

High-Dimensional Methods for Timely and Interpretable Economic Statistics

Luke Mosley, B.Sc.(Hons.), M.Res



Submitted for the degree of Doctor of Philosophy at
Lancaster University.

June 2023

Abstract

The need for a timely and detailed measure of the economy has become increasingly important in recent years due to sudden periods of economic growth and decline. It has become imperative for policymakers and practitioners to accurately capture short-term dynamics and trajectories as we emerge from shocks. However, relying on traditional survey-centric economic indicators presents challenges, as they often lack granularity and are published with significant delays. For this reason, National Statistics Institutes, such as the UK's Office for National Statistics, are undergoing a transformation by incorporating administrative and alternative data-sources as key components of their statistical frameworks. These data sources offer several benefits, including timeliness, cost-effectiveness, reliability, and comprehensive coverage of economic processes. While a sense of awareness for these new data-sources is being established, the statistical methodology to calibrate insights from these sources remains under-developed.

This thesis develops new methodology for producing timely and interpretable economic statistics. Specifically, it proposes novel techniques to address the crucial tasks of temporal disaggregation and nowcasting when dealing with large volumes of high-dimensional data. The frameworks developed in this research leverage the growing body of literature on sparsity-inducing regularisers, allowing for interpretable measures of relevant data-sources. Acknowledging the importance of reproducibility, the work presented is developed into R packages, and a wide range of case studies are extensively covered to showcase applicability of the proposed methodologies.

Acknowledgements

Firstly, I would like to thank my supervisor Alex; someone I view more as a friend rather than a ‘boss’. I am forever grateful for your help, support, and passion for all things LASSO. It has been quite the adventure. From early trips down to Newport, to conferences in Glasgow, London and Paris, these experiences I will never forget and I look forward to keeping in touch.

I would also like to acknowledge other colleagues that have made this PhD possible. Thanks to Idris for your support and guidance in the first half of my PhD. I appreciate the teams call chats during lockdown. Thanks to Hannah, Ben and Duncan at the ONS for making me feel so welcome and introducing me to personnel at the ONS. Thanks to my Italian colleagues Filippo and Barbara for an interesting research collaboration, an engaging nowcasting competition and a fun trip up the Eiffel tower to top it off. Thanks to Kaveh for showing me the ropes on how to develop R packages and more importantly introducing me to the culinary delights of London. Thanks to Tak-Shing for your recent amusing collaboration. I am also very grateful for the funding provided by the STOR-i CDT, EPSRC, ESRC and ONS.

Thank you to STOR-i for making my PhD experience so enjoyable; we really are spoilt with all the cheese, wine, coffee and away days you provide us. I am grateful for the friends I have made here, especially the Barton Road boys for easy-to-win games of FIFA and relaxing rounds of golf. A big shout out to Grad FC. I have had 4 wonderful seasons playing for you and met some great friends along the way; #UTMG. To the

Bolton boys, all I can say is: hoorah to the shoorah my fellow crocs. Thanks also to everyone else who has been with me on this journey.

Last but not least, special thanks to Mum, Dad, Abigail and Callum for making it the most loving home anyone could wish for. Thanks to mum for everything you do and being the best teacher. Thanks to dad for your wise advice and being a masterful wordsmith. Thanks to Abi for our coffee meet-ups and music experiences. Thanks to Callum for your sense of humour and competitive battles.

Declaration

I declare that the work in this thesis has been done by myself and has not been submitted elsewhere for the award of any other degree.

A version of Chapter 3 has been published as Mosley, L., Eckley, I. A., & Gibberd, A. (2022). Sparse temporal disaggregation. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 185(4), 2203-2233.

Case Study 2 of Chapter 3 has been submitted for publication as Guardabascio, B., Moauro, F., & Mosley, L (2022). Indirect estimation of a monthly service turnover indicator in Italy.

A version of Chapter 4 has been submitted for publication as Mosley, L., Chan, T. S. T., & Gibberd, A. (2023). The Sparse Dynamic Factor Model: A Regularised Quasi-Maximum Likelihood Approach. arXiv preprint arXiv:2303.11892.

Luke Mosley

Contents

Abstract	I
Acknowledgements	II
Declaration	IV
Contents	V
List of Figures	X
List of Tables	XVIII
1 Introduction	1
1.1 Main Contribution and Thesis Structure	3
2 Literature Review	6
2.1 Temporal Disaggregation	10
2.1.1 Introduction	10
2.1.2 Temporal Aggregation Constraint	13
2.1.3 Smoothing-based Methods	14
2.1.4 Regression-based Methods with Indicator Series	17
2.1.5 Multivariate Methods with Contemporaneous Constraints	21
2.2 Nowcasting	24

<i>CONTENTS</i>	VI
2.2.1 Introduction	25
2.2.2 Bridge Equations	28
2.2.3 Mixed Data Sampling	31
2.2.4 Dynamic Factor Models	34
2.2.5 Nowcasting with Dynamic Factor Models	39
3 Sparse Temporal Disaggregation	43
3.1 Introduction	44
3.2 Shortcomings of the Classic Temporal Disaggregation Framework	47
3.3 Sparse Temporal Disaggregation	50
3.3.1 A General Regularised M-estimation Framework for Temporal Disaggregation	50
3.3.2 The ℓ_1 -spTD method	52
3.3.3 Dealing with Correlated Indicator Series	58
3.4 Simulation Study	61
3.4.1 Estimating the True Quarterly Series	62
3.4.2 Recovery of the True Regression Coefficient	65
3.4.3 Distribution of AR(1) Parameter Estimate	69
3.4.4 Correlated Design Setting	72
3.5 Case Study 1 - Temporal Disaggregation of UK Gross Domestic Product	73
3.5.1 Results	76
3.6 Case Study 2 - Indirect Estimation of a Monthly Service Turnover Indicator in Italy	80
3.6.1 Results	83
3.7 Conclusion	87
4 The Sparse Dynamic Factor Model: A Regularised Quasi-Maximum Likelihood Approach	90

