapter. Occasionally, for precision, we will be referring to change-point detection in the piecewise-constant model as the "canonical" change-point problem, while our general feature detection problem will sometimes be referred to as a "generalised" change-point problem.

Core to our approach is a particular blend of "global" and "local" treatment of the data Y_t in the search for the multiple features that may be present in f_t, a combination that gives our method a multi-scale character. At the first "global" stage, we randomly draw a number of subsamples $(Y_s, Y_{s+1}, \ldots, Y_e)'$, where $1 \leq s < e \leq T$. On each subsample, we assume, possibly erroneously, that *only one* feature is present and use a tailor-made contrast function derived (according to a universal recipe we provide later) from the likelihood theory to find the most likely location of the feature. We retain those subsamples for which the contrast *exceeds a certain user-specified threshold*, and discard the others. Amongst the retained subsamples, we search for the one drawn on the *narrowest* interval, i.e. one for which $e - s$ is the smallest: it is this step that gives rise to the name *Narrowest-Over-Threshold* (NOT) for our methodology. The focus on the narrowest interval constitutes the "local" part of the method, and is a key ingredient of our approach which ensures that with high probability, at most one feature is present in the selected interval. This key observation gives our methodology a general character and allows it to be used, only with minor modifications, in a wide range of scenarios, including those described in the previous paragraph. Having detected the first feature, the algorithm then proceeds recursively to the left and to the right of it, and stops, on any current interval, if no contrasts can be found that exceed the threshold.

Besides its generic character, other benefits of the proposed methodology include low computational complexity, ease of implementation, accuracy in the detection of the feature locations, and the fact that it enables parametric (and hence: interpretable) estimation of the signal on each section delimited by a pair of neighbouring estimated features. Regarding the computational complexity, the facts that only a limited number of data subsamples, M, need to be drawn (we provide precise bounds later; with finitely many change-points, one can take $M = O(\log T)$ in general), and that typical contrasts are computable in linear time, lead to a computational complexity of $O(MT)$ for the entire procedure. Moreover, the entire threshold-indexed solution path can also be computed efficiently, in typically close-to-linear time, as observed from our numerical experiments. Regarding the estimation accuracy, in the scenarios we consider theoretically, our procedure yields near-optimal rates of convergence for the estimators of feature locations.

Importantly, the flexible character of our methodology leaves it open to possible extensions and modifications. Indeed, borrowing words from Sweldens and Schröder (2000), who advocated "building your own wavelets at home", we also view our proposal as flexible enough to enable the user to "construct their own feature detector at home", e.g. by proposing their own specialised contrast functions, or by data-adaptively choosing the most suitable contrast function from a pre-specified dictionary (which would lead to mixed-type feature detection). Although these extensions are not covered in the current work, we view this modularity and flexibility offered by our methodology as an important aspect of our proposal.

On a broader level, our methodology promotes the idea of "fitting simple models on subsets of the data (the local aspect), and then aggregating the results to obtain the overall fit (the global aspect)", an idea also present in the Wild Binary Segmentation method of Fryzlewicz (2014). However, we emphasise that the way the simple models

(here: models containing *at most one* change-point or other feature) are fitted in the NOT and WBS methods are entirely different and have different aims. Unlike the WBS, the NOT methodology focuses on the *narrowest* intervals of the data on which it is possible to locate the feature of interest. It is this focus on the narrowest intervals that enables NOT to extend well beyond mere change-point detection for a piecewise-constant f_t, the latter being the sole focus of the WBS method. The lack of the narrowest-interval focus in the WBS and BS methods means that it is not applicable to more general feature detection, and we explain the mechanics of this phenomenon briefly in the following simple example.

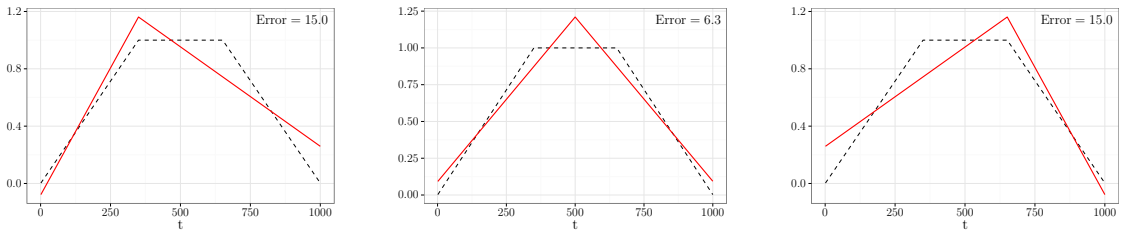
Consider a continuous piecewise-linear signal that has two change-points in its first derivative:

$$f_t = \begin{cases} \frac{1}{350}t, & t = 1, \ldots, 350, \\ 1, & t = 351, \ldots, 650, \\ \frac{1001}{350} - \frac{1}{350}t, & t = 651, \ldots, 1000. \end{cases} \tag{4.2}$$

If we approximate f_t using a piecewise-linear signal with only one change-point in its derivative, then the best approximation (in terms of minimising the ℓ_2 distance) will result in an estimated change-point at $t = 500$, which is away from the true ones at $t = 350$ and $t = 650$, as is illustrated in Figure 4.1. Therefore, taking the entire sample of data starting at $s = 1$ and ending at $e = 1000$, and searching for one of its multiple change-points by fitting, via least squares, a triangular signal with a single change-point, does not make sense. NOT avoids this issue because of its unique feature of picking the *narrowest* intervals which are likely to contain only one change-point. To understand the mechanics of this key feature, imagine that now f_t is observed with noise. Through its pursuit of the narrowest intervals, NOT will ensure that, with high probability, some suitably narrow intervals around the change-points $t = 350$ and $t = 650$ are considered. More precisely, by

construction, they will be *narrow enough to contain only one change-point each*, but wide enough for the designed contrast (see Section 4.2.3.2 for more on contrasts) to indicate the existence of the change-point within both of them. The designed contrast function will indicate the right location of the change-point (modulo the estimation error) if only one change-point is present in the data subsample considered, unlike in the situation described earlier in which multiple change-points were included in the chosen interval. More details on this example are presented in Section 4.3.3.

We note that this example is different from the canonical change-point detection problem (i.e. piecewise-constant signal with multiple change-points), where if we approximate the signal using a piecewise-constant function with only one change-point, the change-point of the fitted signal will always be among the true ones (Venkatraman, 1992). Since the latter property does not hold in most generalised change-point detection problems, this highlights the need for new methods with better localisation of the feature of interest, such as our NOT algorithm. Finally, we remark that Fang et al. (2016) independently considered a related shortest-interval idea in the context of the canonical change-point detection problem. However, they did not consider it as a springboard to more general feature detection problems, which is the key motivation behind NOT and its most valuable contribution.



(a) $\tau = 350$ (b) $\tau = 500$ (c) $\tau = 651$

Figure 4.1: Best ℓ_2 approximation of the true signal (dashed) via a triangular signal with a single change-point, the location of which is fixed at the left change-point (left panel), halfway between the true change-points (middle panel) and at the right change-point (right panel). Approximation errors (in terms of squared ℓ_2 distance) are given in the top-right corners of the corresponding panels.

To summarise, in the NOT approach, we propose a new "modus operandi" in statistical smoothing, by providing a novel, general, flexible framework for feature detection and interpretable signal estimation. The procedure is fast, accurate, easy to code and to extend by the user to tailor to their own needs. Its implementation is provided in the R package **not** (Baranowski et al., 2016b).

The remainder of this chapter is organised as follows. In Section 4.2, we give a more mathematical description of NOT. In particular, we consider NOT in four scenarios, each with a different form of structural change in the mean and/or variance. For the development of both theory and computation, in each scenario, we also introduce the tailor-made contrast function derived from the generalised likelihood ratio (GLR), which is used to detect features within each subsample. Theoretical properties of NOT, such as its consistency and convergence rates are also provided. Section 4.3 deals with the computational aspects of NOT, while a comprehensive simulation study is carried out in Section 4.4, where we compare NOT with the state-of-art change-point detection tools. In Section 4.5, we consider data examples of oil price, global temperature anomalies and London housing data. All proofs can be found in Section 4.6.

4.2 Methodology

4.2.1 Setup

To describe the main framework of NOT, we consider a simplified version of (4.1), where $\mathbf{Y} = (Y_1, \ldots, Y_T)'$ is modelled through

$$Y_t = f_t + \sigma_t \varepsilon_t, \quad t = 1, \ldots, T, \tag{4.3}$$

where f_t is the signal, $\varepsilon_t \overset{\text{i.i.d}}{\sim} \mathcal{N}(0,1)$ is the standardised independent and identically distributed (i.i.d.) Gaussian noise, and where σ_t is the noise's standard deviation at time t. We note that the normality assumption facilitates the technical presentation of our results, but the entire framework can be extended to other noise distributions. Numerical examples involving other noise distributions can be found in Section 4.4.

We assume that (f_t, σ_t) can be partitioned into $q+1$ segments, with q unknown distinct change-points $0 = \tau_0 < \tau_1 < \ldots < \tau_q < \tau_{q+1} = T$. Here the value of q is not pre-specified and can grow with T. For each $j = 1, \ldots, q+1$ and for $t = \tau_{j-1}+1, \ldots, \tau_j$, the structure of (f_t, σ_t) is is modelled parametrically by a local (i.e. depending on j) real-valued d-dimensional parameter vector $\boldsymbol{\Theta}_j$ (with $\boldsymbol{\Theta}_j \neq \boldsymbol{\Theta}_{j-1}$), where d is known and typically small. In addition, we require the minimum distance between consecutive change-points to be greater than d for the purpose of identifiability. In other words, (f_t, σ_t) can be divided into q different segments, each from the same parametric family of much simpler structure. Even if the main goal is not change-point detection, the class of piecewise-parametric functions is rich enough for function estimation, as any function could be approximated arbitrarily well in L_p $(0 < p < \infty)$ by a piecewise-parametric function with enough segments (DeVore, 1998).

Some commonly-encountered scenarios are listed below, where the following holds inside the j-th segment for each $j = 1, \ldots, q+1$:

(S1) **Constant variance, piecewise-constant mean**:

$\sigma_t = \sigma_0$ and $f_t = \theta_j$ for $t = \tau_{j-1}+1, \ldots, \tau_j$.

(S2) **Constant variance, continuous and piecewise-linear mean**:

$\sigma_t = \sigma_0$ and $f_{\tau_{j-1}+1} = \theta_{j,1}$, $f_t = f_{t-1} + \theta_{j,2}$ for $t = \tau_{j-1}+2, \ldots, \tau_j$, with the additional constraint of

$$\theta_{j,1} + \theta_{j,2}(\tau_j - \tau_{j-1} - 1) = \theta_{j+1,1} - \theta_{j+1,2}$$

for $j = 1, \ldots, q$. Therefore, $t \in \{\tau_1, \ldots, \tau_q\}$ if and only if $f_{t-1} + f_{t+1} \neq 2f_t$.

(S3) **Constant variance, piecewise-linear (not necessarily continuous) mean**:

$\sigma_t = \sigma_0$ and $f_{\tau_{j-1}+1} = \theta_{j,1}$, $f_t = f_{t-1} + \theta_{j,2}$ for $t = \tau_{j-1} + 2, \ldots, \tau_j$.

(S4) **Piecewise-constant variance, piecewise-constant mean**:

$f_t = \theta_{j,1}$ and $\sigma_t = \theta_{j,2} > 0$ for $t = \tau_{j-1} + 1, \ldots, \tau_j$.

Since σ_0 in (S1)–(S3) acts as a nuisance parameter, in the rest of this manuscript, for simplicity we assume that its value is known. If it is unknown, then it can be estimated accurately using the Median Absolute Deviation (MAD) method (Hampel, 1974). More specifically, the MAD estimator of σ_0 is defined as $\hat{\sigma} = \text{Median}\{|Y_2 - Y_1|, \ldots, |Y_T - Y_{T-1}|\}/\{\Phi^{-1}(3/4)\sqrt{2}\}$ in Scenario (S1) and as $\hat{\sigma} = \text{Median}\{|Y_1 - 2Y_2 + Y_3|, \ldots, |Y_{T-2} - 2Y_{T-1} + Y_T|\}/\{\Phi^{-1}(3/4)\sqrt{6}\}$ in Scenarios (S2) and (S3), where $\Phi^{-1}(\cdot)$ denotes the quantile function of the standard normal distribution.

Both the methodology and the theory developed below can readily be extended to handle more complicated cases in which the signal within the segments is non-linear (e.g. higher-order-polynomial, a case illustrated in Section 4.4). In all of the above-listed scenarios, we focus on structure changes in the mean or the first two moments in the univariate setting. Nevertheless, our framework can be extended to handle multivariate observations, or other more complex structure changes such as autocovariance in time series. In addition, as mentioned earlier, the normality assumption of the noise can be relaxed as well.

4.2.2 Main idea

We now describe the main idea of NOT formally. In the first step, instead of directly using the entire data sample, we randomly extract subsamples, i.e. vectors $(Y_s, Y_{s+1}, \ldots, Y_e)'$, where s and e are integers drawn (independently with replacement) uniformly from the

set $\{1, \ldots, T\}$ that satisfy $1 \leq s < e \leq T$ and $e - s > 2(d-1)$. Let $\ell(Y_s, \ldots, Y_e; \boldsymbol{\Theta})$ be the likelihood of $\boldsymbol{\Theta}$ given $(Y_s, \ldots, Y_e)'$. We then compute the generalised log-likelihood ratio (GLR) statistic for all potential single change-points within the subsample and pick the maximum, that is,

$$\mathcal{R}^b_{s,e}(\mathbf{Y}) = 2\log\left[\frac{\sup_{\boldsymbol{\Theta}^1,\boldsymbol{\Theta}^2}\left\{\ell(Y_s, \ldots, Y_b; \boldsymbol{\Theta}^1)\ell(Y_{b+1}, \ldots, Y_e; \boldsymbol{\Theta}^2)\right\}}{\sup_{\boldsymbol{\Theta}} \ell(Y_s, \ldots, Y_e; \boldsymbol{\Theta})}\right]; \tag{4.4}$$

$$\mathcal{R}_{s,e}(\mathbf{Y}) = \max_{b \in \{s+d-1, \ldots, e-d\}} \mathcal{R}^b_{s,e}(\mathbf{Y}).$$

If constraints are in place between $\boldsymbol{\Theta}_j$ and $\boldsymbol{\Theta}_{j+1}$ for any $j = 1, \ldots, q$ (e.g. as in (S2)), the supremum in the numerator of (4.4) is taken over the set that only contains elements of form $\boldsymbol{\Theta}^1 \times \boldsymbol{\Theta}^2$ satisfying these constraints. Otherwise, as in (S1), (S3) and (S4), (4.4) can be simplified to

$$\mathcal{R}^b_{s,e}(\mathbf{Y}) = 2\log\left\{\frac{\sup_{\boldsymbol{\Theta}} \ell(Y_s, \ldots, Y_b; \boldsymbol{\Theta}) \sup_{\boldsymbol{\Theta}} \ell(Y_{b+1}, \ldots, Y_e; \boldsymbol{\Theta})}{\sup_{\boldsymbol{\Theta}} \ell(Y_s, \ldots, Y_e; \boldsymbol{\Theta})}\right\}.$$

The above procedure is repeated for M randomly drawn intervals $(s_1, e_1), \ldots, (s_M, e_M)$.

In the second step, we test all the $\mathcal{R}_{s_m,e_m}(\mathbf{Y})$ for $m = 1, \ldots, M$ against a given threshold ζ_T, and pick the one corresponding to the interval $[s_{m^*}, e_{m^*}]$ that has the smallest length. Once a change-point is found in $[s_{m^*}, e_{m^*}]$ (i.e. the b^* that maximises $\mathcal{R}^b_{s_{m^*},e_{m^*}}(\mathbf{Y})$), the same procedure is then repeated recursively to the left and to the right of it, until no further significant GLRs can be found.

After finding all the change-points, one can estimate the signals within each segment using standard methods such as least squares or maximum likelihood. Note that spline regression can be viewed as a multiple change-point detection problem set in the context of polynomial segments that are continuously differentiable but have discontinuous higher order derivatives at the change-points between these segments. From this perspective, one can also think of NOT as an adaptive way of picking the number and the location of

knots from the data for the traditional spline regression.

4.2.3 Log-likelihood ratios and contrast functions

In many applications, the GLR (4.4) in NOT can be simplified with the help of "contrast functions" under the setting of Gaussian noise. More precisely, for every integer triple (s, e, b) with $1 \leq s < e \leq T$, our aim is to find $\mathcal{C}^b_{s,e}(\mathbf{Y})$ such that:

(a) $\operatorname{argmax}_b \mathcal{C}^b_{s,e}(\mathbf{Y}) = \operatorname{argmax}_b \mathcal{R}^b_{s,e}(\mathbf{Y})$,

(b) heuristically speaking, the value of $\mathcal{C}^b_{s,e}(\mathbf{Y})$ is relatively small if there is no change-point in $[s, e]$,

(c) the formulation of $\mathcal{C}^b_{s,e}(\mathbf{Y})$ mainly consists of taking inner products between the data and contrast vectors, which facilitates the development of both computation and theory, particularly if the contrast vectors can be taken to be mutually orthonormal.

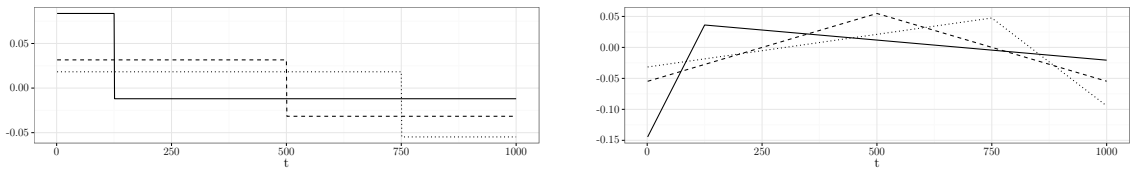
In the following, we give the contrast functions corresponding to (S1)–(S4). We note that this approach recovers the CUSUM statistic in (S1), which is popular in this canonical change-point detection setting. One can view the resulting statistics as generalisations of CUSUM to other scenarios.

4.2.3.1 Scenario (S1)

Here f_t is piecewise-constant. For any integer triple (s, e, b) with $1 \leq s < e \leq T$ and $s \leq b \leq e-1$, we define the contrast vector $\boldsymbol{\psi}^b_{s,e} = \left(\psi^b_{s,e}(1), \ldots, \psi^b_{s,e}(T)\right)'$ with

$$\psi^b_{s,e}(t) = \begin{cases} \sqrt{\frac{e-b}{l(b-s+1)}}, & t = s, \ldots, b \\ -\sqrt{\frac{b-s+1}{l(e-b)}}, & t = b+1, \ldots, e \\ 0, & \text{otherwise}, \end{cases} \tag{4.5}$$

where $l = e - s + 1$. Also, if $b \notin \{s, s+1, \ldots, e-1\}$, then we set $\psi^b_{s,e}(t) = 0$ for all t. As an illustration, plots of $\boldsymbol{\psi}^b_{s,e}$ with different (s, e, b) are shown in Figure 4.2a.



(a) $\boldsymbol{\psi}^b_{s,e}$ (b) $\boldsymbol{\phi}^b_{s,e}$

Figure 4.2: Plots of $\boldsymbol{\phi}^b_{s,e}$ and $\boldsymbol{\psi}^b_{s,e}$ given by, respectively, (4.5) and (4.8) for $s = 1$, $e = 1000$ and several values of b. Solid line: $b = 125$; dotted line: $b = 500$; dashed line: $b = 750$.

For any vector $\mathbf{v} = (v_1, \ldots, v_T)'$ we define the contrast function as $\mathcal{C}^b_{s,e}(\mathbf{v}) = \sqrt{\left\langle \mathbf{v}, \boldsymbol{\psi}^b_{s,e} \right\rangle^2} = \left| \left\langle \mathbf{v}, \boldsymbol{\psi}^b_{s,e} \right\rangle \right|$. Therefore, if $s \leq b \leq e-1$, then

$$\mathcal{C}^b_{s,e}(\mathbf{v}) = \left| \sqrt{\frac{e-b}{l(b-s+1)}} \sum_{t=s}^{b} v_t - \sqrt{\frac{b-s+1}{l(e-b)}} \sum_{t=b+1}^{e} v_t \right|. \tag{4.6}$$

Otherwise, $\mathcal{C}^b_{s,e}(\mathbf{v}) = 0$. This recovers the well-known CUSUM statistic in the change-point detection literature. It can be shown that $[\mathcal{C}^b_{s,e}(\mathbf{Y})]^2 = \sigma_0^2 \mathcal{R}^b_{s,e}(\mathbf{Y})$ for every (s, e, b) with $1 \leq s \leq b < e \leq T$, thus $\mathcal{C}^b_{s,e}(\cdot)$ fulfills the aforementioned requirements for the contrast function.

In addition, with a slight abuse of notation, for any $1 \leq s < e \leq T$, we define the constant vector for the interval $[s, e]$ as

$$\mathbf{1}_{s,e}(t) = \begin{cases} (e-s+1)^{-1/2}, & t = s, \ldots, e \\ 0, & \text{otherwise} \end{cases}, \tag{4.7}$$

and write $\mathbf{1}_{s,e} = \left(\mathbf{1}_{s,e}(1), \ldots, \mathbf{1}_{s,e}(T)\right)'$. Then it is easy to check that $\mathbf{1}_{s,e}$ and $\boldsymbol{\psi}^b_{s,e}$ are orthonormal. This explains why the CUSUM is invariant to shifts in the mean.

4.2.3.2 Scenario (S2)

Here f_t is piecewise-linear and continuous. For any triple (s, e, b) with $1 \leq s < e \leq T$ and $s + 1 \leq b \leq e - 1$, consider the contrast vector $\boldsymbol{\phi}^b_{s,e} = \left(\phi^b_{s,e}(1), \ldots, \phi^b_{s,e}(T)\right)'$ with

$$\phi^b_{s,e}(t) = \begin{cases} \alpha^b_{s,e}\beta^b_{s,e}\Big[\big\{3(b-s+1)+(e-b)-1\big\}t - \big\{b(e-s)+2s(b-s+1)\big\}\Big], & t = s, \ldots, b \\ -\frac{\alpha^b_{s,e}}{\beta^b_{s,e}}\Big[\big\{3(e-b)+(b-s+1)+1\big\}t - \big\{b(e-s)+2e(e-b+1)\big\}\Big], & t = b+1, \ldots, e, \\ 0, & \text{otherwise.} \end{cases} \tag{4.8}$$

where $\alpha^b_{s,e} = \left(\frac{6}{l(l^2-1)(1+(e-b+1)(b-s+1)+(e-b)(b-s))}\right)^{1/2}$, $\beta^b_{s,e} = \left(\frac{(e-b+1)(e-b)}{(b-s)(b-s+1)}\right)^{1/2}$ and $l = e-s+1$. If $b \notin \{s+1, \ldots, e-1\}$, then we set $\phi^b_{s,e}(t) = 0$ for all t. We illustrate the structure of $\boldsymbol{\phi}^b_{s,e}$ in Figure 4.2b. The contrast function is then defined as

$$\mathcal{C}^b_{s,e}(\mathbf{v}) = \sqrt{\left\langle \mathbf{v}, \phi^b_{s,e} \right\rangle^2} = \left| \left\langle \mathbf{v}, \boldsymbol{\phi}^b_{s,e} \right\rangle \right|, \tag{4.9}$$

To explain the rationale behind $\boldsymbol{\phi}^b_{s,e}$, we first define the "linear" vector for the interval $[s, e]$, $\boldsymbol{\gamma}_{s,e} = \left(\gamma_{s,e}(1), \ldots, \gamma_{s,e}(T)\right)'$, as

$$\gamma_{s,e}(t) = \begin{cases} \left\{\frac{1}{12}(e-s+1)(e^2-2es+2e+s^2-2s)\right\}^{-1/2}\left(t - \frac{e+s}{2}\right), & t = s, \ldots, e \\ 0, & \text{otherwise} \end{cases}. \tag{4.10}$$

Then we have that $\boldsymbol{\phi}^b_{s,e}$ is orthonormal to both $\mathbf{1}_{s,e}$ and $\boldsymbol{\gamma}_{s,e}$ (note that $\boldsymbol{\gamma}_{s,e}$ itself is orthonormal to $\mathbf{1}_{s,e}$). The orthonormality of the vectors $\mathbf{1}_{s,e}$, $\boldsymbol{\gamma}_{s,e}$ and $\boldsymbol{\phi}^b_{s,e}$ is important in deriving the identity $\sigma_0^2\mathcal{R}^b_{s,e}(\mathbf{Y}) = \mathcal{C}^b_{s,e}(\mathbf{Y})^2$ below, and helps improve the numerical efficiency and stability in our implementation of NOT. In particular, it means that the contrast function is invariant to both mean shifts and slope shifts on a given interval.

In fact, $\boldsymbol{\phi}^b_{s,e}$ can be derived by (i) applying the Gram–Schmidt process on the following vector (linear with a kink at $b+1$ on $[s,e]$)

$$\tilde{\phi}^b_{s,e}(t) = \begin{cases} t-b, & t = b+1,\ldots,e \\ 0, & \text{otherwise} \end{cases}$$

with respect to $\mathbf{1}_{s,e}$ and $\boldsymbol{\gamma}_{s,e}$, and (ii) normalisation such that $\|\cdot\|_2 = 1$.

Write the restriction of $\mathbf{v}$ on $[s,e]$ as $\mathbf{v}|_{[s,e]} = (0,\ldots,0,v_s,\ldots,v_e,0,\ldots,0)'$. Fix any (s,e,b), given the restriction imposed on $\boldsymbol{\Theta}$ in (S2), the best approximation of $\mathbf{Y}|_{[s,e]}$ (in the ℓ_2 distance) with a single kink at b is a linear combination of $\mathbf{1}_{s,e}$, $\boldsymbol{\gamma}_{s,e}$ and $\boldsymbol{\phi}^b_{s,e}$ (all mutually orthonormal). Therefore,

$$\begin{aligned}
&\sigma_0^2 \mathcal{R}^b_{s,e}(\mathbf{Y}) = \\
&\min_{a_0,a_1\in\mathbb{R}} \|\mathbf{Y}|_{[s,e]} - a_0\mathbf{1}_{s,e} - a_1\boldsymbol{\gamma}_{s,e}\|_2^2 - \min_{a_0,a_1,a_2\in\mathbb{R}} \|\mathbf{Y}|_{[s,e]} - a_0\mathbf{1}_{s,e} - a_1\boldsymbol{\gamma}_{s,e} - a_2\boldsymbol{\phi}^b_{s,e}\|_2^2 \\
&= \|\mathbf{Y}|_{[s,e]} - \langle \mathbf{Y}, \boldsymbol{\gamma}_{s,e}\rangle\boldsymbol{\gamma}_{s,e} - \langle \mathbf{Y}, \mathbf{1}_{s,e}\rangle\mathbf{1}_{s,e}\|_2^2 \\
&- \|\mathbf{Y}|_{[s,e]} - \langle \mathbf{Y}, \boldsymbol{\phi}^b_{s,e}\rangle\boldsymbol{\phi}^b_{s,e} - \langle \mathbf{Y}, \boldsymbol{\gamma}_{s,e}\rangle\boldsymbol{\gamma}_{s,e} - \langle \mathbf{Y}, \mathbf{1}_{s,e}\rangle\mathbf{1}_{s,e}\|_2^2 \\
&= \langle \mathbf{f}, \boldsymbol{\phi}^b_{s,e}\rangle^2 = \mathcal{C}^b_{s,e}(\mathbf{Y})^2,
\end{aligned}$$

i.e. the aforementioned requirements for the contrast function are satisfied.

4.2.3.3 Scenario (S3)

Here f_t is a piecewise-linear but not necessarily continuous function. We use the following contrast function for any $s < b < e$:

$$\mathcal{C}^b_{s,e}(\mathbf{v}) = \left(\left\langle \mathbf{v}, \boldsymbol{\psi}^b_{s,e} \right\rangle^2 + \left\langle \mathbf{v}, \boldsymbol{\gamma}_{s,b} \right\rangle^2 + \left\langle \mathbf{v}, \boldsymbol{\gamma}_{b+1,e} \right\rangle^2 - \left\langle \mathbf{v}, \boldsymbol{\gamma}_{s,e} \right\rangle^2 \right)^{1/2}. \tag{4.11}$$

This construction is justified by noting that

$$\begin{aligned}\sigma_0^2 \mathcal{R}_{s,e}^b(\mathbf{Y}) = & \min_{a_0,a_1\in\mathbb{R}} \|\mathbf{Y}|_{[s,e]} - a_0\mathbf{1}_{s,e} - a_1\boldsymbol{\gamma}_{s,e}\|_2^2 \\ & - \min_{a_0,a_1\in\mathbb{R}} \|\mathbf{Y}|_{[s,b]} - a_0\mathbf{1}_{s,b} - a_1\boldsymbol{\gamma}_{s,b}\|_2^2 \\ & + \min_{a_0,a_1\in\mathbb{R}} \|\mathbf{Y}|_{[b+1,e]} - a_0\mathbf{1}_{b+1,e} - a_1\boldsymbol{\gamma}_{b+1,e}\|_2^2 \\ & = \mathcal{C}_{s,e}^b(\mathbf{Y})^2,\end{aligned}$$

where we also used the orthonormality among $\mathbf{1}_{s,e}$, $\boldsymbol{\psi}_{s,e}^b$, $\boldsymbol{\gamma}_{s,b}$ and $\boldsymbol{\gamma}_{b+1,e}$ in the above derivation.

4.2.3.4 Scenario (S4)

Here both f_t and σ_t are piecewise-constant. For any $1 \le s+1 < b < e-1 \le T$, we propose

$$\begin{aligned}\mathcal{C}_{s,e}^b(\mathbf{Y}) = & (b-s+1)\log\left(\hat{\sigma}_{s,b}(\mathbf{Y})\right) + (e-b)\log\left(\hat{\sigma}_{b+1,e}(\mathbf{Y})\right) \\ & - (e-s+1)\log\left(\hat{\sigma}_{s,e}(\mathbf{Y})\right),\end{aligned} \tag{4.12}$$

where

$$\hat{\sigma}_{s,e}^2(\mathbf{Y}) = \frac{1}{e-s+1}\sum_{t=s}^{e}\left(Y_t - \frac{1}{e-s+1}\sum_{t=s}^{e} Y_t\right)^2 = \left\langle \mathbf{Y}^2, \mathbf{1}_{s,e}^2\right\rangle - \left\langle \mathbf{Y}, \mathbf{1}_{s,e}^2\right\rangle^2.$$

Otherwise, for $b \notin \{s+2,\ldots,e-2\}$, we set $\mathcal{C}_{s,e}^b(\mathbf{Y}) = 0$. In this Scenario, it is straightforward to verify that $\mathcal{C}_{s,e}^b(\mathbf{Y}) = \mathcal{R}_{s,e}^b(\mathbf{Y})$. (N.B. $\mathbf{1}_{s,e}^2 \neq \mathbf{1}_{s,e}$ because of the normalising constant.)

4.2.4 The NOT algorithm

Algorithm 4.6 presents a generic version of the NOT algorithm, described using pseudocode. The main ingredient of the NOT procedure is a contrast function $\mathcal{C}^b_{s,e}(\cdot)$, chosen by the user, depending on the assumed nature of change-points in the data, e.g. as exemplified by our scenarios (S1)–(S4) above. The threshold $\zeta_T > 0$ is a tuning parameter for the method with respect to which the contrast should be tested, while M is the number of the intervals drawn in the procedure. Guidance on the choice of ζ_T and M is given in Section 4.3.

Algorithm 4.6 Narrowest-Over-Threshold algorithm

Input: Data vector $\mathbf{Y} = (Y_1, \ldots, Y_T)'$, F_T^M being a set of M intervals, with start- and end- points drawn independently and uniformly with replacement from $\{1, \ldots, T\}$, $\mathcal{S} = \emptyset$.

Output: Set of estimated change-points $\mathcal{S} \subset \{1, \ldots, T\}$.

procedure NOT(s, e, ζ_T)
  if $e - s < 1$ **then** STOP
  else
    $\mathcal{M}_{s,e} := \left\{m : [s_m, e_m] \in F_T^M, [s_m, e_m] \subset [s, e]\right\}$
    if $\mathcal{M}_{s,e} = \emptyset$ **then** STOP
    else
      $\mathcal{O}_{s,e} := \left\{m \in \mathcal{M}_{s,e} : \max_{s_m \le b \le e_m} \mathcal{C}^b_{s_m,e_m}(\mathbf{Y}) > \zeta_T\right\}$
      if $\mathcal{O}_{s,e} = \emptyset$ **then** STOP
      else
        $m^* :\in \operatorname{argmin}_{m \in \mathcal{O}_{s,e}} |e_m - s_m|$
        $b^* := \operatorname{argmax}_{s_{m^*} \le b \le e_{m^*}} \mathcal{C}^b_{s_{m^*},e_{m^*}}(\mathbf{Y})$
        $\mathcal{S} := \mathcal{S} \cup \{b^*\}$
        NOT(s, b^*, ζ_T)
        NOT$(b^* + 1, e, \zeta_T)$
      end if
    end if
  end if
end procedure

4.2.5 Theoretical properties of NOT

In this section, we analyse the theoretical behaviour of the NOT algorithm in scenarios (S1) and (S2). An attractive feature of our methodology is that proofs for other scenarios can in principle be constructed "at home" by the user, by following the same generic proof strategy as the one we use for these two scenarios. A discussion of the proof strategy, as well its extensions, is given in Section 4.6.2.

First, we revisit the canonical change-point detection problem, (S1), where the signal vector $\mathbf{f} = (f_1, \ldots, f_T)'$ is piecewise-constant. For notational convenience, we set $\sigma_0 = 1$. Again σ_0 is assumed to be known. (If not, one can plug in the MAD estimator, described in Section 4.2.1.)

Theorem 4.2.1. *Suppose Y_t follow (4.3) in Scenario (S1). Let $\delta_T = \min_{j=1,\ldots,q+1}(\tau_j - \tau_{j-1})$, $\Delta_j^{\mathbf{f}} = |f_{\tau_j+1} - f_{\tau_j}|$, $\underline{f}_T = \min_{j=1,\ldots,q} \Delta_j^{\mathbf{f}}$. Furthermore, assume that $\delta_T^{1/2}\underline{f}_T \geq \underline{C}\sqrt{\log T}$ for some large enough $\underline{C}$. Let $\hat{q}$ and $\hat{\tau}_1, \ldots, \hat{\tau}_q$ denote, respectively, the number and locations of change-points, sorted in increasing order, estimated by Algorithm 4.6 with the contrast function given by (4.6). Then there exist constants $C_1, C_2, C_3, C_4 > 0$ (all not depending on T) such that given $C_1\sqrt{\log T} \leq \zeta_T < C_2\delta_T^{1/2}\underline{f}_T$, $M \geq 36T^2\delta_T^{-2}\log(T^2\delta_T^{-1})$, and for sufficiently large T,*

$$\mathbb{P}\left(\hat{q} = q,\ \max_{j=1,\ldots,q}\left(|\hat{\tau}_j - \tau_j|(\Delta_j^{\mathbf{f}})^2\right) \leq C_3 \log T\right) \geq 1 - C_4/T. \tag{4.13}$$

In the simplest case where we have finitely many change-points with $\delta_T \sim T$, we need $M = O(\log T)$ many random intervals for the consistent detection of all the change-points, which leads to a total computational cost of $O(T \log T)$ for the entire procedure. Furthermore, $\max_{j=1,\ldots,q}\left(|\hat{\tau}_j - \tau_j|\right) = O_P(\log T)$, which trails the minimax rate of $O_p(1)$ by only a logarithmic factor.

In addition, we note that the NOT procedure allows for $\delta_T^{1/2}\underline{f}_T$, a quantitity that

characterises the difficulty level of the problem, to be of order $\sqrt{\log T}$. As argued in Chan and Walther (2013) and Fryzlewicz (2014), this is the smallest rate that permits change-point detection for any method, and is thus optimal.

Next, we revisit Scenario (S2), in which the signal is piecewise-linear and continuous. Again, we set $\sigma_0 = 1$ for notational convenience.

Theorem 4.2.2. *Suppose Y_t follow (4.3) in Scenario (S2). Let $\delta_T = \min_{j=1,\ldots,q+1}(\tau_j - \tau_{j-1})$, $\Delta_j^{\mathbf{f}} = |2f_{\tau_j} - f_{\tau_j-1} - f_{\tau_j+1}|$, $\underline{f}_T = \min_{j=1,\ldots,q} \Delta_j^{\mathbf{f}}$. Furthermore, assume that $\delta_T^{3/2}\underline{f}_T \geq \underline{C}\sqrt{\log T}$ for some large enough $\underline{C}$. Let $\hat{q}$ and $\hat{\tau}_1, \ldots, \hat{\tau}_q$ denote, respectively, the number and locations of change-points, sorted in increasing, order estimated by Algorithm 4.6 with the contrast function given by (4.9). Then there exist constants $C_1, C_2, C_3, C_4 > 0$ not depending on T such that given $C_1\sqrt{\log T} \leq \zeta_T < C_2\delta_T^{3/2}\underline{f}_T$, $M \geq 36T^2\delta_T^{-2}\log(T^2\delta_T^{-1})$, and for sufficiently large T,*

$$\mathbb{P}\left(\hat{q} = q,\ \max_{j=1,\ldots,q}\left(|\hat{\tau}_j - \tau_j|(\Delta_j^{\mathbf{f}})^{2/3}\right) \leq C_3(\log T)^{1/3}\right) \geq 1 - C_4/T. \tag{4.14}$$

In the case where we have finitely many change-points with $\delta_T \sim T$, we again need $M = O(\log T)$ many random intervals for the consistent estimation of all the change-points, leading to the total computational cost of $O(T\log T)$. In the most common case of $\underline{f}_T \sim T^{-1}$ (in which the signal f_t is bounded), the resulting change-point detection rate is $O_p(T^{2/3}(\log T)^{1/3})$, which is different from the minimax rate of $O_p(T^{2/3})$ derived by Raimondo (1998) by only a logarithmic factor. Moreover, in more general cases, the difficulty level of the problem in Scenario (S2) can be charaterised by $\delta_T^{3/2}\underline{f}_T$, a quantity analogous to $\delta_T^{1/2}\underline{f}_T$ in the setting of (S1).

Finally, we remark that results similar to Theorem 4.2.1 and Theorem 4.7 can be obtained if we replace the assumption of standard Gaussian noise by $\mathbb{E}\left(\exp(u\varepsilon_t)\right) < \infty$ for some $u > 0$. In essence, we only require the tails of ε_t to be about or lighter than exponential, which can be seen from Step One and Step Two of the proofs in Section 4.6.2

and Section 4.6.3.

4.3 Computational aspects

4.3.1 Computing contrast functions in linear time

The practical performance (in terms of computational cost) of Algorithm 4.6 relies on the fast computation of the contrast functions discussed in Section 4.2.3 on any given interval $[s, e]$. In this section, we show that in all scenarios listed in Section 4.2.3, the cost of computing $\{\mathcal{C}^b_{s,e}(\mathbf{Y})\}_{b=s}^{e-1}$ is $O(e-s+1)$.