4.1	Introduction	91
4.2	The Sparse DFM	95
4.2.1	Consistency and Pervasiveness	96
4.2.2	Identifiability	97
4.3	Estimation	99
4.3.1	A Regularised Expectation Maximisation Algorithm	100
4.3.2	Parameter Tuning	107
4.3.3	Implementation	109
4.4	Simulation Study	109
4.4.1	Recovery of Sparse Loadings	111
4.4.2	Forecasting Performance	114
4.4.3	Computational Efficiency	115
4.5	Case Study 1 - Nowcasting UK Trade in Goods	116
4.5.1	Data	117
4.5.2	Interpreting Factor Structure	119
4.5.3	Pseudo-Real Time Nowcasting Exercise	121
4.6	Case Study 2 - The Dynamic Factors of Energy Consumption Around a University Campus	124
4.6.1	Data and Preprocessing	124
4.6.2	Factor Estimates and Interpretation	125
4.6.3	Forecasting Performance	127
4.7	Conclusion	129
5	Timely Estimates of UK Trade in Services: A Sparse Dynamic Factor Model Approach	134
5.1	Introduction	135
5.2	Problem Set Up	140
5.3	Joint state-space Framework	142

5.3.1	Initialisation	145
5.3.2	Estimation and Convergence	147
5.4	Simulation Study	147
5.5	Monthly Estimation of the Subdivisions of the Service Sector	149
5.6	Examining UK Trade Throughout the Brexit Transition Period	157
5.6.1	Introduction	157
5.6.2	Interrupted Time Series Analysis	160
5.6.3	Results	163
5.7	Conclusion	165
6	Temporal Disaggregation and Nowcasting with Sparsity Constraints in R	168
6.1	DisaggregateTS: An R Package for High-Dimensional Temporal Disaggregation	169
6.1.1	Introduction	169
6.1.2	Package Structure	170
6.1.3	Synthetic Example	173
6.2	sparseDFM: An R Package to Estimate Dynamic Factor Models with Sparse Loadings	178
6.2.1	Introduction	178
6.2.2	Package Structure	180
6.2.3	Empirical Application: Exploring UK Inflation	184
7	Conclusions and Further Work	189
A	Appendix to Chapter 2	197
A.1	Quadratic Optimisation with a Linear Equality Constraint for Temporal Disaggregation	197
A.2	Multivariate and Univariate Kalman Filter and Smoother Equations	199

A.2.1	Multivariate KFS Equations - Shumway and Stoffer (1982) . . .	200
A.2.2	Univariate KFS Equations - Koopman and Durbin (2000)	202
B	Appendix to Chapter 3	206
B.1	Indicator Data Description for Section 3.6	206
B.2	Indicators Selected in stable ℓ_1 -spTD for Section 3.6	206
C	Appendix to Chapter 4	211
C.1	Maximisation Step Derivations for Section 4.3.1	211
C.2	ADMM Algorithm Derivations for Section 4.3.1	216
D	Appendix to Chapter 6	218
D.1	An Extension for AR(1) Idiosyncratic Errors	218
D.1.1	Effect on Kalman Filter and Smoother Computation	219
D.2	Table of Functions of the <code>sparseDFM</code> Package	220
	Bibliography	222

List of Figures

2.0.1	UK GDP Quarterly National Accounts time series: chained volume measures (CVM) and seasonally adjusted (SA). Key dates highlighted: [1] June 23, 2016 Brexit referendum; [2] March 29, 2017 Article 50 of the Treaty on European Union invoked; [3] March 23, 2020 UK lockdown due to COVID-19 pandemic; [4] 30 December, 2020 UK-EU Trade and Cooperation Agreement signed; [5] Autumn, 2021 Energy crisis begins; [6] 24 February, 2022 Russia invade Ukraine.	7
2.1.1	Monthly estimate of UK GDP using Chow-Lin temporal disaggregation with Index of Production and Services as indicator series. Key dates highlighted: [1] June 23, 2016 Brexit referendum; [2] March 29, 2017 Article 50 of the Treaty on European Union invoked; [3] March 23, 2020 UK lockdown due to COVID-19 pandemic; [4] 30 December, 2020 UK-EU Trade and Cooperation Agreement signed; [5] Autumn, 2021 Energy crisis begins; [6] 24 February, 2022 Russia invade Ukraine. . .	12
2.2.1	Visual representation for the task of nowcasting Q1 2023 GDP.	26
3.3.1	Random walk synthetic data example using $n_l = 100$, $p = 150$, $\rho = 0.5$ and true predictor coefficient β with the first 10 values equal to 5 and the remaining 140 equal to 0. (a) Solution paths generated via LARS, plot shows $\hat{\beta}(\lambda^{(l)})$ as a function of step l ; (b) BIC plot for each step of the LARS algorithm indicating the best step and cut-off point.	55

3.4.1 Boxplots comparing RMSE values of $\hat{\mathbf{y}}$ for CL, spTD and spTD_RF in the stationary i.i.d. standard Normal indicator series experiment. Left, middle and right columns represent $p = 30, 90$ and 150 respectively. Top, middle and bottom rows represent AR(1) parameter $\rho = 0.2, 0.5$ and 0.8 respectively. 63

3.4.2 Boxplots comparing RMSE values of $\hat{\mathbf{y}}$ for CL, spTD and spTD_RF in the non-stationary random walk indicator series experiment. Left, middle and right columns represent $p = 30, 90$ and 150 respectively. Top, middle and bottom rows represent AR(1) parameter $\rho = 0.2, 0.5$ and 0.8 respectively. 64

3.4.3 Boxplots for the estimates $\hat{\beta}_1, \dots, \hat{\beta}_p$ using CL, spTD and spTD_RF (left to right respectively) with $\rho = 0.5$ and $p = 30$ (top row) and $p = 90$ (bottom row) with stationary indicator series. 68

3.4.4 Violin plots showing the distribution of $\hat{\rho}$ for CL, spTD and spTD_RF using stationary indicator series. Dimension $p = 30, 90$ and 150 from left to right and true AR parameter ρ indicated by horizontal dashed line. 70

3.4.5 Violin plots showing the distribution of $\hat{\rho}$ for CL, spTD and spTD_RF using non-stationary indicator series. Dimension $p = 30, 90$ and 150 from left to right and true AR parameter ρ indicated by horizontal dashed line. 71

3.4.6 Boxplots for the estimates $\hat{\beta}_1, \dots, \hat{\beta}_p$ using CL, spTD and spTD_RF (left to right respectively) with $\rho = 0.5$ and $p = 30$ (top row) and $p = 90$ (bottom row) with correlated (0.9) indicator series. 73

3.5.1	Comparison of estimated sequences and the ONS' monthly GDP index. a) Estimated high-frequency (monthly) series for 2008-2020. b-d) Absolute difference between the index and the estimate for each month (millions of pounds), b) CL with 10 indicators, c) ℓ_1 -spTD with 10 possible indicators, d) ℓ_1 -spTD with 97 possible indicators.	77
3.6.1	Italian quarterly turnover index in transports	82
3.6.2	Chart of the indicators selected at each extension of the sample window. The y-axis displays the date observed from 2016:Q4 up to 2021:Q4. Refer to Table 3.6.1 for the indicator indexed on the x-axis. The colours are representative of the membership data-set: red = BS, yellow = AA, orange = IT.	84
4.2.1	The impact of rotation on the function $\mathcal{L}(\tilde{\Lambda}(\theta)) + \ \tilde{\Lambda}(\theta)\ _q$ for $q = 0, 1, 2$. In the case of $q = 0, 1$, the set of feasible Λ_0 from 4.2.2 is restricted to the points $\theta \in \{0, \pm\frac{1}{2}\pi, \pm\pi\}$ corresponding to either swapping columns, or flipping signs.	99
4.4.1	Median log-MAE score (top panel) and median F1 score (bottom panel) for recovering factor loadings across 100 experiments with a shaded confidence band of the 25th and 75th percentile. The plots represent a setting with a fixed $n = 100$ and varying number of variables p and where the cross-correlation parameter in the VAR(1) process is set to $\rho = 0$ (left plot), $\rho = 0.6$ (middle plot) and $\rho = 0.9$ (right plot).	113
4.4.2	Average MAE score forecasting, as a function of the level of missing data in the last sample. From left-right: $\rho = 0$, $\rho = 0.6$, $\rho = 0.9$. Plot indicates the 50th percentile of performance across 100 experiments with $n = 100$, $p = 64$	115

4.4.3	Summary of computational cost. Left: as a function of n , with fixed $p = 24$. Right: as a function of p , with fixed $n = 100$. Average performance across 10 experiments.	116
4.5.1	Breakdown of the 9 trade in goods target series.	117
4.5.2	Top left: Information Criteria (4.3.13) as a function of number of retained factors r . Top right: Proportion of variance explained, based on PCA applied to the scaled and pre-processed data set. Bottom: BIC as a function of α , vertical line indicates minimiser and the α used in the subsequent analysis.	120
4.5.3	Factor loadings estimates for all 4 factors using a regular DFM and a sparse DFM. Colour coded by indicator series data group of Table 4.5.1.	122
4.5.4	Estimated factors (black) with original data (grey) as a function of time using the optimal $\alpha \approx 2.984$ chosen according to BIC.	123
4.6.1	Example of time series readings for the 24 days under analysis. The figures present the square root of the consumption in each hour ($\sqrt{6\text{kWh}}$) for different types of building, and illustrate the diverse nature of consumption.	126
4.6.2	Estimated factor loadings for the regular DFM (top) and sparse DFM (bottom). Series are categorised according to one of six building types, triangles indicate the example series plotted in Fig 4.6.1.	127
4.6.3	Estimated factors (black) with original data (grey) as a function of time using the optimal $\alpha = 0.01$ chosen according to BIC. When multiplied by the factor loadings (top) gives the estimated common component.	127
4.6.4	Average factor profile as a function of time-of-day, $t = 0$ corresponding to midnight. The solid line is a pointwise average of the factor $\hat{\mathbf{a}}_{t n}$ across the 18 weekdays in the sample, confidence intervals are constructed as ± 1.96 the standard-deviation.	128

4.6.5 Example of predicted consumption ($\sqrt{\text{kWh}}$) in one (accommodation) building on the campus. Forecasts are in blue and true data in red. The left column represents 1 hour ahead forecasts based on an expanding window, whilst the right column represents 6 hour ahead forecasts. The SDFM and SVAR are tuned on the 24 days of data prior to that presented in the figure. Confidence intervals for the SDFM are based on $1.96 \times [\hat{\mathbf{A}}\mathbf{P}_{t|n}\hat{\mathbf{A}}^\top + \hat{\Sigma}_\epsilon]^{1/2}$ 130

4.6.6 Forecast errors (MAE) for each building for (left column) 1 hour ahead forecast, and (right column) 6 hour ahead forecast. Performance evaluated on one hold out day ($144 - h$ data points). Each bar is colored according to which method performs best for that building. Blue: SDFM, red: AR(1), grey: SVAR. The solid black line indicates average performance across all buildings, the grouping of buildings is indicated via the dashed line under the plots. 131

5.1.1 Contribution of the 11 subdivisions of the service sector in 2021. 136

5.1.2 Quarterly time series of total imports and total exports of UK trade in services during the period Q1 2015 to Q4 2021. 137

5.2.1 Overview of what our problem would look like if we are currently in March 2021. Release of Q1 Trade in Services data not available until June. Our model is able to produce monthly nowcasts in Q1 by using the monthly indicator information available - grouped into data category colours. It also produces a temporally consistent disaggregate series up to December 2020. 141

5.4.1 (Top) Average RMSE scores over 100 iterations for recovery of true disaggregate series \mathbf{Y}_1 , \mathbf{Y}_2 and \mathbf{Y}_3 , with 25% and 75% confidence bands. (Bottom) Average loadings matrix estimate over 100 iterations for the state-space sparse DFM approach. 148

5.5.1	Estimated monthly disaggregate trade series of the subdivisions of the service sector (£billion). Exports trade in red on top and imports trade in blue at the bottom.	152
5.5.2	(Left - £billion) Published monthly total exports and imports time series by ONS with estimated series. (Right - %) Percentage difference between published series, \mathbf{y}^{pub} , and estimated series, \mathbf{y}^{est} , i.e. $(\mathbf{y}^{\text{pub}} - \mathbf{y}^{\text{est}})/\mathbf{y}^{\text{est}} \times 100\%$	154
5.5.3	Estimated factor loadings for all 4 factors in the joint state-space framework trade in services application. Loadings colour coded by indicator group.	154
5.5.4	Estimated factors (black) with original data (grey) as a function of time from the joint state-space framework trade in services application. . .	155
5.5.5	Pseudo-real time nowcasting errors (MAE) for each trade in services target series with bars ordered by the service subdivisions: total; insurance & pension; construction; financial; telecomms; intellectual property; transport; manufacturing & maintenance; other business; personal, cultural & recreational; government; and travel. Grouped by exports and imports. Performance evaluated by estimating observed quarters over the period September 2018 to August 2022. Left plot displays the joint state-space MAE results and the right plot displays spTD. Each bar is coloured according to which method performs best for that target series. Blue: Joint state-space, red: spTD.	156