Note that the key ingredients in $\mathcal{C}^b_{s,e}(\mathbf{Y})$ under the different scenarios are functions of the inner products, i.e. $\left\langle \mathbf{Y}, \boldsymbol{\phi}^b_{s,e} \right\rangle$, $\left\langle \mathbf{Y}, \boldsymbol{\psi}^b_{s,e} \right\rangle$, $\left\langle \mathbf{Y}, \boldsymbol{\gamma}_{s,b} \right\rangle$, $\left\langle \mathbf{Y}, \boldsymbol{\gamma}_{b+1,e} \right\rangle$, $\left\langle \mathbf{Y}, \mathbf{1}^2_{s,b} \right\rangle$, $\left\langle \mathbf{Y}, \mathbf{1}^2_{b+1,e} \right\rangle$, $\left\langle \mathbf{Y}^2, \mathbf{1}^2_{s,b} \right\rangle$ and $\left\langle \mathbf{Y}^2, \mathbf{1}^2_{b+1,e} \right\rangle$ for $b = s, \ldots, e-1$. For a fixed interval $[s, e]$, by simple algebra, we observe that $\left\langle \mathbf{Y}, \boldsymbol{\phi}^b_{s,e} \right\rangle$ and $\left\langle \mathbf{Y}, \boldsymbol{\psi}^b_{s,e} \right\rangle$ can be decomposed as

$$
\begin{aligned}
\left\langle \mathbf{Y}, \boldsymbol{\phi}^b_{s,e} \right\rangle &= \overleftarrow{a}_{\boldsymbol{\phi},b} \sum_{t=s}^{b} Y_t - \overrightarrow{a}_{\boldsymbol{\phi},b} \sum_{t=b+1}^{e} Y_t \\
&:= \overleftarrow{a}_{\boldsymbol{\phi},b} \overleftarrow{\pi}^{(0)}_b(\mathbf{Y}) - \overrightarrow{a}_{\boldsymbol{\phi},b} \overrightarrow{\pi}^{(0)}_b(\mathbf{Y}), \\
\left\langle \mathbf{Y}, \boldsymbol{\psi}^b_{s,e} \right\rangle &= \overleftarrow{a}^{(1)}_{\boldsymbol{\psi},b} \sum_{t=s}^{b} t Y_t - \overrightarrow{a}^{(1)}_{\boldsymbol{\psi},b} \sum_{t=b+1}^{e} t Y_t + \overleftarrow{a}^{(0)}_{\boldsymbol{\psi},b} \sum_{t=s}^{b} Y_t - \overrightarrow{a}^{(0)}_{\boldsymbol{\psi},b} \sum_{t=b+1}^{e} Y_t \\
&:= \overleftarrow{a}^{(1)}_{\boldsymbol{\psi},b} \overleftarrow{\pi}^{(1)}_b(\mathbf{Y}) - \overrightarrow{a}^{(1)}_{\boldsymbol{\psi},b} \overrightarrow{\pi}^{(1)}_b(\mathbf{Y}) + \overleftarrow{a}^{(0)}_{\boldsymbol{\psi},b} \overleftarrow{\pi}^{(0)}_b(\mathbf{Y}) - \overrightarrow{a}^{(0)}_{\boldsymbol{\psi},b} \overrightarrow{\pi}^{(0)}_b(\mathbf{Y}),
\end{aligned}
$$

where $\overleftarrow{a}_{\boldsymbol{\phi},b}$, $\overrightarrow{a}_{\boldsymbol{\phi},b}$, $\overleftarrow{a}^{(1)}_{\boldsymbol{\psi},b}$, $\overrightarrow{a}^{(1)}_{\boldsymbol{\psi},b}$, $\overleftarrow{a}^{(0)}_{\boldsymbol{\psi},b}$ and $\overrightarrow{a}^{(0)}_{\boldsymbol{\psi},b}$ are scalars that do not depend on $\mathbf{Y}$, and can all be computed at the cost of $O(1)$ using equations given in Section 4.2.3. Here for notational convenience, we use overhead arrows to indicate whether a scalar or a function is associated with observations to the left of b (i.e. $[s, b]$, using $\overleftarrow{\cdot}$) or to the right of b (i.e. $[b+1, e]$, using $\overrightarrow{\cdot}$). We also suppress their dependence on s and e in the

notation. In addition, the following recursive formulae hold

$$\overleftarrow{\pi}_{b+1}^{(k)}(\mathbf{Y}) = \overleftarrow{\pi}_{b}^{(k)}(\mathbf{Y}) + (b+1)^k Y_{b+1},$$
$$\overrightarrow{\pi}_{b}^{(k)}(\mathbf{Y}) = \overrightarrow{\pi}_{b+1}^{(k)}(\mathbf{Y}) + (b+1)^k Y_{b+1},$$

with $\overleftarrow{\pi}_{s}^{(k)}(\mathbf{Y}) = \overrightarrow{\pi}_{e}^{(k)}(\mathbf{Y}) = 0$ for $k = 0, 1$. Consequently, $\overleftarrow{\pi}_{b}^{(k)}(\mathbf{Y})$ and $\overrightarrow{\pi}_{b}^{(k)}(\mathbf{Y})$ for all $b \in \{s, \ldots, e-1\}$ and $k = 0, 1$ (thereby $\left\langle \mathbf{Y}, \boldsymbol{\phi}_{s,e}^{b} \right\rangle$ and $\left\langle \mathbf{Y}, \boldsymbol{\psi}_{s,e}^{b} \right\rangle$) can be computed in a single pass through $Y_s, \ldots, Y_e$. Similar approach can be applied to the remaining inner products involved in the definitions of the contrast functions given in Section 4.2.3, which demonstrates that in all these cases the computation of $\{\mathcal{C}_{s,e}^{b}(\mathbf{Y})\}_{b=s}^{e-1}$ scales linearly with the number of observations.

4.3.2 The NOT solution path algorithm

In general, there are at least two ways of choosing a suitable threshold ζ_T in Algorithm 4.6. It can be either done by selecting a ζ_T which guarantees consistent change-point estimation in a given class of segmentation problems with a high probability, or by using one that optimises a loss function or a model selection criterion. The latter approach proves particularly useful when the theoretically "optimal" threshold is either difficult to derive, or depends on some unobserved quantities, which is typically the case. Denote by $\mathcal{T}(\zeta_T) = \{\hat{\tau}_1(\zeta_T), \ldots, \hat{\tau}_{\hat{q}}(\zeta_T)\}$ the locations of change-points estimated by Algorithm 4.6 with threshold ζ_T (where we suppress the dependence of $\hat{q}$ on ζ_T for notational convenience) and define the solution path as the family of sets $\{\mathcal{T}(\zeta_T)\}_{\zeta_T \geq 0}$. In this section, we present a fast algorithm that computes the entire solution path of Algorithm 4.6. Being able to compute the solution path quickly is essential in Section 4.3.4, where we study a data-driven approach to the choice of ζ_T.

The solution path seen as the function $\zeta_T \mapsto \mathcal{T}(\zeta_T)$ changes only at discrete points, i.e.

Algorithm 4.7 NOT solution path

Input: Intervals $[s_m, e_m]$ and

$$b_m := \mathrm{argmax}_{s_m \le b \le e_m} \mathcal{C}^b_{s_m,e_m}(\mathbf{Y}), \quad c_m := \mathcal{C}^{b_m}_{s_m,e_m}(\mathbf{Y}), \quad l_m := e_m - s_m + 1$$

for all $m \in F_T^M$.

Output: Thresholds $0 = \zeta_T^{(1)} < \ldots < \zeta_T^{(N)}$ and sets of the estimated change-points $\mathcal{T}(\zeta_T^{(1)}), \ldots, \mathcal{T}(\zeta_T^{(N)})$.

procedure BuildBinaryTree(s, e, ζ_T, N)
  $\mathcal{M}_{s,e}$:= set of those $m \in \{1, \ldots, M\}$ such that $[s_m, e_m] \subset [s, e]$
  $\mathcal{O}_{s,e}$:= set of $m \in \mathcal{M}_{s,e}$ such that $c_m > \zeta_T$
  if $\mathcal{O}_{s,e} = \emptyset$ **then** N = NULL
  else
    k := any elements of $\mathrm{argmin}_{m \in \mathcal{O}_{s,e}} l_m$
    N.b := b_k, N.c := c_k, N.Left := NULL, N.Right := NULL
    BuildBinaryTree(s, N.b, ζ_T, N.Left)
    BuildBinaryTree(N.b + 1, e, ζ_T, N.Right)
  end if
end procedure

procedure UpdateBinaryTree(s, e, ζ_T, N)
  if N.c $\le \zeta_T$ **then**
    BuildBinaryTree(s, e, ζ_T, N)
  else
    if N.Left $\neq$ NULL **then**
      UpdateBinaryTree(s, N.b, ζ_T, N.Left)
    end if
    if N.Right $\neq$ NULL **then**
      UpdateBinaryTree(N.b + 1, e, ζ_T, N.Right)
    end if
  end if
end procedure

procedure SolutionPath()
  Set $\mathrm{N_r}$:= NULL, $i := 1$, $\zeta_T^{(1)} := 0$
  BuildBinaryTree(1, T, $\zeta_T^{(1)}$, $\mathrm{N_r}$)
  while $\mathrm{N_r} \neq$ NULL **do**
    $\mathcal{D}$:= {$\mathrm{N_r}$ and all its children nodes}
    $\mathcal{T}(\zeta_T^{(i)}) := \{\mathrm{N.b} | \mathrm{N} \in \mathcal{D}\}$
    $\zeta_T^{(i+1)} := \min_{\mathrm{N} \in \mathcal{D}}\{\mathrm{N.c}\}$
    UpdateBinaryTree(1, T, $\zeta_T^{(i+1)}$, R)
    $i := i + 1$
  end while
end procedure

there exist $0 \leq \zeta_T^{(1)} < \ldots < \zeta_T^{(N)}$, such that $\mathcal{T}(\zeta_T^{(i)}) \neq \mathcal{T}(\zeta_T^{(i+1)})$ for any $i = 1, \ldots, N-1$, and $\mathcal{T}(\zeta_T) = \mathcal{T}(\zeta_T^{(i)})$ for any $\zeta_T \in [\zeta_T^{(i)}, \zeta_T^{(i+1)})$. Furthermore, we have that $T(\zeta_T) = \emptyset$ for any $\zeta_T \geq \zeta_T^{(N)}$. Thresholds $\zeta_T^{(i)}$ are unknown and depend on the data, therefore applying Algorithm 4.6 on a range of pre-specified thresholds typically does not recover the entire solution path. From the computational point of view, repeated application of Algorithm 4.6 to find the solution path is not optimal either, because intuitively one would expect the solutions for $\zeta_T^{(i+1)}$ and $\zeta_T^{(i)}$ to be similar for most i.

We propose our Algorithm 4.7 that computes the entire solution path $\{\mathcal{T}(\zeta_T)\}_{\zeta_T \geq 0}$. Its construction stems from the following two observations. First, for any fixed threshold ζ_T, Algorithm 4.6 implies a binary tree data structure that is constructed according to the order of the detection of each change-point. More specifically, in our implementation, each tree node `N` contains information on the location of the detected change-point `N.b` over the interval of interest, $[\texttt{N.s}, \texttt{N.e}]$, along with the maximum achieved value of the contrast function over all intervals in F_T^M that are subsets of $[\texttt{N.s}, \texttt{N.e}]$ (the largest value and its location are denoted by `N.c` and `N.b`, respectively). Moreover, we define `N.Left` and `N.Right` pointing to the nodes of the next detected change-points in $[\texttt{N.s}, \texttt{N.b}]$ and $[\texttt{N.b} + 1, \texttt{N.e}]$, respectively. We then treat the first detected change-point over $[1, T]$ as the root of the tree and construct its branches in a recursive fashion afterwards. Second, suppose that we have already constructed the tree for ζ_T with root $\texttt{N}_\texttt{r}$. For $\zeta_T' > \zeta_T$, the new tree's root is unchanged if $\texttt{N}_\texttt{r}\texttt{.c} > \zeta_T'$. This observation remains valid for $\texttt{N}_\texttt{r}\texttt{.Left}$ and $\texttt{N}_\texttt{r}\texttt{.Right}$ and all subsequent nodes. Therefore, a branch of the tree has to be reconstructed only if $\texttt{N.c} \leq \zeta_T'$ for some node N. In this way, the tree constructed for ζ_T can be used as a starting point to finding the tree corresponding to ζ_T', thus significantly reducing the computational time in comparison to constructing the tree from scratch. See the pseudo-code of Algorithm 4.7 for more details.

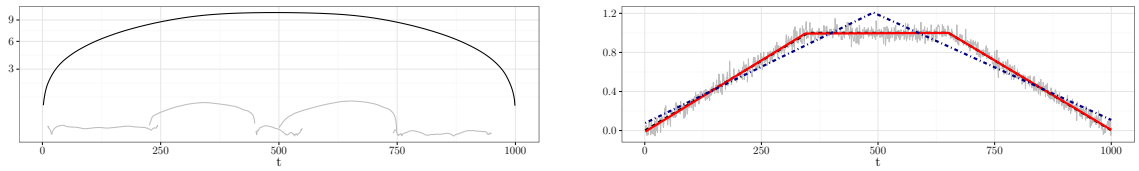
4.3.3 An illustrative example

In this part, we revisit the example shown in the Introduction, and provide a simple illustration of how Algorithm 4.6 and Algorithm 4.7 work on a simulated dataset. Figure 4.3 shows the generated data $\{Y_t\}_{t=1}^{1000}$ following Scenario (S2), where the signal f_t is as in (4.2) and $\sigma_t = 0.05$. The contrast function (4.9) is evaluated for 5 intervals. We observe that the contrast function corresponding to $[1, 1000]$, being the longest interval here, attains its maximum at $b = 490$, which is far from the true change-points located at $\tau = 350$ and $\tau = 650$. Furthermore, $\max_{1\le b\le 1000} \mathcal{C}^b_{s,e}(\mathbf{Y})$ is much larger than the corresponding value for the other intervals considered in Table 4.1. However, thanks to the fact that we focus on the narrowest-over-threshold intervals, Algorithm 4.6 (for any $\zeta_T \in (0.08, 0.83)$) picks at its first iteration an interval with exactly one change-point (depending on ζ_T, it is either $[225, 450]$ or $[500, 750]$) and the maximum of the contrast function computed is close to one of the true change-points.

s	e	$e - s + 1$	$\operatorname{argmax}_{s\le b\le e} \mathcal{C}^b_{s,e}(\mathbf{Y})$	$\max_{s\le b\le e} \mathcal{C}^b_{s,e}(\mathbf{Y})$
1	1000	1000	490	10.19
10	245	236	43	0.08
225	450	226	344	0.76
500	750	251	651	0.83
740	950	211	746	0.03
450	550	101	471	0.07

Table 4.1: Intervals considered in Figure 4.3a and corresponding maxima of the contrast function $\mathcal{C}^b_{s,e}(\mathbf{Y})$ given by (4.9), all calculated for a sample path of Y_t, $t = 1, \ldots, 1000$ generated from model (4.1) with the signal f_t given by (4.2) and the noise $\varepsilon_t \sim \mathcal{N}(0, 0.05^2)$.

Figure 4.4 shows how Algorithm 4.7 proceeds in the example presented in Figure 4.3. At the initial stage that can be seen in Figure 4.4a, the threshold is set to $\zeta_T^{(1)} = 0$ and $b = 417$, the maximum of the contrast function computed for the shortest interval $[450, 550]$ is taken as the root of the binary tree. Then we construct its left and right branches by considering only those intervals specified in Table 4.1 whose endpoints $[s, e] \subset [1, 471]$ and $[s, e] \subset [472, 1000]$, respectively, and the procedure continues for



(a) (b)

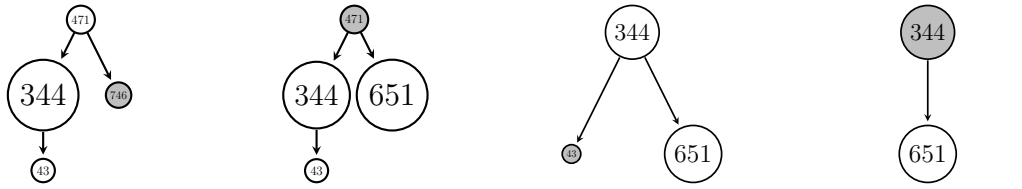
Figure 4.3: An application of the NOT methodology to Y_t generated from model (4.1) with the signal f_t given by (4.2) and i.i.d. $\varepsilon_t \sim \mathcal{N}(0, 0.05^2)$. Figure 4.3a: contrast function $\mathcal{C}^b_{s,e}(\mathbf{Y})$ given by (4.9) evaluated for all $b \in [s, e]$ and intervals $[s, e]$ specified in Table 4.1. For intervals containing one change-point, $\mathcal{C}^b_{s,e}(\mathbf{Y})$ attains its maximum at b close to the change-point. When there are two change-points (black solid line), the maximum is far from both change-points, despite $\max_{s \le b \le e} \mathcal{C}^b_{s,e}(\mathbf{Y})$ being large. Figure 4.3b: observed Y_t (thin grey), true signal (thick dashed black), signal estimated picking the change-point candidate based on the interval corresponding to the largest contrast function (dotted-dashed navy) and the *narrowest-over-threshold* intervals (dashed red).

the resulting nodes. Next, the node with the smallest value of the contrast function is determined ($b = 746$) and the threshold is set to the corresponding minimum $\zeta_T^{(2)} = 0.03$. This guarantees that as Algorithm 4.7 proceeds, there will be at least one update in the binary tree. In our example, the $b = 746$ node is removed and, as the maximum for $[500, 750] \subset [472, 1000]$ exceeds the threshold, the $b = 651$ node is inserted its place. Subsequently, we identify the node with the smallest contrast again ($b = 471$), update the threshold to $\zeta_T^{(3)} = 0.07$ and reconstruct the entire tree, as $b = 471$ in Figure 4.4b constitutes its root. Algorithm 4.7 keeps running until the resulting tree shrinks to `NULL`. In this example, the fourth solution on the path (Figure 4.4d) contains exactly two nodes being close to the true change-points.

4.3.4 Parameter choice

4.3.4.1 Choice of M

We recommend setting $M = 10000$ when the number of observations is of the order of thousands. Our empirical evidence shows that setting a much higher M does not improve



(a) $\zeta_T^{(1)} = 0$ (b) $\zeta_T^{(2)} = 0.03$ (c) $\zeta_T^{(3)} = 0.07$ (d) $\zeta_T^{(4)} = 0.08$

Figure 4.4: First four segmentation trees obtained by Algorithm 4.7 applied to a $Y_1, \ldots, Y_{1000}$ presented in Figure 4.3. The larger the node, the larger the corresponding value of $\max_{s \leq b \leq e} \mathcal{C}_{s,e}^b(\mathbf{Y})$ given by (4.9). The grey nodes correspond to the smallest contrast function for each tree and are updated as Algorithm 4.7 proceeds.

the practical performance of the method in these circumstances. With this value of M, the implementation of Algorithm 4.6 provided in the R **not** package (Baranowski et al., 2016b) achieves the average computation time not longer than 2 seconds in all examples discussed in Section 4.4.2 using a single core of an Intel Xeon 3.6 GHz CPU. This can be accelerated further, as the **not** package allows for computing the contrast function over the intervals drawn in parallel using all available CPU cores.

4.3.4.2 Choice of the threshold ζ_T

Algorithm 4.6 can be applied to a wide range of change-point detection problems with various contrast functions, hence it seems challenging (at least from a theoretical perspective) to find a universal threshold that works well in all settings. In the piecewise-constant and piecewise-linear cases, based on Theorem 4.2.1 and Theorem 4.2.2, respectively, we could take ζ_T of the lowest admissible order (i.e. $\sqrt{\log T}$). Here, our ambition is to come up with a more general data-driven choice of ζ_T based on Algorithm 4.7. Let $\mathcal{T}(\zeta^{(1)}), \ldots, \mathcal{T}(\zeta^{(N)})$ be the NOT solution path, i.e. the collection of candidate models produced by Algorithm 4.7. We propose to select $\mathcal{T}(\zeta^{(k)})$ minimising the Schwarz Information Criterion (SIC) defined as follows. Let $k = 1, \ldots, N$, $\hat{q}_k = |\mathcal{T}(\zeta_T^{(k)})|$ and $\hat{\boldsymbol{\Theta}}_1, \ldots, \hat{\boldsymbol{\Theta}}_{\hat{q}_k+1}$ be the maximum likelihood estimators of the segment parameters in model (4.3) with the estimated change-points $\hat{\tau}_1, \ldots, \hat{\tau}_{\hat{q}_k} \in \mathcal{T}(\zeta_T^{(k)})$. Denote by n_k the total num-

ber of estimated parameters, including the number of free parameters in $\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_{\hat{q}_k+1}$ (this can be different from the dimensionality of each $\boldsymbol{\Theta}_j$ multiplied by the number of segments, as e.g. in (S2)), and $\hat{q}_k$. The SIC criterion is given by

$$\mathrm{SIC}(k) = -2 \sum_{j=1}^{\hat{q}_k+1} \ell(Y_{\hat{\tau}_{j-1}+1}, \ldots, Y_{\hat{\tau}_j}; \hat{\boldsymbol{\Theta}}_j) + n_k \log(T), \tag{4.15}$$

with $\hat{\tau}_0 = 0$ and $\hat{\tau}_{\hat{q}_k+1} = T$. In practice it may not be necessary to calculate SIC for all k, if the number of change-points in the data is expected to be rather moderate. In all applications presented in this work we compute SIC only for k such that $\mathcal{T}(\zeta_T^{(k)}) \leq q_{max}$ with $q_{max} = 25$. In general, solutions on the path corresponding to very small values of ζ_T contain many estimated change-points, especially when M is large. Such solutions are unlikely to minimise (4.15), therefore by considering $\mathcal{T}(\zeta_T^{(k)}) \leq q_{max}$ we achieve computational gains, without adversely impacting the overall performance of the methodology.

4.3.5 Computational complexity of the NOT and NOT solution path algorithms

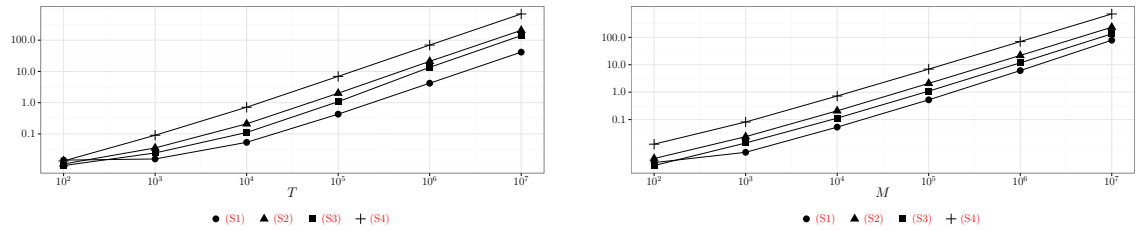
Here we elaborate on the computational complexity of Algorithms 4.6 and 4.7. For both algorithms, the task of computation can be divided into two main parts. First, we need to evaluate a chosen contrast function for all points in the M randomly picked intervals with their endpoints in $\{1, \ldots, T\}$. In the second part, we find potential locations of the change-points for a single threshold ζ_T in the case of Algorithm 4.6 and for all possible thresholds in the case of Algorithm 4.7.

Naturally, the total computational complexity of the first part depends on the cost of computing the contrast function for a single interval. In all scenarios studied in this chapter, this cost is linear in the length of an interval, as shown in Section 4.3.1. The

intervals drawn in the procedures have approximately $T/4$ points on average, therefore the computational complexity of the first part of the computations is $O(MT)$ in a typical application. Importantly, as the calculations for one interval are completely independent of the calculations for another, it is straightforward to run these computations in parallel. Implementation of the NOT methodology available from the R package **not** (Baranowski et al., 2016b) uses to this end the OpenMP framework (Dagum and Menon, 1998), allowing for the efficient use of multiple cores that modern CPUs offer.

As we explain in Section 4.3.2, finding solutions of Algorithm 4.6 for a single threshold ζ_T is equivalent to the construction of a binary tree, which can be performed with the BuildBinaryTree routine given in Algorithm 4.7. Computational cost of this operation is no larger than $O(MK_{\zeta_T})$, where K_{ζ_T} denotes the height of the constructed binary tree with the threshold ζ_T. The computational complexity of finding the entire solution path using Algorithm 4.7 is therefore (in the worst case) of the order $O(MKN)$, where N and K are, respectively, the number of solutions and the maximum tree depth over the entire solution path. However, this is a rough estimate which assumes that for each threshold on the path the binary tree has a different root node, which, from our empirical experience, is highly unlikely to occur in practice. Typically, the consecutive trees on the path differ just slightly, see e.g. Figure 4.6, which significantly reduces the amount of computation that Algorithm 4.7 requires. Finally, we remark that the memory complexity of Algorithm 4.7 is $O(MT)$, which combined with its low computational complexity implies that our approach can handle problem of size T in the millions.

Figure 4.5 shows execution times for the implementation of Algorithm 4.7 available from the R package **not**, with the data Y_t, $t = 1, \ldots, T$, being i.i.d. $\mathcal{N}(0, 1)$. The running times appears to scale linearly both in T (Figure 4.5b) and in M (Figure 4.5b), which provides evidence that the computational complexity of Algorithm 4.7 in this example is practically of the order $O(MT)$.



(a) fixed $M = 10000$ (b) fixed $T = 10000$

Figure 4.5: Execution times (in seconds) for the implementation of Algorithm 4.7 available from R package **not** (Baranowski et al., 2016b), for various feature detection problems with the data Y_t, $t = 1, \ldots, T$ being i.i.d. $\mathcal{N}(0, 1)$. In a single run, computations for the input of the algorithm are performed in parallel, using 8 virtual cores of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM. The computation times are averaged over 10 runs in each case.

4.4 Simulation study

We compare the performance of the R package **not** implementing the NOT methodology against the best competitors available on CRAN. The R code for all simulations can be downloaded from our GitHub repository (Baranowski et al., 2016a). We consider examples following (S1)–(S4) introduced in Section 4.2.3, as well as an extra example satisfying

(S5) $\sigma_t = \sigma_0$ and f_t is a piecewise-quadratic function of t.

Calculations required to derive the contrast function in (S5) are similar to those shown in Section 4.2.3 for (S3); we omit them here.

4.4.1 Simulation methods

To the best of our knowledge, none of the competing packages can be applied in all of the scenarios (S1)–(S5). For change-point detection in the mean, the competitors are: **changepoint** (Killick and Eckley, 2014) implementing the PELT methodology proposed by Killick et al. (2012a), **changepoint.np** (Haynes et al., 2016b) implementing a nonparametric extension of the PELT methodology studied in Haynes et al. (2016a),

wbs (Baranowski and Fryzlewicz, 2015)) implementing the Wild Binary Segmentation proposed by Fryzlewicz (2014), **ecp** (James and Matteson, 2014) implementing the e.cp3o method proposed by James and Matteson (2015), **strucchange** (Zeileis et al., 2002) implementing the methodology of Bai and Perron (2003), **Segmentor3IsBack** (Cleynen et al., 2013) implementing the technique proposed by Rigaill (2010), **nmcdr** (Zou and Lancezhange, 2014), implementing the NMCD methodology of Zou et al. (2014), **stepR** (Hotz and Sieling, 2016), implementing the SMUCE method proposed by Frick et al. (2014). We refer to the corresponding methods as, respectively, PELT, NP-PELT, WBS, e.cp3o, B&P, S3IB, NMCD and SMUCE. All techniques but B&P, WBS, S3IB and SMUCE can be also used for change-point detection in (S4), where change-points occur in the mean and variance of the data.

Only the B&P method allows for change-point detection in piecewise-linear and piecewise-quadratic signals, hence we also study the performance of the trend filtering methodology of Kim et al. (2009) termed as TF hereafter, using the implementation available from the R package **genlasso** (Taylor and Tibshirani, 2014), to have a broader comparison. The TF method aims to estimate a piecewise polynomial signal from the data, not focusing on the change-point detection problem directly. Let $\hat{f}_t^{(TF)}$ denote the TF estimate of the true signal f_t, then the TF estimates of the change-points in (S2) are defined as those τ for which $|2\hat{f}_\tau^{(TF)} - \hat{f}_{\tau-1}^{(TF)} - \hat{f}_{\tau+1}^{(TF)}| > \epsilon$, where $\epsilon > 0$ is a very small number being the numerical tolerance level (more precisely, we set $\epsilon = 1.11 \times 10^{-15}$). In the piecewise-polynomial case, the change-points are defined as those τ for which the third order differences $|\hat{f}_{\tau+2}^{(TF)} - 3\hat{f}_{\tau+1}^{(TF)} + 3\hat{f}_\tau^{(TF)} - \hat{f}_{\tau-1}^{(TF)}| > \epsilon$. Finally, we note that both B&P and TF require a substantial amount of computational resources, with B&P being the slowest among all methods considered in this study. Owing to this, below we consider signals of moderate lengths not exceeding a few thousand, however, as demonstrated in Section 4.3.5, our proposal can be applied even if T is of the order of 10^7.

In this section, we apply Algorithm 4.7 to compute the NOT solution path and always pick the solution minimising the SIC criterion introduced in Section 4.3.4. The number of intervals drawn in the procedure and the maximum number of change-points for SIC are set to $M = 10000$ and $q_{max} = 25$, respectively. In each simulated example, we use the contrast function designed to detect change-points in the scenario that the example follows, derived in Section 4.2.3 under the assumption that ε_t is Gaussian. The resulting method is referred to simply as 'NOT'. The tuning parameters for the competing methods are set to the values recommended by the authors of the corresponding R packages.

The simulation results below show that the NOT methodology with the Gaussian contrast functions is fairly robust against the misspecification of the distribution of the noise. Nevertheless, to illustrate how its performance can be improved further in the presence of heavy-tailed noise, in simulation models for Scenario (S1) we apply Algorithm 4.7 with an additional contrast function, defined for $\mathbf{Y}$ and $1 \leq s \leq b < e < T$ as

$$\mathcal{C}^b_{s,e}(\mathbf{Y}) = \left\langle \mathcal{S}_{s,e}(\mathbf{Y}), \boldsymbol{\psi}^b_{s,e} \right\rangle, \tag{4.16}$$

where for any vector $\mathbf{v} = (v_1, \ldots, v_T)'$ the i-component of $\mathcal{S}_{s,e}(\mathbf{v})$ is given by $\mathcal{S}_{s,e}(\mathbf{v})_i = \operatorname{sign}\left(v_i - (e-s+1)^{-1}\sum_{t=s}^{e} v_t\right)$ and $\boldsymbol{\psi}^b_{s,e}$ is defined by (4.5). The rationale behind (4.16) is as follows. Suppose Y_t satisfies (4.1) with the piecewise-constant signal f_t and let $[s, e)$ be any interval containing exactly one change-point at $\tau \in [s, e)$. For $i = s, \ldots, e$, consider $\tilde{Y}_i = \operatorname{sign}\left(Y_t - (e-s+1)^{-1}\sum_{t=s}^{e} f_t\right)$. Then $\tilde{Y}_i$ decomposes as $\tilde{Y}_i = \tilde{f}_t + \tilde{\varepsilon}_t$, where $\tilde{f}_t = \mathrm{E}\operatorname{sign}\left(Y_t - (e-s+1)^{-1}\sum_{t=s}^{e} f_t\right)$ also has exactly one change-point at τ, while the distribution of $\tilde{\varepsilon}_t$ is binomial (regardless of the distribution for the original noise ε_t), hence its tails are light. In this setting, as argued in Section 4.2.5, (4.6) can be used to identify the location of the change-point in $\tilde{Y}_s, \ldots, \tilde{Y}_e$. As the true signal is unknown, we use $\bar{\mathbf{Y}}_{s,e} := (e-s+1)^{-1}\sum_{t=s}^{e} Y_t$ as a proxy for $(e-s+1)^{-1}\sum_{t=s}^{e} f_t$ when computing

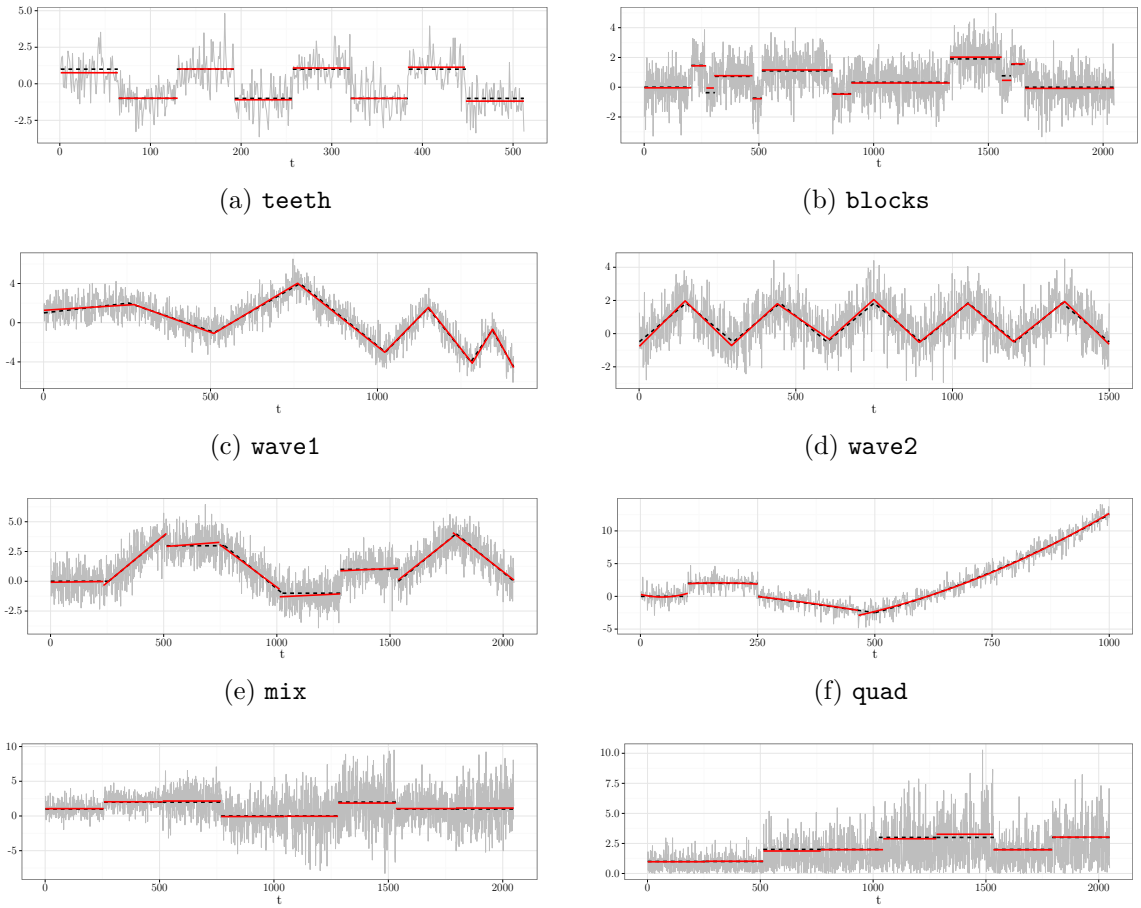
(4.6) for the data $\mathbf{Y}$. In essence, we assign $Y_s - \bar{\mathbf{Y}}_{s,e}, \ldots, Y_e - \bar{\mathbf{Y}}_{s,e}$ (i.e. residuals for fitting a curve with no change-point on a given interval) into two classes (± 1), and apply the contrast function to their labels. Algorithm 4.7 combined with (4.16) and SIC is termed 'NOT HT', where 'HT' stands for heavy tails. We expect that the theoretical properties of NOT HT can be shown along the lines of Theorem 4.2.1, because the tails of $\tilde{\varepsilon}_t$ are lighter than exponential. Finally, we note that the contrast functions addressing the issue of heavy-tails in the noise can be also constructed for (S2)–(S5). For example, when the distribution of the noise is known, this can be achieved by considering GLR given by (4.4) with the correct likelihood function. Otherwise, on any given interval $[s, e]$, one could again consider the vector of residuals from fitting a corresponding curve with no change-point, and truncate the residuals on that interval by a small proportion before plugging it (instead of $\mathbf{Y}$) into the contrast function. This approach is robust, and intuitively preserves more information than using just the sign operator and could be useful for determining the location of a change-point in segments of a more complicated parametric form.

4.4.2 Simulation models

We simulate data according to equation (4.3) using the following test signals.

(M1) `teeth`: piecewise-constant f_t (in Scenario (S1)), $T = 512$, $q = 7$ change-points at $\tau = 64, 128, \ldots, 448$, with the corresponding jump sizes $-2, 2, -2, \ldots, -2$, starting intercept $f_1 = 1$, $\sigma_t = 1$ for $t = 1, \ldots, T$.

(M2) `blocks`: piecewise-constant f_t (in Scenario (S1)), $T = 2024$, $q = 11$ change-points at $\tau = 205, 267, 308, 472, 512, 820, 902, 1332, 1557, 1598, 1659$, with the corresponding jump sizes $1.464, -1.830, 1.098, -1.464, 1.830, -1.537, 0.768, 1.574, -1.135, 0.769, -1.537$, starting intercept $f_1 = 0$, $\sigma_t = 1$ for $t = 1, \ldots, T$. This signal is widely analysed in the literature, see e.g. Donoho and Johnstone (1994).



(a) `teeth` (b) `blocks`

(c) `wave1` (d) `wave2`

(e) `mix` (f) `quad`

(g) `vol`: f_t (h) `vol`: σ_t

Figure 4.6: Examples of data generated from simulation models studied in Section 4.4.2. Figure 4.6a– 4.6g: data series Y_t (thin grey), true signal f_t (dashed black), $\hat{f}_t$ being the OLS estimate of f_t with the change-points estimated by NOT (thick red). Figure 4.6h: centered data $|Y_t - \hat{f}_t|$ (thick grey), true standard deviation σ_t (dashed black) and the estimated standard deviation $\hat{\sigma}_t$ between the change-points detected by NOT (thick red).

(M3) `wave1`: piecewise-linear f_t without jumps in the intercept (in Scenario (S2)), $T = 1408$, $q = 7$ change-points at $\tau = 256, 512, 768, 1024, 1152, 1280, 1344$, with the corresponding changes in slopes $1 \cdot 2^{-6}, -2 \cdot 2^{-6}, 3 \cdot 2^{-6} \ldots, -7 \cdot 2^{-6}$, starting intercept $f_1 = 1$ and slope $f_2 - f_1 = 2^{-8}$, $\sigma_t = 1$ for $t = 1, \ldots, T$.

(M4) `wave2`: piecewise-linear f_t without jumps in the intercept (in Scenario (S2)), $T = 1500$, $q = 9$ change-points at $\tau = 150, 300, \ldots, 1350$, with the corresponding changes in slopes $2^{-5}, -2^{-5}, 2^{-5}, \ldots, -2^{-5}$, starting intercept $f_1 = 2^{-1}$ and slope $f_2 - f_1 = 2^{-6}$, $\sigma_t = 1$ for $t = 1, \ldots, T$.

(M5) `mix`: piecewise-linear f_t with jumps in the intercept (in Scenario (S3)), length $T = 2048$, $q = 7$ change-points at $\tau = 256, 512, \ldots, 1792$, with the corresponding changes in the intercept $0, -1, 0, 0, 2, -1, 0$ and in the slope 2^{-6}, -2^{-6}, -2^{-6}, 2^{-6}, 0, 2^{-6}, -2^{-5}, starting value for the intercept $f_1 = 0$ and slope $f_2 - f_1 = 0$, $\sigma_t = 1$ for $t = 1, \ldots, T$.

(M6) `vol`: piecewise-constant f_t and σ_t (in Scenario (S4)), $T = 2048$, $q = 7$ changes at $\tau = 256, 512, \ldots, 1792$ with the corresponding jumps in f_t and σ_t being 1,0,−2,0,2,−1,0 and 0,1,0,1,0,−1,1, respectively, initial values $f_1 = \sigma_1 = 1$.