5.6.1 Monthly UK trade with EU countries (blue) and with non-EU countries (red). Top left: published trade in goods exports. Top right: published trade in goods imports. Bottom left: estimated trade in services exports from joint state-space model. Bottom right: estimated trade in services imports from joint state-space model. Grey shaded region starts in January 2021 when the UK officially transitioned out of the EU. . . . 159

5.6.2 Visual interpretation of a multi-group ITSM with a treatment group (top line) and a control group (bottom line) using simulated data with a policy intervention at time $t = 20$. Coefficients $\beta_0 - \beta_7$ correspond to those in equation 5.6.1. A similar figure can be found in Linden (2015). 161

6.2.1 Heatmap of the estimated factor loadings in the inflation empirical example. 188

B.2.1 Indicators selected when fixing the number of indicators selected to be 8. Refer to Table B.2.1 for the indicator on the x-axis. 209

B.2.2 Indicators selected when fixing the number of indicators selected to be 24. Refer to Table B.2.2 for the indicator on the x-axis. 210

D.1.1 Average time from 10 experiments the EM algorithm (setting `alg = "EM"`) takes with the multivariate and univariate KFS methods with `err = "IID"` on the left and `err = "AR1"` on the right. 220

List of Tables

3.4.1	Metrics for the performance of $\hat{\beta}$ using CL, spTD and spTD_RF across $p = 30, 90, 150$ and $\rho = 0.2, 0.5, 0.8$ in both the stationary and non-stationary indicators setting. Metrics include the mean (standard deviation) across 1000 iterations of RMSE, ℓ_∞ and FP.	67
3.5.1	Summary of indicators used in GDP analysis	74
3.5.2	Coefficient estimates of GDP analysis with 10 indicator series using Chow-Lin (CL), ℓ_1 -spTD and adaptive extension	79
3.5.3	Non-zero coefficient estimates of GDP analysis with 97 indicator series using ℓ_1 -spTD and adaptive lasso extension	79
3.6.1	Selected indicators across the 21 sparse disaggregations. Number in brackets refer to the x-axis index in Figure 3.6.2.	84
3.6.2	Pseudo-real time nowcasting results	86
4.5.1	Description of data used in the model including goods target series and monthly indicator series.	118
4.5.2	Pseudo real-time nowcasting exercise results.	123
5.1.1	Description of data used in the model including quarterly target series and monthly indicator series. No. = number of indicators. Delay = publication delay in months.	138

5.6.1	Interrupted time series model coefficients (standard errors) [p-value] for the UK-EU trade in goods monthly and quarterly series. Significant coefficients (i.e. p-value < 0.05) in bold.	165
5.6.2	Interrupted time series model coefficients (standard errors) [p-value] for the UK-EU trade in services monthly and quarterly series. Significant coefficients (i.e. p-value < 0.05) in bold.	165
B.1.1	Monthly Business Survey - The Table reports 56 series downloaded related to Business survey (BS) in transportation and storage (NACE H) and land transport and transport via pipelines (NACE H49): balance (s) and frequency of reply marked as improved/increased (a), remained unchanged (n) and deteriorated (b) on 7 questions.	207
B.1.2	Asso-Aeroporti - The Table reports 18 monthly series downloaded from Assaeroporti (AA).	207
B.1.3	Turnover in Industry The Table reports all 69 monthly series representative of the Industrial Turnover Indices (IT), The list includes the tern total/domestic/foreign turnover	208
B.2.1	Selected indicators across the 21 sparse disaggregations. Number in brackets refer to the indicator on the x-axis of Figure B.2.1.	208
B.2.2	Selected indicators across the 21 sparse disaggregations. Number in brackets refer to the indicator on the x-axis of Figure B.2.2.	209

Chapter 1

Introduction

The world's economic landscape has faced many changes in recent years, both on a global and localised scale. Much of this change may be attributed to the coronavirus pandemic and related restrictions on trade and movement of people. Prior to the pandemic, which rapidly spread at the end of 2019 (reaching UK shores early 2020), the last global financial shock to meaningfully impact the UK was that of the financial crisis of 2008. The cause of these shocks are drastically different, as are the potential consequences and trajectories of the economy as we emerge. Fundamentally, to examine the impact of economic events and assess these trajectories, we first need to accurately measure activity across the economy.

In the UK, the Office for National Statistics (ONS) has a primary responsibility for this task, and reports on a wide range of statistics of interest, from headline Gross Domestic Product (GDP) figures, inflation indices, to novel indicators of economic activity and social change (ONS, 2022a). These latter measures have been more recently introduced, in part as a response to the 2016 Independent Review of UK Economic Statistics (Bean, 2016) that identified increasing the timeliness and granularity of economic statistics as key to improving the situational awareness of policy makers, government agencies and statistical organisations. Similar reports (e.g. Pfeffermann,

2015; Eurostat, 2017; Hand et al., 2018) have explored the way in which official statistics are gathered and motivated attempts to move away from the traditional survey-centric approach. In particular, there has been a notable shift towards embracing alternative¹ and administrative data sources that offer timeliness, cost-effectiveness, and reliability, while also capturing numerous processes (Fuleky, 2019). While traditional, low-frequency data sources will remain critical, their role will evolve to complement and enhance the measurement capabilities of the new higher-frequency sources (Jarmin, 2019).

The ONS has made great strides in improving the timeliness of its key statistics, for instance through integrating Value Added Tax (VAT) return data to construct a monthly indicator index of GDP (ONS, 2022b,d), and developments into using supermarket scanner data for constructing consumer inflation indices (ONS, 2021). However, many statistics of interest are not produced on a timescale relevant to understanding the fast moving economic environment that we have faced in the last few years, and the statistical methodology to calibrate insights from large collections of alternative data sources remains under-developed.

This thesis explores new methods for producing timely and interpretable economic statistics. In particular, I propose novel techniques for the tasks of temporal disaggregation and nowcasting when in the presence of large (high-dimensional) amounts of data. Temporal disaggregation (Dagum and Cholette, 2006) is a method for estimating a high-frequency (e.g. monthly) version of a low-frequency (e.g. quarterly) time series using related high-frequency indicator series (if available). The resulting high-frequency series is constrained to be temporally consistent with the low-frequency series for every low-frequency period observed. Temporal disaggregation is an important task for National Statistical Institutes (NSIs) because it provides a coherent way to combine precise but not very timely aggregate data from National Accounts with timely

¹Alternative data sources include a wide range of examples, such as satellite imagery for address registry (ONS, 2017b), road traffic sensor data, and ship tracking (ONS, 2019a), as well as internet search data from Google Trends (Ferrara and Simoni, 2022). Chapter 1 of Fuleky (2019) provides numerous additional illustrations.

indicators capable of revealing short-term dynamics. Nowcasting, on the other hand, is a technique for estimating current or very recent values of a time series based on available information. Popularised in economics by Giannone et al. (2008), nowcasting has become an important tool to provide real-time information on the current state of the economy, which consequently can be used to make timely and informed decisions. As the majority of official statistics are published with delay, nowcasts can be used as an early estimate and avoid the long delay.

With the prevalence of large volumes of high-frequency administrative and alternative data sources NSIs endeavour to incorporate into the production of official statistics, it becomes imperative to draw upon the extensive body of literature concerning high-dimensional statistics (Bühlmann and Van De Geer, 2011) and how this can be used in conjunction with the well-established temporal disaggregation and nowcasting literature. A crucial aspect of the methodology developed in this context is interpretability. How does one interpret estimates of economic statistics obtained from a potentially overwhelming amount of data? Transparency is crucial in today’s fast-paced economy, where decisions have to be made in short spaces of time. Identifying relevant variables gives the user knowledge on the main driving forces behind estimates, helping with future data revisions and the understanding of policy effects and shocks. If appropriate variables and features can be isolated, then estimates are likely to be more accurate and more consistent as noise from irrelevant variables is mitigated.

1.1 Main Contribution and Thesis Structure

The primary contribution of this thesis is to provide novel, coherent and reproducible methodologies for the tasks of temporal disaggregation and nowcasting in high-dimensional scenarios. The two main methodologies of Chapter 3 and 4 have been made into R packages available on the Comprehensive R Archive Network (CRAN) for reproducibility.

They should allow the reader a practical way to implement and interpret a high-dimensional amount of time series for the purpose of producing timely and accurate estimates of target variables. Throughout the thesis, we shall see several case studies across a range of important real data applications. The structure of the thesis is as follows.

Chapter 2 aims to provide the background knowledge and statistical tools required for the main contributions of the thesis. It begins by exploring the fundamental concepts of time series analysis. Subsequently, a thorough literature review is presented, focusing on the methods and motivations behind two crucial topics: temporal disaggregation and nowcasting.

Chapter 3 introduces a new method, named Sparse Temporal Disaggregation (spTD), that extends the well-established [Chow and Lin \(1971\)](#) family of methods for temporal disaggregation to allow for a high-dimensional set of indicators. A regularised M-estimation framework that is able to simultaneously select the relevant indicators and estimate their impact is established. The advantages of the proposed method is demonstrated through simulation study. The chapter then presents two case studies implementing spTD using a high-dimensional mixture of traditional and alternative data-sources. The first attempts to accurately disaggregate quarterly UK GDP and inform us on the most relevant indicators for this task. The second attempts to disaggregate the quarterly index of service turnover for transportation in Italy and compares the accuracy and stability of spTD with competing methods in a pseudo-real time nowcasting exercise.

Chapter 4 formally introduces a new class of sparse Dynamic Factor Models (DFMs). DFMs are a very popular tool for modelling high-dimensional time series and in particular for the task of nowcasting, however, it is often hard to interpret the meaning of the latent factors. In the proposed methodology, DFMs are estimated using an expectation-maximisation (EM) algorithm to enable estimation of model parameters using a regularised quasi-maximum likelihood. The regularisation (ℓ_1 -norm) is placed on

the loadings parameter of the DFM to obtain a sparse amount of non-zero loadings for the estimated latent factors. This provides much greater interpretation into the factor structure, allowing us to identify which series, or group of series, are influencing the dynamics of the system. Synthetic experiments demonstrate consistency in terms of estimating the loading structure, and superior predictive performance where a low-rank factor structure may be appropriate. The utility of the method is further illustrated by two case studies in distinct fields. The first is an econometric application that looks to nowcast 9 of the main trade commodities the UK exports worldwide. The second is an energy application that forecasts electricity consumption across a large set of smart meters across a University campus.

Chapter 5 presents results of a case study attempting to produce timely estimates of the subdivisions of UK Trade in Services, with particular focus on how the trajectory may have varied throughout the months of the coronavirus pandemic and Brexit transition period. For this task, a large state-space framework is constructed combining the sparse dynamic factor model with multivariate temporal disaggregation to produce a quarterly-to-monthly disaggregation of quarterly service data with the ability to obtain real-time nowcasts - avoiding the 3 month publication delay service data currently possess. In addition, an intervention study employing an interrupted time series model is conducted, which explores the potential impacts that Brexit has exerted on UK trade with EU countries.

Chapter 6 provides details on the two R packages available on CRAN called `DisaggregateTS` and `sparseDFM` implementing the methods in Chapter 3 and Chapter 4 respectively. The core structure and functionality of the packages are presented, as are detailed walk-through examples on synthetic and real data applications on how one might use the packages in practice.

Chapter 7 gives concluding remarks, highlighting the main takeaways from each chapter and presents several avenues for future work.

Chapter 2

Literature Review

A time series (Hamilton, 2020) is a collection of observations made sequentially through time. In the context of this thesis, the focus lies on the domain of official statistics, encompassing datasets published by NSIs such as the UK's ONS. These datasets predominantly comprise of information derived from surveys, censuses, and administrative data-sources, covering a wide range of vital indicators such as GDP, unemployment rates, inflation measures, crime statistics, and demographic data. It is important to note that all time series examined in this study are discrete, implying that observations are recorded at specific, evenly spaced intervals, such as months, quarters, or years. This discrete nature contrasts with continuous time series, where observations are continuously gathered over time.

Figure 2.0.1 illustrates a prominent time series that is used throughout the literature review. This particular time series corresponds to quarterly GDP data sourced from the UK's Quarterly National Accounts (QNA), comprising the period between 2016 and 2022 with key economic events in this period highlighted. The QNA is a vital publication in the UK, providing a comprehensive and consistent depiction of the economy. Notably, policymakers and analysts heavily rely on the QNA within government circles and various industries. At the Bank of England, the country's central bank, the QNA plays