(M7) `quad`: piecewise-quadratic f_t (in Scenario (S5)), $T = 1000$, $q = 3$ change-points at $\tau = 100, 250, 500$, with the corresponding changes in the intercept $2, -2, 0$, in the slope $0, -10^{-1}, 10^{-1}$ and in the quadratic coefficient $0, 0, 2 \times 10^{-5}$, the initial values $f_1 = f_2 - f_1 = f_3 - 2f_2 + f_1 = 0$, $\sigma_t = 1$ for all $t = 1, \ldots, T$.

Figure 4.6 shows the examples of the data generated from models (M1)–(M7), as well as estimates produced by NOT, for the i.i.d. $\mathcal{N}(0, 1)$ noise ε_t.

Method	Model	$\hat{q}-q$ ≤ -3	-2	-1	0	1	2	≥ 3	MSE	$d_H \times 10^2$	time
B&P	(M1)	70	8	1	21	0	0	0	0.703	11.39	0.27
e-cp3o		0	0	0	**100**	0	0	0	0.052	**0.48**	2.32
NMCD		0	0	0	**96**	4	0	0	0.093	0.76	1.38
NOT		0	0	0	**99**	1	0	0	0.053	0.54	0.08
NOT HT		0	0	0	**99**	1	0	0	0.055	**0.51**	0.1
NP-PELT		0	0	0	86	11	2	1	0.068	0.85	0.03
PELT		0	0	0	**100**	0	0	0	0.052	**0.48**	0
S3IB		0	0	0	**92**	6	2	0	0.055	0.67	0.11
SMUCE		0	0	0	**100**	0	0	0	0.083	0.57	0.22
WBS		0	0	0	**97**	3	0	0	0.054	0.58	0.11
B&P	(M2)	100	0	0	0	0	0	0	0.314	12.56	4.29
e-cp3o		100	0	0	0	0	0	0	0.127	5.69	188.84
NMCD		0	5	64	31	0	0	0	0.035	1.82	4.92
NOT		0	4	61	35	0	0	0	0.026	1.56	0.11
NOT HT		2	8	54	28	8	0	0	0.033	2.08	0.23
NP-PELT		0	0	27	44	15	9	5	0.029	2.13	0.49
PELT		11	33	45	11	0	0	0	0.035	2.97	0.01
S3IB		0	2	49	**49**	0	0	0	0.024	**1.42**	0.51
SMUCE		59	36	5	0	0	0	0	0.069	3.44	0.03
WBS		0	1	45	**53**	0	1	0	0.026	**1.31**	0.22
B&P	(M3)	0	0	100	0	0	0	0	0.218	3.78	147.23
NOT		0	0	0	**99**	1	0	0	0.015	**0.99**	0.63
TF		0	0	0	0	0	0	100	0.019	8.33	63.98
B&P	(M4)	0	1	3	**96**	0	0	0	0.072	2.59	168.12
NOT		0	0	0	**100**	0	0	0	0.016	**1.21**	0.53
TF		0	0	0	0	0	0	100	0.016	4.3	64.81
B&P	(M5)	0	0	0	**100**	0	0	0	0.02	**2.42**	382.96
NOT		0	0	0	**99**	1	0	0	0.02	**2.42**	0.51
TF		0	0	0	0	0	0	100	0.026	6.03	77.09
e-cp3o	(M6)	94	3	0	3	0	0	0	0.378	16.83	11.35
NMCD		0	0	7	83	8	2	0	0.057	2.54	4.8
NOT		0	0	4	**94**	2	0	0	0.049	**1.69**	1.22
NP-PELT		0	0	0	20	30	19	31	0.123	2.96	0.61
PELT		9	15	28	48	0	0	0	0.074	8	0.02
B&P	(M7)	0	0	0	**100**	0	0	0	0.021	**1.94**	44.14
NOT		0	0	0	**100**	0	0	0	0.02	**1.78**	0.31
TF		0	0	0	0	0	0	100	0.049	23.33	59.56

Table 4.2: Distribution of $\hat{q}-q$ for data generated according to (4.3) with the noise term ε_t being i.i.d. $\mathcal{N}(0,1)$ for various choices of f_t and σ_t given in Section 4.4.2 and competing methods introduced in Section 4.4. Also, the average Mean-Square Error of the resulting estimate of the signal f_t, average Hausdorff distance d_H given by (4.18) and average computation time in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM, all calculated over 100 simulated data sets. Bold: methods with the largest empirical frequency of $\hat{q}-q=0$ or smallest average d_H and those within 10% of the highest, or, respectively, within 10% of the lowest.

Method	Model	$\hat{q}-q$ ≤ -3	-2	-1	0	1	2	≥ 3	MSE	$d_H \times 10^2$	time
B&P	(M1)	82	9	2	7	0	0	0	0.832	14.15	0.26
e-cp3o		0	0	0	**100**	0	0	0	0.109	**1.02**	2.15
NMCD		0	0	0	**98**	2	0	0	0.149	1.43	1.28
NOT		0	0	0	**99**	1	0	0	0.112	**1.05**	0.08
NOT HT		0	0	0	**97**	3	0	0	0.127	1.35	0.09
NP-PELT		0	0	0	73	24	2	1	0.131	1.43	0.04
PELT		0	0	0	**100**	0	0	0	0.11	**1.04**	0
S3IB		0	0	0	**94**	5	1	0	0.113	1.17	0.11
SMUCE		0	1	15	84	0	0	0	0.192	2.23	0.23
WBS		0	0	0	**98**	2	0	0	0.11	**1.05**	0.11
B&P	(M2)	100	0	0	0	0	0	0	0.358	14.34	5.64
e-cp3o		100	0	0	0	0	0	0	0.142	8.12	194.18
NMCD		37	31	26	5	1	0	0	0.073	4.02	5.06
NOT		27	28	25	17	2	1	0	0.062	**3.48**	0.11
NOT HT		42	27	23	7	1	0	0	0.076	4.23	0.23
NP-PELT		1	12	26	**25**	17	16	3	0.067	3.91	0.54
PELT		92	7	0	1	0	0	0	0.106	7.28	0.01
S3IB		35	23	24	17	0	1	0	0.065	3.94	0.53
SMUCE		100	0	0	0	0	0	0	0.139	5.72	0.04
WBS		30	26	27	16	1	0	0	0.064	**3.64**	0.22
B&P	(M3)	0	0	100	0	0	0	0	0.246	3.94	146.74
NOT		0	0	0	**99**	1	0	0	0.032	**1.47**	0.54
TF		0	0	0	0	0	0	100	0.032	8.42	63.71
B&P	(M4)	16	55	28	1	0	0	0	0.336	6.48	167.31
NOT		0	0	0	**98**	2	0	0	0.039	**2.08**	0.47
TF		0	0	0	0	0	0	100	0.031	4.44	64.41
B&P	(M5)	0	0	8	**92**	0	0	0	0.044	**3.31**	380.84
NOT		0	0	5	**93**	2	0	0	0.045	**3.52**	0.48
TF		0	0	0	0	0	0	100	0.041	5.89	78.46
e-cp3o	(M6)	95	2	0	3	0	0	0	0.372	16.55	11.67
NMCD		0	0	15	79	6	0	0	0.058	3.35	4.78
NOT		0	0	10	**89**	1	0	0	0.045	**2.07**	1.22
NP-PELT		0	0	0	22	24	22	32	0.12	2.97	0.61
PELT		11	15	28	44	2	0	0	0.075	7.83	0.02
B&P	(M7)	0	0	35	**65**	0	0	0	0.066	6.47	44.26
NOT		0	1	37	**62**	0	0	0	0.064	**5.78**	0.31
TF		0	0	0	0	0	1	99	0.075	22.71	60.17

Table 4.3: Distribution of $\hat{q}-q$ for data generated according to (4.3) with the noise term ε_t being i.i.d. $\mathcal{N}(0,2)$ for various choices of f_t and σ_t given in Section 4.4.2 and competing methods introduced in Section 4.4. Also, the average Mean-Square Error of the resulting estimate of the signal f_t, average Hausdorff distance d_H given by (4.18) and average computation time in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM, all calculated over 100 simulated data sets. Bold: methods with the largest empirical frequency of $\hat{q}-q=0$ or smallest average d_H and those within 10% of the highest, or, respectively, within 10% of the lowest.

Method	Model	$\hat{q}-q$ ≤ -3	-2	-1	0	1	2	≥ 3	MSE	$d_H \times 10^2$	time
B&P	(M1)	76	4	1	19	0	0	0	0.745	13.04	0.25
e-cp3o		0	0	0	**100**	0	0	0	0.097	0.87	2.13
NMCD		0	0	0	**94**	6	0	0	0.141	1.35	1.28
NOT		0	1	0	**95**	3	1	0	0.107	1.19	0.08
NOT HT		0	0	0	**99**	0	1	0	0.093	**0.79**	0.09
NP-PELT		0	0	0	71	22	6	1	0.141	1.57	0.04
PELT		0	0	0	69	13	14	4	0.145	1.4	0
S3IB		0	1	0	76	10	9	4	0.136	1.47	0.11
SMUCE		0	0	1	52	23	14	10	0.155	2.6	0.21
WBS		0	0	0	64	4	23	9	0.151	1.91	0.11
B&P	(M2)	100	0	0	0	0	0	0	0.311	12.55	5.36
e-cp3o		100	0	0	0	0	0	0	0.147	9.1	191.73
NMCD		15	36	37	12	0	0	0	0.06	**3.37**	5.06
NOT		51	21	17	9	2	0	0	0.079	4.8	0.11
NOT HT		23	26	36	15	0	0	0	0.054	**3.08**	0.23
NP-PELT		0	4	10	19	27	19	21	0.077	4.03	0.51
PELT		20	21	19	14	14	6	6	0.108	5.02	0.01
S3IB		88	8	2	2	0	0	0	0.13	10.22	0.5
SMUCE		14	16	23	**22**	6	8	11	0.108	6.02	0.03
WBS		21	12	12	15	15	10	15	0.104	4.98	0.22
B&P	(M3)	0	0	100	0	0	0	0	0.261	4.16	147.23
NOT		0	0	1	**96**	1	1	1	0.037	**1.89**	0.52
TF		0	0	0	0	0	0	100	0.035	8.42	64.08
B&P	(M4)	16	44	37	3	0	0	0	0.323	6.27	171.88
NOT		0	0	0	**96**	3	1	0	0.042	**2.24**	0.44
TF		0	0	0	0	0	0	100	0.032	4.38	66.53
B&P	(M5)	0	1	6	**93**	0	0	0	0.045	**3.44**	384.72
NOT		0	1	2	**90**	3	3	1	0.047	**3.48**	0.5
TF		0	0	0	0	0	0	100	0.041	5.91	78.1
e-cp3o	(M6)	96	3	1	0	0	0	0	0.481	17.95	11.91
NMCD		1	28	38	30	2	0	1	0.098	9.45	4.83
NOT		1	10	42	**35**	9	1	2	0.188	8.17	1.24
NP-PELT		0	1	4	14	22	16	43	0.359	**5.34**	0.75
PELT		22	22	35	17	3	1	0	0.215	12.8	0.03
B&P	(M7)	0	0	41	**59**	0	0	0	0.066	**5.93**	44.19
NOT		0	2	51	44	2	1	0	0.077	7.7	0.32
TF		0	0	0	0	0	0	100	0.075	22.42	60.33

Table 4.4: Distribution of $\hat{q}-q$ for data generated according to (4.3) with the noise term ε_t being i.i.d. Laplace $\left(0, (\sqrt{2})^{-1}\right)$ (N.B. $\mathrm{Var}(\varepsilon_t) = 1$ here) for various choices of f_t and σ_t given in Sectopm 4.4.2 and competing methods introduced in Section 4.4. Also, the average Mean-Square Error of the resulting estimate of the signal f_t, average Hausdorff distance d_H given by (4.18) and average computation time in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM, all calculated over 100 simulated data sets. Bold: methods with the largest empirical frequency of $\hat{q}-q=0$ or smallest average d_H and those within 10% of the highest, or, respectively, within 10% of the lowest.

Method	Model	$\hat{q}-q$ ≤ -3	-2	-1	0	1	2	≥ 3	MSE	$d_H \times 10^2$	time
B&P	(M1)	65	12	0	23	0	0	0	0.67	10.76	0.26
e-cp3o		0	0	0	**100**	0	0	0	0.044	**0.39**	2.22
NMCD		0	0	0	**94**	6	0	0	0.092	0.81	1.31
NOT		0	0	0	**94**	5	1	0	0.046	0.57	0.08
NOT HT		0	0	0	**98**	2	0	0	0.045	0.47	0.1
NP-PELT		0	0	0	73	14	11	2	0.082	1.37	0.03
PELT		0	0	0	63	6	16	15	0.092	1.68	0
S3IB		0	0	0	54	7	20	19	0.096	1.84	0.11
SMUCE		0	0	0	45	22	19	14	0.091	2.53	0.21
WBS		0	0	0	44	3	28	25	0.105	2.44	0.11
B&P	(M2)	100	0	0	0	0	0	0	0.302	11.98	4.28
e-cp3o		100	0	0	0	0	0	0	0.126	5.87	197.26
NMCD		0	4	66	**29**	0	1	0	0.032	**1.92**	5.13
NOT		2	16	33	**31**	14	3	1	0.032	4.09	0.11
NOT HT		1	7	62	**28**	2	0	0	0.027	**1.9**	0.23
NP-PELT		0	0	6	22	20	23	29	0.048	3.91	0.46
PELT		0	3	16	19	20	12	30	0.066	3.98	0.01
S3IB		29	10	26	20	4	11	0	0.065	4.38	0.49
SMUCE		0	5	11	25	14	13	32	0.056	5.36	0.03
WBS		0	3	15	11	21	15	35	0.067	4.7	0.22
B&P	(M3)	0	0	100	0	0	0	0	0.217	3.63	149.51
NOT		0	0	0	**99**	1	0	0	0.015	**1**	0.63
TF		0	0	0	0	0	0	100	0.017	8.4	66.66
B&P	(M4)	0	0	10	**90**	0	0	0	0.081	2.78	175.34
NOT		0	0	0	**94**	5	1	0	0.019	**1.51**	0.54
TF		0	0	0	0	0	0	100	0.017	4.44	68.33
B&P	(M5)	0	0	0	**100**	0	0	0	0.019	**2.29**	392
NOT		0	0	0	**96**	4	0	0	0.019	**2.33**	0.53
TF		0	0	0	0	0	0	100	0.026	6.01	80.41
e-cp3o	(M6)	91	2	2	4	0	1	0	0.327	14.05	11.51
NMCD		0	12	47	**36**	5	0	0	0.053	8.56	4.94
NOT		0	4	17	**35**	25	12	7	0.08	6.1	1.26
NP-PELT		0	0	2	9	22	19	48	0.205	**5.1**	0.66
PELT		7	14	26	**33**	15	5	0	0.112	8.88	0.03
B&P	(M7)	0	0	0	**99**	1	0	0	0.021	**2.5**	45.59
NOT		0	0	8	79	11	2	0	0.03	4.28	0.32
TF		0	0	0	0	0	0	100	0.05	23.32	62.79

Table 4.5: Distribution of $\hat{q}-q$ for data generated according to (4.3) with the noise term ε_t being i.i.d. $(3/5)^{1/2}t_5$ (N.B. $\mathrm{Var}(\varepsilon_t) = 1$ here) for various choices of f_t and σ_t given in Section 4.4.2 and competing methods introduced in Section 4.4. Also, the average Mean-Square Error of the resulting estimate of the signal f_t, average Hausdorff distance d_H given by (4.18) and average computation time in seconds using a single core of an Intel Xeon 3.6 GHz CPU with 16 GB of RAM, all calculated over 100 simulated data sets. Bold: methods with the largest empirical frequency of $\hat{q}-q=0$ or smallest average d_H and those within 10% of the highest, or, respectively, within 10% of the lowest.

4.4.3 Results and discussion

Tables 4.2–4.5 summarise the results for the four different distributions of the noise ε_t. For each method, we show a frequency table for the distribution of $\hat{q} - q$, where $\hat{q}$ is the number of the estimated change-points and q denotes the true number of change-points. We also report Monte-Carlo estimates of the Mean-Square error

$$\mathrm{MSE} = \frac{1}{T}\sum_{t=1}^{T} \mathrm{E}\left(f_t - \hat{f}_t\right)^2. \tag{4.17}$$

For all methods but TF, $\hat{f}_t$ is calculated by finding the OLS approximation of the signal of the appropriate type depending on the true f_t, between each consecutive pair of estimated change-points. For TF, $\hat{f}_t$ used in the definition of the MSE is the penalised least squares estimate of f_t returned by the TF algorithm. To assess the performance of each method in terms of the accuracy of the estimated locations of the change-points, we also report estimates of the (scaled) Hausdorff distance defined as

$$d_H = T^{-1}\,\mathrm{E}\max\left\{\max_{j=0,\ldots,q+1}\min_{k=0,\ldots,\hat{q}+1}|\tau_j - \hat{\tau}_k|, \max_{k=0,\ldots,\hat{q}+1}\min_{j=0,\ldots,q+1}|\hat{\tau}_k - \tau_j|\right\}, \tag{4.18}$$

where $0 = \tau_0 < \tau_1 < \ldots \tau_q < \tau_{q+1} = T$ and $0 = \hat{\tau}_0 < \hat{\tau}_1 < \ldots \hat{\tau}_q < \hat{\tau}_{q+1} = T$ denote, respectively, true and estimated locations of the change-points. From the definition above, it follows that that $0 \leq d_H \leq 1$. An estimator is regarded to perform well when its d_H is close to 0. However, when the number of change-points is under-estimated or some of the estimated change-points are not close to the real ones, d_H is closer to 1.

The points below, grouped according to the scenario for the type of segmentation problem, discuss the results.

(S1) Two simulation models follow this scenario: (M1) `teeth` and (M2) `blocks`. The `teeth` signal with the $\mathcal{N}(0,1)$ noise is a relatively easy setting, where all methods but B&P always detect all change-points. PELT, SMUCE and e-cp3o per-

form exceptionally well here, always finding exactly 7 change-points close to the true locations. NMCD, NOT, NOT HT, S3IB and WBS overestimate q sporadically, while NP-PELT shows a tendency of detecting some additional change-points. The performance of NP-PELT and SMUCE deteriorates in (M1) when $\varepsilon_t \sim \mathcal{N}(0, 2)$; SMUCE underestimates q, while NP-PELT overestimates q more frequently than in the $\mathcal{N}(0, 1)$ case. In the heavy-tailed scenarios ($\varepsilon_t \sim (3/5)^{1/2} t_5$ and $\varepsilon_t \sim \text{Laplace}\left(0, (\sqrt{2})^{-1}\right)$), NOT, NOT HT, NMCD and e-cp3o, offer the best performance, while the other methods but B&P tend to slightly overestimate q.

For the `blocks` signal with $\mathcal{N}(0, 1)$ noise, WBS performs the best, S3IB is the second best, while NOT is the third best method, which can be seen from the corresponding values of the Hausdorff distance d_H and MSE. B&P, e-cp3o and SMUCE underestimate, while NP-PELT tends to overestimate the number of change-points. In the $\mathcal{N}(0, 2)$ case, NOT performs the best in terms of d_H and MSE, while WBS is the second best. In the heavy-tailed noise cases, performance of NOT HT and NMCD stands out, with the former achieving the best d_H and MSE, while PELT, NP-PELT, SMUCE tend to overestimate q.

Overall, we observe that only three methods, namely NMCD, NOT and NOT HT, perform reasonably well across all the examples with a piecewise constant signal.

(S2) Two signals follow this scenario: (M3) `wave1` and (M4) `wave2`. For the `wave1` signal, we observe a pattern common across all considered scenarios for ε_t: typically B&P underestimates the number of changes in the slope coefficient, TF largely overestimates q while NOT tends to find the correct number of the change-points. The NOT estimates lie close to the true locations of the change-points, which can be seen from very low values of d_H. Moreover, NOT estimates of the underlying signal yields MSEs comparable to or even lower than the corresponding values for TF, despite the latter procedure having been designed solely for the estimation of

f_t.

In (M4), NOT performs the best across all scenarios for ε_t, most often identifying the correct number of change-points. In the case of $\varepsilon_t \sim \mathcal{N}(0,1)$ and $\varepsilon_t \sim (3/5)^{1/2} t_5$ B&P performs reasonably well, while in the remaining two scenarios it frequently fails to identify some of the change-points.

Finally, the NOT estimates are orders of magnitude quicker to compute than the competing estimators.

(S3) The (M5) `mix` signal follows this scenario. In the case of $\varepsilon_t \sim \mathcal{N}(0,1)$, NOT performs slightly better than B&P, always correctly identifying the number of change-points. TF performs well in terms of the average MSE, but it largely overestimates the number of change-points. On the other hand, NOT identifies the correct number of change-points more frequently than B&P when the noise $\varepsilon_t \sim \mathcal{N}(0,2)$, but B&P achieves a slightly lower d_H in that scenario. In the heavy-tailed examples, B&P performs very well, while NOT slightly overestimates the number of change-points. However, we emphasise again that NOT is much quicker to compute than the competing methods.

(S4) The (M6) `vol` signal follows this scenario. In the cases of $\varepsilon_t \sim \mathcal{N}(0,1)$ and $\varepsilon_t \sim \mathcal{N}(0,2)$, NOT most frequently estimates the number of change-points correctly and achieves the lowest average d_H, while NMCD is the second best. In the heavy-tailed scenarios, NP-PELT achieves the best d_H, but it exhibits an overall tendency of overestimating the number of change-points. Besides, e-c3po and PELT in all cases underestimate q.

(S5) The (M7) `quad` signal follows this scenario. In the case of $\varepsilon_t \sim \mathcal{N}(0,1)$, both NOT and B&P always correctly estimate the number of change-points, however, NOT estimates are on average closer to the true locations. The problem becomes

more challenging for $\varepsilon_t \sim \mathcal{N}(0, 2)$, where all methods frequently fail to identify one change-point, with NOT being marginally better than B&P and significantly better than TF. The challenge here is that the signal between $t = 251$ to $t = 1000$ can be approximated by a quadratic function reasonably well, therefore SIC and other criteria may prefer a simpler model without a change-point at $t = 500$ when the standard deviation of the noise is relatively large. In the heavy-tailed cases, NOT slightly overestimates the number of change-points, however its performance in terms of d_H remains reasonably close to the performance of B&P, which is the best in these examples.

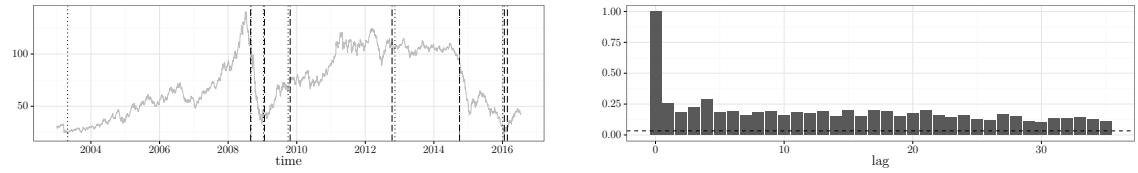
In all simulated scenarios, NOT is always either the best or not far from the best method. Importantly, it is quick to compute, which gives it a particular advantage over its competitors in Scenarios (S2), (S3) and (S5), where the computational complexity of the competing methods is polynomial, which is prohibitive for large sample sizes. Furthermore, NOT with the contrast function derived under the assumption that the noise is Gaussian is relatively robust against the misspecification in the distribution of ε_t.

4.5 Real data analysis

We present applications of the NOT methodology to three real data sets: oil price log-returns, temperature anomalies data and the UK House Price Index. All R code used in this section is available from our GitHub repository (Baranowski et al., 2016a).

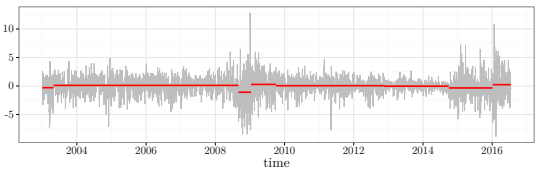
4.5.1 OPEC Reference Basket oil price

We perform change-point analysis on the daily Organisation of the Petroleum Exporting Countries (OPEC) Reference Basket oil price from 1 January, 2003 to 15 July, 2016. The data were obtained from the OPEC database through the R package **Quandl** (McTaggart

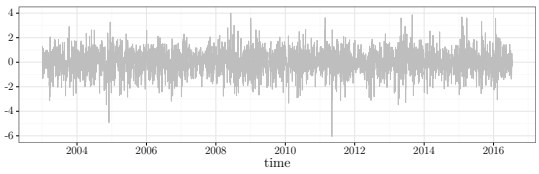


(a) daily oil price

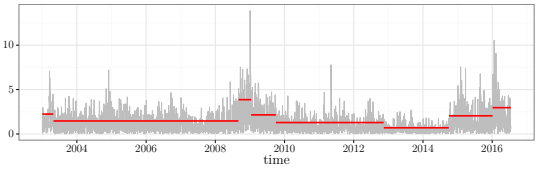
(b) autocorrelation of the squared log-returns



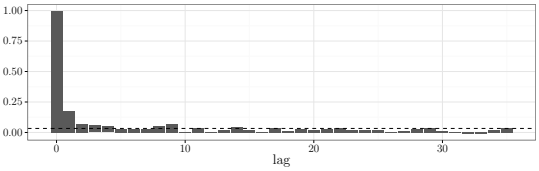
(c) log-returns



(d) residuals



(e) centred log-returns



(f) autocorrelation of the squared residuals

Figure 4.7: Change-point analysis on the daily OPEC Reference Basket oil price in USD from 1 January, 2003 to 15 July, 2016. Figure 4.7a: price series P_t (thin grey), locations of the change-points detected with NOT (vertical dotted lines) and NMCD (vertical dashed lines). Figure 4.7b: autocorrelation function of Y_t^2. Figure 4.7c: log-returns $Y_t = 100 \log(P_t/P_{t-1})$ (thin grey), the fitted piecewise-constant mean $\hat{f}_t$ (thick red). Figure 4.7d residuals $\hat{\varepsilon}_t = (Y_t - \hat{f}_t)/\hat{\sigma}_t$. Figure 4.7e: the centred log-returns $|Y_t - \hat{f}_t|$ (thin grey), fitted piecewise-constant volatility $\hat{\sigma}_t$ (thick red). Figure 4.7f: autocorrelation of $\hat{\varepsilon}_t^2$. The exact locations of the change-points detected with NOT are given in Table 4.6.

NOT	NMCD	Event
29 April 2003	N/A	Invasion of Iraq
1 September 2008	28 August 2008	critical stage of the subprime mortgage crisis
27 January 2009	22 January 2009	tensions in the Gaza Strip
1 October 2009	23 October 2009	
12 November 2012	12 October 2012	beginning of a period of low volatility
30 September 2014	1 October 2014	
5 January 2016	21 January 2016	beginning of a sell-off leading the price to 12-year low
N/A	22 February 2016	

Table 4.6: Change-points detected using NOTWBS and NMCD methods in the daily OPEC Reference Basket oil price data from 1 January 2003 to 15 July 2016, with the majority of them dated.

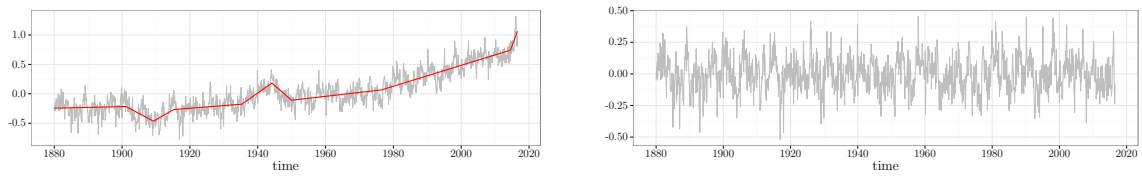
et al., 2016). Instead of working with the raw price series, we analyse the log-returns series $Y_t = 100 \log(P_t/P_{t-1})$, where P_t denotes the daily oil price. One of the stylised facts of the financial time series data is that the autocorrelation of assets returns are weak, while squared returns tend to exhibit strong autocorrelation, which is the case for the oil price time series (see Figure 4.7b). This phenomenon can be possibly explained by the existence of the structural breaks in the mean and variance structure of the data series (Fryzlewicz et al., 2006; Mikosch and Stărică, 2004). In this study, we apply NOT with the contrast function given by (4.12), which is designed to detect changes in both the mean and the volatility. For comparison, we also report change-points detected with the NMCD method of Zou et al. (2014), which was the second best method for change-point detection in Scenario (S4) in the simulation study of Section 4.4.

We apply Algorithm 4.7 to compute the NOT solution path and choose the model achieving the lowest SIC given by (4.15), setting the number of intervals drawn $M = 10000$ and the maximum number of change-points $q_{max} = 25$. Computations for the solution path and model selection are performed using the R package **not** (Baranowski et al., 2016b). For the NMCD procedure, we use the `nmcd` routine from the R package **nmcdr** (Zou and Lancezhange, 2014), setting the maximum number of change-points to $q_{max} = 25$ as well.

Figure 4.7 illustrates the results of our analysis. The oil price time series and the locations of the change-points identified by NOT and NMCD can be seen in Figure 4.7a. Both methods discover 7 change-points, largely agreeing on their locations, in the sense that for 6 out of 7 NOT estimates, NMCD detects a change-point nearby. However, NMCD does not indicate any change-point around the first change-point identified by NOT on 29 April 2003. This date can be clearly related to the end of the 2003 invasion of Iraq, which initiated the upward trend in the oil price lasting almost ceaselessly until the beginning of the 2008–09 financial crisis. On the other hand, NMCD indicates two

change-points in the first quarter of 2016, while NOT finds a single change-point in that period. Table 4.6 lists the exact locations of the change-points detected by the two methods and the events that can be related to some of them. Figure 4.7f shows the autocorrelation function for the squared residuals obtained by subtracting the sample mean and dividing by the standard deviations from the data in each segment. It appears that there is little autocorrelation in the squares of the residuals, meaning that (S4) models the data in this example reasonably well.

4.5.2 Temperature anomalies



(a) data (b) residuals

Figure 4.8: Change-point analysis for the GISSTEMP data set introduced in Section 4.5.2. Figure 4.8a: the data series Y_t (thin grey) and $\hat{f}_t$ estimated using change-points returned by NOT (thick red). Figure 4.8b: residuals $\hat{\varepsilon}_t = Y_t - \hat{f}_t$.

For the second application, we analyse the GISS Surface Temperature anomalies data set available from GISTEMP Team (2016), consisting of monthly temperature anomalies recorded from January 1880 to June 2016. The anomaly here is defined as the difference between the average global temperature in a given month and the baseline value, being the average calculated for that time of the year over the 30-year period from 1951 to 1980; for more details see Hansen et al. (2010). This and similar anomalies series are frequently studied in literature with a particular focus on identifying change-points in the data, see e.g. Ruggieri (2013) or James and Matteson (2015).

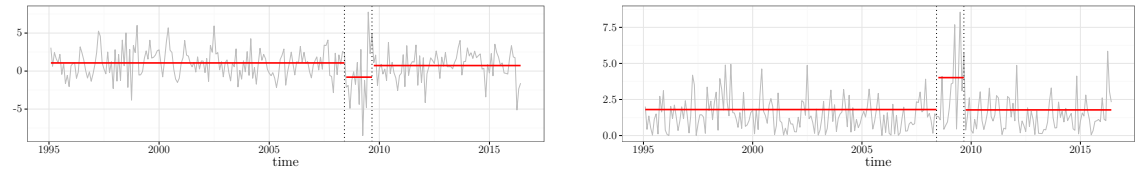
The plot of the data (Figure 4.8a) clearly indicates the presence of a linear trend with several change-points in the temperature anomalies series. The corresponding changes

are not abrupt, therefore we believe that Scenario (S2) with change-points in the slope of the trend is most appropriate here. To detect the locations of the change-points, we apply Algorithm 4.7 with the contrast given by (4.9), combined with the SIC criterion to determine the best model on the solution path. The maximum number of change-points for NOT is set to $q_{max} = 25$ and $M = 50000$.

Figure 4.8 shows the data, the NOT estimate of the piecewise-linear trend and the empirical residuals. We identify 8 change-points located at the following dates: March 1901, December 1910, July 1915, June 1935, April 1944, December 1946, June 1976 and May 2015. Previous studies conducted on similar temperature anomalies series (observed at a yearly frequency and obtained from a different source), report change-points around 1910, 1945 and 1976 (see Ruggieri (2013) for an overview of a number of related analyses). In addition to the change-points around these dates, NOT identifies two periods, 1901–1915 and 1935–1946, where local deviations from the baseline trend are clearly visible. We also observe a long-lasting upward trend in the anomalies series starting in December 1946. NOT estimates indicate that the slope of the trend is increasing, with the most recent change-point in May 2015.

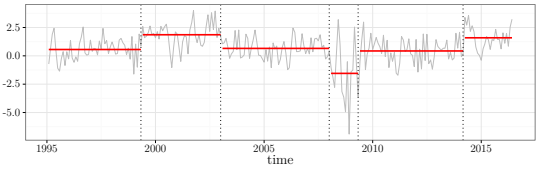
4.5.3 UK House Price Index

In our final example, we analyse monthly percentage changes in the UK House Price Index (HPI) which provides an overall estimate of the changes in house prices across the UK. The data and a detailed description of how the index is calculated are available online from UK Land Registry (2016). Fryzlewicz (2016), who proposed a method for signal estimation and change-point dectection in Scenario (S1), used this data set to illustrate the performance of his methodology. We perform similar analysis, assuming the more flexible Scenario (S4), allowing for changes both in the mean and the variance of the series, which, we argue, leads to some additional insights and better-interpretable

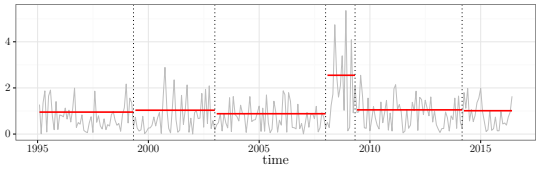


(a) Hackney: $\hat{f}_t$

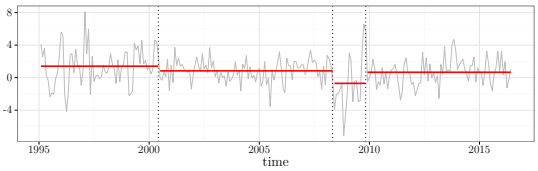
(b) Hackney: $\hat{\sigma}_t$



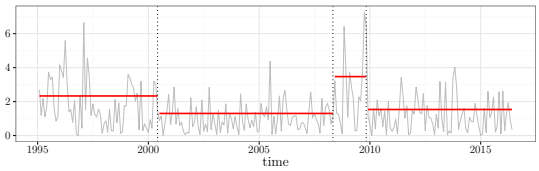
(c) Newham: $\hat{f}_t$



(d) Newham: $\hat{\sigma}_t$



(e) Tower Hamlets: $\hat{f}_t$



(f) Tower Hamlets: $\hat{\sigma}_t$

Figure 4.9: Change-point analysis for the monthly percentage changes in the UK House Price Index from January 1995 to May 2016. Figure 4.9a, 4.9c and 4.9e: the monthly percentage changes Y_t and the fitted piecewise-constant mean $\hat{f}_t$, between the change-points estimated with NOT. Figure 4.9b, 4.9d and 4.9f: $|Y_t - \hat{f}_t|$ and the fitted piecewise-constant standard deviation $\hat{\sigma}_t$, between the change-points estimated with NOT.

estimates in this case.

As in Fryzlewicz (2016), we analyse the percentage changes in the HPI for three London boroughs, namely Hackney, Newham and Tower Hamlets, all of which are located in East London. Hackney and Tower of Hamlets border on the City of London, a major business and financial district, with the latter being a home to Canary Wharf, another important financial centre. On the other hand, Newham, located to the east of Hackney and Tower Hamlets, hosted the London 2012 Olympic Games which involved large-scale investment in that borough.

Figure 4.9 shows monthly percentage changes in HPI for the analysed boroughs and the corresponding NOT estimates, obtained using the contrast function (4.12). As recommended in Section 4.3.4, we set the number of intervals drawn in the procedure to $M = 10000$ and choose the threshold that minimises the SIC criterion (4.15). For better comparability, NOT is applied with the same random seed for each data series.

In contrast to Fryzlewicz (2016), whose TGUH method estimates at least 10 change-points in each HPI series, we detect just a few change-points in the data, facilitating the interpretation of the results. Furthermore, for all three boroughs, NOT estimates two change-points (one around March 2008 and one around September 2009) that can clearly be linked to the 2008–2009 financial crisis and the concurrent collapse of the housing market. Estimated standard deviations for that period are much larger than the estimates corresponding to the other segments of piecewise-constancy, suggesting that in this example Scenario (S4) may be more relevant than (S1) considered in Fryzlewicz (2016). It is also interesting to observe that, with the exception of Tower Hamlets from January 1995 to April 2000 and the 2008–2009 financial crisis for all boroughs, the estimated standard deviations oscillate around a baseline level (different for each series).

The period of a larger volatility for Tower Hamlets in Figure 4.9f, observed from January 1995 to April 2000, can possibly be explained by developments in Canary

Wharf, which in the past was a dock complex closed in 1980. Gordon (2001) claims that the project of converting Canary Wharf into a business district "was politically controversial and widely regarded as a planning disaster" which "(in 1992) failed as a result of six factors: a recession in the London property market, competition from the City of London, poor transport links, few British tenants, complicated finances and developer overconfidence". Over the 1995–2000 period, the situation in the London property reversed, which combined with a development of new public transport lines in Canary Wharf led to the success of the project. According to Gordon (2001), "when the Jubilee underground line opened in 2000, Canary Wharf's resurrection was complete".

Finally, it is interesting to observe that over two periods, namely March 1991 to November 2002 and January 2014 to May 2016, the HPI for Newham (Figure 4.9c) was increasing at a rate higher than for the other two boroughs.

4.6 Proofs

4.6.1 Some useful lemmas

4.6.1.1 The piecewise constant case

Lemma 4.6.1. *Let* $g(x,y) = \frac{xy}{x+y}$ *and suppose that* $\min(x,y) > 0$. *Then*

$$g(x,y) \geq \frac{1}{2}\min(x,y).$$

Proof. Without loss of generality, assume that $x \geq y$. Then $g(x,y) \geq \frac{xy}{2x} \geq y/2 = \min(x,y)/2$. □

Lemma 4.6.2. *Suppose* $\mathbf{f} = (f_1, \ldots, f_T)'$ *is piecewise-constant vector as in Scenario (S1), and* $\tau_1, \ldots, \tau_q$ *are the locations of the change-points. Suppose* $1 \leq s < e \leq T$, *such that* $\tau_{j-1} < s \leq \tau_j < e \leq \tau_{j+1}$ *for some* $j = 1 \ldots, q$. *Let* $\eta = \min\{\tau_j - s + 1, e - \tau_j\}$ *and*

$\Delta_j^{\mathbf{f}} = |f_{\tau_j+1} - f_{\tau_j}|$. *Then*

$$\mathcal{C}_{s,e}^{\tau_j}(\mathbf{f}) = \max_{s \le b < e} \mathcal{C}_{s,e}^{b}(\mathbf{f}) \begin{cases} \ge \frac{1}{\sqrt{2}} \eta^{1/2} \Delta_j^{\mathbf{f}}, \\ \le \eta^{1/2} \Delta_j^{\mathbf{f}}. \end{cases}$$

Proof. For any $s \le b < e$, by simple algebra, we have

$$\mathcal{C}_{s,e}^{b}(\mathbf{f}) = \begin{cases} \sqrt{\frac{b-s+1}{l(e-b)}}(e-\tau_j)|f_{\tau_j+1} - f_{\tau_j}|, & b \le \tau_j; \\ \sqrt{\frac{(\tau_j - s+1)(e-\tau_j)}{l}}|f_{\tau_j+1} - f_{\tau_j}|, & b = \tau_j; \\ \sqrt{\frac{e-b}{l(b-s+1)}}(\tau_j - s + 1)|f_{\tau_j+1} - f_{\tau_j}|, & b \ge \tau_j, \end{cases} \tag{4.19}$$

where $l = s - e + 1$. Now $\mathcal{C}_{s,e}^{\tau_j}(\mathbf{f}) = \max_{s \le b \le e} \mathcal{C}_{s,e}^{b}(\mathbf{f})$ follows from the fact that $\mathcal{C}_{s,e}^{b}(\mathbf{f})$ is increasing (as a function of b) for $1 \le b \le \tau_j$ and decreasing for $\tau_j \le b \le e$. To prove the lower bound, we set $\eta_L = \tau_j - s + 1$ and $\eta_R = e - \tau_j$ and observe that $\eta_L \ge \eta$ and $\eta_R \ge \eta$. Therefore by Lemma 4.6.1, $\frac{\eta_L \eta_R}{\eta_L + \eta_R} \ge \frac{\eta}{2}$. Noting that $l = \eta_L + \eta_R$ we bound

$$\mathcal{C}_{s,e}^{\tau_j}(\mathbf{f}) = \sqrt{\frac{(\tau_j - s + 1)(e - \tau_j)}{l}}|f_{\tau_j+1} - f_{\tau_j}| \begin{cases} \ge (\eta/2)^{1/2} \Delta_j^{\mathbf{f}}; \\ \le \eta^{1/2} \Delta_j^{\mathbf{f}}. \end{cases}$$

which completes the proof. □

Lemma 4.6.3. *Suppose* $\mathbf{f} = (f_1, \ldots, f_T)'$ *is piecewise-constant vector as in Scenario (S1), and* $\tau_1, \ldots, \tau_q$ *are the locations of the change-points. Suppose* $1 \le s < e \le T$ *such that* $\tau_{j-1} < s \le \tau_j$ *and* $\tau_{j+1} < e \le \tau_{j+2}$ *for some* $j = 1 \ldots, q-1$. *Then*

$$\max_{s \le b < e} \mathcal{C}_{s,e}^{b}(\mathbf{f}) \le (\tau_j - s + 1)^{1/2} \Delta_j^{\mathbf{f}} + (e - \tau_{j+1})^{1/2} \Delta_{j+1}^{\mathbf{f}}$$

where $\Delta_j^{\mathbf{f}} = |f_{\tau_j+1} - f_{\tau_j}|$.

Proof. Suppose that $b^* = \operatorname{argmax}_{s\le b<e} \mathcal{C}^b_{s,e}(\mathbf{f})$. Then

$$\begin{aligned}
0 &\le \|\mathbf{f} - \langle \mathbf{f}, \boldsymbol{\psi}^{b^*}_{s,e}\rangle \boldsymbol{\psi}^{b^*}_{s,e} - \langle \mathbf{f}, \mathbf{1}_{s,e}\rangle \mathbf{1}_{s,e}\|^2 = \|\mathbf{f} - \langle \mathbf{f}, \mathbf{1}_{s,e}\rangle \mathbf{1}_{s,e}\|^2 - \langle \mathbf{f}, \boldsymbol{\psi}^{b^*}_{s,e}\rangle^2 \\
&\le \|\mathbf{f} - f_{\tau_j+1}\sqrt{s-e+1}\mathbf{1}_{s,e}\|^2 - \langle \mathbf{f}, \boldsymbol{\psi}^{b^*}_{s,e}\rangle^2 \\
&= (\tau_j - s + 1)(\Delta^{\mathbf{f}}_j)^2 + (e - \tau_{j+1})(\Delta^{\mathbf{f}}_{j+1})^2 - \left(\max_{s\le b<e} \mathcal{C}^b_{s,e}(\mathbf{f})\right)^2.
\end{aligned}$$

It then follows that

$$\begin{aligned}
\max_{s\le b<e} \mathcal{C}^b_{s,e}(\mathbf{f}) &\le \sqrt{(\tau_j - s + 1)(\Delta^{\mathbf{f}}_j)^2 + (e - \tau_{j+1})(\Delta^{\mathbf{f}}_{j+1})^2} \\
&\le (\tau_j - s + 1)^{1/2}\Delta^{\mathbf{f}}_j + (e - \tau_{j+1})^{1/2}\Delta^{\mathbf{f}}_{j+1}.
\end{aligned}$$

□

Lemma 4.6.4. *Suppose* $\mathbf{f} = (f_1, \dots, f_T)'$ *is piecewise-constant vector as in Scenario (S1). Pick any interval* $[s, e] \subset [1, T]$ *such that* $[s, e-1]$ *contains exactly one change-point* τ_j*. Let* $\rho = |\tau_j - b|$, $\Delta^{\mathbf{f}}_j = |f_{\tau_j+1} - f_{\tau_j}|$, $\eta_L = \tau_j - s + 1$ *and* $\eta_R = e - \tau_j$*. Then,*

$$\|\boldsymbol{\psi}^b_{s,e}\langle \mathbf{f}, \boldsymbol{\psi}^b_{s,e}\rangle - \boldsymbol{\psi}^{\tau_j}_{s,e}\langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e}\rangle\|_2^2 = (\mathcal{C}^{\tau_j}_{s,e}(\mathbf{f}))^2 - (\mathcal{C}^b_{s,e}(\mathbf{f}))^2.$$

Moreover,

1. *for any* $\tau_j \le b < e$, $(\mathcal{C}^{\tau_j}_{s,e}(\mathbf{f}))^2 - (\mathcal{C}^b_{s,e}(\mathbf{f}))^2 = \frac{\rho\,\eta_L}{\rho+\eta_L}(\Delta^{\mathbf{f}}_j)^2$;
2. *for any* $s \le b < \tau_j$, $(\mathcal{C}^{\tau_j}_{s,e}(\mathbf{f}))^2 - (\mathcal{C}^b_{s,e}(\mathbf{f}))^2 = \frac{\rho\,\eta_R}{\rho+\eta_R}(\Delta^{\mathbf{f}}_j)^2$.

Proof. First, we note that since there is only one change-point in $[s, e-1]$, the restriction of $\mathbf{f}$ on $[s, e]$, i.e. $\mathbf{f}|_{[s,e]} = (0, \dots, 0, f_s, \dots, f_e, 0, \dots, 0)'$ can be decomposed into

$$\mathbf{f}|_{[s,e]} = \boldsymbol{\psi}^{\tau_j}_{s,e}\langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e}\rangle + \mathbf{1}_{s,e}\langle \mathbf{f}, \mathbf{1}_{s,e}\rangle,$$

where we also used the fact that $\boldsymbol{\psi}_{s,e}^{\tau_j}$ and $\mathbf{1}_{s,e}$ are orthonormal. Note that $\boldsymbol{\psi}_{s,e}^{b}$ and $\mathbf{1}_{s,e}$ are also orthonormal, it follows that

$$\langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{b} \rangle = \langle \mathbf{f}|_{[s,e]}, \boldsymbol{\psi}_{s,e}^{b} \rangle = \left\langle \boldsymbol{\psi}_{s,e}^{\tau_j} \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{\tau_j} \rangle + \mathbf{1}_{s,e} \langle \mathbf{f}, \mathbf{1}_{s,e} \rangle, \boldsymbol{\psi}_{s,e}^{b} \right\rangle = \langle \boldsymbol{\psi}_{s,e}^{\tau_j}, \boldsymbol{\psi}_{s,e}^{b} \rangle \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{\tau_j} \rangle.$$

Therefore,

$$\langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{b} \rangle^2 = \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{b} \rangle \langle \boldsymbol{\psi}_{s,e}^{\tau_j}, \boldsymbol{\psi}_{s,e}^{b} \rangle \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{\tau_j} \rangle,$$

and thus

$$\begin{aligned} \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{\tau_j} \rangle^2 - \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{b} \rangle^2 &= \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{\tau_j} \rangle^2 + \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{b} \rangle^2 - 2 \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{b} \rangle \langle \boldsymbol{\psi}_{s,e}^{\tau_j}, \boldsymbol{\psi}_{s,e}^{b} \rangle \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{\tau_j} \rangle \\ &= \| \boldsymbol{\psi}_{s,e}^{b} \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{b} \rangle - \boldsymbol{\psi}_{s,e}^{\tau_j} \langle \mathbf{f}, \boldsymbol{\psi}_{s,e}^{\tau_j} \rangle \|_2^2. \end{aligned}$$

Here in the above final step, we used the fact that $\|\boldsymbol{\psi}_{s,e}^{\tau_j}\|_2^2 = \|\boldsymbol{\psi}_{s,e}^{b}\|_2^2 = 1$.

Second, for the sake of brevity, we only prove the case of $b \geq \tau_j$. Let $l = e - s + 1$, $x = b - s + 1$, and thus $\rho = x - \eta_L$. Using (4.19), we get

$$\begin{aligned} (\mathcal{C}_{s,e}^{\tau_j}(\mathbf{f}))^2 - (\mathcal{C}_{s,e}^{b}(\mathbf{f}))^2 &= \left(\frac{\eta_L(l - \eta_L)}{l} - \frac{\eta_L^2(l - x)}{lx} \right) |f_{\tau_j+1} - f_{\tau_j}|^2 \\ &= \frac{\eta_L(x - \eta_L)}{x} (\Delta_j^{\mathbf{f}})^2 = \left(\frac{\rho \eta_L}{\eta_L + \rho} \right) (\Delta_j^{\mathbf{f}})^2. \end{aligned}$$

□

4.6.1.2 The piecewise linear continuous case

Lemma 4.6.5. *Suppose* $\mathbf{f} = (f_1, \ldots, f_T)'$ *is piecewise-linear vector as in Scenario (S2), and* $\tau_1, \ldots, \tau_q$ *are the locations of the change-points. Suppose* $1 \leq s < e \leq T$, *such that* $\tau_{j-1} \leq s < \tau_j < e \leq \tau_{j+1}$ *for some* $j = 1 \ldots, q$. *Let* $\eta = \min\{\tau_j - s, e - \tau_j\}$ *and*

$\Delta_j^{\mathbf{f}} = |2f_{\tau_j} - f_{\tau_j-1} - f_{\tau_j+1}|$. *Then*

$$\mathcal{C}_{s,e}^{\tau_j}(\mathbf{f}) = \max_{s<b<e} \mathcal{C}_{s,e}^{b}(\mathbf{f}) \begin{cases} \geq \frac{1}{\sqrt{24}}\eta^{3/2}\Delta_j^{\mathbf{f}}, \\ \leq \frac{1}{\sqrt{3}}(\eta+1)^{3/2}\Delta_j^{\mathbf{f}}. \end{cases}$$

Proof. First, we show that $\mathcal{C}_{s,e}^{b}(\mathbf{f})$ is maximised at $b = \tau_j$. Using the notation from the proof of Lemma 4.6.4, we have that

$$\mathbf{f}|_{[s,e]} = \boldsymbol{\phi}_{s,e}^{\tau_j}\langle \mathbf{f}, \boldsymbol{\phi}_{s,e}^{\tau_j}\rangle + \boldsymbol{\gamma}_{s,e}\langle \mathbf{f}, \mathbf{1}_{s,e}\rangle + \mathbf{1}_{s,e}\langle \mathbf{f}, \mathbf{1}_{s,e}\rangle.$$

Therefore, it follows that

$$\|\mathbf{f}|_{[s,e]}\|_2^2 = \langle \mathbf{f}, \boldsymbol{\phi}_{s,e}^{\tau_j}\rangle^2 + \langle \mathbf{f}, \boldsymbol{\gamma}_{s,e}\rangle^2 + \langle \mathbf{f}, \mathbf{1}_{s,e}\rangle^2. \tag{4.20}$$

For any $b \in \{s+1, \ldots, \tau_j - 1, \tau_j + 1, \ldots, e-1\}$, it is clear that $\mathbf{f}|_{[s,e]}$ does not lie in the span of $\boldsymbol{\phi}_{s,e}^{b}$, $\boldsymbol{\gamma}_{s,e}$ and $\mathbf{1}_{s,e}$. Consequently, by projecting $\mathbf{f}|_{[s,e]}$ onto these three bases, we have that

$$\|\mathbf{f}|_{[s,e]}\|^2 > \langle \mathbf{f}, \boldsymbol{\phi}_{s,e}^{b}\rangle^2 + \langle \mathbf{f}, \boldsymbol{\gamma}_{s,e}\rangle^2 + \langle \mathbf{f}, \mathbf{1}_{s,e}\rangle^2. \tag{4.21}$$

Comparing (4.21) with (4.20) entails that $\left|\langle \mathbf{f}, \boldsymbol{\phi}_{s,e}^{\tau_j}\rangle\right| > \left|\langle \mathbf{f}, \boldsymbol{\phi}_{s,e}^{b}\rangle\right|$ for any $b \neq \tau_j$.

Secondly, set $\eta_L = \tau_j - s$ and $\eta_R = e - \tau_j$. After some calculation, we get that

$$\mathcal{C}_{s,e}^{\tau_j}(\mathbf{f}) = \left\{ \frac{\eta_L(\eta_L+1)\eta_R(\eta_R+1)(2\eta_L\eta_R + \eta_L + \eta_R + 2)}{6l(l^2-1)} \right\} \Delta_j^{\mathbf{f}},$$

where $l = e - s + 1$. Also, we have $\eta_L \geq \eta$, $\eta_R \geq \eta$ and $l = \eta_L + \eta_R + 1$. To prove the

lower bound, we observe that

$$\left\{\frac{\eta_L(\eta_L+1)\eta_R(\eta_R+1)(2\eta_L\eta_R+\eta_L+\eta_R+2)}{6l(l^2-1)}\right\}$$
$$\geq \left\{\frac{1}{6}\frac{(\eta_L+1)\eta_R}{l}\frac{\eta_L(\eta_R+1)}{l}\frac{2\min(\eta_L,\eta_R)\{\max(\eta_L,\eta_R)+1\}}{l}\right\} \geq \left\{\frac{\eta^3}{24}\right\},$$

where the last inequality is obtained applying Lemma 4.6.1 three times. For the upper bound, we notice that $2\eta_L\eta_R+\eta_L+\eta_R+2 \leq 2(\eta_L+1)(\eta_R+1)$ which implies

$$\left\{\frac{\eta_L(\eta_L+1)\eta_R(\eta_R+1)(2\eta_L\eta_R+\eta_L+\eta_R+2)}{6l(l^2-1)}\right\} \leq \left\{\frac{1}{3}\frac{\eta_L\eta_R(\eta_L+1)^2(\eta_R+1)^2}{(l-1)l^2}\right\}$$
$$\leq \left\{\frac{(\eta+1)^3}{3}\right\}.$$

□

Lemma 4.6.6. *Suppose* $\mathbf{f}=(f_1,\ldots,f_T)'$ *is piecewise-linear vector as in Scenario (S2), and* $\tau_1,\ldots,\tau_q$ *are the locations of the change-points. Suppose* $1\leq s<e\leq T$ *such that* $\tau_{j-1}\leq s\leq\tau_j$ *and* $\tau_{j+1}\leq e\leq\tau_{j+2}$ *for some* $j=1\ldots,q-1$. *Then*

$$\max_{s\leq b<e}\mathcal{C}^b_{s,e}(\mathbf{f}) \leq \frac{1}{\sqrt{3}}(\tau_j-s+1)^{3/2}\Delta^{\mathbf{f}}_j + \frac{1}{\sqrt{3}}(e-\tau_{j+1}+1)^{3/2}\Delta^{\mathbf{f}}_{j+1},$$

where $\Delta^{\mathbf{f}}_j=|2f_{\tau_j}-f_{\tau_j-1}-f_{\tau_j+1}|$.

Proof. Suppose that $b^*=\operatorname{argmax}_{s\leq b\leq e}\mathcal{C}^b_{s,e}(\mathbf{f})$. Then

$$\begin{aligned}
0 &\leq \|\mathbf{f}-\langle\mathbf{f},\boldsymbol{\phi}^{b^*}_{s,e}\rangle\boldsymbol{\phi}^{b^*}_{s,e}-\langle\mathbf{f},\boldsymbol{\gamma}_{s,e}\rangle\boldsymbol{\gamma}_{s,e}-\langle\mathbf{f},\mathbf{1}_{s,e}\rangle\mathbf{1}_{s,e}\|^2\\
&= \|\mathbf{f}-\langle\mathbf{f},\boldsymbol{\gamma}_{s,e}\rangle\boldsymbol{\gamma}_{s,e}-\langle\mathbf{f},\mathbf{1}_{s,e}\rangle\mathbf{1}_{s,e}\|^2-\langle\mathbf{f},\boldsymbol{\phi}^{b^*}_{s,e}\rangle^2\\
&= \frac{1}{6}(\tau_j-s)(\tau_j-s+1)(2\tau_j-2s+1)(\Delta^{\mathbf{f}}_j)^2\\
&\quad+\frac{1}{6}(e-\tau_{j+1})(e-\tau_{j+1}+1)(2e-2\tau_{j+1}+1)(\Delta^{\mathbf{f}}_{j+1})^2-\left(\max_{s\leq b<e}\mathcal{C}^b_{s,e}(\mathbf{f})\right)^2.
\end{aligned}$$

It then follows that

$$\max_{s\le b<e} \mathcal{C}^b_{s,e}(\mathbf{f}) \le \left\{(\tau_j - s + 1)^3(\Delta^{\mathbf{f}}_j)^2/3 + (e - \tau_{j+1} + 1)^3(\Delta^{\mathbf{f}}_{j+1})^2/3\right\}$$
$$\le \frac{1}{\sqrt{3}}(\tau_j - s + 1)^{3/2}\Delta^{\mathbf{f}}_j + \frac{1}{\sqrt{3}}(e - \tau_{j+1} + 1)^{3/2}\Delta^{\mathbf{f}}_{j+1}.$$

□

Lemma 4.6.7. *Suppose* $\mathbf{f} = (f_1, \ldots, f_T)'$ *is piecewise-linear vector as in Scenario (S2), and* $\tau_1, \ldots, \tau_q$ *are the locations of the change-points. Suppose* $1 \le s < e \le T$*, such that* $\tau_{j-1} \le s < \tau_j < e \le \tau_{j+1}$ *for some* $j = 1 \ldots, q$*. Let* $\rho = |\tau_j - b|$*,* $\eta_L = \tau_j - s$*,* $\eta_R = e - \tau_j$ *and* $\Delta^{\mathbf{f}}_j = |2f_{\tau_j} - f_{\tau_j - 1} - f_{\tau_j+1}|$*. Then,*

$$\|\boldsymbol{\phi}^b_{s,e}\langle \mathbf{f}, \boldsymbol{\phi}^b_{s,e}\rangle - \boldsymbol{\phi}^{\tau_j}_{s,e}\langle \mathbf{f}, \boldsymbol{\phi}^{\tau_j}_{s,e}\rangle\|_2^2 = (\mathcal{C}^{\tau_j}_{s,e}(\mathbf{f}))^2 - (\mathcal{C}^b_{s,e}(\mathbf{f}))^2. \tag{4.22}$$

Moreover,

1. *for any* $\tau_j \le b < e$*,* $(\mathcal{C}^{\tau_j}_{s,e}(\mathbf{f}))^2 - (\mathcal{C}^b_{s,e}(\mathbf{f}))^2 \ge \frac{1}{63}\min(\rho, \eta_L)^3(\Delta^{\mathbf{f}}_j)^2$*;*
2. *for any* $s < b \le \tau_j$*,* $(\mathcal{C}^{\tau_j}_{s,e}(\mathbf{f}))^2 - (\mathcal{C}^b_{s,e}(\mathbf{f}))^2 \ge \frac{1}{63}\min(\rho, \eta_R)^3(\Delta^{\mathbf{f}}_j)^2$*.*

Proof. The proof of (4.22) is very similar to that shown in Lemma 4.6.4, so is omitted for brevity. In the following, we only deal with the case of $\tau_j \le b < e$. Note that

$$\|\boldsymbol{\phi}^b_{s,e}\langle \mathbf{f}, \boldsymbol{\phi}^b_{s,e}\rangle - \boldsymbol{\phi}^{\tau_j}_{s,e}\langle \mathbf{f}, \boldsymbol{\phi}^{\tau_j}_{s,e}\rangle\|_2^2 = \left\|\boldsymbol{\phi}^b_{s,e}\langle \mathbf{f}, \boldsymbol{\phi}^b_{s,e}\rangle + \boldsymbol{\gamma}_{s,e}\langle \mathbf{f}, \boldsymbol{\gamma}_{s,e}\rangle + \mathbf{1}_{s,e}\langle \mathbf{f}, \mathbf{1}_{s,e}\rangle - \mathbf{f}|_{[s,e]}\right\|_2^2$$
$$\ge \min_{a_0,a_1\in\mathbb{R}} \left\|\mathbf{f}|_{[s,b]} - a_0\mathbf{1}_{s,b} - a_1\boldsymbol{\gamma}_{s,b}\right\|_2^2 + \min_{a_0,a_1\in\mathbb{R}} \left\|\mathbf{f}|_{[b+1,e]} - a_0\mathbf{1}_{b+1,e} - a_1\boldsymbol{\gamma}_{b+1,e}\right\|_2^2$$
$$\ge \min_{a_0,a_1\in\mathbb{R}} \left\|\mathbf{f}|_{[s,b]} - a_0\mathbf{1}_{s,b} - a_1\boldsymbol{\gamma}_{s,b}\right\|_2^2.$$

Recalling the definitions of $\alpha^{\tau_j}_{s,b}$ and $\beta^{\tau_j}_{s,b}$ in (4.8), and writing $d = b - s + 1$. After some calculations (similar to what has already been carried out in deriving $\boldsymbol{\phi}^b_{s,e}$), we obtain

that

$$\min_{a_0,a_1\in\mathbb{R}} \|\mathbf{f}|_{[s,b]} - a_0\mathbf{1}_{s,b} - a_1\boldsymbol{\gamma}_{s,b}\|_2^2 = \left[(3\eta_L+\rho+2)\alpha_{s,b}^{\tau_j}\beta_{s,b}^{\tau_j} + (3\rho+\eta_L+2)\alpha_{s,b}^{\tau_j}(\beta_{s,b}^{\tau_j})^{-1}\right]^{-2}(\Delta_j^{\mathbf{f}})^2$$
$$= \frac{1}{6}(\Delta_j^{\mathbf{f}})^2 d(d^2-1)[1+\rho\eta_L+(\rho+1)(\eta_L+1)]\times$$
$$\left[(d+2\eta_L+1)^2\frac{\rho(\rho+1)}{\eta_L(\eta_L+1)} + (d+2\rho+1)^2\frac{\eta_L(\eta_L+1)}{\rho(\rho+1)} + 2(d+2\eta_L+1)(d+2\rho+1)\right]^{-1}.$$

Notice that the above equation is symmetric with respect to η_L and ρ. Without loss of generality, here we proceed by assuming that $\eta_L \geq \rho$. Since $(d+2\eta_L+1)+(d+2\rho+1) = 4d$, it follows that $(d+2\eta_L+1)(d+2\rho+1) \leq 4d^2$. Therefore,

$$\begin{aligned}
&\min_{a_0,a_1\in\mathbb{R}} \left\|\mathbf{f}|_{[s,b]} - a_0\mathbf{1}_{s,b} - a_1\boldsymbol{\gamma}_{s,b}\right\|_2^2 \\
&\geq \frac{1}{6}(\Delta_j^{\mathbf{f}})^2 d(d^2-1)[2(\eta_L+1)\rho]\left[(3d)^2 + (2d)^2\frac{(\eta_L+1)^2}{\rho^2} + 8d^2\right]^{-1} \\
&\geq \frac{1}{6}(\Delta_j^{\mathbf{f}})^2 d^2(d-1)[2(\eta_L+1)\rho]\left[21d^2\frac{(\eta_L+1)^2}{\rho^2}\right]^{-1} \geq \frac{1}{63}\rho^3(\Delta_j^{\mathbf{f}})^2,
\end{aligned}$$

where in the last step, we used the fact that $\frac{d-1}{\eta_L+1} \geq 1$ for $\rho \geq 1$ (and note that the last above-displayed equation also holds if $\rho = 0$).

Finally, we remark that the case of $s < b \leq \tau_j$ can also be handled by symmetry. □

4.6.2 Proof of Theorem 4.2.1

Here we informally discuss our proof strategy, which could be generalised to other scenarios. Intuitively speaking, lemmas from Section 4.6.1 deal with noiseless versions of the change-point estimation problems. In order to apply these results to show the consistency of estimated number of change-points, we need to control $\|\mathcal{C}_{s,e}^b(\mathbf{Y}) - \mathcal{C}_{s,e}^b(\mathbf{f})\|$ for every (s,e,b), which can be achieved using Bonferroni in Step 1. Note that for any fixed interval with start-point s and end-point e, to decide whether b_1 or b_2 is a more suitable change-point candidate inside this interval, we only need to look at the

value of $\mathcal{C}_{s,e}^{b_1}(\mathbf{Y}) - \mathcal{C}_{s,e}^{b_2}(\mathbf{Y})$. Therefore, when establishing the convergence rate of the estimated change-point location , we control the distance between $\mathcal{C}_{s,e}^{b_1}(\mathbf{Y}) - \mathcal{C}_{s,e}^{b_2}(\mathbf{Y})$ and its noiseless analogue $\mathcal{C}_{s,e}^{b_1}(\mathbf{f}) - \mathcal{C}_{s,e}^{b_2}(\mathbf{f})$ (after proper normalisation) for all tuples (s, e, b_1, b_2) in Step 2. In Step 3, we show that given a properly chosen threshold and a large enough M, both bounds in Step 1 and Step 2 hold, and for each change-point τ_j, there exists an interval from F_T^M that contains only this change-point and both its start- and end- points are sufficiently far away from other change-points. Since we are dealing with the narrowest-over-threshold intervals, the actual intervals that our NOT algorithm pick must have length no longer than the ones we considered in Step 3, thus could only contain precisely one change-point. So in Step 4, it suffices to investigate a single change-point detection problem, where we can use lemmas from Section 4.6.1 and the bound in Step 2 to establish the convergence rate for its location estimation. Finally, in Step 5, we show that after detecting all the change-points, the NOT algorithm stops with no further detection. This is because the remaining elements $[s, e] \in F_T^M$ to be considered either have no change-point inside, or have one/two change-points that are very close to its start- or/and end- points, thus their corresponding $\max_b \mathcal{C}_{s,e}^b(\mathbf{Y})$ cannot exceed the given threshold in views of the property of its noiseless analogue and the bound from Step 1.

Now we proceed to the technical details.

Proof. We shall prove the following more specific result, which in turn implies (4.13).

$$\mathbb{P}\left(\hat{q} = q,\ \max_{j=1,\ldots,q}\left(|\hat{\tau}_j - \tau_j|(\Delta_j^{\mathbf{f}})^2\right) \leq C_3 \log T\right) \geq 1 - T^{-1}/(6\sqrt{\pi}) \tag{4.23}$$

$$- T\delta_T^{-1}(1 - \delta_T^2 T^{-2}/36)^M, \tag{4.24}$$

Step One.

Let $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_T)'$ and $\lambda_T = \sqrt{8 \log T}$. Define the set

$$A_T = \left\{ \max_{s,b,e:1 \leq s \leq b < e \leq T} |\mathcal{C}^b_{s,e}(\boldsymbol{\varepsilon})| \leq \lambda_T \right\}.$$

Note that for any $1 \leq s \leq b < e \leq T$, $\mathcal{C}^b_{s,e}(\boldsymbol{\varepsilon})$ follows a standard normal distribution. Therefore, using the Bonferroni bound, we get

$$\mathbb{P}(A^c_T) \leq \frac{T^3}{6} \frac{2e^{-(\sqrt{8 \log T})^2/2}}{\sqrt{8 \log T}\sqrt{2\pi}} \leq \frac{T^{-1}}{12\sqrt{\pi}}.$$

Moreover, because $\mathcal{C}^b_{s,e}(\mathbf{Y}) - \mathcal{C}^b_{s,e}(\mathbf{f}) = \mathcal{C}^b_{s,e}(\boldsymbol{\varepsilon})$, so A_T also implies that

$$\left\{ \max_{s,b,e:1 \leq s \leq b < e \leq T} |\mathcal{C}^b_{s,e}(\mathbf{Y}) - \mathcal{C}^b_{s,e}(\mathbf{f})| \leq \lambda_T \right\}.$$

Step Two.

Define the set

$$B_T = \left\{ \max_{j=1,\ldots,q} \max_{\substack{\tau_{j-1} < s \leq \tau_j \\ \tau_j < e \leq \tau_{j+1} \\ s \leq b < e}} \frac{\left| \left\langle \boldsymbol{\psi}^b_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^b_{s,e} \rangle - \boldsymbol{\psi}^{\tau_j}_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e} \rangle, \boldsymbol{\varepsilon} \right\rangle \right|}{\| \boldsymbol{\psi}^b_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^b_{s,e} \rangle - \boldsymbol{\psi}^{\tau_j}_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e} \rangle \|_2} \leq \lambda_T \right\}.$$

Again, for any $1 \leq s \leq b < e \leq T$, $\frac{|\langle \boldsymbol{\psi}^b_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^b_{s,e} \rangle - \boldsymbol{\psi}^{\tau_j}_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e} \rangle, \boldsymbol{\varepsilon} \rangle|}{\| \boldsymbol{\psi}^b_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^b_{s,e} \rangle - \boldsymbol{\psi}^{\tau_j}_{s,e} \langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e} \rangle \|_2}$ follows a standard normal distribution, so using a similar argument, we get

$$\mathbb{P}(B^c_T) \leq \frac{T^{-1}}{12\sqrt{\pi}}.$$

Step Three.

To fix the ideas, for $j = 1, \dots, q$, we define intervals

$$\mathcal{I}_j^L = (\tau_j - \delta_T/3, \tau_j - \delta_T/6) \tag{4.25}$$

$$\mathcal{I}_j^R = (\tau_j + \delta_T/6, \tau_j + \delta_T/3) \tag{4.26}$$

Note that these intervals all contain at least one integer as long as $\delta_T > 6$. This is always true for sufficiently large T, as it follows from Conditions 1 and 2 that $\delta_T > \underline{C} \log T / \underline{f}$. Recall that F_T^M is the set of M randomly drawn intervals with endpoints in $\{1, \dots, T\}$. Denote by $[s_1, e_1], \dots, [s_M, e_M]$ the elements of F_T^M and let

$$D_T^M = \Big\{ \forall j = 1, \dots, q, \exists k \in \{1, \dots, M\}, \text{ s.t. } s_k \times e_k \in \mathcal{I}_j^L \times \mathcal{I}_j^R \Big\}. \tag{4.27}$$

We have that

$$\begin{aligned} \mathbb{P}\left((D_T^M)^c\right) &\leq \sum_{j=1}^{q} \Pi_{m=1}^{M} \Big(1 - \mathbb{P}\left(s_m \times e_m \in \mathcal{I}_j^L \times \mathcal{I}_j^R \right) \Big) \\ &\leq q \left(1 - \frac{\delta_T^2}{6^2 T^2} \right)^M \leq \frac{T}{\delta_T} \left(1 - \frac{\delta_T^2}{36 T^2} \right)^M. \end{aligned}$$

Therefore, $\mathbb{P}\left(A_T \cap B_T \cap D_T^M \right) \geq 1 - T^{-1}/(6\sqrt{\pi}) - T\delta_T^{-1}(1 - \delta_T^2 T^{-2}/36)^M$.

In the rest of the proof, we assume that A_T, B_T and D_T^M all hold. We give the constants as follows:

$$C_1 = 2\sqrt{C_3} + \sqrt{8}, \quad C_2 = \frac{1}{\sqrt{6}} - \frac{2\sqrt{2}}{\underline{C}}, \quad C_3 = 32\sqrt{2} + 48.$$

These constants could be further refined by applying the Bonferroni bound more carefully. But since our main aim is to establish the rate, we chose not to pursue this direction further. In addition, here we need to make sure that $\underline{C} C_2 > C_1$, and thus $C_2 \delta_T^{1/2} \underline{f}_T > C_1 \sqrt{\log T}$,

i.e. we can select $\zeta_T \in [C_1\sqrt{\log T}, C_2\delta_T^{1/2}\underline{f}_T)$. This is indeed the case because $\underline{C}$ is sufficiently large.

Step Four.

We focus on a generic interval $[s, e]$ such that

$$\exists j \in \{1, \ldots, q\},\ \exists k \in \{1, \ldots, M\},\ \text{s.t. } [s_k, e_k] \subset [s, e] \text{ and } s_k \times e_k \in \mathcal{I}_j^L \times \mathcal{I}_j^R \quad (4.28)$$

Fix such an interval $[s, e]$ and let $j \in \{1, \ldots, q\}$ and $k \in \{1, \ldots, M\}$ be such that (4.28) is satisfied. Let $b_k^* = \text{argmax}_{s_k \le b \le e_k} \mathcal{C}_{s_k,e_k}^b(\mathbf{Y})$. By construction, $[s_k, e_k]$ satisfies $\tau_j - s_k + 1 \ge \delta_T/6$ and $e_k - \tau_j > \delta_T/6$. Denote by

$$\mathcal{M}_{s,e} = \left\{m : [s_m, e_m] \in F_T^M, [s_m, e_m] \subset [s, e]\right\};$$
$$\mathcal{O}_{s,e} = \{m \in \mathcal{M}_{s,e} : \max_{s_m \le b < e_m} \mathcal{C}_{s_m,e_m}^b(\mathbf{Y}) > \zeta_T\}$$

Our first aim is to show that $\mathcal{O}_{s,e}$ is non-empty. This follows from Lemma 4.6.2 and the calculation below.

$$\begin{aligned}
\mathcal{C}_{s_k,e_k}^{b_k^*}(\mathbf{Y}) &\ge \mathcal{C}_{s_k,e_k}^{\tau_j}(\mathbf{Y}) \\
&\ge \mathcal{C}_{s_k,e_k}^{b_k^*}(\mathbf{f}) - \lambda_T \ge \left(\frac{\delta_T}{6}\right)^{1/2} |f_{\tau_j+1} - f_{\tau_j}| - \lambda_T \ge \left(\frac{\delta_T}{6}\right)^{1/2} \underline{f}_T - \lambda_T \\
&= \left(\frac{1}{\sqrt{6}} - \frac{\lambda_T}{\delta_T^{1/2}\underline{f}_T}\right) \delta_T^{1/2}\underline{f}_T \ge \left(\frac{1}{\sqrt{6}} - \frac{2\sqrt{2}}{\underline{C}}\right) \delta_T^{1/2}\underline{f}_T = C_2\delta_T^{1/2}\underline{f}_T > \zeta_T.
\end{aligned}$$

Let $m^* = \text{argmin}_{m \in \mathcal{O}_{s,e}}(e_m - s_m + 1)$ and $b^* = \text{argmax}_{s_{m^*} \le b < e_{m^*}} \mathcal{C}_{s_{m^*},e_{m^*}}^b(\mathbf{Y})$. Observe that $[s_{m^*}, e_{m^*})$ must contain at least one change-point. Indeed, if that was not the case,

we would have $\mathcal{C}^{b}_{s_{m^*},e_{m^*}}(\mathbf{f}) = 0$ and

$$\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{Y}) = |\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{Y}) - \mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{f})| \leq \lambda_T \leq \zeta_T$$

which contradicts $\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{Y}) > \zeta_T$. On the other hand, $[s_{m^*}, e_{m^*})$ cannot contain more than one change-points, because $e_{m^*} - s_{m^*} + 1 \leq e_k - s_k + 1 \leq \delta_T$, as we picked the *narrowest*-over-threshold interval.

Without loss of generality, assume $\tau_j \in [s_{m^*}, e_{m^*}]$. Denote by $\eta_L = \tau_j - s_{m^*} + 1$ $\eta_R = e_{m^*} - \tau_j$ and $\eta_T = (C_1 - \sqrt{8})^2(\Delta^{\mathbf{f}}_j)^{-2}\log T$, where $\Delta^{\mathbf{f}}_j = |f_{\tau_j+1} - f_{\tau_j}|$. We claim that $\min(\eta_L, \eta_R) > \eta_T$, because $\min(\eta_L, \eta_R) \leq \eta_T$ and Lemma 4.6.2 result in

$$\begin{aligned}\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{Y}) &\leq \mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{f}) + \lambda_T \leq \mathcal{C}^{\tau_j}_{s_{m^*},e_{m^*}}(\mathbf{f}) + \lambda_T \leq \eta_T^{1/2}\Delta^{\mathbf{f}}_j + \lambda_T \\ &= (C_1 - \sqrt{8} + \sqrt{8})\sqrt{\log T} = C_1\sqrt{\log T} \leq \zeta_T,\end{aligned}$$

which contradicts $\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{Y}) > \zeta_T$.

We are now in the position to prove $|b^* - \tau_j| \leq C_3 \log T/(\Delta^{\mathbf{f}}_j)^2$. The arguments we use here are simpler and slightly more general than Lemma A.3 of Fryzlewicz (2014). Our aim is to find ϵ_T such that for any $b \in \{s_{m^*}, s_{m^*}+1, \ldots, e_{m^*}-1\}$ with $|b - \tau_j| > \epsilon_T$, we always have

$$(\mathcal{C}^{\tau_j}_{s_{m^*},e_{m^*}}(\mathbf{Y}))^2 - (\mathcal{C}^{b}_{s_{m^*},e_{m^*}}(\mathbf{Y}))^2 > 0. \tag{4.29}$$

This would then imply that $|b^* - \tau_j| \leq \varepsilon_T$. By expansion and rearranging the terms

(using the fact that $f_t = Y_t + \varepsilon_t$), we see that (4.29) is equivalent to

$$\langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \rangle^2 - \langle \mathbf{f}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \rangle^2 > \langle \boldsymbol{\varepsilon}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \rangle^2 - \langle \boldsymbol{\varepsilon}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \rangle^2 + 2\Big\langle \boldsymbol{\varepsilon}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \langle \mathbf{f}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \rangle - \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \rangle \Big\rangle. \tag{4.30}$$

In the following, we assume that $b \geq \tau_j$. The case that $b < \tau_j$ can be handled in a similar fashion. By Lemma 4.6.4, we have

$$\langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \rangle^2 - \langle \mathbf{f}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \rangle^2 = (\mathcal{C}^{\tau_j}_{s^*,e^*}(\mathbf{f}))^2 - (\mathcal{C}^{b}_{s_{m^*},e_{m^*}}(\mathbf{f}))^2 = \frac{|b-\tau_j|\eta_L}{|b-\tau_j|+\eta_L}(\Delta^{\mathbf{f}}_j)^2 := \kappa.$$

In addition, since A_T and B_T hold, we have that

$$\begin{aligned}
&\langle \boldsymbol{\varepsilon}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \rangle^2 - \langle \boldsymbol{\varepsilon}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \rangle^2 \leq \lambda_T^2, \\
&2\Big\langle \boldsymbol{\varepsilon}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \langle \mathbf{f}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \rangle - \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \rangle \Big\rangle \\
&\qquad \leq 2\|\boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \langle \mathbf{f}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}} \rangle - \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}} \rangle\|_2 \lambda_T = 2\kappa^{1/2}\lambda_T,
\end{aligned}$$

where the last equality also comes from Lemma 4.6.4. Consequently, (4.30) can be deducted from the stronger inequality $\kappa - 2\lambda_T\kappa^{1/2} - \lambda_T^2 > 0$. This quadratic inequality is implied by $\kappa > (\sqrt{2}+1)^2\lambda_T^2$, and could be restricted further to

$$\frac{2|b-\tau_j|\eta_L}{|b-\tau_j|+\eta_L} \geq \min(|b-\tau_j|, \eta_L) > (32\sqrt{2}+48)(\Delta^{\mathbf{f}}_j)^{-2}\log T = C_3(\Delta^{\mathbf{f}}_j)^{-2}\log T. \tag{4.31}$$

But since

$$\eta_L \geq \eta_T = (C_1 - \sqrt{8})^2(\Delta^{\mathbf{f}}_j)^{-2}\log T = (2\sqrt{C_3})^2(\Delta^{\mathbf{f}}_j)^{-2}\log T > C_3(\Delta^{\mathbf{f}}_j)^{-2}\log T,$$

we see that (4.31) is equivalent to $|b-\tau_j| > C_3(\Delta^{\mathbf{f}}_j)^{-2}\log T$. To sum up, $|b^*-\tau_j|(\Delta^{\mathbf{f}}_j)^2 >$

$C_3 \log T$ would result in (4.29), a contradiction. So we have proved that $|b^* - \tau_j|(\Delta_j^{\mathbf{f}})^2 \leq C_3 \log T$.