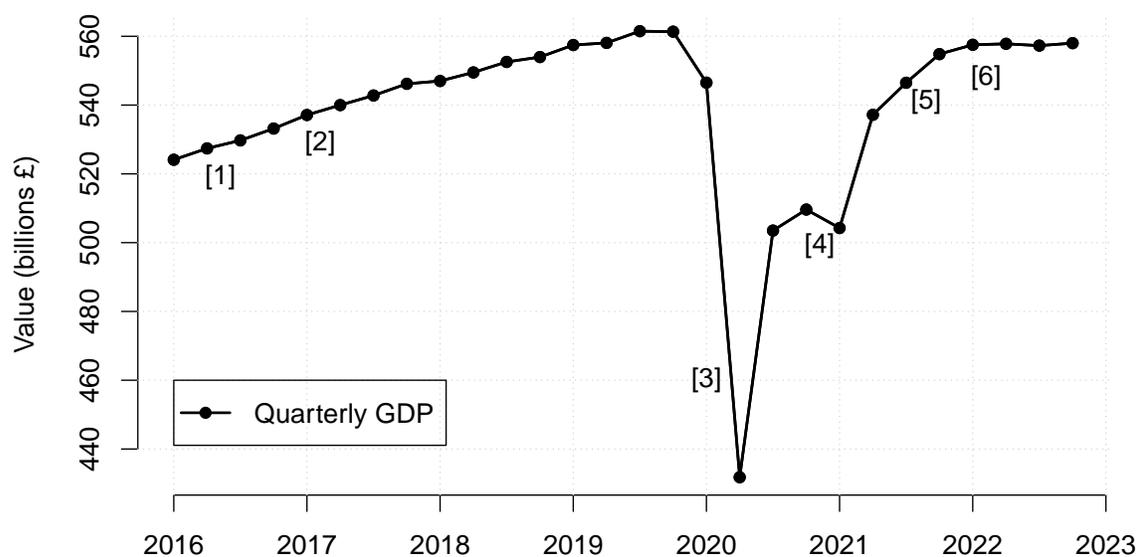


Figure 2.0.1: UK GDP Quarterly National Accounts time series: chained volume measures (CVM) and seasonally adjusted (SA). Key dates highlighted: [1] June 23, 2016 Brexit referendum; [2] March 29, 2017 Article 50 of the Treaty on European Union invoked; [3] March 23, 2020 UK lockdown due to COVID-19 pandemic; [4] 30 December, 2020 UK-EU Trade and Cooperation Agreement signed; [5] Autumn, 2021 Energy crisis begins; [6] 24 February, 2022 Russia invade Ukraine.

a crucial role in informing the Monetary Policy Committee’s decisions regarding interest rates. Furthermore, the Office for Budget Responsibility utilises the QNA for economic growth projections and forecasting public sector debt. For detailed insights into the construction of the QNA specific to the UK, please refer to ONS (2022c).

Time series data exhibits a distinctive characteristic whereby consecutive time points are not independently sampled. Consequently, the presence of correlation between these time points, referred to as autocorrelation, must be accounted for by time series models when performing inference. In these models, the current observations are typically expressed as a function of their preceding observations. Notably, observations that are closer in time tend to exhibit a stronger interdependency relationship compared to those

with larger time separations.

A popular class of time series models are the autoregressive-moving-average (ARMA) models (Hamilton, 2020). For a given univariate time series $\{x_t : t \in \mathbb{N}\}$, these models combine an autoregressive component (AR), capturing the linear relationship between current and past observations, with the moving-average (MA) component capturing the influence of past error terms $\{\epsilon_t : t \in \mathbb{N}\}$. Formally, an ARMA(p, q) model is defined as:

$$x_t = \phi_1 x_{t-1} + \cdots + \phi_p x_{t-p} + \epsilon_t + \psi_1 \epsilon_{t-1} + \cdots + \psi_q \epsilon_{t-q},$$

for p AR lags with AR coefficients ϕ_1, \dots, ϕ_p , and q MA lags with MA coefficients ψ_1, \dots, ψ_q . The error terms are assumed to be serially uncorrelated and identically distributed white noise processes. When dealing with multivariate time series, these models can be extended to vector autoregressive-moving-average (VARMA) models (Tsay, 2013), enabling the measurement of autocorrelation properties across multiple time series.

The concept of stationarity is essential in time series analysis. Stationarity refers to the property of a time series where its statistical properties, such as mean, variance, and autocovariance structure, remain constant over time. Formally, a time series x_t is considered stationary (or weakly stationary) if it satisfies the following conditions:

- $\mathbb{E}[x_t] = \mu$ for all t ;
- $\mathbb{E}[x_t^2] < \infty$ for all t ;
- $\text{Cov}(x_t, x_{t-k}) = \gamma(k)$, for all t and k .

A strictly stationary time series maintains an invariant joint probability distribution over time, while a weakly stationary time series with Gaussian distribution is also strictly stationary (Brockwell and Davis, 2009). When stationarity is discussed throughout the thesis, we can assume this is in the weak-sense. Stationarity plays a crucial role in

simplifying and interpreting time series analysis, serving as a prerequisite for reliable statistical inference. ARMA models are highly popular because, according to Wold's decomposition theorem (Hamilton, 2020), any stationary process can be approximated by a stationary ARMA model¹.

Most time series data exhibit trends, representing long-term increases or decreases, and seasonality, which refers to repeated cycles at fixed frequencies. These characteristics make time series inherently non-stationary. To ensure stationarity when modeling, considerable effort is invested in preparing and publishing official time series data. One simple technique used to remove trends from a time series is differencing, which involves computing the differences between consecutive observations. By differencing the data, non-stationary time series may be transformed into stationary ones. A canonical example is the ARIMA(p, d, q) that takes on an ARMA(p, q) after differencing d times.

In addition to trends, seasonality can also be addressed through differencing. By differencing the series at fixed intervals corresponding to the occurrence of seasonality, the seasonal patterns can be eliminated. The X13-ARIMA-SEATS methodology (U.S. Census Bureau, 2022) is a highly regarded technique for seasonal adjustment that also considers holiday effects, and is implemented by several NSIs.

When multiple non-stationary time series are modeled together, it is possible that they share a stable relationship. Although these series may exhibit independent movements, their average distance remains relatively constant. This property is known as cointegration (Johansen, 1988). Cointegration allows for the joint modeling of non-stationary variables, as they will demonstrate a stationary relationship in the long run. This is particularly useful for capturing the interdependencies and long-term dynamics among multiple time series.

For a more detailed understanding of time series theory and analysis, refer to the

¹ARMA(p, q) models are stationary when the roots of the characteristic equation (the equation that relates the current observation to past observations) lie outside the unit circle. In other words, all the roots of the autoregressive polynomial must have magnitudes greater than one. For example, the frequently used AR(1) model (i.e. ARMA(1,0)) in this thesis is stationary when $|\phi_1| < 1$.

comprehensive books by Brockwell and Davis (2009), Tsay (2013), Box et al. (2015) and Hamilton (2020). The rest of the chapter provides a detailed literature review for two important time series techniques in production and publication of official statistics: temporal disaggregation and nowcasting.