Step Five.

Using the arguments given above which are valid on the event $A_T \cap B_T \cap D_T^M$, we can now proceed with the proof of the theorem as follows. At the start of Algorithm 4.6 we have $s = 1$ and $e = T$ and, provided that $q \geq 1$, condition (4.28) is satisfied. Therefore the algorithm detects a change-point b^* in that interval such that $|b^* - \tau_j| \leq C_3 \log T(\Delta_j^{\mathbf{f}})^{-2}$. By construction, we also have that $|b^* - \tau_j| < 2/3\delta_T$. This in turn implies that for all $l = 1, \ldots, q$ such that $\tau_l \in [s, e]$ and $l \neq j$ we have either $\mathcal{I}_l^L, \mathcal{I}_l^R \subset [s, b^*]$ or $\mathcal{I}_l^L, \mathcal{I}_l^R \subset [b^* + 1, e]$. Therefore (4.28) is satisfied within each segment containing at least one change-point. Note that before all q change-points are detected, each change-point will not be detected twice. To see this, we suppose that τ_j has already been detected by b, then for all intervals $[s_k, e_k] \subset [\tau_j - C_3 \log T(\Delta_j^{\mathbf{f}})^{-2} + 1, \tau_j - C_3 \log T(\Delta_j^{\mathbf{f}})^{-2} + 2/3\delta_T + 1] \cup [\tau_j + C_3 \log T(\Delta_j^{\mathbf{f}})^{-2} - 2/3\delta_T, \tau_j + C_3 \log T(\Delta_j^{\mathbf{f}})^{-2}]$, Lemma 4.6.2, together with the event A_T, guarantees that

$$\begin{aligned} s_k \leq b < e_k \mathcal{C}_{s_k,e_k}^b(\mathbf{Y}) &\leq \max_{s \leq b < e} \mathcal{C}_{s_k,e_k}^b(\mathbf{f}) + \sqrt{8 \log T} \leq \sqrt{C_3 \log T(\Delta_j^{\mathbf{f}})^{-2}} \Delta_j^{\mathbf{f}} + \sqrt{8 \log T} \\ &\leq C_1 \sqrt{\log T} \leq \zeta_T. \end{aligned}$$

Once all the change-points are detected, we then only need to consider $[s_k, e_k]$ such that

$$[s_k, e_k] \subset [\tau_j - C_3 \log T(\Delta_j^{\mathbf{f}})^{-2} + 1, \tau_{j+1} + C_3 \log T(\Delta_{j+1}^{f})^{-2}]$$

for $j = 0, \ldots, q$, where we set $\Delta_0^f = \Delta_{q+1}^f = \infty$ for notational convenience. It follows from Lemma 4.6.3 (within A_T) that

$$\begin{aligned}\max_{s_k \le b < e} \mathcal{C}^b_{s_k,e_k}(\mathbf{Y}) &\le \max_{s \le b < e} \mathcal{C}^b_{s_k,e_k}(\mathbf{f}) + \sqrt{8\log T} \\ &\le \sqrt{C_3 \log T (\Delta_j^{\mathbf{f}})^{-2}}\Delta_j^{\mathbf{f}} + \sqrt{C_3 \log T (\Delta_{j+1}^f)^{-2}}\Delta_{j+1}^{\mathbf{f}} + \sqrt{8\log T} \\ &< (2\sqrt{C_3} + \sqrt{8})\sqrt{\log T} = C_1\sqrt{\log T} \le \zeta_T.\end{aligned}$$

Hence the algorithm terminates and no further change-points are detected. □

4.6.3 Proof of Theorem 4.2.2

Proof. The proof proceeds in analogy to the proof of Theorem 4.2.1. In five steps we shall establish the following result,

$$\mathbb{P}\left(\hat{q} = q,\ \max_{j=1,\ldots,q}\left(|\hat{\tau}_j - \tau_j|(\Delta_j^{\mathbf{f}})^{2/3}\right) \le C_3(\log T)^{1/3}\right) \ge 1 - T^{-1}/(6\sqrt{\pi}) \tag{4.32}$$

$$- T\delta_T^{-1}(1 - \delta_T^2 T^{-2}/36)^M, \tag{4.33}$$

which in turn implies (4.14).

Step One and Step Two

We define the following two events

$$A_T = \left\{\max_{s,b,e:1\le s\le b<e\le T} |\mathcal{C}^b_{s,e}(\boldsymbol{\varepsilon})| \le \lambda_T\right\},$$

$$B_T = \left\{\max_{j=1,\ldots,q}\ \max_{\substack{\tau_{j-1}<s\le\tau_j \\ \tau_j<e\le\tau_{j+1} \\ s\le b<e}} \frac{\left|\left\langle \boldsymbol{\psi}^b_{s,e}\langle \mathbf{f}, \boldsymbol{\psi}^b_{s,e}\rangle - \boldsymbol{\psi}^{\tau_j}_{s,e}\langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e}\rangle, \boldsymbol{\varepsilon}\right\rangle\right|}{\|\boldsymbol{\psi}^b_{s,e}\langle \mathbf{f}, \boldsymbol{\psi}^b_{s,e}\rangle - \boldsymbol{\psi}^{\tau_j}_{s,e}\langle \mathbf{f}, \boldsymbol{\psi}^{\tau_j}_{s,e}\rangle\|_2} \le \lambda_T\right\},$$

where $\lambda_T = \sqrt{8 \log T}$. Arguments as those used in Step One and Step Two of the proof of Theorem 4.13 show that $\mathbb{P}(A_T^c) \leq \frac{T^{-1}}{12\sqrt{\pi}}$ and $\mathbb{P}(B_T^c) \leq \frac{T^{-1}}{12\sqrt{\pi}}$.

Step Three

In the rest of the proof, we assume that A_T, B_T and D_T^M all hold, where the last event is given by (4.27). Exactly as in the proof of Theorem 4.13, we show that $\mathbb{P}\left(A_T \cap B_T \cap D_T^M\right) \geq 1 - T^{-1}/(6\sqrt{\pi}) - T\delta_T^{-1}(1 - \delta_T^2 T^{-2}/36)^M$.

We give the constants as follows:

$$C_1 = 2\sqrt{\frac{2}{3}}C_3^{3/2} + \sqrt{8}, \quad C_2 = \frac{1}{72} - \frac{2\sqrt{2}}{\underline{C}}, \quad C_3 = 2\sqrt[3]{7}\left(3\left(1+\sqrt{2}\right)\right)^{2/3}.$$

We require $\underline{C}$ to be sufficiently large such that $\underline{C}C_2 > C_1$. Consequently it is possible to select $\zeta_T \in \left[C_1\sqrt{\log T}, C_2\delta_T^{3/2}\underline{f}_T\right)$.

Step Four

Consider a generic interval $[s, e]$ satisfying

$$\exists j \in \{1, \ldots, q\},\ \exists k \in \{1, \ldots, M\},\ \text{s.t. } [s_k, e_k] \subset [s, e] \text{ and } s_k \times e_k \in \mathcal{I}_j^L \times \mathcal{I}_j^R \tag{4.34}$$

and define events

$$\mathcal{M}_{s,e} = \left\{m : [s_m, e_m] \in F_T^M, [s_m, e_m] \subset [s, e]\right\},$$
$$\mathcal{O}_{s,e} = \{m \in \mathcal{M}_{s,e} : \max_{s_m \leq b < e_m} \mathcal{C}_{s_m,e_m}^b(\mathbf{Y}) > \zeta_T\}.$$

Let $b_k^* = \operatorname{argmax}_{s_k \le b \le e_k} \mathcal{C}^b_{s_k,e_k}(\mathbf{Y})$. We have

$$\begin{aligned}\mathcal{C}^{b_k^*}_{s_k,e_k}(\mathbf{Y}) &\ge \mathcal{C}^{\tau_j}_{s_k,e_k}(\mathbf{Y}) \\ &\ge \mathcal{C}^{b_k^*}_{s_k,e_k}(\mathbf{f}) - \lambda_T \ge \frac{1}{\sqrt{24}}(\delta_T/6)^{3/2}\Delta_j^{\mathbf{f}} - \lambda_T \ge \frac{1}{72}\delta_T^{3/2}\underline{f}_T - \lambda_T \\ &= \left(\frac{1}{72} - \frac{\lambda_T}{\delta_T^{3/2}\underline{f}_T}\right)\delta_T^{1/2}\underline{f}_T \ge \left(\frac{1}{72} - \frac{2\sqrt{2}}{\underline{C}}\right)\delta_T^{3/2}\underline{f}_T = C_2\delta_T^{3/2}\underline{f}_T > \zeta_T,\end{aligned}$$

where the third inequality above follows from Lemma 4.6.5, therefore $\mathcal{O}_{s,e}$ is non-empty.

Let $m^* = \operatorname{argmin}_{m\in\mathcal{O}_{s,e}}(e_m - s_m + 1)$ and $b^* = \operatorname{argmax}_{s_{m^*}\le b<e_{m^*}} \mathcal{C}^b_{s_{m^*},e_{m^*}}(\mathbf{Y})$. Arguing exactly as in Step Four in the proof of Theorem 4.2.1, we show that $[s_{m^*}, e_{m^*})$ must contain exactly one change-point. Without loss of generality, assume that $\tau_j \in [s_{m^*}, e_{m^*})$. Let $\eta_L = \tau_j - s_{m^*}$, $\eta_R = e_{m^*} - \tau_j$ and $\eta_T = \left(\sqrt{3}(C_1 - \sqrt{8})\sqrt{\log T}(\Delta_j^{\mathbf{f}})^{-1})\right)^{2/3} - 1$. We observe that $\min(\eta_L, \eta_R) > \eta_T$, as $\min(\eta_L, \eta_R) \le \eta_T$ and Lemma 4.6.5 implies that

$$\begin{aligned}\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{Y}) &\le \mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{f}) + \lambda_T \le \mathcal{C}^{\tau_j}_{s_{m^*},e_{m^*}}(\mathbf{f}) + \lambda_T \le \frac{1}{\sqrt{3}}(\eta_T+1)^{3/2}\Delta_j^{\mathbf{f}} + \lambda_T \\ &= (C_1 - \sqrt{8} + \sqrt{8})\sqrt{\log T} = C_1\sqrt{\log T} \le \zeta_T,\end{aligned}$$

contradicting $\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}(\mathbf{Y}) > \zeta_T$.

We are now in position to prove that $|b^* - \tau_j| \le C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3} := \epsilon_T$. Let $b \in \{s_{m^*}+1, \ldots, e_{m^*}-2\}$ and define $\kappa = ((\mathcal{C}^{\tau_j}_{s_k,e_k}(\mathbf{f}))^2 - (\mathcal{C}^b_{s_k,e_k}(\mathbf{f}))^2$. We claim that

$$(\mathcal{C}^{\tau_j}_{s_{m^*},e_{m^*}}(\mathbf{Y}))^2 - (\mathcal{C}^b_{s_{m^*},e_{m^*}}(\mathbf{Y}))^2 > 0, \tag{4.35}$$

when $|b - \tau_j| > \epsilon_T$. Inequality (4.35) does not hold for $b = b^*$, so proving the claim suffices to demonstrate that $|b^* - \tau_j| \le \epsilon_T$. Without loss of generality, we consider the case of $b > \tau_j$. Using arguments as those in Step Four of the proof of Theorem 4.2.1 we can show that (4.35) is implied by $\kappa > (\sqrt{2}+1)^2\lambda_T^2$, where $\kappa = (\mathcal{C}^{\tau_j}_{s_{m^*},e_{m^*}}(\mathbf{f}))^2 - (\mathcal{C}^b_{s_{m^*},e_{m^*}}(\mathbf{f}))^2$. By

Lemma 4.6.7, $\kappa > (\sqrt{2}+1)^2\lambda_T^2$ is implied by

$$\min\left(|b-\tau_j|, \eta_L\right) > \left(63(\Delta_j^{\mathbf{f}})^{-2} \cdot 8(\sqrt{2}+1)^2 \log T\right)^{1/3} = C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3}$$

However, for sufficiently large T,

$$\begin{aligned}\eta_L > \eta_T &= (\sqrt{3}(C_1-\sqrt{8}))^{2/3}(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3} - 1 > (C_1-\sqrt{8})^{2/3}(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3} \\ &= (C_3^{3/2}+\sqrt{8}-\sqrt{8})^{2/3}(\Delta_j^{\mathbf{f}})^{-2/3} = C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3} = \epsilon_T,\end{aligned}$$

hence $|b-\tau_j| > \epsilon_T$ implies (4.35), so it must hold that $|b^*-\tau_j| \leq \epsilon_T$.

Step Five

Using the arguments given above which are valid on the event $A_T \cap B_T \cap D_T^M$, we can now proceed with the proof of the theorem as follows. At the start of Algorithm 4.6 we have $s = 1$ and $e = T$ and, provided that $q \geq 1$, condition (4.28) is satisfied. Therefore the algorithm detects a change-point b^* in that interval such that $|b^*-\tau_j| \leq C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3}$. By construction, we also have that $|b^*-\tau_j| < 2/3\delta_T$. This in turn implies that for all $l = 1,\ldots,q$ such that $\tau_l \in [s,e]$ and $l \neq j$ we have either $\mathcal{I}_l^L, \mathcal{I}_l^R \subset [s,b^*]$ or $\mathcal{I}_l^L, \mathcal{I}_l^R \subset [b^*+1,e]$. Therefore (4.28) is satisfied within each segment containing at least one change-point. Note that before all q change-points are detected, each change-point will not be detected twice. To see this, we suppose that τ_j has already been detected by b, then for all intervals $[s_k,e_k] \subset [\tau_j-\epsilon_T+1, \tau_j-\epsilon_T+2/3\delta_T+1] \cup [\tau_j+\epsilon_T-2/3\delta_T, \tau_j+\epsilon_T]$,

Lemma 4.6.5, together with the event A_T, guarantees that

$$\begin{aligned}\max_{s_k \le b < e_k} \mathcal{C}^b_{s_k,e_k}(\mathbf{Y}) &\le \max_{s \le b < e} \mathcal{C}^b_{s_k,e_k}(\mathbf{f}) + \sqrt{8\log T} \\ &\le \frac{1}{\sqrt{3}}(C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3} + 1)^{3/2}\Delta_j^{\mathbf{f}} + \sqrt{8\log T} \\ &\le (2\sqrt{\frac{2}{3}}C_3^{3/2} + \sqrt{8})\sqrt{\log T} = C_1\sqrt{\log T} \le \zeta_T\end{aligned}$$

Once all the change-points are detected, we then only need to consider $[s_k, e_k]$ such that

$$[s_k, e_k] \subset [\tau_j - C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3} + 1, \tau_{j+1} + C_3(\Delta_{j+1}^{\mathbf{f}})^{-2/3}(\log T)^{1/3}]$$

for $j = 0, \ldots, q$, where we set $\Delta_0^f = \Delta_{q+1}^f = \infty$ for notational convenience. It follows from Lemma 4.6.6 (within A_T) that

$$\begin{aligned}\max_{s_k \le b < e} \mathcal{C}^b_{s_k,e_k}(\mathbf{Y}) &\le \max_{s \le b < e} \mathcal{C}^b_{s_k,e_k}(\mathbf{f}) + \sqrt{8\log T} \\ &\le \frac{1}{\sqrt{3}}(C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3})^{3/2}\Delta_j^{\mathbf{f}} \\ &\quad + \frac{1}{\sqrt{3}}(C_3(\Delta_j^{\mathbf{f}})^{-2/3}(\log T)^{1/3})^{3/2}\Delta_{j+1}^{\mathbf{f}} + \sqrt{8\log T} \\ &= (\frac{2}{\sqrt{3}}C_3^{3/2} + \sqrt{8})\sqrt{\log T} \le C_1\sqrt{\log T} \le \zeta_T.\end{aligned}$$

Hence the algorithm terminates and no further change-points are detected. □

Chapter 5

Multiscale autoregression on adaptively detected timescales

5.1 Introduction

Let X_t be the univariate time series representing return on a financial asset, observed at a mid- or high- frequency, e.g. every ten minutes. In this chapter, we propose Adaptive Multiscale Autoregressive (AMAR) time series models, where X_t linearly depends on its own past averages calculated over unknown timespans. Formally, the AMAR(q) model is defined as

$$X_t = \alpha_1 \frac{1}{\tau_1}(X_{t-1} + \ldots + X_{t-\tau_1}) + \ldots + \alpha_q \frac{1}{\tau_q}(X_{t-1} + \ldots + X_{t-\tau_q}) + \varepsilon_t, \quad t = 1, \ldots, T, \tag{5.1}$$

where the timescales $1 \leq \tau_1 < \tau_2 < \ldots < \tau_q$ and the scale coefficients $\alpha_1, \ldots, \alpha_q \in \mathbb{R}$ are unknown, the number of scales q is much smaller than the largest timescale τ_q and ε_t is a white-noise-like innovation.

The key idea behind the AMAR(q) model is that, in the hope of improving the

forecasting accuracy, we model X_t using only some features of the past (in (5.1) these are the average calculated at possible large timescales), as opposed to using all information as in e.g. the Autoregressive time series model given by (5.2) below. Similar concepts have been previously studied in in the context of modelling of multivariate time series in Reinsel (1983) and Ahn and Reinsel (1988). Reinsel (1983) consider the multivariate autoregressive index models, where the multivariate time series of interest $\mathbb{Y}_t$ depends linearly on a small number of the index variables which are linear combinations of the lagged values of $\mathbb{Y}_t$. The averages of X_t in (5.1) serve as an example of the index variables. However, in contrast to our setting, the index variables in Reinsel (1983) are assumed the be known. Ferreira et al. (2006) consider a multi-scale time series model in a Bayesian context. In their model, the time series of interest observed at a coarser timescale depends on the averages observed at the finer timescales, which is exactly the opposite to the dependence structure in AMAR(q). Another class of multi-scale time series models is proposed in Ghysels et al. (2004), where time series observed at finer scales are used to model the one observed at the lower frequency.

We propose the estimation procedure for fitting AMAR(q) models from the data, which is motivated as follows. Observe that for any $p > \tau_q$, AMAR(q) is an instance of the sparsely parametrised Autoregressive (AR) time series model, therefore (5.1) can be rewritten as

$$X_t = \beta_1 X_{t-1} + \ldots \beta_p X_{t-p} + \varepsilon_t, \quad t = 1, \ldots, T, \tag{5.2}$$

$$\beta_j = \sum_{k:\tau_k \geq j} \frac{\alpha_k}{\tau_k}, \quad j = 1, \ldots, p, \tag{5.3}$$

where ε_t is again a white-noise-like sequence. We refer to (5.2) with the coefficients given by (5.1) as to the AR(p) representation of the AMAR(q) process. Let $\hat{\boldsymbol{\beta}} = (\hat{\beta}, \ldots, \hat{\beta}_p)'$ be the Ordinary Least Squares (OLS) estimator of $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)'$. Then $\hat{\beta}_j$'s trivially

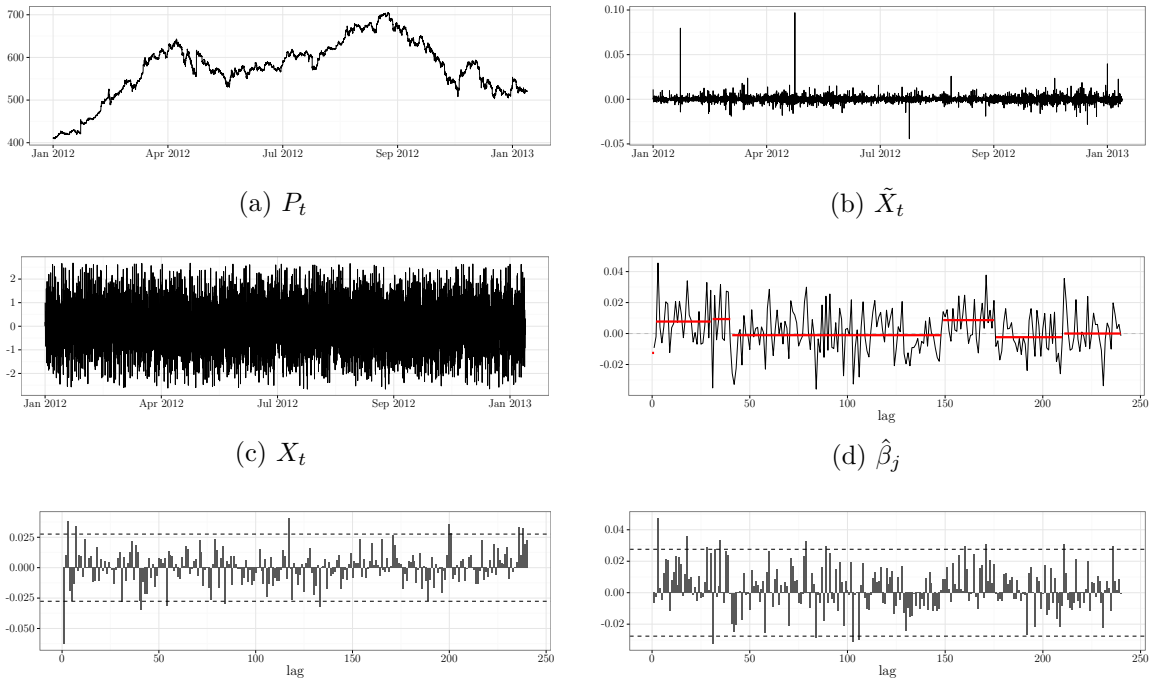
decompose as

$$\hat{\beta}_j = \beta_j + (\hat{\beta}_j - \beta_j), \quad j = 1, \ldots, p. \tag{5.4}$$

The coefficients $\beta_1, \ldots, \beta_p$ form the piecewise-constant vector with the change-points at the timescales $\tau_1, \ldots, \tau_q$. Consequently, (5.4) follows the 'piecewise-constant signal + noise' model, with the noise sequence $\hat{\beta}_j - \beta_j$, $j = 1, \ldots, p$, which implies that the timescales can be estimated by identifying the change-points in (5.4) and motivates the following procedure. First, we choose a large p and find the OLS estimates of the autoregressive coefficients in the AR(p) representation of the AMAR(q) process. Subsequently, we estimate the time-scales by identifying the change-points in (5.4), using to this end the Narrowest-Over-Threshold approach introduced in Chapter 4. Once the time-scales are estimated, we estimate the scale coefficients, using to this end OLS again. As an illustration, an example of the resulting estimates is shown in Figure 5.1d.

From the theoretical point of view, our main contributions can be summarised as follows. We demonstrate that our proposal recovers the locations of the timescales with a large probability, under the framework in which the timescales are allowed to diverge with the growing sample size T. As a side result, we provide an explicit bound on the tail probability of the ℓ_2 norm of the difference between the autoregressive coefficients an their OLS estimates in the AR(p) model with i.i.d. Gaussian noise. The bound can be used to study consistency of the OLS estimators when both the order p and the autoregressive coefficients depend on the sample size T.

We also show that AMAR(q) models estimated with our procedure offer relatively good predictive power in terms of out-of-sample forecasting of high- and mid- frequency financial returns, in an application to stock price series for a number of companies listed on New York Stock Exchange (NYSE). The R package **amar** (Baranowski and Fryzlewicz, 2016a) provides an efficient implementation of our proposal. The most computationally



(a) P_t (b) $\tilde{X}_t$

(c) X_t (d) $\hat{\beta}_j$

(e) acf $\tilde{X}_t$ (f) acf X_t

Figure 5.1: Example of high-frequency trades data for Apple Inc., observed from January 2012 to January 2013, obtained from the New York Stock Exchange Trades and Quotes Database through Wharton Research Data Services. Figure 5.1a: price of the stock P_t observed every 10 minutes. Figure 5.1b: log-returns $\tilde{X}_t = \log(P_t/P_{t-1})$. Figure 5.1c: the normalised log-returns X_t (see Section 5.4.1 for details). Figure 5.1d: the OLS estimates of the AR(p) coefficients with $p = 240$ (thin black) and the piecewise-constant estimate of the coefficients computed with our proposal (red). Figure 5.1e and 5.1f: the autocorrelation function for, respectively, $\tilde{X}_t$ and X_t.

intensive parts of the computations have been coded in C, with a focus on ensuring optimal use of the available computational resources. The R code used in all numerical examples reported in this chapter is available from our GitHub repository (Baranowski and Fryzlewicz, 2016b).

The remainder of the chapter is organised as follows. In Section 5.2, we introduce an estimation procedure for AMAR(q) models and study its theoretical properties. Section 5.3 discusses the choice of the parameters of the procedure and demonstrates its finite-sample performance on simulated data. In Section 5.4, we apply our proposal in order to forecast high-frequency returns for several stocks listed on NYSE. Sections 5.5 and 5.6 contain proofs of all our theoretical results.

5.2 Methodology and theory

5.2.1 Notation

In a typical application of the AMAR(q) model, the number of timescales q is considered to be small in comparison to the maximum timescale τ_q. In order to model this phenomenon, we work in a framework where the timescales $\tau_1, \ldots, \tau_q$ possibly diverge with, and the coefficients $\alpha_1, \ldots, \alpha_q$ depend on the sample size T. However, for economy of notation we suppress the dependence of T on α_j, τ_j, q and X_t in the remainder of this chapter. Apart from the overall amount of the noise in (5.4), the following two quantities will measure how difficult a change-point problem is:

$$\delta_T = \min_{j=1,\ldots,q} |\tau_{j+1} - \tau_j| \tag{5.5}$$

$$\underline{\alpha}_T = \min_{j=1,\ldots,q} |\beta_{\tau_j+1} - \beta_{\tau_j}| = \min_{j=1,\ldots,q} |\alpha_j| \tau_j^{-1} \tag{5.6}$$

Let $\mathbb{C}$ denote the complex plane. For any AR(p) process, we define its characteristic polynomial by

$$b(z) = 1 - \sum_{j=1}^{p} \beta_j z^j, \tag{5.7}$$

where $z \in \mathbb{C}$. Furthermore, the unit circle is denoted by

$$\mathbb{T} = \{z \in \mathbb{C} : |z| = 1\}, \tag{5.8}$$

Finally, for any vector $\mathbf{v} = (v_1, \ldots, v_k)' \in \mathbb{R}^k$ the Euclidean norm is denoted by $\|\mathbf{v}\| = \sqrt{\sum_{j=1}^{k} v_k^2}$.

5.2.2 Large deviations for the OLS estimator in AR(p)

In this section, we obtain a tail probability bound on the Euclidean norm of the difference between the OLS estimator $\hat{\boldsymbol{\beta}}$ of the autoregressive parameters $\boldsymbol{\beta}$ in model (5.2), with all bounds explicitly depending on T, p and other parameters of the AR(p) process. The following theorem holds.

Theorem 5.2.1. *Suppose X_t, $t = 1, \ldots, T$, follow the AR(p) model given by (5.2) and assume that the innovations $\varepsilon_1, \ldots, \varepsilon_T$ are i.i.d. $\mathcal{N}(0,1)$ distributed. Suppose that the initial conditions $X_t = 0$ a.s. for $t = 0, -1, \ldots, -p+1$ and all roots of the characteristic polynomial $b(z)$ given by (5.7) lie within the unit circle. Let $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)'$ be the OLS estimate of the vector of the autoregressive coefficients $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)'$. Then there exist universal constants κ_1, κ_2, $\kappa_3 > 0$ not depending on T, p and $\boldsymbol{\beta}$ s.t. if $\sqrt{T} > \kappa_2 p \log(T)$, we have*

$$\mathbb{P}\left(\left\| \hat{\boldsymbol{\beta}} - \boldsymbol{\beta} \right\| \leq \kappa_1 (\underline{b}/\overline{b})^2 \|\boldsymbol{\beta}\| \frac{p \log(T) \sqrt{\log(T+p)}}{\sqrt{T} - \kappa_2 p \log(T)} \right) \geq 1 - \frac{\kappa_3}{T}, \tag{5.9}$$

where $\underline{b} = \min_{z\in\mathbb{T}} |b(z)|$ *and* $\overline{b} = \max_{z\in\mathbb{T}} |b(z)|$.

Theorem 5.2.1 implies that, with high probability, the differences $\hat{\beta}_j - \beta_j$ in (5.4) diverge to zero with $T \to \infty$, provided that $p = o(T^{1/2})$. This property justifies why (5.4) can be seem as the 'piecewise-constant signal + noise' problem.

We remark that in a classic setting, where both the order p and the autoregressive coefficients in model (5.2) do not depend on the sample size T, properties of the OLS estimators are well-established. For example, Bercu et al. (2008) provides an exponential inequality for the OLS estimats in the AR(1) model with the i.i.d. Gaussian noise. Lai and Wei (1983) show that, without any assumptions on the roots of the characteristic polynomial $b(z)$, hence both in a stationary and non-stationary case, the OLS estimates are strongly consistent, provided that ε_t is a martingale difference sequence with conditional second moments bounded from below and above. (Barabanov, 1983) obtains similar results independently, under slightly stronger assumptions on the noise sequence.

5.2.3 Estimation of the timescales with NOT

In order to estimate the locations of change-points in (5.4), we propose to use the Narrowest-Over-Threshold approach introduced in Chapter 4 with the contrast function (4.6) designed to detect change-points in piecewise-constant signals. This choice is, first of all, motivated by the fact that owing to its modular structure, NOT can be easily adopted to the problem of estimation of the timescales in (5.4). Second, recall that in Section 4.2.5 NOT has been shown to approximately recover the locations of the change-points at optimal rates in the 'piecewise-constant signal + i.i.d. Gaussian noise' model. Although it is challenging to establish the corresponding optimal rates in (5.4) due to the serial dependence in the noise $\hat{\beta}_j - \beta_j$, we show later in Section (5.2.4) that NOT estimates in (5.4) enjoy properties similar to those established in the i.i.d. Gaussian setting.

Algorithm 5.8 NOT algorithm for estimation of the time-scales in AMAR models

Input: Estimates of the autoregressive coefficients $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)'$, F_T^M being a set of M independently drawn intervals with the endpoints in $\{1, \ldots, p\}$, $\mathcal{S} = \emptyset$.

Output: Set of estimated scales $\mathcal{S} \subset \{1, \ldots, p\}$.

procedure NOT$(\hat{\boldsymbol{\beta}}, s, e, \zeta_T)$
  if $e - s < d$ **then** STOP
  else
    $\mathcal{M}_{s,e} := \left\{m : [s_m, e_m] \in F_T^M, [s_m, e_m] \subset [s, e]\right\}$
    if $\mathcal{M}_{s,e} = \emptyset$ **then** STOP
    else
      $\mathcal{O}_{s,e} := \left\{m \in \mathcal{M}_{s,e} : \max_{s_m \le b \le e_m} \mathcal{C}^b_{s_m, e_m}\left(\hat{\boldsymbol{\beta}}\right) > \zeta_T\right\}$
      if $\mathcal{O}_{s,e} = \emptyset$ **then** STOP
      else
        $m^* :\in \operatorname{argmin}_{m \in \mathcal{O}_{s,e}} |e_m - s_m + 1|$
        $b^* := \operatorname{argmax}_{s^*_m \le b \le e^*_m} \mathcal{C}^b_{s^*_m, e^*_m}\left(\hat{\boldsymbol{\beta}}\right)$
        $\mathcal{S} := \mathcal{S} \cup \{b^*\}$
        NOT$(\hat{\boldsymbol{\beta}}, s, b^*, \zeta_T)$
        NOT$(\hat{\boldsymbol{\beta}}, b^* + 1, e, \zeta_T)$
      end if
    end if
  end if
end procedure

Let $\zeta_T > 0$ be the threshold which we use to identify large CUSUMS and F_T^M be set of M randomly drawn subsamples with the endpoint in $\{1, \ldots, p\}$. The NOT procedure for estimation of the timescales in the AMAR(q) model is described in Algorithm 5.8 using pseudocode.

5.2.4 AMAR algorithm and its theoretical properties.

Having introduced all the ingredients of our proposal, we now introduce the AMAR estimation procedure. Its key steps are described in Algorithm 5.9, again using pseudocode. An efficient implementation of the procedure is available from the R package **amar** (Baranowski and Fryzlewicz, 2016a).

Algorithm 5.9 AMAR algorithm

Input: Data $X_1, \ldots, X_T$, threshold ζ_T and M, p.

Output: Estimates of the relevant scales $\hat{\tau}_1, \ldots, \hat{\tau}_{\hat{q}}$ and the corresponding AMAR coefficients $\hat{\alpha}_1, \ldots, \hat{\alpha}_{\hat{q}}$.

procedure AMAR($X_1, \ldots, X_T$, p, ζ_T)

Step 1 Find $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)'$ being the OLS estimates of the autoregressive coefficients in the AR(p) representation of AMAR(p).

Step 2 Call NOT($\hat{\boldsymbol{\beta}}, 1, p, \zeta_T$) given in Algorithm 5.8 to find the estimates of the timescales $\hat{\tau}_1, \ldots, \hat{\tau}_{\hat{q}}$.

Step 3 With the timescales in (5.1) set to $\hat{\tau}_1, \ldots, \hat{\tau}_{\hat{q}}$, find $\hat{\alpha}_1, \ldots, \hat{\alpha}_{\hat{q}}$ being the OLS estimates of the scale coefficients $\alpha_1, \ldots, \alpha_q$.

end procedure

Studying the theoretical properties of the estimates of the timescales obtained with Algorithm 5.9, we make the following assumptions on the time series X_t following the AMAR(q) model and p being the order of its AR(p) representation.

(A1) Assume that the process X_t follows the AMAR(q) model given (5.1) with the innovations ε_t i.i.d. $\mathcal{N}(0, 1)$ and the initial conditions satisfy $X_t = 0$ a.s. for $t < 0$.

(A2) Suppose that $p > \tau_q$ and there exist constants $\theta < \frac{1}{2}$ and $c_1 > 0$ such that $p < c_1 T^\theta$ for all T.

(A3) Assume that all roots of the characteristic polynomial $b(z)$ given by (5.7) lie strictly inside the unit circle. Furthermore, suppose that there exist two constants $c_2, c_3 > 0$ such that $c_2 \leq \min_{z\in\mathbb{T}} |b(z)| \leq \max_{z\in\mathbb{T}} |b(z)| \leq \sqrt{1 + \|\boldsymbol{\beta}\|^2} \leq c_3$ uniformly in T.

(A4) Set $\lambda_T = c_4 T^{\theta-\frac{1}{2}} (\log(T))^{3/2}$, where θ is as in (A2) and $c_4 > 0$ is certain constant depending on c_1, c_2, c_3 given in (A2) and (A3). Assume that $\delta_T^{1/2} \underline{\alpha}_T \geq \underline{c}\lambda_T$ for a sufficiently large $\underline{c} > 0$, where δ_T and $\underline{\alpha}_T$ are given by (5.5) and (5.6), respectively.

The Gaussianity Assumption (A1) is made to simplify the theoretical arguments of the proof of Theorem 5.2.1, which is subsequently used to justify Theorem 5.2.2 given below. As we argue in Section 5.2.2, Theorem 5.2.1, could be possibly extended to cover more complicated scenarios, where e.g. ε_t is a martingale difference sequence following a non-Gaussian distribution. However, the Gaussianity assumption appears to be reasonable from the point of view of the applications of AMAR(q) in forecasting high-frequency returns in Section 5.4. In the applications, we first remove the volatility from the data and subsequently apply AMAR(q) modelling to the resulting residuals, an example of which can be seen in Figure 5.1c.

Condition (A2) imposes the restriction on both p and the maximum time-scale τ_q, which is allowed to increase with $T \to \infty$, but at the rate not faster than $T^{1/2}$. Similar condition on p being the order of AR(p) approximations of an AR(∞) processes can be found in e.g. Ing and Wei (2005). Assumption (A3) implies that for the AMAR(q) process X_t, $t = 1, \ldots, T$ is stationary for all T. The requirement that $\min_{z\in\mathbb{T}} |b(z)|$ is bounded from below implies that the roots of the characteristic polynomial do not approach the unit circle $\mathbb{T}$ when $T \to \infty$, which in turn ensures that the X_t process is, heuristically speaking, sufficiently far from the unit root process.

Assumption (A4) controls both the minimum spacing between the timescales and the size of the jumps in (5.3). The quantity $\delta_T^{1/2} \underline{\alpha}_T$ displayed here is well-known in the

change-point detection literature and characterises the difficulty of the problem, as e.g. in the problems discussed in Section 4.2.5.

Under the introduced assumptions, the following result holds for the AMAR estimation procedure.

Theorem 5.2.2. *Suppose that the assumptions (A1), (A2), (A3) and (A4) are met. Let* $\hat{q}$ *and* $\hat{\tau}_1, \ldots, \hat{\tau}_q$ *denote, respectively, the number and the locations of the timescales sorted in increasing order estimated with Algorithm 5.9. There exist constants* $C_1, C_2, C_3, C_4 > 0$ *such that if* $C_1 \lambda_T \leq \zeta_T \leq C_2 \delta_T^{1/2} \underline{\alpha}_T$*, and* $M \geq 36 T \delta_T^{-2} \log(T \delta_T^{-1})$*, then for all sufficiently large* T *we have*

$$\mathbb{P}\left(\hat{q} = q, \max_{j=1,\ldots,q} |\hat{\tau}_j - \tau_j| \leq \epsilon_T\right) \geq 1 - C_4 T^{-1}, \tag{5.10}$$

with $\epsilon_T = C_3 \lambda_T^2 \underline{\alpha}_T^{-2}$.

The main conclusion of Theorem 5.2.2 is that Algorithm 5.9 estimates the number of the time-scales correctly, while the corresponding locations of the estimates lie *close* to the true timescales, all with a high probability. Under certain circumstances, Algorithm 5.9 recovers the exact locations of the time-scales. Consider e.g. the case when both the number of scales q and the scale coefficients $\alpha_1, \ldots, \alpha_q$ in (5.1) are fixed, while the time-scales increase with T such that $\delta_T \sim p \sim T^\theta$ (recall that '$\sim$' means that the quantities grow at the same rate with $T \to \infty$). This is a challenging setting, in which $\underline{\alpha}_T \sim T^{-\theta}$ and $\|\boldsymbol{\beta}\| \sim T^{-\theta/2}$, where the coordinates of $\boldsymbol{\beta}$ are given by (5.3), so the signal strength decreases to 0 when $T \to \infty$. Here $\delta_T^{1/2} \underline{\alpha}_T \sim T^{-\theta/2}$, consequently (A4) can be met only if θ in (A2) satisfies an additional requirement $\theta \leq \frac{1}{3}$. The distance between the true timescales and their estimates is then not larger than $\epsilon_T \sim T^{4\theta-1}(\log(T))^3$, which converges to zero provided that $\theta < \frac{1}{4}$. In this case, (5.10) simplifies to $\mathbb{P}\left(\hat{q} = q, \forall j = 1, \ldots, q\ \hat{\tau}_j = \tau_j\right) \geq 1 - C_4 T^{-1}$, when T is sufficiently large.

5.3 Practicalities and simulated examples

5.3.1 Parameter choice and other practicalities

In this section, we elaborate on the problem of the parameter choice for Algorithm 5.9, which requires specification of the maximum timescale p, the number of subsamples drawn M and the threshold ζ_T, the latter two being used in the NOT procedure. We also discuss the computational complexity of the proposed estimation procedure.

5.3.1.1 Choice of the threshold ζ_T

The lower bound for the the admissible thresholds in Theorem 5.2.2 is of the order of $O(T^{\theta-1/2}(\log(T))^{3/2})$, regardless of the value of δ_T and $\underline{\alpha}_T$. This motivates the use of the thresholds of the form $\zeta_T = CT^{\theta-1/2}(\log(T))^{3/2}$ in Algorithm 5.9, with $C > 0$ being the user-specified constant and θ as in (A2). As the value of θ is unknown, in our simulations we use $\theta = 0$ aiming to ensure that the NOT procedure does not underestimate the number of the timescales, which inevitably happens when the threshold is too large. From the practical point of view, it is difficult to propose a particular choice of C, as the constant in Theorem 5.2.2 depends on the unknown constants that are listed in Assumptions (A1)–(A4), which can be seen from the proofs of Theorem 5.2.1 and Theorem 5.2.2. In the simulation study of Section 5.3.2 we present the results for the threshold $\zeta_T = CT^{-1/2}(\log(T))^{3/2}$ with $C = 0.25$ in $C = 0.5$, the both of which appear to work well for large sample sizes. However, in real world applications we suggest to use a data-adaptive approach for the choice of ζ_T. For any $\zeta_T > 0$, denote by $\hat{X}_t(\zeta_T)$ the forecast of X_t constructed with Algorithm 5.9 and by $\hat{q}(\zeta_T)$ the number of the estimated timescales. We propose to select the thresholds that minimises the Schwarz Information

Criterion (SIC) defined as follows:

$$\mathrm{SIC}(\zeta_T) = T \log \left(\sum_{t=1}^{T} (X_t - \hat{X}_t(\zeta_T))^2 \right) + 2\hat{q}(\zeta_T) \log T, \tag{5.11}$$

where (5.11) is minimised over ζ_T such that $\hat{q}(\zeta_T) \leq q_{max} = 10$. The thresholds which produce solutions satisfying this requirement can be quickly computed using Algorithm 4.7. Another possible way of choosing the appropriate ζ_T is discussed in Section 5.4, where we apply AMAR(q) modelling to forecast high-frequency financial returns.

5.3.1.2 Choice of p

In real-data applications we suggest to choose p corresponding to a large "natural" time span. For example, if X_t represents 5-minute returns, we can take p equal to the length of a trading week or trading month expressed in the number of 5-minute intervals,for which trading activity occurs. In principle, the SIC criterion (5.11) can minimised with respect to both ζ_T and p, but this would increase the total computational burden involved in the procedure, hence we do not pursue this direction.

5.3.1.3 Choice of M

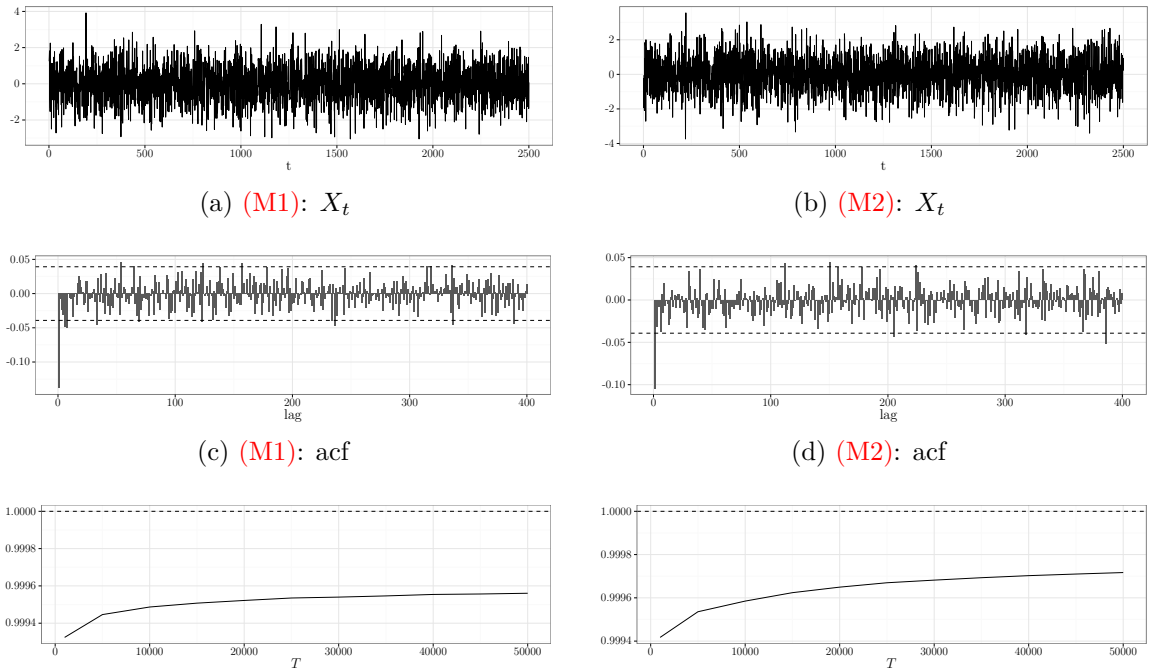
Regarding the choice of the number of subsamples drawn in the NOT procedure, we follow the recommendation given in Section 4.3.4 and set $M = 10000$.

5.3.1.4 Computational complexity.

From the computational point of view, Algorithm 5.9 involves the two main operations. In Step 1 and Step 3, we calculate the OLS estimates with T data points and at most p predictors, the cost of which is typically of the order of $O(Tp^2)$. In Step 2 of Algorithm 5.9, we estimate the change-points in the p-element vector using the NOT procedure. As shown in Section 4.3.5, (4.6) for any pair of integers (s, e) can be computed in $O(e - s)$

time, the computational complexity of Step 2 is therefore $O(Mp)$. The $O(Mp)$ term is typically dominated by $O(Tp^2)$, hence the computational complexity of the entire procedure is $O(Tp^2)$. In practice, Step 1 requires most of the computational time. In order to quickly compute all OLS estimates, the **amar** package uses an efficient implementation of the OLS available from the R package **RcppEigen** (Bates and Eddelbuettel, 2013).

5.3.2 Simulation study



(a) (M1): X_t (b) (M2): X_t

(c) (M1): acf (d) (M2): acf

(e) (M1): $\max_{z:b(z)=0} |z|$ (f) (M2): $\max_{z:b(z)=0} |z|$

Figure 5.2: Example of the data generated according to (5.1) with parameters specified by (M1) and (M2) and $T = 2500$ observations.

To illustrate the finite sample performance of our proposal, we apply Algorithm 5.9 to simulated data. All computations are performed with the **amar** package. The R code used in this section is available from our GitHub repository (Baranowski and Fryzlewicz, 2016b). The data are simulated according to (5.1) for the following two scenarios.

(M1) Three timescales $\tau_1 = 1$, $\tau_2 = \lfloor 20\log(T) \rfloor$, $\tau_3 = \lfloor 40\log(T) \rfloor$ with the corresponding coefficients $\alpha_1 = -0.115$, $\alpha_2 = -2.15$ and $\alpha_3 = -15$ and i.i.d. $\mathcal{N}(0,1)$ noise ε_t.

(M2) Four timescales $\tau_1 = 1$, $\tau_2 = \lfloor 20\log(T) \rfloor$, $\tau_3 = \lfloor 10\log(T)^2 \rfloor$, $\tau_4 = \lfloor 20(\log(T))^2 \rfloor$ with the corresponding coefficients $\alpha_1 = -0.115$, $\alpha_2 = -3.15$, $\alpha_3 = -15$, $\alpha_4 = 10$ and i.i.d. $\mathcal{N}(0,1)$ noise ε_t.

Figure 5.2 shows sample the paths and the estimated autocorrelation function for both scenarios with $T = 2500$ observations. Here we observe that, apart from lag 1, the sample autocorrelation function fails to detect the serial dependence in the data. Figure 5.2e and 5.2f show the largest modulus of the roots of characteristic polynomials for, respectively, (M1) and (M2), depending on the sample size T. The modulus in both cases is always lower than 1, which shows that the corresponding AMAR(q) processes are stationary.

We look at the two aspects of the estimates obtained with Algorithm 5.9. In order to assess the performance of the method in terms of (in-sample) forecasting accuracy, we consider the Relative Prediction Error (RPE), defined as follows:

$$\mathrm{RPE} = \frac{\sum_{t=1}^{T}(\hat{X}_t - \mu_t)^2}{\sum_{t=1}^{T}\mu_t^2}, \tag{5.12}$$

where $\hat{X}_t = \hat{\alpha}_1 \frac{1}{\hat{\tau}_1}(X_{t-1}+\ldots+X_{t-\hat{\tau}_1})+\ldots+\hat{\alpha}_{\hat{q}} \frac{1}{\hat{\tau}_{\hat{q}}}(X_{t-1}+\ldots+X_{t-\hat{\tau}_{\hat{q}}})$ is a AMAR estimate of the conditional mean $\mu_t = \alpha_1 \frac{1}{\tau_1}(X_{t-1}+\ldots+X_{t-\tau_1})+\ldots+\alpha_q \frac{1}{\tau_q}(X_{t-1}+\ldots+X_{t-\tau_q}) = X_t - \varepsilon_t$. We furthermore investigate the accuracy of Algorithm 5.9 at estimation of the timescales $\tau_1, \ldots, \tau_q$. To this end, we consider the following three measures:

$$\mathrm{TP}_\eta = \left|\{j : \exists_k |\hat{\tau}_k - \tau_j| \leq \eta\}\right|, \tag{5.13}$$

$$\mathrm{FN}_\eta = \left|\{j : \nexists_k |\hat{\tau}_k - \tau_j| \leq \eta\}\right|, \tag{5.14}$$

$$\mathrm{FP}_\eta = \hat{q} - \mathrm{TP}_\eta, \tag{5.15}$$

Method	Model	T	p	FP_0	TP_0	FN_0	$\text{FP}_{\log(T)}$	$\text{TP}_{\log(T)}$	$\text{FN}_{\log(T)}$	RPE
SIC	(M1)	2500	412	2.0	1.8	1.1	1.2	2.6	0.4	0.119
		5000	440	1.6	2.1	0.9	0.8	2.9	0.1	0.047
		10000	468	1.1	2.4	0.6	0.5	3.0	0.0	0.018
		25000	505	0.4	2.8	0.2	0.2	3.0	0.0	0.006
		50000	532	0.3	2.9	0.1	0.2	3.0	0.0	0.002
$C = 0.25$	(M1)	2500	412	1.4	1.8	1.2	0.7	2.5	0.5	0.106
		5000	440	0.8	1.9	1.1	0.3	2.5	0.5	0.074
		10000	468	0.3	2.0	1.0	0.0	2.3	0.7	0.054
		25000	505	0.2	2.1	0.9	0.0	2.3	0.7	0.049
		50000	532	0.1	2.0	1.0	0.0	2.1	0.9	0.057
$C = 0.5$	(M1)	2500	412	0.4	1.3	1.7	0.1	1.6	1.4	0.411
		5000	440	0.3	1.5	1.5	0.1	1.7	1.3	0.289
		10000	468	0.2	1.8	1.2	0.0	2.0	1.0	0.169
		25000	505	0.1	2.0	1.0	0.0	2.1	0.9	0.066
		50000	532	0.0	2.0	1.0	0.0	2.0	1.0	0.061
SIC	(M2)	2500	1324	2.5	1.3	2.7	1.8	2.0	2.0	0.259
		5000	1550	2.7	1.4	2.6	1.5	2.5	1.5	0.161
		10000	1796	2.8	1.9	2.1	1.6	3.0	1.0	0.080
		25000	2150	2.3	2.2	1.8	1.1	3.5	0.6	0.038
		50000	2441	1.5	2.7	1.3	0.6	3.6	0.4	0.018
$C = 0.25$	(M2)	2500	1324	10.0	1.2	2.8	9.1	2.1	1.9	0.475
		5000	1550	3.6	1.5	2.5	2.7	2.4	1.6	0.177
		10000	1796	2.6	1.6	2.4	1.4	2.8	1.1	0.095
		25000	2150	1.5	2.2	1.8	0.6	3.1	0.9	0.048
		50000	2441	0.8	2.5	1.5	0.3	3.1	0.9	0.043
$C = 0.5$	(M2)	2500	1324	1.4	1.1	2.9	0.9	1.6	2.5	0.366
		5000	1550	0.9	1.2	2.8	0.4	1.7	2.3	0.292
		10000	1796	0.6	1.3	2.7	0.1	1.8	2.2	0.253
		25000	2150	0.5	1.7	2.3	0.0	2.2	1.8	0.140
		50000	2441	0.2	2.0	2.0	0.0	2.2	1.8	0.129

Table 5.1: Simulation results for the data following (5.1) with parameters given in Section 5.3.2 for a growing sample size T, with averages of the Relative Prediction Error, TP_η, FN_η and FP_η given by, respectively, (5.12), (5.13), (5.14) and (5.15), all calculated over 100 simulated data sets.

with $\eta = 0$ and $\eta = \log T$. For $\eta = 0$, TP_η, FN_η and FP_η are the number of, respectively, true positives, false negatives and false positives.

We apply Algorithm 5.9 with the threshold $\zeta_T = CT^{-1/2}(\log(T))^{3/2}$ for $C = 0.25$ and $C = 0.5$ (the corresponding methods are termed 'THR $C = 0.25$' and 'THR $C = 0.5$', respectively), and with the threshold chosen using the SIC criterion given by (5.11) (termed simply as 'SIC'). The order of the AR(p) representation is set to $p = \lfloor 40 \log(T) \rfloor + 100$ in (M1) and $p = \lfloor 20(\log(T))^2 \rfloor + 100$ in (M2), while $M = 10000$, as recommended in Section 5.3.1. Table 5.1 shows the results. We observe that for all methods, the average RPE decreases with T growing, however, SIC performs the best in this aspect, achieving the lowest RPE in almost all cases. In terms of the estimation of the timescales, we also observe that the performance of all methods improves for the larger sample sizes, with SIC yielding the best results. For example, in (M1) and with $T \geq 10000$, SIC always identifies three timescales close to the true ones, as $\mathrm{TP}_{\log(T)} = 3$ in those cases. For $T \geq 25000$, SIC also very often recovers the exact locations of the timescales, as the average TP_0 is close to 3.

5.4 Application to high-frequency data from NYSE TAQ database

In this section, we apply our proposal to the returns series for a number of stocks listed on the New York Stock Exchange. The chosen companies, shown in Table 5.2, represent various industries and are liquid enough to analyse them at a high-frequency. We download the price tick-by-tick data from the NYSE Trades and Quotes database through Wharton Research Data Services, for the time span covering 10 years from January 2004 to December 2013. The R code used to obtain the results we discuss in this section is available from our GitHub repository (Baranowski and Fryzlewicz, 2016b).

Ticker	Company	Industry
AAPL	Apple Inc.	Computer Hardware
BAC	Bank of America Corp	Banks
CVX	Chevron Corp.	Oil & Gas Exploration & Production
CSCO	Cisco Systems	Networking Equipment
F	Ford Motor	Automobile Manufacturers
GE	General Electric	Industrial Conglomerates
GOOG	Alphabet Inc.	Internet Software & Services
MSFT	Microsoft Corp.	Systems Software
T	AT&T Inc.	Telecommunications

Table 5.2: Ticker symbols and the industries for the companies analysed in Section 5.4.

5.4.1 Data preprocessing

The data are preprocessed in the following three steps. First, as the TAQ database contain some erroneous observations, the tick-by-tick data need to be cleaned. To this end, we use the methodology proposed by Brownlees and Gallo (2006), using the implementation available from the R package **TAQMNGR** (Calvori et al., 2015).

Second, the tick-by-tick data are observed in irregular time intervals. To obtain the price series observed the required frequency, we divide the trading day into time intervals of equal length (we consider 5-minute and 10-minute intervals). For each interval, the price process P_t is defined as the price of the last trade observed in that bin. When there are no trades in an interval, P_t is set to the price of the latest available trade. Computations for this step are also performed with the **TAQMNGR** package.

Third, we remove the volatility from the log-returns $\tilde{X}_t = \log(P_t/P_{t-1})$, using to this end the NoVaS transformation approach (Politis, 2003, 2007). The NoVaS estimate of the (squared) volatility is defined as

$$\hat{\sigma}_t^2(\lambda) = (1-\lambda)\hat{\sigma}_{t-1}^2(\lambda) + \lambda \tilde{X}_t^2, \tag{5.16}$$

with the initial value $\hat{\sigma}_0^2(\lambda) = 1$ and $\lambda \in (0,1)$ being the tuning parameter. The NoVaS transformation is similar to the ordinary exponential smoothing (ES, Gardner (1985);

Taylor (2004)), where σ_t^2 is estimated as the weighted average of the squared returns with exponentially decaying weights. However, the ES estimator depends only on observations prior time t, while (5.16) involves also the current observation. The evolution of the intra-day volatility follows certain periodic patterns (Hecq et al., 2012), e.g. it is typically higher in the morning, when the trading starts, and shortly before the close of the market. The simple ES estimator cannot capture such patterns, as it gives a very small weight to observation where the pattern has been observed for the last time. Judging from the residuals obtained using (5.16), an example of which can be seen in Figure 5.1c, it appears that the NoVaS transformation captures the daily patterns in the volatility reasonably well.

In order to choose an appropriate λ for (5.16), Politis and Thomakos (2013) recommends to find $\lambda \in (0, 1)$ such that that the resulting residuals $\frac{\tilde{X}_t}{\hat{\sigma}_t(\lambda)}$ match a desired distribution. All our theoretical results discussed in Section 5.2 are derived under the Gaussianity assumption for the noise, which in turn implies Gaussianity of the considered AMAR(q) processes. Consequently, we aim to 'Gaussianise' $\tilde{X}_t$, by minimising the Jarque-Bera test statistic (Jarque and Bera, 1980), defined as

$$\mathrm{JB}(\lambda) = \frac{n}{6}\left(\hat{\gamma}(\lambda)^2 + \frac{1}{4}(\hat{\kappa}(\lambda) - 3)^2\right), \tag{5.17}$$

where $\hat{\gamma}(\lambda)$ and $\hat{\kappa}(\lambda)$ denote, respectively, the sample skewness and the sample kurtosis both computed for the residuals $\frac{\tilde{X}_t}{\hat{\sigma}_t(\lambda)}$, computed on the validation set defined in the next section.

5.4.2 Rolling window analysis

We conduct the rolling window analysis, where we compare the forecasts obtained with AMAR(q) models against the predictions obtained with the classic AR(p) model. A detailed description of the procedure applied for a single window is given in Algorithm 5.10.

Algorithm 5.10 AMAR train algorithm

Input: Price series P_t observed at a chosen frequency; the maximum order p of the AR(p) approximation; the number of subsamples M.

Output: The estimated returns $\hat{X}_t$.

procedure TRAINAMAR($P_1, \ldots, P_T$, p, q_{max})

Step 1 Set $\mathcal{S}_{train} = \{1, \ldots, \lfloor 0.5T \rfloor\}$, $\mathcal{S}_{validate} = \{\lfloor 0.5T \rfloor + 1, \ldots, \lfloor 0.75T \rfloor + 1\}$, and $\mathcal{S}_{test} = \{\lfloor 0.75T \rfloor + 1, \ldots, T\}$.

Step 2 Set $\tilde{X}_t = \log(P_t/P_{t-1})$ for $t = 1, \ldots, T$.

Step 3 Find $\lambda^* = \operatorname{argmin}_{\lambda \in (0,1)} \mathrm{JB}(\lambda)$, where $JB(\lambda)$ given by (5.17) is calculated using $\tilde{X}_t$ s.t. $t \in \mathcal{S}_{train}$. Set $X_t = \frac{\tilde{X}_t}{\hat{\sigma}(\lambda^*)}$, $t = 1, \ldots, T$.

Step 4 Using $\tilde{X}_t$ for $t \in \mathcal{S}_{train}$, find $\hat{\beta}_1, \ldots, \hat{\beta}_p$, the OLS estimates of the autoregressive coefficients for the AR(p) model.

Step 5 Apply $\mathrm{NOT}(\hat{\boldsymbol{\beta}}, 1, p, \zeta_T^{(k)})$ for all thresholds $\zeta_T^{(k))}$ such there are at most q_{max} time scales. Denote by $\mathcal{T}_1, \ldots, \mathcal{T}_N$ the resulting sets of timescales.

Step 6 For each $\mathcal{T}_k$, find the OLS estimates of $\alpha_1, \ldots, \alpha_q$, using X_t, $t \in \mathcal{S}_{train}$. Using those estimates, construct predictions $X_t^{(k)}$ for $t \in \mathcal{S}_{validate}$. Find $k^* = \operatorname{argmax}_{k=1,\ldots,N} \mathrm{R}^2_{validate}(k)$, where $\mathrm{R}^2_{validate}(k)$ is given by (5.18) computed for $X_t^{(k)}$ for $t \in \mathcal{S}_{validate}$.

Step 7 Find the OLS estimates of the AMAR coefficients for the timescales $\mathcal{T}_{k^*}$ using X_t such that $t \in \mathcal{S}_{validate}$.

Step 8 Using the model obtained in the previous step, find predictions $\hat{X}_t$ for $t \in \mathcal{S}_{test}$. Record R^2_{test} and HR_{test}.

end procedure

The window size is set to 252 days, which is approximately the number of the trading days on NYSE each year. For each window, the data are split into three parts. The first half (approximately 6 months) is used as the training set on which we estimate the parameters for the analysed candidate models. The subsequent 3 months are used as the validation set, on which we select the model yielding best forecasting (in terms of R^2 introduced below). The last three months serve as the test set, where we use the model selected on the validation set to construct the out-of-sample forecasts of the normalised returns X_t. Once the forecast are calculated, we move the entire window such that the old test set becomes the new validation set.

Let $\hat{X}_t$ be a forecast of X_t for $t = 1, \ldots, T$. The main criterion we use to asses the predictions is defined as follows:

$$R^2 = 1 - \frac{\sum_{t=1}^{T}(X_t - \hat{X}_t)^2}{\sum_{t=1}^{T} X_t^2}. \tag{5.18}$$

Naturally, $R^2 = 0$ for $\hat{X}_t \equiv 0$, therefore $R^2 > 0$ implies that the given forecast beats the 'zeros only' benchmark, which is a difficult task in the context of financial returns. From the point of view of constructing trading strategies involving the forecasts, it is also interesting to investigate how often the sign of the forecast agrees with the sign of the observed return. To this end we consider the hit-rate defined as

$$\mathrm{HR} = \frac{|\{t = 1, \ldots, T : \mathrm{sgn}(\hat{X}_t) = \mathrm{sgn}(X_t), X_t \neq 0\}|}{|\{t = 1, \ldots, T : X_t \neq 0\}|}. \tag{5.19}$$

5.4.3 Results and discussion

Tables 5.3–5.6 show the results of our analysis for the data observed every 5 and 10 minutes, with the p in Algorithm 5.10 set to 6 and 12 days. We observe that, overall, both AR and AMAR achieve positive average R^2, which means that they typically beat

Ticker	Method	R^2	HR − 50%	pct. zeroes
AAPL	MZAR	**0.00042**	**1.6**	2.07
	AR	-0.00036	1.2	
BAC	MZAR	**0.00231**	**2.2**	9.36
	AR	0.00186	0.9	
CVX	MZAR	**-0.00002**	**0.7**	4.24
	AR	-0.00026	0.5	
CSCO	MZAR	**0.00273**	**2.5**	9.32
	AR	0.00235	1.8	
F	MZAR	0.00586	**3.8**	16.75
	AR	**0.00601**	1.8	
GE	MZAR	**0.00208**	**2.2**	9.63
	AR	0.00197	2.1	
GOOG	MZAR	**0.00111**	**1.9**	1.09
	AR	0.00066	1.9	
MSFT	MZAR	**0.00393**	**2.9**	8.79
	AR	0.00386	2.2	
T	MZAR	0.00321	**2.2**	9.68
	AR	**0.00358**	0.7	

Table 5.3: Averages of the measures introduced in Section 5.4 evaluating the out-of sample performance of the forecasts obtained with the MZAR methodology and the classic AR model. The returns X_t are observed every 5 minutes, while the maximum time-scale and the maximum order for MZAR and AR are both set to $p = 480$ (6 trading days expressed in 5-minute intervals). For each pair of a characteristic and company ticker, bold font indicates the better method.

Ticker	Method	R^2	HR − 50%	pct. zeroes
AAPL	MZAR	**0.00038**	**1.7**	2.07
	AR	-0.00038	1.4	
BAC	MZAR	**0.00227**	**2.3**	9.35
	AR	0.00204	1.5	
CVX	MZAR	**0.00017**	**0.8**	4.23
	AR	-0.00034	0.5	
CSCO	MZAR	0.00287	**2.5**	9.36
	AR	**0.00302**	1.9	
F	MZAR	**0.00602**	**3.8**	16.73
	AR	0.00595	1.6	
GE	MZAR	**0.00206**	**2.2**	9.61
	AR	0.00184	2.1	
GOOG	MZAR	**0.00088**	**1.8**	1.08
	AR	0.00064	1.8	
MSFT	MZAR	**0.00375**	**2.8**	8.83
	AR	0.00347	1.8	
T	MZAR	0.00315	**2.2**	9.69
	AR	**0.00350**	0.9	

Table 5.4: Averages of the measures introduced in Section 5.4 evaluating the out-of sample performance of the forecasts obtained with the AMAR methodology and the classic AR model. The returns X_t are observed every 5 minutes, while the maximum time-scale and the maximum order for AMAR and AR are both set to $p = 960$ (12 trading days expressed in 5-minute intervals). For each pair of a characteristic and company ticker, bold font indicates the better method.

Ticker	Method	R^2	HR − 50%	pct. zeroes
AAPL	MZAR	**0.00043**	**1.9**	1.41
	AR	-0.00034	1.5	
BAC	MZAR	**0.00063**	**1.9**	6.83
	AR	0.00011	0.7	
CVX	MZAR	**-0.00065**	**0.5**	3.12
	AR	-0.00075	0.2	
CSCO	MZAR	**0.00181**	**1.8**	6.45
	AR	0.00124	0.5	
F	MZAR	**0.00277**	**2.7**	12.52
	AR	0.00235	0.3	
GE	MZAR	**0.00202**	**1.8**	6.95
	AR	0.00177	1.4	
GOOG	MZAR	**0.00095**	**1.7**	0.72
	AR	0.00072	1.6	
MSFT	MZAR	0.00208	**2.1**	6.12
	AR	**0.00241**	1.2	
T	MZAR	**0.00309**	**1.9**	7.40
	AR	0.00260	0.7	

Table 5.5: Averages of the measures introduced in Section 5.4 evaluating the out-of sample performance of the forecasts obtained with the AMAR methodology and the classic AR model. The returns X_t are observed every 10 minutes, while the maximum time-scale and the maximum order for AMAR and AR are both set to $p = 240$ (6 trading days expressed in 10-minute intervals). For each pair of a characteristic and company ticker, bold font indicates the better method.

Ticker	Method	R^2	HR − 50%	pct. zeroes
AAPL	MZAR	**0.00039**	**1.6**	1.41
	AR	-0.00050	1.5	
BAC	MZAR	**0.00074**	**1.9**	6.82
	AR	0.00033	0.7	
CVX	MZAR	**0.00001**	**0.8**	3.10
	AR	-0.00076	0	
CSCO	MZAR	**0.00154**	**1.8**	6.45
	AR	0.00138	0.6	
F	MZAR	**0.00288**	**3**	12.50
	AR	0.00242	-0.3	
GE	MZAR	0.00230	**2.3**	6.92
	AR	**0.00240**	1.5	
GOOG	MZAR	0.00060	1.5	0.71
	AR	**0.00093**	**1.6**	
MSFT	MZAR	**0.00233**	**2.2**	6.16
	AR	0.00220	1.5	
T	MZAR	**0.00321**	**2**	7.40
	AR	0.00278	0.6	

Table 5.6: Averages of the measures introduced in Section 5.4 evaluating the out-of sample performance of the forecasts obtained with the AMAR methodology and the classic AR model. The returns X_t are observed every 10 minutes, while the maximum time-scale and the maximum order for AMAR and AR are both set to $p = 480$ (12 trading days expressed in 10-minute intervals). For each pair of a characteristic and company ticker, bold font indicates the better method.

the 'zeros only' benchmark. However, in the majority of cases, AMAR is better than AR in terms of R^2 and it is always better in terms of the average hit-rate.

5.4.4 Simulated data with real volatility

In this section, we illustrate the performance of Algorithm 5.10 on a simulated data aiming to resemble 5 minutes log-returns on a stock price. The simulated returns are generated according to the following equation

$$Y_t = \sigma_t X_t, \quad t = 1, \dots, T, \tag{5.20}$$

where X_t follows (5.1), ε_t are i.i.d. $\mathcal{N}(0, \sigma^2)$, while the volatility σ_t is either constant in t or extracted from the real data using the NoVaS procedure described in Section 5.4.1. Scales τ_j for X_t correspond to one trading hour, one trading day and one trading week, all expressed in 5-five minute intervals. The following list summarises all examples studied in this section.

(MV1) Timescales $\tau_1 = 1$, $\tau_2 = 12$ (1 hour), $\tau_3 = 400$ (1 trading week), $\alpha_1 = -0.15$, $\alpha_1 = 0.25$, $\alpha_3 = -3.15$, , $\sigma_t = 1$, $\sigma^2 = 2$,

(MV2) All parameters as in (MV1) expect for σ_t which is simulated from the real data.

(MV3) Timescales $\tau_1 = 1$, $\tau_2 = 12$ (1 hour), $\tau_3 = 80$ (1 trading day), $\tau_4 = 400$ (1 trading week), $\alpha_1 = -0.15$, $\alpha_1 = 0.25$, $\alpha_3 = -3.15$, $\alpha_4 = -5.5$, $\sigma_t = 1$, $\sigma^2 = 2$.

(MV4) All parameters as in (MV3) expect for σ_t which is simulated from the real data.

Table 5.7 shows the results. Clearly, the AMAR models fitted with Algorithm 5.10 offer better forecasting accuracy than the AR model, which can be seen from the lower corresponding values of RPE.

Model	days	p	$\hat{q}$	$\hat{p}$	RPE MZAR	RPE AR	time MZAR	time AR
(MV1)	90	480	4.5	10.3	0.155	0.376	0.25	40.32
	120	480	4.5	10.7	0.111	0.336	0.32	54.15
	180	480	4.2	14.5	0.062	0.326	0.47	83.76
	250	480	4.1	14.0	0.042	0.287	0.65	118.15
(MV2)	90	480	4.5	8.0	0.461	0.571	0.28	41.52
	120	480	4.9	9.5	0.445	0.561	0.34	54.44
	180	480	4.5	10.9	0.424	0.548	0.51	83.91
	250	480	4.5	11.3	0.408	0.541	0.68	115.36
(MV3)	90	480	4.6	74.1	0.104	0.384	0.28	39.06
	120	480	4.6	80.7	0.082	0.266	0.37	53.60
	180	480	4.9	84.2	0.048	0.189	0.50	83.00
	250	480	5.3	84.4	0.026	0.150	0.67	115.47
(MV4)	90	480	4.3	53.6	0.510	0.634	0.28	38.42
	120	480	4.4	69.7	0.501	0.576	0.37	52.90
	180	480	5.3	82.4	0.497	0.523	0.53	82.70
	250	480	6.0	86.3	0.486	0.512	0.76	121.60

Table 5.7: Simulation results for the data following (5.20) with parameters given in Section 5.4.4 for a growing sample size T, with averages of the Relative Prediction Error given by (5.12) for the predictions obtained with the MZAR and classic AR methodology, all calculated over 100 simulated data sets.

5.5 Large deviations for LSE estimators in stationary AR(p) models

Suppose X_t follows the AR(p) model given by (5.2), where the initial conditions satisfy $X_0 = \ldots = X_{-p+1} = 0$ almost surely. The aim of this note is to provide a probabilistic and non-asymptotic bound on the Euclidean norm $\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|$, where $\hat{\boldsymbol{\beta}}$ is the OLS estimator of $\boldsymbol{\beta}$. Conveniently, the AR(p) can be rewritten as the VAR(1) model

$$\mathbf{Y}_t = \mathbf{B}\mathbf{Y}_{t-1} + \varepsilon_t \mathbf{u}, \quad t = 1, \ldots, T, \tag{5.21}$$

where $\mathbf{Y}_t = (X_t, X_{t-1}, ..., X_{t-p+1})'$, the matrix of the coefficients

$$\mathbf{B} = \begin{pmatrix} \beta_1 & \beta_2 & \cdots & \beta_p \\ & \mathbf{I}_{p-1} & & 0 \end{pmatrix} \tag{5.22}$$

and $\mathbf{u} = (1, 0, \ldots, 0)' \in \mathbb{R}^p$. We assume that X_t is stationary, i.e. the modulus of all roots of the characteristic polynomial $b(z)$ give by (5.7) is strictly larger than 1.

5.5.1 Some properties of the B matrix

In this section, we provide some useful facts about various quadratic forms involving matrix $\mathbf{B}$ defined by (5.22), which are essential in proving Theorem 5.2.1. First we recall some well-known facts.

Theorem 5.5.1 (Parseval's Theorem, Theorem 1.9 in Duoandikoetxea (2000))**.** *For any complex-valued sequence* $\{f_k\}_{k\in\mathbb{Z}}$ *such that* $\sum_{k\in\mathbb{Z}} |f_k|^2 < \infty$*, the following identity holds*

$$\sum_{k\in\mathbb{Z}} |f_k|^2 = \int_{\mathbb{T}} |f(z)|^2 dm(z), \tag{5.23}$$

where $a(z) = \sum_{k\in\mathbb{Z}} a_k z^k$, $\mathbb{T} = \{z \in \mathbb{C} : \ |z| = 1\}$, $dm(z) = \frac{d|z|}{2\pi}$.

Lemma 5.5.1 (Cauchy's integral formula)**.** *Let* $\mathbf{M} \in \mathbb{R}^{p\times p}$ *be a real- or complex- valued matrix. Then for any curve* Γ *enclosing all eigenvalues of* $\mathbf{M}$ *and any* $j \in \mathbb{N}$ *the following holds*

$$\mathbf{M}^j = \frac{1}{2\pi i} \int_\Gamma z^j (z\mathbf{I}_p - \mathbf{M})^{-1} dz = \frac{1}{2\pi i} \int_\Gamma z^{j-1} (\mathbf{I}_p - z^{-1}\mathbf{M})^{-1} dz. \tag{5.24}$$

Lemma 5.5.2. *Let* $\mathbf{B}$ *given by* (5.22) *be the matrix of coefficients of a stationary* $AR(p)$

process and let $\mathbf{v} \in \mathbb{R}^p$. *For all* $z \in \mathbb{C}$ *such that* $\sum_{i=0}^{\infty} |\langle \mathbf{v}, \mathbf{B}^i \mathbf{u} \rangle| |z^i| < \infty$, *we have*

$$b(z) \sum_{i=0}^{\infty} \left\langle \mathbf{v}, \mathbf{B}^i \mathbf{u} \right\rangle z^i = b(z) \left\langle \mathbf{v}, (\mathbf{I}_p - z\mathbf{B})^{-1} \right\rangle = v(z), \tag{5.25}$$

where $v(z) = v_1 + v_2 z + \ldots + v_p z^{p-1}$, $b(z)$ *is given by* (5.7).