2.1 Temporal Disaggregation

This section aims to provide a detailed literature review for temporal disaggregation methods. For alternative reviews, see Dagum and Cholette (2006) and Pavía-Miralles et al. (2010) for surveys of the literature, and Chen (2007) for an empirical comparison of popular methods using 60 series of annual data from the national accounts of the U.S. Bureau of Economic Analysis. The concept of temporal disaggregation is motivated and explained in Section 2.1.1. In Section 2.1.2, the temporal aggregation matrix that ensures disaggregate estimates are temporally consistent with low-frequency observations is defined. Section 2.1.3 reviews smoothing methods used for temporal disaggregation that aim to derive a smooth path for the unobserved series. Section 2.1.4 reviews regression-based temporal disaggregation methods that make use of a collection of related indicator series. This section opens the door for the methodology discussed in Chapter 3. Section 2.1.5 concludes with a review on multivariate temporal disaggregation methods where contemporaneous constraints may be present.

2.1.1 Introduction

Temporal disaggregation is a vital tool for National Statistical Institutes (NSIs) such as the UK's Office for National Statistics (ONS) to transform data from a lower to higher frequency. NSIs often face the challenge of having information not directly observable at the temporal frequency they desire and are increasingly asked to provide more frequent publications of headline statistics like GDP, inflation rates, and employment.

For example, when compiling a country's Quarterly National Accounts, NSIs often use comprehensive data sources obtained from annual industry and household surveys. The typical motivation for providing these more frequent measures is to paint a more timely picture of the economic health of a country. Data at lower frequencies may miss short-term fluctuations and trends that could be significant for policy makers when making informed decisions.

The temporal disaggregation problem is perhaps best explained from the perspective of a practitioner working in official statistics. The practitioner observes a low frequency data-stream of a key statistic (such as GDP) that they wish to disaggregate to a higher frequency. For the sake of clarity, we can assume that this low-frequency data-stream is at a quarterly scale and denote this as the vector $\bar{\mathbf{y}} \in \mathbb{R}^{n_l \times 1}$, containing n_l quarterly observations. Note, we use the bar notation above variables that are at the lower temporal frequency and n_l will represent the number of low-frequency data points. The practitioner seeks to construct a disaggregated version of $\bar{\mathbf{y}}$ at a sub-quarter time resolution. In addition, it is desirable that the disaggregated series be temporally consistent during each quarter and not contain spurious jumps between quarters. Consider, by way of example, a setting where one seeks to produce a monthly version and denote this by $\mathbf{y} \in \mathbb{R}^{n \times 1}$, where $n = 3n_l$ in the case of quarterly-to-monthly disaggregation. The disaggregation challenge lies in developing a principled approach to interpolate or distribute between each observed quarterly data point.

A common approach to achieve this goal is to use indicator series² recorded at the desired high-frequency (monthly) that are believed to indicate the intra-quarter movements and help with their estimation. We denote a set of high-frequency indicator series as the matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ where each column is a monthly time series representing one of p indicators. By way of example, consider the task of disaggregating the quarterly series of UK GDP we saw in Figure 2.0.1 to a monthly frequency using the Index of

²Indicator series may also be referred to as related, covariate, predictor or explanatory series.

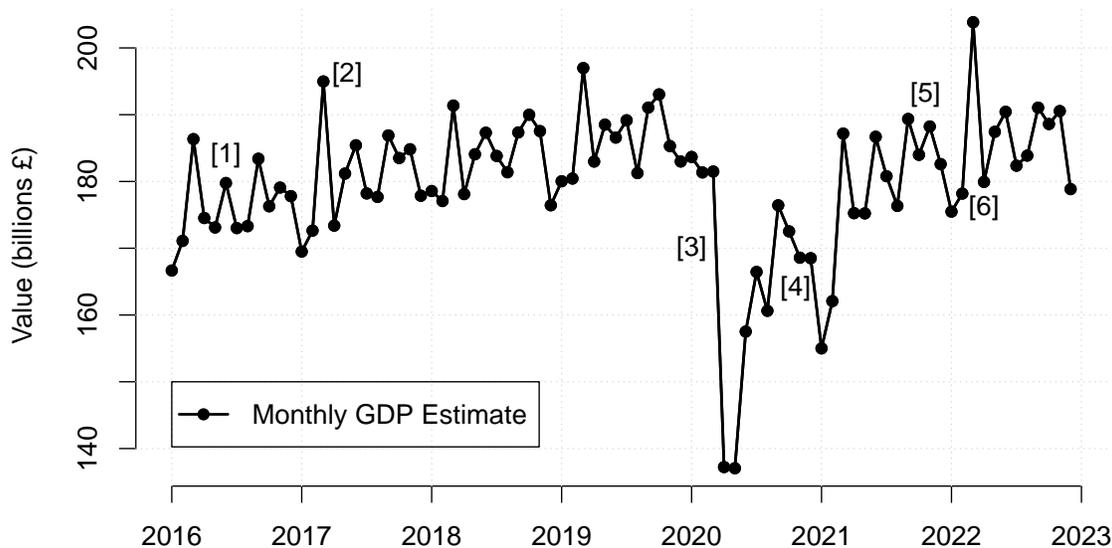


Figure 2.1.1: Monthly estimate of UK GDP using Chow-Lin temporal disaggregation with Index of Production and Services as indicator series. Key dates highlighted: [1] June 23, 2016 Brexit referendum; [2] March 29, 2017 Article 50 of the Treaty on European Union invoked; [3] March 23, 2020 UK lockdown due to COVID-19 pandemic; [4] 30 December, 2020 UK-EU Trade and Cooperation Agreement signed; [5] Autumn, 2021 Energy crisis begins; [6] 24 February, 2022 Russia invade Ukraine.

Production and Services for the UK as two monthly indicator series. Using the well-established temporal disaggregation method by [Chow and Lin \(1971\)](#), that is explained in detail in [Section 2.1.4](#), we obtain a monthly disaggregate series of UK GDP displayed in [Figure 2.1.1](#). Key events affecting the UK economy are again added to the figure. Having GDP at the monthly frequency allows us to better understand the short-term movements around these key dates, informing us on months of economic decline and recovery. The disaggregate series estimate is constrained such that three monthly values in a quarter sum to the corresponding observed quarterly figure from the Quarterly National Accounts. This temporal aggregation constraint is discussed in the next section.