Proof. As $\sum_{i=0}^{\infty} |\langle \mathbf{v}, \mathbf{B}^i \mathbf{u} \rangle| |z^i| < 0$, we can change the order of summation in the left-hand side of (5.25)

$$(1 - \beta_1 z - \ldots - \beta_p z^p) \sum_{i=0}^{\infty} \left\langle \mathbf{v}, \mathbf{B}^i \mathbf{u} \right\rangle z^i = \left\langle \mathbf{v}, \left(\sum_{i=0}^{\infty} (1 - \beta_1 z - \ldots - \beta_p z^p) z^i \mathbf{B}^i \right) \mathbf{u} \right\rangle$$

Define $\beta_0 = -1$, $\beta_k = 0$ for $k > p$. By direct algebra

$$\sum_{i=0}^{\infty} (1 - \beta_1 z - \ldots - \beta_p z^p) z^i \mathbf{B}^i = -\sum_{i=0}^{\infty} \left(\sum_{k=0}^{i} \beta_k \mathbf{B}^{i-k} \right) z^i := -\sum_{i=0}^{\infty} \mathbf{D}_i z^i$$

The characteristic polynomial of $\mathbf{B}$ is given by $\phi(z) = (-1)^{p+1} \sum_{k=0}^{p} \beta_k z^{p-k}$. From the Cayley-Hamilton theorem , $\mathbf{B}$ is a root ϕ, and, consequently for $i \geq p$

$$\mathbf{D}_i = \mathbf{B}^{i-p} \sum_{k=0}^{i} \beta_k \mathbf{B}^{p-k} = \mathbf{B}^{i-p} \sum_{k=0}^{p} \beta_k \mathbf{B}^{p-k} = 0.$$

It remains to demonstrate that $\langle \mathbf{v}, \mathbf{D}_i \mathbf{u} \rangle = -v_{i+1}$ for $i = 0, \ldots, p-1$, which we show by induction. For $i = 0$, $\langle \mathbf{v}, \mathbf{D}_i \mathbf{u} \rangle = \beta_0 \langle \mathbf{v}, \mathbf{u} \rangle = -v_1$. When $i \geq 1$, matrices $\mathbf{D}_i$ satisfy $\mathbf{D}_i = \mathbf{B}\mathbf{D}_{i-1} + \beta_i \mathbf{I}_p$, therefore

$$\begin{aligned}
\langle \mathbf{v}, \mathbf{D}_i \mathbf{u} \rangle &= \langle \mathbf{v}, \mathbf{B}\mathbf{D}_{i-1} \mathbf{u} \rangle + \beta_i \langle \mathbf{v}, \mathbf{u} \rangle = \langle \mathbf{B}'\mathbf{v}, \mathbf{D}_{i-1} \mathbf{u} \rangle + \beta_i \langle \mathbf{v}, \mathbf{u} \rangle \\
&= \langle v_1(\beta_1, \ldots, \beta_p)' + (v_2, \ldots, v_p, 0)', \mathbf{D}_{i-1} \mathbf{u} \rangle + \beta_i \langle \mathbf{v}, \mathbf{u} \rangle = -v_1 \beta_i - v_{i+1} + v_1 \beta_i \\
&= -v_{i+1},
\end{aligned}$$

which finishes the proof. □

5.5.2 Two useful lemmas

Lemma 5.5.3. *Let $Z_1, Z_2, \ldots$ be a sequence of i.i.d. $\mathcal{N}(0,1)$ random variables. Then for any integers $l \neq 0$ and $k > 0$, the following exponential probability bound holds*

$$\mathbb{P}\left(\left|\sum_{t=1}^{k} Z_t Z_{t+l}\right| > kx\right) \leq 2\exp\left(-\frac{1}{8}\frac{kx^2}{6+x}\right). \tag{5.26}$$

Proof. For brevity, we will only show that $\mathbb{P}\left(\sum_{t=1}^{k} Z_t Z_{t+l} > kx\right) \leq \exp\left(-\frac{1}{8}\frac{kx^2}{6+x}\right)$. Proof of $\mathbb{P}\left(\sum_{t=1}^{k} Z_t Z_{t+l} < kx\right) \leq \exp\left(-\frac{1}{4}kx\right)$ is similar and, combined with the former inequality, implies (5.26). By Markov's inequality, for any $x > 0$ and $\lambda > 0$ it holds that

$$\mathbb{P}\left(\sum_{t=1}^{k} Z_t Z_{t+l} > kx\right) \leq \exp\left(-kx\lambda\right) \mathrm{E}\exp\left(\lambda \sum_{t=1}^{k} Z_t Z_{t+l}\right).$$

Naturally for any $\lambda > 0$ function $y \mapsto \exp(\lambda y)$ is convex, therefore by Theorem 1 in Vershynin (2011), the expectation above is bounded by

$$\mathrm{E}\exp\left(\lambda \sum_{t=1}^{k} Z_t Z_{t+l}\right) \leq \mathrm{E}\exp\left(4\lambda \sum_{t=1}^{k} Z_t \tilde{Z}_t\right),$$

where $\tilde{Z}_1, \ldots, \tilde{Z}_k$ are independent copies of $Z_1, \ldots, Z_k$. Using the independence by direct computation we get

$$\mathrm{E}\exp\left(4\lambda \sum_{t=1}^{k} Z_t \tilde{Z}_t\right) = \left(\mathrm{E}\exp\left(4\lambda Z_1 \tilde{Z}_1\right)\right)^k = \left(\mathrm{E}\exp\left(8\lambda^2 \tilde{Z}_1^2\right)\right)^k = \left(1 - 16\lambda^2\right)^{-\frac{1}{2}k}$$

provided that $0 < \lambda < \frac{1}{4}$, therefore

$$\mathbb{P}\left(\sum_{t=1}^{k} Z_t Z_{t+l} > kx\right) \leq \exp\left(-kx\lambda - \frac{k}{2}\log\left(1 - 16\lambda^2\right)\right).$$

Taking $\lambda = \frac{-2+\sqrt{4+x^2}}{4x}$ minimises the right-hand side of the inequality above. Substituting

this value of λ and using simple bound $\log(x) \leq x - 1$ we further get

$$\begin{aligned}
\mathbb{P}\left(\sum_{t=1}^{k} Z_t Z_{t+l} > kx\right) &\leq \exp\left(\frac{k}{4}\left(2 - \sqrt{x^2+4} + 2\log\left(\frac{1}{4}\left(\sqrt{x^2+4}+2\right)\right)\right)\right) \\
&\leq \exp\left(\frac{k}{4}\left(2 - \sqrt{x^2+4} + \frac{1}{2}\left(\sqrt{x^2+4}+2\right) - 2\right)\right) \\
&= \exp\left(\frac{k}{8}\left(2 - \sqrt{x^2+4}\right)\right) = \exp\left(-\frac{1}{8}\frac{kx^2}{2+\sqrt{x^2+4}}\right) \\
&\leq \exp\left(-\frac{1}{8}\frac{kx^2}{6+x}\right)
\end{aligned}$$

which finishes the proof. □

Lemma 5.5.4 (Lemma 1 in Laurent and Massart (2000)). *Let $Z_1, Z_2, \ldots$ be a sequence of i.i.d. $\mathcal{N}(0,1)$ random variables. For any integer $k > 0$ and $x \in \mathbb{R}$ s.t. $x > 0$, the following exponential probability bounds hold*

$$\mathbb{P}\left(\sum_{t=1}^{k} Z_t^2 \geq k + 2\sqrt{kx} + 2x\right) \leq \exp\left(-x\right), \tag{5.27}$$

$$\mathbb{P}\left(\sum_{t=1}^{k} Z_t^2 \leq k - 2\sqrt{kx}\right) \leq \exp\left(-x\right). \tag{5.28}$$

5.5.3 Proof of Theorem 5.2.1

Proof. For $\mathbf{C}_T = \sum_{t=1}^{T-1} \mathbf{Y}_t \mathbf{Y}_t'$ and $\mathbf{A}_T = \sum_{t=1}^{T-1} \varepsilon_{t+1} \mathbf{Y}_t$, we have $\hat{\boldsymbol{\beta}} - \boldsymbol{\beta} = \mathbf{C}_T^{-1} \mathbf{A}_T$. Consequently,

$$\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\| \leq \lambda_{max}(\mathbf{C}_T^{-1}) \left\|\mathbf{A}_T\right\| = \lambda_{min}^{-1}(\mathbf{C}_T) \left\|\mathbf{A}_T\right\|, \tag{5.29}$$

where $\lambda_{min}(\mathbf{M})$ and $\lambda_{max}(\mathbf{M})$ denote, respectively, the smallest and the largest eigenvalue of a symmetric matrix $\mathbf{M}$. To provide an upper bound on $\left\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|$ given in Theorem 5.2.1, we will bound $\lambda_{min}(\mathbf{C}_T)$ from below and $\|\mathbf{A}_T\|$ from above, working on a set whose

probability is large. Here we will show result more specific than (5.9), i.e.

$$\|\mathbf{A}_T\| \leq \left(32\underline{b}^{-2}\sqrt{1+\|\boldsymbol{\beta}\|^2}\right) p\log(T)\sqrt{(1+\log(T+p))T}, \tag{5.30}$$

$$\lambda_{min}(\mathbf{C}_T) \geq \overline{b}^{-2}\left(T - p(1+32\log(T)\sqrt{T})\right), \tag{5.31}$$

on the the following event

$$\mathcal{E}_T = \mathcal{E}_T^{(1)} \cap \mathcal{E}_T^{(2)} \cap \mathcal{E}_T^{(3)}, \tag{5.32}$$

where

$$\begin{aligned}
\mathcal{E}_T^{(1)} &= \bigcap_{1\leq i<j\leq p}\left\{\left|\sum_{t=1}^{T-\max(i,j)}\varepsilon_t\varepsilon_{t+|i-j|}\right| < 32\log(T)\sqrt{T-\max(i,j)}\right\},\\
\mathcal{E}_T^{(2)} &= \bigcap_{j=1}^{T}\left\{\left|\sum_{t=1}^{T-j}\varepsilon_t\varepsilon_{t+j}\right| < 32\log(T)\sqrt{T-j}\right\},\\
\mathcal{E}_T^{(3)} &= \left\{\sum_{t=1}^{T-p}\varepsilon_t^2 > T-p-2\sqrt{\log(T)(T-p)}\right\}.
\end{aligned}$$

Finally, we will demonstrate that $\mathcal{E}_T$ satisfies

$$\mathbb{P}\left(\mathcal{E}_T\right) \geq 1 - \frac{5}{T}. \tag{5.33}$$

Naturally, (5.29), (5.30), (5.31) and (5.33) combined together imply the statement of Theorem 5.2.1. We note that constants appearing in the right-hand side of (5.30) and (5.31) can be improved, however, here we are interested in the rates, hence this direction is not pursued.

The remaining part of the proof is split into three parts, where we subsequently show (5.30), (5.31) and (5.33). In the calculations below, we will repeatedly use the following

representation of $\mathbf{Y}_t$, which follows from applying (5.21) recursively:

$$\mathbf{Y}_t = \sum_{j=1}^{t} \varepsilon_j \mathbf{B}^{t-j}\mathbf{u} = \sum_{j=1}^{t} \varepsilon_{t-j+1}\mathbf{B}^{j-1}\mathbf{u}, \quad t = 1, 2, \ldots, T. \tag{5.34}$$

Upper bound for $\|\mathbf{A}_T\|$

The Euclidean norm satisfies $\|\mathbf{A}_T\| = \sup_{\mathbf{v}:\in\mathbb{R}^p \|\mathbf{v}\|=1} |\langle \mathbf{v}, \mathbf{A}_T\rangle|$, therefore we consider inner products $\langle \mathbf{v}, \mathbf{A}_T\rangle$ where $\mathbf{v} \in \mathbb{R}^p$ is any unit vector. By (5.34), we have

$$\langle \mathbf{v}, \mathbf{A}_T\rangle = \sum_{t=1}^{T-1} \langle \mathbf{v}, \mathbf{Y}_t\rangle \varepsilon_{t+1} = \sum_{t=1}^{T-1}\sum_{j=1}^{T-1} \left\langle \mathbf{v}, \mathbf{B}^{j-1}\mathbf{u}\right\rangle \varepsilon_{t-j+1}\varepsilon_{t+1} = \sum_{j=1}^{T-1} \left\langle \mathbf{v}, \mathbf{B}^{j-1}\mathbf{u}\right\rangle a_j,$$

where $a_j = \sum_{t=1}^{T-1} \varepsilon_{t-j+1}\varepsilon_{t+1} = \sum_{t=1}^{T-j} \varepsilon_t\varepsilon_{t+j}$. Lemma 5.5.1 and Lemma 5.5.2 applied to the equation above yields

$$\begin{aligned}
\sum_{j=1}^{T-1} \left\langle v, \mathbf{B}^{j-1}\mathbf{u}\right\rangle a_j &= \frac{1}{2\pi i}\int_{\mathbb{T}} \left(\sum_{j=1}^{T-1} z^{j-1}a_j\right) \left\langle \mathbf{v}, (z\mathbf{I}_p - \mathbf{B})^{-1}\right\rangle dz \\
&= \frac{1}{2\pi i}\int_{\mathbb{T}} \left(\sum_{j=1}^{T-1} z^{j-1}a_j\right) \left(\sum_{j=1}^{p} z^{p-j}v_j\right) q(z)dz \\
&= \frac{1}{2\pi i}\int_{\mathbb{T}} \left(\sum_{j=0}^{T+p-1} z^{j}c_j\right) q(z)dz
\end{aligned}$$

where $q(z) = (z^p b(z^{-1}))^{-1}$ and $c_j = \sum_{i=0}^{j} a_{i+1}v_{p-j+i}$. Integrating by parts, we get

$$\frac{1}{2\pi i}\int_{\mathbb{T}} \left(\sum_{j=0}^{T+p-1} z^{j}c_j\right) q(z)dz = -\frac{1}{2\pi i}\int_{\mathbb{T}} \left(\sum_{j=0}^{T+p-1} z^{j+1}\frac{c_j}{j+1}\right) q'(z)dz.$$

Combining the calculations above and Cauchy's inequality we obtain the following bound.

$$\langle \mathbf{v}, \mathbf{A}_T\rangle \leq \sqrt{\sum_{j=0}^{T+p-1} \left(\frac{c_j}{j+1}\right)^2} \sqrt{\int_{\mathbb{T}} |q'(z)|^2 dm(z)} \tag{5.35}$$

To further bound the first term on the right-hand side of (5.35), we recall that on the event $\mathcal{E}_T$ coefficients $|a_j| \le 32\log(T)\sqrt{T}$, hence

$$\begin{aligned}\sqrt{\sum_{j=0}^{T+p-1}\left(\frac{c_j}{j+1}\right)^2} &= \sqrt{\sum_{j=0}^{T+p-1}\frac{1}{(j+1)^2}\left(\sum_{i=0}^{j}a_{i+1}v_{p-j+i}\right)^2}\\ &\le \max_{j=0,\dots,T+p-1}|a_j|\sqrt{\sum_{j=0}^{T+p-1}\frac{1}{(j+1)^2}\left(\sum_{i=0}^{j}|v_{p-j+i}|\right)^2}\\ &\le 32\log(T)\sqrt{T}\sqrt{\sum_{j=0}^{T+p-1}\frac{\max(j+1,p)}{(j+1)^2}}\\ &\le 32\log(T)\sqrt{(1+\log(T+p))T}.\end{aligned}$$

For the second term in (5.35), we calculate the derivative $q'(z) = -\frac{pz^{p-1}-\sum_{j=1}^{p}(p-j)\beta_j z^{p-j-1}}{(z^p b(z^{-p}))^2}$ and bound

$$\begin{aligned}\sqrt{\int_{\mathbb{T}}|q'(z)|^2 dm(z)} &= \sqrt{\int_{\mathbb{T}}\left|\frac{pz^{p-1}-\sum_{j=1}^{p}(p-j)\beta_j z^{p-j}}{(z^p b(z^{-p}))^2}\right|^2 dm(z)}\\ &\le \frac{\sqrt{\int_{\mathbb{T}}\left|pz^{p-1}-\sum_{j=1}^{p}(p-j-1)\beta_j z^{p-j-1}\right|^2 dm(z)}}{\min_{|z|=1}|(z^p b(z^{-p}))|^2} =\\ &= \underline{b}^{-2}\sqrt{\left(p^2+\sum_{j=1}^{p}(p-j)^2\beta_j^2\right)} \le \underline{b}^{-2}p\sqrt{1+\|\boldsymbol{\beta}\|^2}.\end{aligned}$$

Combining bounds on the two terms, we obtain

$$\langle \mathbf{v}, \mathbf{A}_T\rangle \le \left(32\underline{b}^{-2}\sqrt{1+\|\boldsymbol{\beta}\|^2}\right)p\log(T)\sqrt{(1+\log(T+p))T}.$$

Taking supremum over $\mathbf{v}\in\mathbb{R}^p$ such that $\|\mathbf{v}\| = 1$ proves (5.30).

Lower bound for $\lambda_{min}(\mathbf{C}_T)$

Let $\mathbf{v} = (v_1, \ldots, v_p)'$ be a unit vector in $\mathbb{R}^p$. We begin the proof by establishing the following inequality

$$\langle \mathbf{v}, \mathbf{C}_T \mathbf{v} \rangle \geq \bar{b}^{-2} \sum_{i,j=1}^{p} v_i v_j \sum_{t=1}^{T-1} \varepsilon_{t-j+1} \varepsilon_{t-i+1}, \tag{5.36}$$

where $\varepsilon_t = 0$ for $t \leq 0$ and $\bar{b} = \max_{z \in \mathbb{T}} |b(z)|$. By Theorem 5.5.1 and (5.34), we rewrite the quadratic form on the left-hand side of (5.36) to

$$\langle \mathbf{v}, \mathbf{C}_T \mathbf{v} \rangle = \sum_{t=1}^{T-1} \langle \mathbf{v}, \mathbf{Y}_t \rangle^2 = \int_{\mathbb{T}} \left| \sum_{t=1}^{T-1} \left\langle v, \sum_{j=1}^{t} \varepsilon_j B^{t-j} u \right\rangle z^t \right|^2 dm(z) \tag{5.37}$$

$$= \int_{\mathbb{T}} \left| \sum_{t=1}^{T-1} \sum_{j=1}^{T-1} \varepsilon_j \omega_{t-j} z^t \right|^2 dm(z) \tag{5.38}$$

where $\omega_j = \langle \mathbf{v}, \mathbf{B}^j u \rangle$ for $j \geq 0$, $\omega_j = 0$ for $j < 0$. Changing the order of summation and by a simple substitution we get

$$\sum_{t=1}^{T-1} \sum_{j=1}^{T-1} \varepsilon_j \omega_{t-j} z^t = \sum_{j=1}^{T-1} \varepsilon_j z^j \sum_{t=1}^{T-1} \omega_{t-j} z^{t-j} = \sum_{j=1}^{T-1} \varepsilon_j z^j \sum_{t=0}^{T-j-1} \omega_t z^t. \tag{5.39}$$

Using the definition of ω_j, the fact that all eigenvalues of B have modulus strictly lower than one and Lemma 5.5.2, (5.39) simplifies to

$$\begin{aligned}
\sum_{j=1}^{T-1} \varepsilon_j z^j \sum_{t=0}^{T-j-1} \omega_t z^t &= \sum_{j=1}^{T-1} \varepsilon_j z^j \left\langle \mathbf{v}, (\mathbf{I}_p - (\mathbf{B}z)^{T-j})(\mathbf{I}_p - \mathbf{B}z)^{-1} \right\rangle \\
&= \sum_{j=1}^{T-1} \varepsilon_j \left(z^j \left\langle \mathbf{v}, (\mathbf{I}_p - \mathbf{B}z)^{-1} \right\rangle - z^T \left\langle \mathbf{B}^{T-j} \mathbf{v}, (\mathbf{I}_p - \mathbf{B}z)^{-1} \right\rangle \right) \\
&= b(z)^{-1} \sum_{j=1}^{T-1} \varepsilon_j \left(z^j v(z) - z^T w_j(z) \right),
\end{aligned}$$

where $v(z) = \sum_{k=1}^{p} v_k z_{k-1}$ and $w_j(z) = \sum_{k=1}^{p} (\mathbf{B}^{T-j} v)_k z^{k-1}$ for $j = 0, \dots, n-1$. The equation above, (5.37) and (5.39) combined together imply the following inequality

$$\begin{aligned}\langle \mathbf{v}, \mathbf{C}_T \mathbf{v} \rangle &= \int_{\mathbb{T}} \left| b(z)^{-1} \sum_{j=1}^{T-1} \varepsilon_j \left(z^j v(z) - z^T w_j(z) \right) \right|^2 dm(z) \\ &\geq \bar{b}^{-2} \int_{\mathbb{T}} \left| \sum_{j=1}^{T-1} \varepsilon_j \left(z^j v(z) - z^T w_j(z) \right) \right|^2 dm(z).\end{aligned}$$

Observe that $\sum_{j=1}^{T-1} \varepsilon_j \left(z^j v(z) - z^T w_j(z) \right) = \sum_{j=1}^{T-1} \varepsilon_j \left(z^j v(z) - z^T w_j(z) \right) = \sum_{t=1}^{T+p-1} c_t z^t$ is a trigonometric polynomial, therefore by Theorem 5.5.1 and simple calculations

$$\begin{aligned}\int_{\mathbb{T}} \left| \sum_{j=1}^{T-1} \varepsilon_j \left(z^j v(z) - z^T w_j(z) \right) \right| dm(z) &= \sum_{t=1}^{T+p-1} |c_t|^2 \geq \sum_{t=1}^{n-1} |c_t|^2 = \sum_{t=1}^{T-1} \left(\sum_{j=1}^{p} v_j \varepsilon_{t-j+1} \right)^2 = \\ &= \sum_{i,j=1}^{p} v_j v_i \sum_{t=1}^{T-1} \varepsilon_{t-j+1} \varepsilon_{t-i+1},\end{aligned}$$

which proves (5.36).

We are now in position to bound $\langle \mathbf{v}, \mathbf{C}_T \mathbf{v} \rangle$ from below. Rearranging terms in (5.36) yields

$$\begin{aligned}\langle \mathbf{v}, \mathbf{C}_T \mathbf{v} \rangle &\geq \bar{b}^{-2} \left(\sum_{i=1}^{p} v_i^2 \sum_{t=1}^{n-i} \varepsilon_t^2 + \sum_{1 \leq i < j \leq p} v_i v_j \sum_{t=1}^{T-\max(i,j)} \varepsilon_t \varepsilon_{t+|j-i|} \right) \\ &\geq \bar{b}^{-2} \left(\sum_{t=1}^{T-p} \varepsilon_t^2 \sum_{i=1}^{p} v_i^2 - \max_{1 \leq i < j \leq p} \left| \sum_{t=1}^{T-\max(i,j)} \varepsilon_t \varepsilon_{t+|j-i|} \right| \left(\left(\sum_{i=1}^{p} |v_i| \right)^2 - \sum_{i=1}^{p} v_i^2 \right) \right) \\ &\geq \bar{b}^{-2} \left(\sum_{t=1}^{T-p} \varepsilon_t^2 - (p-1) \max_{1 \leq i < j \leq p} \left| \sum_{t=1}^{T-\max(i,j)} \varepsilon_t \varepsilon_{t+|j-i|} \right| \right).\end{aligned}$$

Now, recalling the definition $\mathcal{E}_T$, we conclude that on this event

$$\begin{aligned}\langle \mathbf{v}, \mathbf{C}_T \mathbf{v} \rangle &\geq \bar{b}^{-2} \left(T - p - 2\sqrt{\log(T)(T-p)} - (p-1) 32 \log(T) \sqrt{T} \right) \\ &\geq \bar{b}^{-2} \left(T - p(1 + 32 \log(T) \sqrt{T}) \right).\end{aligned}$$

Taking infimum over $\mathbf{v} \in \mathbb{R}^p$ such that $\|\mathbf{v}\| = 1$ in the inequality above proves (5.31).

Lower bound for $\mathbb{P}(\mathcal{E}_T)$

Recalling (5.32) and using simple Bonferroni bound, we get

$$\begin{aligned}\mathbb{P}(\mathcal{E}_T^c) &\leq p^2 \max_{1\leq i<j\leq p} \mathbb{P}\left(\left|\sum_{t=1}^{T-\max(i,j)} \varepsilon_t \varepsilon_{t+|i-j|}\right| \geq 32\log(T)\sqrt{T-\max(i,j)}\right) \\ &+ T \max_{1\leq j\leq T} \mathbb{P}\left(\left|\sum_{t=1}^{T-j} \varepsilon_t \varepsilon_{t+j}\right| < 32\log(T)\sqrt{T-j}\right) \\ &+ \mathbb{P}\left(\sum_{t=1}^{T-p} \varepsilon_t^2 > T - p - 2\sqrt{\log(T)(T-p)}\right) \\ &:= p^2 \max_{1\leq i<j\leq p} P_{i,j}^{(1)} + T \max_{1\leq j\leq T} P_j^{(2)} + P^{(3)}.\end{aligned}$$

Lemma 5.5.3 implies that

$$\begin{aligned}P_{i,j}^{(1)} &\leq 2\exp\left(-\frac{1}{8}\frac{(32\log(T))^2}{6+(\sqrt{T-\max(i,j)})^{-1}32\log(T)}\right) \leq 2\exp\left(-2\log(T)\right) = \frac{2}{T^2}, \\ P_j^{(2)} &\leq 2\exp\left(-\frac{1}{8}\frac{(32\log(T))^2}{6+(\sqrt{T-j})^{-1}32\log(T)}\right) \leq 2\exp\left(-2\log(T)\right) = \frac{2}{T^2}.\end{aligned}$$

Moreover, by Lemma 5.5.4, $P^{(3)} \leq \exp(-\log(T)) = \frac{1}{T}$, hence, given that $p^2 < T$, $\mathbb{P}(\mathcal{E}_T^c) \leq \frac{5}{T}$, which finishes the proof.

□

5.6 Proof of Theorem 5.2.2

Proof. The proof follows the structure of the proof of Theorem 4.2.1 given in Section 4.6.2.

Step One.

Consider the event $\left\{\left\|\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}\right\| \leq \kappa_1(\underline{b}/\overline{b})^2 \left\|\boldsymbol{\beta}\right\| \frac{p\log(T)\sqrt{\log(T+p)}}{\sqrt{T}-\kappa_2 p\log(T)}\right\}$ where κ_1, κ_2 are as in Theorem 5.2.1. Assumption (A3) imply that $\underline{b}/\overline{b}$ and $\|\boldsymbol{\beta}\|$ are bounded from above by a constant. Furthermore, by (A2), $p \leq C_1 T^{\theta}$, which implies that

$$\kappa_1(\underline{b}/\overline{b})^2 \left\|\boldsymbol{\beta}\right\| \frac{p\log(T)\sqrt{\log(T+p)}}{\sqrt{T}-\kappa_2 p\log(T)} \leq CT^{a-1/2}(\log(T))^{3/2} =: \lambda_T$$

for some constant $C > 0$ and sufficiently large T. Define now

$$A_T = \left\{\left\|\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}\right\| \leq \lambda_T\right\} \tag{5.40}$$

By Theorem 5.2.1,

$$\mathbb{P}\left(A_T\right) \geq \mathbb{P}\left(\left\|\hat{\boldsymbol{\beta}}_T-\boldsymbol{\beta}\right\| \leq \kappa_1(\underline{b}/\overline{b})^2 \left\|\boldsymbol{\beta}\right\| \frac{p\log(T)\sqrt{\log(T+p)}}{\sqrt{T}-\kappa_2 p\log(T)}\right) \geq 1-\kappa_3 T^{-1}, \tag{5.41}$$

for some constant $\kappa_3 > 0$.

Step Two.

To fix the ideas, for $j = 1, \ldots, q$ we define intervals

$$\mathcal{I}_j^L = (\tau_j - \delta_T/3, \tau_j - \delta_T/6) \tag{5.42}$$

$$\mathcal{I}_j^R = (\tau_j + \delta_T/6, \tau_j + \delta_T/3) \tag{5.43}$$

Recall that F_T^M is the set of M randomly drawn intervals with endpoints in $\{1, \ldots, p\}$. Denote by $[s_1, e_1], \ldots, [s_M, e_M]$ the elements of F_T^M and let

$$D_T^M = \left\{\forall j = 1, \ldots, q, \exists k \in \{1, \ldots, M\}, \text{ s.t. } s_k \times e_k \in \mathcal{I}_j^L \times \mathcal{I}_j^R\right\}. \tag{5.44}$$

We have that

$$\mathbb{P}\left((D_T^M)^c\right) \leq \sum_{j=1}^{q} \Pi_{m=1}^{M}\left(1-\mathbb{P}\left(s_m \times e_m \in \mathcal{I}_j^L \times \mathcal{I}_j^R\right)\right)$$
$$\leq q\left(1-\frac{\delta_T^2}{6^2p^2}\right)^M \leq \frac{p}{\delta_T}\left(1-\frac{\delta_T^2}{36p^2}\right)^M.$$

Therefore, $\mathbb{P}\left(A_T \cap D_T^M\right) \geq 1-\kappa_3 T^{-1} - T\delta_T^{-1}(1-\delta_T^2 p^{-2}/36)^M$.

In the rest of the proof, we assume that A_T and D_T^M all hold. We give the constants as follows:

$$C_1 = 2\sqrt{C_3}+1, \quad C_2 = \frac{1}{\sqrt{6}} - \frac{2\sqrt{2}}{\underline{C}}, \quad C_3 = (4\sqrt{2}+6).$$

But since our main aim is to establish the rate, we chose not to pursue this direction further. In addition, here we need to make sure that $\underline{C}C_2 > C_1$, and thus $C_2\delta_T^{1/2}\underline{f}_T > C_1\sqrt{\log(T)}$, i.e., we can select $\zeta_T \in [C_1\sqrt{\log(T)}, C_2\delta_T^{1/2}\underline{f}_T)$. This is indeed the case because $\underline{C}$ is sufficiently large.

Step Three

We focus on a generic interval $[s,e]$ such that

$$\exists j \in \{1,\ldots,q\},\ \exists k \in \{1,\ldots,M\},\ \text{s.t. } [s_k,e_k] \subset [s,e] \text{ and } s_k \times e_k \in \mathcal{I}_j^L \times \mathcal{I}_j^R \quad (5.45)$$

Fix such an interval $[s,e]$ and let $j \in \{1,\ldots,q\}$ and $k \in \{1,\ldots,M\}$ be such that (5.45) is satisfied. Let $b_k^* = \operatorname{argmax}_{s_k \leq b \leq e_k} \mathcal{C}_{s_k,e_k}^b\left(\hat{\boldsymbol{\beta}}\right)$. By construction, $[s_k,e_k]$ satisfies $\tau_j - s_k + 1 \geq \delta_T/6$ and $e_k - \tau_j > \delta_T/6$. Denote by

$$\mathcal{M}_{s,e} = \left\{m : [s_m,e_m] \in F_T^M, [s_m,e_m] \subset [s,e]\right\};$$
$$\mathcal{O}_{s,e} = \{m \in \mathcal{M}_{s,e} : \max_{s_m \leq b < e_m} \mathcal{C}_{s_m,e_m}^b\left(\hat{\boldsymbol{\beta}}\right) > \zeta_T\}$$

Our first aim is to show that $\mathcal{O}_{s,e}$ is non-empty. This follows from Lemma 4.6.2 and the calculation below.

$$\begin{aligned}\mathcal{C}^{b_k^*}_{s_k,e_k}\left(\hat{\boldsymbol{\beta}}\right) &\geq \mathcal{C}^{\tau_j}_{s_k,e_k}\left(\hat{\boldsymbol{\beta}}\right)\\ &\geq \mathcal{C}^{b_k^*}_{s_k,e_k}\left(\boldsymbol{\beta}\right) - \lambda_T \geq \left(\frac{\delta_T}{6}\right)^{1/2}|\alpha_j\tau_j^{-1}| - \lambda_T \geq \left(\frac{\delta_T}{6}\right)^{1/2}\underline{\alpha}_T - \lambda_T\\ &= \left(\frac{1}{\sqrt{6}} - \frac{\lambda_T}{\delta_T^{1/2}\underline{\alpha}_T}\right)\delta_T^{1/2}\underline{\alpha}_T \geq \left(\frac{1}{\sqrt{6}} - \frac{2\sqrt{2}}{\underline{C}}\right)\delta_T^{1/2}\underline{\alpha}_T = C_2\delta_T^{1/2}\underline{\alpha}_T > \zeta_T.\end{aligned}$$

Let $m^* = \operatorname{argmin}_{m\in\mathcal{O}_{s,e}}(e_m - s_m + 1)$ and $b^* = \operatorname{argmax}_{s_m^*\leq b<e_m^*}\mathcal{C}^{b}_{s_m^*,e_m^*}\left(\hat{\boldsymbol{\beta}}\right)$. Observe that $[s_{m^*}, e_{m^*})$ must contain at least one change-point. Indeed, if that was not the case, we would have $\mathcal{C}^{b}_{s_{m^*},e_{m^*}}\left(\boldsymbol{\beta}\right) = 0$ and

$$\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}\left(\hat{\boldsymbol{\beta}}\right) = |\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}\left(\hat{\boldsymbol{\beta}}\right) - \mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}\left(\boldsymbol{\beta}\right)| \leq \lambda_T < C_1\lambda_T \leq \zeta_T$$

which contradicts $\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}\left(\hat{\boldsymbol{\beta}}\right) > \zeta_T$. On the other hand, $[s_{m^*}, e_{m^*})$ cannot contain more than one change-points, because $e_{m^*} - s_{m^*} + 1 \leq e_k - s_k + 1 \leq \delta_T$, as we picked the *narrowest*-over-threshold interval.

Without loss of generality, assume $\tau_j \in [s_{m^*}, e_{m^*}]$. Denote by $\eta_L = \tau_j - s_{m^*} + 1$ $\eta_R = e_{m^*} - \tau_j$ and $\eta_T = (C_1 - 1)^2\alpha_j^2\tau_j^{-2}\lambda_T^2$, where $\Delta_j^{\mathbf{f}} = |f_{\tau_j+1} - f_{\tau_j}|$. We claim that $\min(\eta_L, \eta_R) > \eta_T$, because $\min(\eta_L, \eta_R) \leq \eta_T$ and Lemma 4.6.2 result in

$$\begin{aligned}\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}\left(\hat{\boldsymbol{\beta}}\right) &\leq \mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}\left(\boldsymbol{\beta}\right) + \lambda_T \leq \mathcal{C}^{\tau_j}_{s_{m^*},e_{m^*}}\left(\boldsymbol{\beta}\right) + \lambda_T \leq \eta_T^{1/2}|\alpha_j\tau_j^{-1}| + \lambda_T\\ &= (C_1 - 1 + 1)\lambda_T = C_1\lambda_T \leq \zeta_T,\end{aligned}$$

which contradicts $\mathcal{C}^{b^*}_{s_{m^*},e_{m^*}}\left(\hat{\boldsymbol{\beta}}\right) > \zeta_T$.

We are now in the position to prove $|b^* - \tau_j| \leq C_3\lambda_T\underline{\alpha}_T^{-2}$. Our aim is to find ϵ_T such

that for any $b \in \{s_{m^*}, s_{m^*}+1, \ldots, e_{m^*}-1\}$ with $|b-\tau_j| > \epsilon_T$, we always have

$$(\mathcal{C}^{\tau_j}_{s_{m^*},e_{m^*}}\left(\hat{\boldsymbol{\beta}}\right))^2 - (\mathcal{C}^{b}_{s_{m^*},e_{m^*}}\left(\hat{\boldsymbol{\beta}}\right))^2 > 0. \tag{5.46}$$

This would then imply that $|b^* - \tau_j| \leq \varepsilon_T$. By expansion and rearranging the terms, we see that (5.46) is equivalent to

$$\begin{aligned}&\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\rangle^2 - \langle \boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\rangle^2 > \langle \hat{\boldsymbol{\beta}}-\boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\rangle^2 - \langle \hat{\boldsymbol{\beta}}-\boldsymbol{\beta}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\rangle^2 \\ &+2\Big\langle \hat{\boldsymbol{\beta}}-\boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\rangle - \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\rangle\Big\rangle. \end{aligned} \tag{5.47}$$

In the following, we assume that $b \geq \tau_j$. The case that $b < \tau_j$ can be handled in a similar fashion. By Lemma 4.6.4, we have

$$\begin{aligned}\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\rangle^2 - \langle \boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\rangle^2 &= (\mathcal{C}^{\tau_j}_{s^*,e^*}(\boldsymbol{\beta}))^2 - (\mathcal{C}^{b}_{s_{m^*},e_{m^*}}(\boldsymbol{\beta}))^2 \\ &= \frac{|b-\tau_j|\eta_L}{|b-\tau_j|+\eta_L}(\alpha_j\tau_j^{-1})^2 := \kappa.\end{aligned}$$

In addition, since A_T holds

$$\begin{aligned}&\langle \hat{\boldsymbol{\beta}}-\boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\rangle^2 - \langle \hat{\boldsymbol{\beta}}-\boldsymbol{\beta}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\rangle^2 \leq \lambda_T^2, \\ &2\Big\langle \hat{\boldsymbol{\beta}}-\boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\rangle - \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\rangle\Big\rangle \\ &\qquad \leq 2\|\boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{b}_{s_{m^*},e_{m^*}}\rangle - \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\langle \boldsymbol{\beta}, \boldsymbol{\psi}^{\tau_j}_{s_{m^*},e_{m^*}}\rangle\|_2\lambda_T = 2\kappa^{1/2}\lambda_T,\end{aligned}$$

where the last equality also comes from Lemma 4.6.4. Consequently, (5.47) can be deducted from the stronger inequality $\kappa - 2\lambda_T\kappa^{1/2} - \lambda_T^2 > 0$. This quadratic inequality is implied by $\kappa > (\sqrt{2}+1)^2\lambda_T^2$, and could be restricted further to

$$\frac{2|b-\tau_j|\eta_L}{|b-\tau_j|+\eta_L} \geq \min(|b-\tau_j|, \eta_L) > (4\sqrt{2}+6)(\underline{\alpha}\tau_j^{-1})^{-2}\lambda_T^2 = C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2. \tag{5.48}$$

But since

$$\eta_L \geq \eta_T = (C_1 - 1)^2(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2 = (2\sqrt{C_3})^2(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2 > C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2,$$

we see that (5.48) is equivalent to $|b - \tau_j| > C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2$. To sum up, $|b^* - \tau_j| > C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2$ would result in (5.46), a contradiction. So we have proved that $|b^* - \tau_j| \leq C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2$.

Step Four

Using the arguments given above which are valid on the event $A_T \cap B_T \cap D_T^M$, we can now proceed with the proof of the theorem as follows. At the start of Algorithm 5.8 we have $s = 1$ and $e = T$ and, provided that $q \geq 1$, condition (5.45) is satisfied. Therefore the algorithm detects a change-point b^* in that interval such that $|b^* - \tau_j| \leq C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2$. By construction, we also have that $|b^* - \tau_j| < 2/3\delta_T$. This in turn implies that for all $l = 1, \ldots, q$ such that $\tau_l \in [s, e]$ and $l \neq j$ we have either $\mathcal{I}_l^L, \mathcal{I}_l^R \subset [s, b^*]$ or $\mathcal{I}_l^L, \mathcal{I}_l^R \subset [b^* + 1, e]$. Therefore (5.45) is satisfied within each segment containing at least one change-point. Note that before all q change points are detected, each change point will not be detected twice. To see this, we suppose that τ_j has already been detected by b, then for all intervals $[s_k, e_k] \subset [\tau_j - C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2 + 1, \tau_j - C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2 + 2/3\delta_T + 1] \cup [\tau_j + C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2 - 2/3\delta_T, \tau_j + C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2]$, Lemma 4.6.2, together with the event A_T, guarantees that

$$\begin{aligned}
\max_{s_k \leq b < e} \mathcal{C}_{s_k,e_k}^b\left(\hat{\boldsymbol{\beta}}\right) &\leq \max_{s \leq b < e} \mathcal{C}_{s_k,e_k}^b\left(\boldsymbol{\beta}\right) + \lambda_T \\
&\leq \sqrt{C_3(\alpha_j\tau_j^{-1})^{-2}\lambda_T^2}\alpha_j\tau_j^{-1} + \sqrt{C_3(\alpha_{j+1}\tau_{j+1}^{-1})^{-2}\lambda_T^2}\alpha_{j+1}\tau_{j+1}^{-1} + \lambda_T \\
&< (2\sqrt{C_3} + 1)\lambda_T = C_1\lambda_T \leq \zeta_T.
\end{aligned}$$

Once all the change-points are detected, we then only need to consider $[s_k, e_k]$ such that

$$[s_k, e_k] \subset [\tau_j - C_3(\alpha_j \tau_j^{-1})^{-2} + 1, \tau_{j+1} + C_3(\alpha_{j+1}\tau_{j+1}^{-1})^{-2}]$$

for $j = 1, \ldots, q$. For such intervals, we have

$$\max_{s_k \le b < e_k} \mathcal{C}^b_{s_k, e_k}\left(\hat{\boldsymbol{\beta}}\right) \le \max_{s \le b < e} \mathcal{C}^b_{s_k, e_k}(\boldsymbol{\beta}) + \lambda_T \le \sqrt{C_3(\alpha_j \tau_j^{-1})^{-2}\lambda_T^2}\alpha_j\tau_j^{-1} + \lambda_T \le C_1\lambda_T \le \zeta_T.$$

Hence the algorithm terminates and no further change-points are detected. □

Chapter 6

Conclusions

Chapters 3, 4 and 5 propose methods to solve statistical problems arising in analysis of three different types of data. Below, we provide a summary of these contributions, as well as a discussion of possible directions for future research.

Chapter 3 introduces the concept of Ranking-Based Variable Selection, as an alternative to variable selection that is achieved through optimisation of a prediction-oriented criterion. We propose the RBVS algorithm, which aims to recover the set of covariates which non-spuriously appears at the top of a chosen variable ranking, and show that it is a consistent procedure within a general statistical framework. In order to address the issue of possible high correlations between the covariates in the linear model, we propose IRVBS, an iterative extension of RVBS, which in our extensive simulation studies consistently outperforms its competitors.

In Chapter 4, we propose the Narrowest-Over-Threshold methodology, a generic framework for detection of multiple generalised change-points in univariate time series. Under the assumption that the noise in the data is i.i.d. Gaussian and for two important scenarios for the type of the change-points, we show that the NOT procedure estimates the number of the change-points consistently and is near-optimal in terms of estimation of their locations. We provide an extensive study of computational aspects related to the

NOT algorithm, demonstrating how to compute its entire solution path in close-to-linear time. A competitive practical performance of the NOT methodology is illustrated in a number of applications to simulated and real-world data.

Chapter 5 introduces Adaptive Multiscale Autoregressive time series models where the conditional mean of the process of interest depends linearly on its averages calculated over unknown timescales. Applying the methodology of Chapter 4, we propose an estimation procedure in order to recover the timescales from the data and establish its theoretical properties. A particularly appealing feature of AMAR models estimated with the proposed method is that they appear to offer a good predictive power in out-of-sample forecasting of high-frequency financial returns, which is illustrated by the application to the data from NYSE TAQ.

All proposed methodologies are accompanied by easy to use software, which is crucial from the practical point of view. Importantly, the R packages implementing the proposed methods use low-level and parallel programming techniques in order to ensure high computational efficiency and are available free of charge.

There is a number of interesting problems related to the methodology of Chapter 3 that can be a topic of future research. In particular, it appears compelling to find a data-adaptive methods for the choice of the subsample size m used in Algorithm 3.3 and 3.4. The work of Götze and Račkauskas (2001) can serve as a starting point for this piece of research. Another interesting avenue is to extend the IRBVS algorithm to the cases where the relationship between the response and the predictors is non-linear. This can be achieved by replacing the original measure $\hat{\omega}_j$ with $\mathrm{E}\,\hat{\omega}_j|(X_{ij},\ i = 1, \ldots, n,\ j \in \mathcal{S})$ at each iteration of Algorithm 3.4. In general, finding $\mathrm{E}\,\hat{\omega}_j|(X_{ij},\ i = 1, \ldots, n,\ j \in \mathcal{S})$ is difficult, however, it can be relatively easily computed in the Generalised Linear Models as shown in Barut et al. (2015).

The Narrowest-Over-Threshold methodology is also open to many possible extensions.

Choosing an appropriate contrast function in Algorithm 4.6, NOT can be easily extended to identify multiple (generalised) change-points in multivariate time series. For example, in the high-dimensional case and if the mean of the data is piecewise-constant, we can use the idea of Cho and Fryzlewicz (2015), who define the contrast as the sum of the component-wise calculated CUSUM statistics (given by (4.6)) that exceed certain threshold. Analogues definition of the contrast function can be used in order to identify the changes in multivariate trends. Another possible solution that can be applied in the multivariate piecewise-constant scenario can be found in Wang and Samworth (2016), who propose to use certain projection technique that transforms the data into one-dimensional vector, preserving the locations of the change-points. In this case, the contrast can be defined in similar fashion to (4.16), i.e. as the inner product of the transformed data and $\boldsymbol{\psi}^b_{s,e}$ defined by (4.5).

Finally, the work of Chapter 5 also provokes many interesting research questions. One of the crucial assumptions in our theoretical results presented in this chapter is that p , i.e. the rate of the AR(p) representation of the considered AMAR(q) process, satisfies $p = o(T^{1/2})$, where T denotes the number of observations. From both theoretical and practical point of view, it is interesting to investigate the possibility of proposing an estimation procedure for the case of p growing faster than $T^{1/2}$ or even the case of $p \sim T$. Some clues how to approach this challenging task can be found in McMurry et al. (2015), who study the problem of estimation of the autocovariance matrix in the $p \sim T$ case. Another possible extension is to modify the AMAR models in order to incorporate the multiscale structure in the volatility of the process. For example, for $t = 1, \ldots, T$ we can consider

$$X_t = \alpha_1 \frac{1}{\tau_1}(X_{t-1} + \ldots + X_{t-\tau_1}) + \ldots + \alpha_q \frac{1}{\tau_q}(X_{t-1} + \ldots + X_{t-\tau_q}) + \sigma_t \varepsilon_t,$$
$$\sigma_t^2 = \gamma_0 + \gamma_1 \frac{1}{\eta_1}(\varepsilon_{t-1}^2 + \ldots + \varepsilon_{t-\eta_1}^2) + \ldots + \gamma_r \frac{1}{\eta_r}(\varepsilon_{t-1}^2 + \ldots + \varepsilon_{t-\eta_r}^2),$$

where $1 \leq \alpha_1 \leq \ldots \leq \alpha_q$, $1 \leq \eta_1 \leq \ldots, \leq \eta_r$ are the unknown timescales and $\alpha_1, \ldots, \alpha_q$, $\gamma_1, \ldots, \gamma_q$ are the coefficients timescale-coefficients and ε_t's are i.i.d. $\mathcal{N}(0, 1)$

References

S. K. Ahn and G. C. Reinsel. Nested reduced-rank autoregressive models for multiple time series. *Journal of the American Statistical Association*, 83:849–856, 1988. 181

Y. Aït-Sahalia and M. W. Brandt. Variable selection for portfolio choice. *The Journal of Finance*, 56:1297–1351, 2001. 22

H. Akaike. Information theory and an extension of the Maximum Likelihood principle. In *Selected Papers of Hirotugu Akaike*, pages 199–213. Springer, 1998. 26

S. Alliney and S. A. Ruzinsky. An algorithm for the minimization of mixed ℓ_1 and ℓ_2 norms with application to Bayesian estimation. *IEEE Transactions on Signal Processing*, 42:618–627, 1994. 26

E. Andreou and E. Ghysels. Detecting multiple breaks in financial market volatility dynamics. *Journal of Applied Econometrics*, 17:579–600, 2002. 31

E. Andreou, E. Ghysels, and A. Kourtellos. Should macroeconomic forecasters use daily financial data and how? *Journal of Business and Economic Statistics*, 31:240–251, 2013. 46

A. Antoniadis, J. Bigot, and S. Lambert-Lacroix. Peaks detection and alignment for mass spectrometry data. *Journal de la Société Française de Statistique*, 151:17–37, 2010. 113

M. A. Arcones and E. Giné. Limit theorems for U-processes. *The Annals of Probability*, 21:1494–1542, 1993. 106

S. Arlot, A. Celisse, et al. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79, 2010. 30

T. B. Arnold and R. J. Tibshirani. Efficient implementations of the generalized Lasso dual path algorithm. *Journal of Computational and Graphical Statistics*, 25:1–27, 2016. 41

A. Aue and L. Horváth. Structural breaks in time series. *Journal of Time Series Analysis*, 34:1–16, 2013. 31

J. Bai and P. Perron. Estimating and testing linear models with multiple structural changes. *Econometrica*, 66:47–78, 1998. 9, 40, 114

J. Bai and P. Perron. Computation and analysis of multiple structural change models. *Journal of Applied Econometrics*, 18:1–22, 2003. 40, 141

J. Bai, E. Ghysels, and J. H. Wright. State space models and MIDAS regressions. *Econometric Reviews*, 32:779–813, 2013. 46

A. E. Barabanov. On strong convergence of the method of least squares. *Avtomatika i Telemekhanika*, 44:119–127, 1983. 186

R. Baranowski. **rbvsGPU**: Ranking-Based Variable Selection on GPU, 2016. URL https://github.com/rbaranowski/rbvsGPU. R package version 1.0. 19, 83, 96

R. Baranowski and P. Fryzlewicz. **wbs**: Wild Binary Segmentation for multiple change-point detection, 2015. URL https://CRAN.R-project.org/package=wbs. R package version 1.3. 141

R. Baranowski and P. Fryzlewicz. **amar**: Adaptive Multiscale Autoregressive time series models, 2016a. URL https://github.com/rbaranowski/amar. R package version 1.00. 20, 182, 188

R. Baranowski and P. Fryzlewicz. Adaptive Multiscale Autoregressive time series models: simulation code, 2016b. URL https://github.com/rbaranowski/amar-num-ex. 184, 193, 196

R. Baranowski, P. Breheny, and I. Turner. **rbvs**: Ranking-Based Variable Selection. CRAN, 2015. URL https://CRAN.R-project.org/package=rbvs. R package version 1.0.2. 19, 51

R. Baranowski, Y. Chen, and P. Fryzlewicz. Narrowest-Over-Threshold detection of multiple change-points and change-point-like features: simulation code, 2016a. URL https://github.com/rbaranowski/not-num-ex. 140, 153

R. Baranowski, Y. Chen, and P. Fryzlewicz. **not**: Narrowest-Over-Threshold change-point detection, 2016b. URL https://CRAN.R-project.org/package=not. R package version 1.0. 20, 119, 137, 139, 140, 155

A. E. Barut. *Variable Selection and Prediction in High Dimensional Problems*. PhD thesis, Princeton University, 2013. URL http://arks.princeton.edu/ark:/88435/dsp01pz50gw21f. 65

E. Barut, J. Fan, and A. Verhasselt. Conditional Sure Independence Screening. *Journal of the American Statistical Association*, 2015. Just-accepted. 224

M. Basseville, I. V. Nikiforov, et al. *Detection of Abrupt Changes: Theory and Application*, volume 104. Prentice Hall Englewood Cliffs, 1993. 31, 33

D. Bates and D. Eddelbuettel. Fast and elegant numerical linear algebra using the **RcppEigen** package. *Journal of Statistical Software*, 52:1–24, 2013. 193

C. Beaulieu, J. Chen, and J. L. Sarmiento. Change-point analysis as a tool to detect abrupt climate variations. *Philosophical Transactions of the Royal Society: Series A (Mathematical, Physical and Engineering Sciences)*, 370:1228–1249, 2012. 31

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 57:289–300, 1995. 28

B. Bercu, A. Touati, et al. Exponential inequalities for self-normalized martingales with applications. *The Annals of Applied Probability*, 18:1848–1869, 2008. 186

D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific Belmont, MA, 1995. 34

P. J. Bickel, J. B. Brown, H. Huang, and Q. Li. An overview of recent developments in genomics and associated statistical methods. *Philosophical Transactions of the Royal Society: Series A (Mathematical, Physical and Engineering Sciences)*, 367:4313–4337, 2009. 22

P. J. Bickel, F. Götze, and W. R. van Zwet. *Resampling Fewer Than n Observations: Gains, Losses, and Remedies for Losses*. Springer, 2012. 58, 64

M. Bogdan, J. K. Ghosh, and R. Doerge. Modifying the Schwarz Bayesian information criterion to locate multiple interacting quantitative trait loci. *Genetics*, 167:989–999, 2004. 26

M. Bogdan, E. van den Berg, C. Sabatti, W. Su, and E. J. Candès. SLOPE – adaptive variable selection via convex optimization. *The Annals of Applied Statistics*, 9:1103, 2015. 27

P. Breheny and J. Huang. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The Annals of Applied Statistics*, 5:232–253, 2011. 67, 101

E. Brodsky and B. S. Darkhovsky. *Nonparametric Methods in Change Point Problems*, volume 243. Springer Science & Business Media, 2013. 31, 42

L. Broemeling. Bayesian procedures for detecting a change in a sequence of random variables. *Metron*, 30:1–14, 1972. 38

L. D. Broemeling. Bayesian inferences about a changing sequence of random variables. *Communications in Statistics – Theory and Methods*, 3:243–255, 1974. 38

C. T. Brownlees and G. M. Gallo. Financial econometric analysis at ultra-high frequency: Data handling concerns. *Computational Statistics and Data Analysis*, 51:2232–2245, 2006. 197

P. Bühlmann, M. Kalisch, and L. Meier. High-dimensional statistics with a view toward applications in biology. *Statistics*, 1:255–278, 2014. 18, 22

P. Bühlmann, M. Kane, and M. van der Laan. *Handbook of Big Data*. Chapman and Hall/CRC, 2016. 23

P. L. Bühlmann, , and S. van de Geer. *Statistics for High-Dimensional Data*. Springer Heidelberg, 2011. 23, 27

N. Cahill, S. Rahmstorf, and A. C. Parnell. Change points of global temperature. *Environmental Research Letters*, 10:084002, 2015. 31, 113

F. Calvori, F. Cipollini, and G. M. Gallo. **TAQMNGR**: Manage tick-by-tick transaction data, 2015. URL http://CRAN.R-project.org/package=TAQMNGR. R package version 2015.2-1. 197

E. Candes and T. Tao. The Dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics*, 35:2313–2351, 2007. 26

E. Carlstein. Nonparametric change-point estimation. *The Annals of Statistics*, 16: 188–197, 1988. 44

H. P. Chan and G. Walther. Detection with the scan and the average likelihood ratio. *Statistica Sinica*, 23:409–428, 2013. 130

J. Chen and Z. Chen. Extended Bayesian information criteria for model selection with large model spaces. *Biometrika*, 95:759–771, 2008. 26

J. Chen and A. K. Gupta. Testing and locating variance changepoints with application to stock prices. *Journal of the American Statistical Association*, 92:739–747, 1997. 43

J. Chen and A. K. Gupta. *Parametric Statistical Change Point Analysis*. Springer Science and Business Media, 2011. 31, 33

H. Chernoff and S. Zacks. Estimating the current mean of a normal distribution which is subjected to changes in time. *The Annals of Mathematical Statistics*, 35:999–1018, 1964. 38

H. Cho. Change-point detection in panel data via double CUSUM statistic. *Electronic Journal of Statistics*, 10:2000–2038, 2016. 36

H. Cho and P. Fryzlewicz. Multiscale interpretation of taut string estimation and its connection to Unbalanced Haar wavelets. *Statistics and Computing*, 21:671–681, 2011. 42

H. Cho and P. Fryzlewicz. High dimensional variable selection via tilting. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74:593–622, 2012a. 48, 80

H. Cho and P. Fryzlewicz. Multiscale and multilevel technique for consistent segmentation of nonstationary time series. *Statistica Sinica*, 22:207–229, 2012b. 36

H. Cho and P. Fryzlewicz. Multiple-change-point detection for high dimensional time series via Sparsified Binary Segmentation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77:475–507, 2015. 36, 225

M. P. Clements and A. B. Galvão. Forecasting US output growth using leading indicators: An appraisal using MIDAS models. *Journal of Applied Econometrics*, 24:1187–1206, 2009. 46

A. Cleynen, G. Rigaill, and M. Koskas. **Segmentor3IsBack**: A fast segmentation algorithm, 2013. URL https://CRAN.R-project.org/package=Segmentor3IsBack. R package version 1.8. 141

L. Dagum and R. Menon. OpenMP: an industry standard API for shared-memory programming. *IEEE Computational Science and Engineering*, 5:46–55, 1998. 139

B. Darkhovskh. A nonparametric method for the a posteriori detection of the "disorder" time of a sequence of independent random variables. *Theory of Probability and Its Applications*, 21:178–183, 1976. 44

W. W. Davis. Robust methods for detection of shifts of the innovation variance of a time series. *Technometrics*, 21:313–320, 1979. 33

C. De Boor. *A Practical Guide to Splines*. Springer, 2001. 113

A. Delaigle and P. Hall. Effect of heavy tails on ultra high dimensional variable ranking methods. *Statistica Sinica*, 22:909–932, 2012. 56

R. A. DeVore. Nonlinear approximation. *Acta Numerica*, 7:51–150, 1998. 120

D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Challenges Lecture*, 2000. 23

D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994. 143

P. Du Jardin. Predicting bankruptcy using neural networks and other classification methods: The influence of variable selection techniques on model accuracy. *Neurocomputing*, 73:2047–2060, 2010. 23

J. Duoandikoetxea. *Fourier Analysis*, volume 29. American Mathematical Society, 2000. 207

C. Erdman and J. W. Emerson. A fast Bayesian change point analysis for the segmentation of microarray data. *Bioinformatics*, 24:2143–2148, 2008. 38

B. T. Ewing and F. Malik. Volatility transmission between gold and oil futures under structural breaks. *International Review of Economics and Finance*, 25:113–121, 2013. 31

J. Fan and Y. Fan. High dimensional classification using features annealed independence rules. *The Annals of Statistics*, 36:2605–2637, 2008. 78

J. Fan and I. Gijbels. *Local Polynomial Modelling and Its Applications: Monographs on Statistics and Applied Probability*. CRC Press, 1996. 113

J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1360, 2001. 28, 49

J. Fan and J. Lv. Sure Independence Screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70:849–911, 2008. 28, 29, 48, 49, 52, 57, 62, 65, 66, 67, 69, 71, 78

J. Fan and J. Lv. A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, 20:101–148, 2010. 21, 23

J. Fan and R. Song. Sure Independence Screening in generalized linear models with NP-dimensionality. *The Annals of Statistics*, 38:3567–3604, 2010. 48

J. Fan, R. Samworth, and Y. Wu. Ultrahigh dimensional feature selection: beyond the linear model. *Journal of Machine Learning Research*, 10:2013–2038, 2009. 97

J. Fan, Y. Feng, and R. Song. Nonparametric independence screening in sparse ultra-high-dimensional additive models. *Journal of the American Statistical Association*, 106:544–557, 2011. 48

J. Fan, Y. Ma, and W. Dai. Nonparametric Independence Screening in sparse ultra-high dimensional varying coefficient models. *Journal of the American Statistical Association*, 109:1270–1284, 2014. 80

Y. Fan and C. Y. Tang. Tuning parameter selection in high dimensional penalized likelihood. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75:531–552, 2013. 30

X. Fang, J. Li, and D. Siegmund. Segmentation and estimation of change-point models. *arXiv preprint arXiv:1608.03032*, 2016. 118

M. A. Ferreira and H. K. Lee. *Multiscale Modeling: a BayesianPerspective*. Springer Science & Business Media, 2007. 44

M. A. Ferreira, M. West, H. K. Lee, and D. M. Higdon. Multi-scale and hidden resolution time series models. *Bayesian Analysis*, 1:947–967, 2006. 9, 45, 181

M. A. Ferreira, A. I. Bertolde, and S. H. Holan. Analysis of economic data with multiscale spatio-temporal models. In *Handbook of Applied Bayesian Analysis*, pages 295–318. Oxford University Press, 2010. 44

L. Forsberg and E. Ghysels. Why do absolute returns predict volatility so well? *Journal of Financial Econometrics*, 5:31–67, 2007. 46

K. Frick, A. Munk, and H. Sieling. Multiscale change point inference. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76:495–580, 2014. 113, 141

J. Friedman, T. Hastie, and R. Tibshirani. *The Elements of Statistical Learning*, volume 1. Springer series in statistics Springer, Berlin, 2001. 30

P. Fryzlewicz. Wild Binary Segmentation for multiple change-point detection. *The Annals of Statistics*, 42:2243–2281, 2014. 36, 38, 114, 116, 130, 141, 172

P. Fryzlewicz. Tail-greedy bottom-up data decompositions and fast multiple change-point detection. *Preprint*, 2016. URL http://stats.lse.ac.uk/fryzlewicz/tguh/tguh.pdf. 38, 157, 159

P. Fryzlewicz and S. Subba Rao. Multiple-change-point detection for auto-regressive conditional heteroscedastic processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76:903–924, 2014. 36

P. Fryzlewicz, T. Sapatinas, and S. S. Rao. A Haar–Fisz technique for locally stationary volatility estimation. *Biometrika*, 93:687–704, 2006. 155

A. R. Gallant and W. A. Fuller. Fitting segmented polynomial regression models whose join points have to be estimated. *Journal of the American Statistical Association*, 68: 144–147, 1973. 39

E. S. Gardner. Exponential smoothing: The state of the art. *Journal of Forecasting*, 4: 1–28, 1985. 197

E. Ghysels, P. Santa-Clara, and R. Valkanov. The MIDAS touch: Mixed data sampling regression models. Technical report, University of North Carolina and UCLA, 2004. 45, 181

E. Ghysels, A. Sinko, and R. Valkanov. MIDAS regressions: Further results and new directions. *Econometric Reviews*, 26:53–90, 2007. 46

G. Ghysels and R. Valkanov. Linear time series processes with mixed data sampling and midas regression models. *Social Science Research Network*, 2006. 46

W. R. Gilks. *Markov Chain Monte Carlo*. Wiley Online Library, 2005. 45

GISTEMP Team. GISS Surface Temperature Analysis (GISTEMP), 2016. URL http://data.giss.nasa.gov/gistemp/. accessed 1-July-2016. 156

D. L. Gordon. The resurrection of Canary Wharf. *Planning Theory and Practice*, 2: 149–168, 2001. 160

F. Götze and A. Račkauskas. Adaptive choice of bootstrap sample sizes. *Lecture Notes – Monograph Series*, 31:286–309, 2001. 64, 224

P. Hall and H. Miller. Using Generalized Correlation to effect variable selection in very high dimensional problems. *Journal of Computational and Graphical Statistics*, 18: 533–550, 2009a. 9, 30, 48, 49, 50, 51, 61, 62, 70, 78

P. Hall and H. Miller. Using the bootstrap to quantify the authority of an empirical ranking. *The Annals of Statistics*, 37:3929–3959, 2009b. 49, 64

P. Hall and J.-H. Xue. On selecting interacting features from high-dimensional data. *Computational Statistics and Data Analysis*, 71:694–708, 2014. 70, 78

F. R. Hampel. The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69:383–393, 1974. 121

J. Hansen, R. Ruedy, M. Sato, and K. Lo. Global surface temperature change. *Reviews of Geophysics*, 48:1–29, 2010. 156

Z. Harchaoui and C. Lévy-Leduc. Multiple change-point estimation with a total variation penalty. *Journal of the American Statistical Association*, 105:1480–1493, 2012. 38

Z. Harchaoui, E. Moulines, and F. R. Bach. Kernel change-point analysis. In *Advances in Neural Information Processing Systems*, pages 609–616, 2009. 44

D. Harrison and D. L. Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5:81–102, 1978. 80

T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: the Lasso and Generalizations*. CRC Press, 2015. 23

D. M. Hawkins. Testing a sequence of observations for a shift in location. *Journal of the American Statistical Association*, 72:180–186, 1977. 33

D. M. Hawkins. Fitting multiple change-point models to data. *Computational Statistics and Data Analysis*, 37:323–341, 2001. 113

D. M. Hawkins and K. Zamba. A change-point model for a shift in variance. *Journal of Quality Technology*, 37:21–31, 2005. 43

K. Haynes, P. Fearnhead, and I. A. Eckley. A computationally efficient nonparametric approach for changepoint detection. *arXiv preprint arXiv:1602.01254*, 2016a. 44, 114, 140

K. Haynes, R. Killick, P. Fearnhead, and I. Eckley. **changepoint.np**: Methods for nonparametric changepoint detection, 2016b. URL https://CRAN.R-project.org/package=changepoint.np. R package version 0.0.2. 140

X. He, L. Wang, and H. G. Hong. Quantile-adaptive model-free variable screening for high-dimensional heterogeneous data. *The Annals of Statistics*, 41:342–369, 2013. 49

A. Hecq, S. Laurent, and F. C. Palm. Common intraday periodicity. *Journal of Financial Econometrics*, 10:325–353, 2012. 198

A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970. 27

L. Horváth. The Maximum Likelihood method for testing changes in the parameters of normal observations. *The Annals of Statistics*, 21:671–680, 1993. 43

T. Hotz and H. Sieling. **stepR**: Fitting step-functions, 2016. URL http://CRAN.R-project.org/package=stepR. R package version 1.0-4. 141

H. Hsieh. Nonparametric tests for scale shift at an unknown time point. *Communications in Statistics – Theory and Methods*, 13:1335–1355, 1984. 43

D. Hsu. Detecting shifts of parameter in gamma sequences with applications to stock price and air traffic flow analysis. *Journal of the American Statistical Association*, 74:31–40, 1979. 43

D.-A. Hsu. Tests for variance shift at an unknown time point. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 26:279–284, 1977. 43

D. J. Hudson. Fitting segmented curves whose join points have to be estimated. *Journal of the American Statistical Association*, 61:1097–1129, 1966. 39

C. Inclan and G. C. Tiao. Use of cumulative sums of squares for retrospective detection of changes of variance. *Journal of the American Statistical Association*, 89:913–923, 1994. 36, 43

C. Ing and C. Wei. Order selection for same-realization predictions in Autoregressive processes. *The Annals of Statistics*, 33:2423–2474, 2005. 189

N. A. James and D. S. Matteson. **ecp**: An R package for nonparametric multiple change point analysis of multivariate data. *Journal of Statistical Software*, 62:1–25, 2014. 141

N. A. James and D. S. Matteson. Change points via probabilistically pruned objectives. *arXiv preprint arXiv:1505.04302*, 2015. 44, 141, 156

V. Jandhyala, S. Fotopoulos, I. MacNeill, and P. Liu. Inference for single and multiple change-points in time series. *Journal of Time Series Analysis*, 2013. 38

C. M. Jarque and A. K. Bera. Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters*, 6:255–259, 1980. 198

M. Kalisch and P. Bühlmann. Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research*, 8:613–636, 2007. 56, 58

Y. Kawahara and M. Sugiyama. Sequential change-point detection based on direct density-ratio estimation. *Statistical Analysis and Data Mining*, 5:114–127, 2012. 33

R. Killick and I. A. Eckley. **changepoint**: An R package for changepoint analysis. *Journal of Statistical Software*, 58:1–19, 2014. 140

R. Killick, I. A. Eckley, K. Ewans, and P. Jonathan. Detection of changes in variance of oceanographic time-series using changepoint analysis. *Ocean Engineering*, 37:1120–1126, 2010. 31

R. Killick, P. Fearnhead, and I. Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107:1590–1598, 2012a. 34, 43, 114, 140

R. Killick, C. Nam, J. Aston, and I. Eckley. The changepoint repository, 2012b. URL http://changepoint.info. 114

R. Killick, I. Eckley, P. Jonathan, et al. A wavelet-based approach for detecting changes in second order structure within nonstationary time series. *Electronic Journal of Statistics*, 7:1167–1183, 2013. 31

S.-J. Kim, K. Koh, S. Boyd, and D. Gorinevsky. ℓ_1 trend filtering. *SIAM Review*, 51: 339–360, 2009. 41, 141

I. Koch. On the asymptotic performance of median smoothers in image analysis and nonparametric regression. *The Annals of Statistics*, 24:1648–1666, 1996. 113

E. D. Kolaczyk. Nonparametric estimation of gamma-ray burst intensities using Haar wavelets. *The Astrophysical Journal*, 483:340–349, 1997. 113

K. Korkas and P. Fryzlewicz. Multiple change-point detection for non-stationary time series using Wild Binary Segmentation. *Preprint*, 2016. URL http://stats.lse.ac.uk/fryzlewicz/WBS_LSW/WBS_LSW.pdf. 38

D. Korobilis. Var forecasting using bayesian variable selection. *Journal of Applied Econometrics*, 28:204–230, 2013. 22

T. Lai and C. Wei. Asymptotic properties of general autoregressive models and strong consistency of least-squares estimates of their parameters. *Journal of Multivariate Analysis*, 13:1–23, 1983. 186

T. L. Lai. *Sequential Analysis*. Wiley Online Library, 2001. 33

B. Laurent and P. Massart. Adaptive estimation of a quadratic functional by model selection. *The Annals of Statistics*, 28:1302–1338, 2000. 210

M. Lavielle. Using penalized contrasts for the change-point problem. *Signal Procesing*, 85:1501–1510, 2005. 113

C.-B. Lee. Estimating the number of change points in exponential families distributions. *Scandinavian Journal of Statistics*, 24:201–210, 1997. 113

F. Leisch and E. Dimitriadou. **mlbench**: Machine learning benchmark problems, 2010. URL https://CRAN.R-project.org/package=mlbench. R package version 2.1-1. 80

F. Leonardi and P. Bühlmann. Computationally efficient change point detection for high-dimensional regression. *arXiv preprint arXiv:1601.03704*, 2016. 40

G. Li, H. Peng, J. Zhang, and L. Zhu. Robust rank correlation based screening. *The Annals of Statistics*, 40:1846–1877, 2012a. 30, 49, 56, 66, 71

R. Li, W. Zhong, and L. Zhu. Feature screening via distance correlation learning. *Journal of the American Statistical Association*, 107:1129–1139, 2012b. 30, 49, 56, 67

K. Lin, J. Sharpnack, A. Rinaldo, and R. J. Tibshirani. Approximate recovery in changepoint problems, from ℓ_2 estimation error rates. *arXiv preprint arXiv:1606.06746*, 2016. 38, 41, 114

J. Liu, W. Zhong, and R. Li. A selective overview of feature screening for ultrahigh-dimensional data. *Science China Mathematics*, 58:1–22, 2015. 30, 49

J. Lokhorst. The Lasso and generalised linear models. *Honors Project, The University of Adelaide, Australia*, 1999. 27

D. Luebke. CUDA: Scalable parallel programming for high-performance scientific computing. In *5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 836–838. IEEE, 2008. 83

R. Maidstone, T. Hocking, G. Rigaill, and P. Fearnhead. On optimal multiple changepoint algorithms for large data. *Statistics and Computing*, 2016. 34

E. Mammen and S. van de Geer. Locally adaptive regression splines. *The Annals of Statistics*, 25:387–413, 1997. 41

D. S. Matteson and N. A. James. A nonparametric approach for multiple change point analysis of multivariate data. *Journal of the American Statistical Association*, 109: 334–345, 2014. 36, 44

P. McCullagh and J. A. Nelder. *Generalized Linear Models*. London England Chapman and Hall, 1989. 24

T. L. McMurry, D. N. Politis, et al. High-dimensional autocovariance matrices and optimal linear prediction. *Electronic Journal of Statistics*, 9:753–788, 2015. 225

R. McTaggart, G. Daroczi, and C. Leung. **Quandl**: API wrapper for Quandl.com, 2016. URL https://CRAN.R-project.org/package=Quandl. R package version 2.8.0. 153

N. Meinshausen and P. Bühlmann. Stability Selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72:417–473, 2010. 27, 30, 50, 52, 57, 61, 62, 67, 69, 78, 83

U. Menzefricke. A Bayesian analysis of a change in the precision of a sequence of independent normal random variables at an unknown time point. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 30:141–146, 1981. 43

T. Mikosch and C. Stărică. Nonstationarities in financial time series, the long-range dependence, and the IGARCH effects. *Review of Economics and Statistics*, 86:378–390, 2004. 155

G. Nason. *Wavelet Methods in Statistics with R*. Springer Science and Business Media, 2010. 45

A. B. Olshen, E. Venkatraman, R. Lucito, and M. Wigler. Circular Binary Segmentation for the analysis of array-based DNA copy number data. *Biostatistics*, 5:557–572, 2004. 31, 36, 113, 114

H. C. Ombao, J. A. Raz, R. von Sachs, and B. A. Malow. Automatic statistical analysis of bivariate nonstationary time series. *Journal of the American Statistical Association*, 96:543–560, 2001. 31

R. K. Pace and O. W. Gilley. Using the spatial configuration of the data to improve estimation. *Journal of Real Estate Finance and Economics*, 14:333–340, 1997. 81

M. Y. Park and T. Hastie. ℓ_1-regularization path algorithm for Generalized Linear Models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69:659–677, 2007. 27

A. Pettitt. A non-parametric approach to the change-point problem. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28:126–135, 1979. 44

B. Pickering. *Changepoint Detection for Acoustic Sensing Signals*. PhD thesis, Lancaster University, 2016. URL http://eprints.lancs.ac.uk/81171/1/2016PickeringPhd.pdf. 31

N. Pochet, F. De Smet, J. A. K. Suykens, and B. L. R. De Moor. Systematic benchmarking of microarray data classification: assessing the role of non-linearity and dimensionality reduction. *Bioinformatics*, 20:3185–3195, 2004. 78

D. Politis. A normalizing and variance-stabilizing transformation for financial time series. In *Recent Advances and Trends in Nonparametric Statistics*, pages 335–347. Elsevier, 2003. 197

D. N. Politis. Model-free versus model-based volatility prediction. *Journal of Financial Econometrics*, 5:358–359, 2007. 197

D. N. Politis and D. D. Thomakos. NoVaS transformations: flexible inference for volatility forecasting. In *Recent Advances and Future Directions in Causality, Prediction, and Specification Analysis*, pages 489–525. Springer, 2013. 198

P. Radchenko and G. M. James. Forward-lasso with adaptive shrinkage. *The Annals of Applied Statistics*, 5:427–448, 2010. 80

M. Raimondo. Minimax estimation of sharp change points. *The Annals of Statistics*, 26: 1379–1397, 1998. 130

A. Rakotomamonjy. Variable selection using svm-based criteria. *Journal of machine learning research*, 3:1357–1370, 2003. 23

G. Reinsel. Some results on Multivariate Autoregressive Index models. *Biometrika*, 70: 145–156, 1983. 181

G. Rigaill. Pruned dynamic programming for optimal multiple change-point detection. *arXiv preprint arXiv:1004.0887*, 2010. 34, 141

M. J. Rosa, L. Portugal, T. Hahn, A. J. Fallgatter, M. I. Garrido, J. Shawe-Taylor, and J. Mourao-Miranda. Sparse network-based models for patient classification using fmri. *Neuroimage*, 105:493–506, 2015. 22

G. J. Ross and N. M. Adams. Two nonparametric control charts for detecting arbitrary distribution changes. *Journal of Quality Technology*, 44:102–116, 2012. 44

E. Ruggieri. A Bayesian approach to detecting change points in climatic records. *International Journal of Climatology*, 33:520–528, 2013. 31, 40, 156, 157

E. Ruggieri and M. Antonellis. An exact approach to Bayesian sequential change point detection. *Computational Statistics and Data Analysis*, 97:71–86, 2016. 40

D. F. Saldana and Y. Feng. **SIS**: An R package for Sure Independence Screening in ultrahigh dimensional statistical models. *Preprint*, 2014. URL http://www.stat.columbia.edu/~diego/PapersandDraft/SIS.pdf. 67, 101

A. L. Schröder. *Methods for Change-Point Detection with Additional Interpretability*. PhD thesis, London School of Economics and Political Science, 2016. 43

A. L. Schröder and P. Fryzlewicz. Adaptive trend estimation in financial time series via multiscale change-point-induced basis recovery. *Statistics and Its Interface*, 6:449–461, 2013. 31, 113

A. L. Schröder and H. Ombao. FreSpeD: Frequency-specific change-point detection in epileptic seizure multi-channel EEG data. *Preprint*, 2015. 31, 36

A. Schwartzman, R. F. Dougherty, J. Lee, D. Ghahremani, and J. E. Taylor. Empirical null and false discovery rate analysis in neuroimaging. *Neuroimage*, 44:71–82, 2009. 22

G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6:461–464, 1978. 26, 34

S. L. Scott and H. R. Varian. Bayesian variable selection for nowcasting economic time series. Technical report, National Bureau of Economic Research, 2013. 22

A. Sen and M. S. Srivastava. On tests for detecting change in mean. *The Annals of Statistics*, 3:98–108, 1975. 33

R. D. Shah and R. J. Samworth. Variable selection with error control: another look at Stability Selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75:55–80, 2013. 50

X. Shao and J. Zhang. Martingale difference correlation and its use in high-dimensional variable screening. *Journal of the American Statistical Association*, 109:1302–1318, 2014. 49

S. K. Shevade and S. S. Keerthi. A simple and efficient algorithm for gene selection using sparse logistic regression. *Bioinformatics*, 19:2246–2253, 2003. 27

J. S. Simonoff. *Smoothing Methods in Statistics*. Springer, 2012. 113

D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, and J. P. Richie. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1:203–209, 2002. 22, 78

J. H. Stock and M. W. Watson. Forecasting using principal components from a large number of predictors. *Journal of the American statistical association*, 97:1167–1179, 2002. 22

W. Su, E. Candes, et al. SLOPE is adaptive to unknown sparsity and asymptotically minimax. *The Annals of Statistics*, 44:1038–1068, 2016. 28

W. Sweldens and P. Schröder. Building your own wavelets at home. In *Wavelets in the Geosciences*, pages 72–107. Springer, 2000. 116

G. J. Szekely and M. L. Rizzo. Hierarchical clustering via joint between-within distances: Extending ward's minimum variance method. *Journal of Classification*, 22:151–183, 2005. 44

G. J. Székely and M. L. Rizzo. Brownian distance covariance. *The Annals of Applied Statistics*, 3:1236–1265, 2009. 49

A. B. Taylor and R. J. Tibshirani. **genlasso**: Path algorithm for generalized Lasso problems, 2014. URL https://CRAN.R-project.org/package=genlasso. R package version 1.3. 38, 141

J. W. Taylor. Volatility forecasting with smooth transition exponential smoothing. *International Journal of Forecasting*, 20:273–286, 2004. 198

S. Tian, Y. Yu, and H. Guo. Variable selection and corporate bankruptcy forecasts. *Journal of Banking and Finance*, 52:89–100, 2015. 22

R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 58:267–288, 1996. 26, 49, 66

R. Tibshirani. Regression shrinkage and selection via the Lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73:273–282, 2011. 27

R. J. Tibshirani. Adaptive piecewise polynomial estimation via trend filtering. *The Annals of Statistics*, 42:285–323, 2014. 41, 114

F. Topsøe. Some bounds for the logarithmic function. *Inequality Theory and Applications*, 4:137–151, 2004. 110

UK Land Registry. UK house price index, 2016. URL http://landregistry.data.gov.uk/app/ukhpi. [Online; accessed 1-August-2016]. 157

P. A. Valdés-Sosa, J. M. Sánchez-Bornot, A. Lage-Castellanos, M. Vega-Hernández, J. Bosch-Bayard, L. Melie-García, and E. Canales-Rodríguez. Estimating brain functional connectivity with sparse multivariate autoregression. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 360:969–981, 2005. 22

S. A. Van de Geer. High-dimensional generalized linear models and the Lasso. *The Annals of Statistics*, 36:614–645, 2008. 27

E. Venkatraman and A. B. Olshen. A faster circular binary segmentation algorithm for the analysis of array CGH data. *Bioinformatics*, 23:657–663, 2007. 36

E. S. Venkatraman. *Consistency results in multiple change-point problems*. PhD thesis, Stanford University, 1992. URL https://statistics.stanford.edu/research/consistency-results-multiple-change-point-problems. 36, 118

R. Vershynin. A simple decoupling inequality in probability theory. Technical report, University of Michigan, 2011. URL http://www-personal.umich.edu/~romanv/papers/decoupling-simple.pdf. 209

L. Vostrikova. Detection of the disorder in multidimensional random processes. *Soviet Mathematics – Doklady*, 259:270–274, 1981. 34, 35, 114

M. P. Wand and M. C. Jones. *Kernel Smoothing*. CRC Press, 1994. 113

H. Wang, R. Killick, and X. Fu. Distributional change of monthly precipitation due to climate change: comprehensive examination of dataset in southeastern united states. *Hydrological Processes*, 28:5212–5219, 2014. 31

T. Wang and R. J. Samworth. High-dimensional changepoint estimation via sparse projection. *arXiv preprint arXiv:1606.06246*, 2016. 36, 225

K. Worsley. Confidence regions and tests for a change-point in a sequence of exponential family random variables. *Biometrika*, 73:91–104, 1986. 33

Y. Wu. *Inference for Change Point and Post Change Means after a CUSUM Test*, volume 180. Springer Science & Business Media, 2007. 31

Y.-C. Yao. Estimating the number of change-points via Schwarz' criterion. *Statistics and Probability Letters*, 6:181–189, 1988. 33, 113

Y.-C. Yao and S. T. Au. Least-squares estimation of a step function. *Sankhya: The Indian Journal of Statistics, Series A*, 51:370–381, 1989. 113

S. Zacks. Classical and Bayesian approaches to the change-point problem : fixed sample and sequential procedures. *Statistique et Analyse des Données*, 7:48–81, 1982. 39

A. Zeileis, F. Leisch, K. Hornik, and C. Kleiber. **strucchange**: An R package for testing for structural change in linear regression models. *Journal of Statistical Software*, 7: 1–38, 2002. 141

C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38:894–942, 2010. 28, 49, 66

C.-H. Zhang and J. Huang. The sparsity and bias of the Lasso selection in high-dimensional linear regression. *The Annals of Statistics*, 36:1567–1594, 2008. 56

N. R. Zhang and D. O. Siegmund. A modified Bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics*, 63:22–32, 2007. 31, 34

L.-P. Zhu, L. Li, R. Li, and L.-X. Zhu. Model-free feature screening for ultrahigh-dimensional data. *Journal of the American Statistical Association*, 106:1464–1475, 2011. 49, 66

C. Zou and Lancezhange. **nmcdr**: Non-parametric multiple change-points detection, 2014. URL https://CRAN.R-project.org/package=nmcdr. R package version 0.3.0. 141, 155

C. Zou, G. Yin, L. Feng, and Z. Wang. Nonparametric maximum likelihood approach to multiple change-point problems. *The Annals of Statistics*, 42:970–1002, 2014. 44, 141, 155

H. Zou. The adaptive Lasso and its oracle properties. *Journal of the American Statistical Association*, 101:1418–1429, 2006. 27

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67:301–320, 2005. 27