2.1.2 Temporal Aggregation Constraint

The disaggregate estimates of a low-frequency time series are constrained to be temporally consistent for each low-frequency period. For example, in a quarterly-to-monthly disaggregation, three monthly estimates of a quarter must correctly aggregate to their respective observed quarter figure. The type of aggregation that must be adhered to depends on the nature of the data in question. With official statistics, data can be typically classified as either stock or flow. Stock data measures the quantity/level of a variable at a specific point in time, for example, capital stock at the end of a year. Flow data measure how much of something has happened over a period of time, for example, production over a month. Temporal disaggregation of stock data is essentially an interpolation problem that is constrained to pass through the low-frequency observations. Typically, the last disaggregate estimate in a low-frequency period is made equal to the corresponding low-frequency observation. I.e.

$$\bar{\mathbf{y}}_\tau = \mathbf{y}_t, \quad \text{for } t = 3\tau,$$

for quarterly-to-monthly disaggregation of all quarters $\tau = 1, \dots, n_l$. For disaggregation of flow data, high-frequency estimates are typically summed or averaged over low-frequency periods. I.e. in quarterly-to-monthly disaggregation we have

$$\bar{\mathbf{y}}_\tau = \sum_{i=0}^2 \mathbf{y}_{t-i} \quad \text{or} \quad \bar{\mathbf{y}}_\tau = \frac{1}{3} \sum_{i=0}^2 \mathbf{y}_{t-i}, \quad \text{for } t = 3\tau, \quad \tau = 1, \dots, n_l,$$

for sum and average constraints, respectively.

In general, the disaggregate series, \mathbf{y} , must satisfy the temporal aggregation constraint given by

$$\bar{\mathbf{y}} = \mathbf{C}\mathbf{y},$$

where \mathbf{C} is a $n_l \times n$ temporal aggregation matrix defined by

$$\mathbf{C} = \mathbf{I}_{n_l} \otimes \mathbf{c} = \begin{pmatrix} \mathbf{c} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{c} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{c} \end{pmatrix},$$

where $\mathbf{0}$ and \mathbf{c} are vectors of length $k = n/n_l$, with $\mathbf{0}$ containing zeros and \mathbf{c} taking various forms:

- $\mathbf{c} = (1, \dots, 1)$ for summing flow series;
- $\mathbf{c} = (\frac{1}{k}, \dots, \frac{1}{k})$ for averaging flow series;
- $\mathbf{c} = (0, \dots, 0, 1)$ for interpolating stock series.

The applications considered in this thesis aim to find timely estimates of flow data, where the sum of high-frequency estimates between low-frequency periods are constrained to equal the corresponding low-frequency observation. Henceforth, the aggregation matrix is $\mathbf{C} = \mathbf{I}_n \otimes \mathbf{c}$, where $\mathbf{c} = (1, \dots, 1)$, the k -dimensional vector of ones.

2.1.3 Smoothing-based Methods

There are several methods that exist to perform the task of temporal disaggregation. The choice of method ultimately depends on what related information is available at the high-frequency and the preferences of the practitioner. The most naive approach would be to simply distribute the unobserved high-frequency values pro-rata, i.e. set each monthly value in a quarter to be one-third of the observed quarter value. The problem with this simplistic approach is that it would result in large, spurious jumps between each quarter and not capture the more realistic smoother, gradual temporal change economic time series possess.

When no related high-frequency information (indicator series) are available, the preferred option is to use constrained smoothing methods that aim to fit a smooth and continuous curve through high-frequency data points ensuring that temporal additivity constraints are satisfied. A commonly used smoothing procedure is cubic spline smoothing. A cubic spline is a series of cubic (third-degree) polynomials that interpolate a tabulated function at certain data points called knots such that the first and second derivatives of the overall curve at each knot are continuous to guarantee smoothness. [Quenneville et al. \(2013\)](#) demonstrate how cubic splines can be used for temporal disaggregation, as the low-frequency observations act as ‘knots’, and, as each cubic polynomial linking up these knots is a function of time, sub-period estimates can then be simply derived from each spline. The levels and slopes at adjacent low-frequency periods are set equal to avoid spurious jumps, and the sum of the values of the spline over each low-frequency period is set equal to the respective low-frequency value to meet temporal additivity constraints. The ONS use cubic splines when producing the Index of Services ([ONS, 2017a](#)).

A different smoothing approach in the absence of indicator data is to set up a constrained quadratic optimisation problem trying to minimise the period-to-period changes in the disaggregate series subject to observed low-frequency temporal constraints. [Boot et al. \(1967\)](#) consider minimising the sum of squares of the differences between successive high-frequency values. Formally, letting $d_t = y_t - y_{t-1}$ be the first-difference process for $t = 2, \dots, n$, and \mathbf{d} be the full vector for all t , they solve the constrained minimisation given by

$$\min_{\mathbf{y}} \quad \mathbf{d}^\top \mathbf{d} = \mathbf{y}^\top \mathbf{D}^\top \mathbf{D} \mathbf{y} \quad \text{s.t.} \quad \bar{\mathbf{y}} = \mathbf{C} \mathbf{y}, \quad (2.1.1)$$

where \mathbf{D} is the $n \times n$ matrix of first-differences:

$$\mathbf{D} = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ -1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & -1 & 1 \end{pmatrix}. \quad (2.1.2)$$

The solution of the constrained minimisation (2.1.1) is

$$\mathbf{y} = (\mathbf{D}^\top \mathbf{D})^{-1} \mathbf{C}^\top (\mathbf{C} (\mathbf{D}^\top \mathbf{D})^{-1} \mathbf{C}^\top)^{-1} \bar{\mathbf{y}}. \quad (2.1.3)$$

Appendix A.1 provides details for solving a general quadratic optimisation problem with a linear equality constraint for the purpose of temporal disaggregation. The general solution derived encompasses solution (2.1.3) for the Boot et al. (1967) smoothing method, as well as other popular temporal disaggregation methods described in the subsequent text.

Stram and Wei (1986) generalise the smoothing optimisation method of Boot et al. (1967) by assuming the latent high-frequency series evolves according to an ARIMA(p_1, d_1, q_1) process. If $p_1 = q_1 = 0$ and $d = 1$, then this is identical to the Boot et al. (1967) approach. Stram and Wei (1986) attempt to find solutions of the constrained minimisation given by

$$\min_{\mathbf{y}} \mathbf{d}^\top \mathbf{V}_d^{-1} \mathbf{d} = \mathbf{y}^\top \mathbf{D}^\top \mathbf{V}_d^{-1} \mathbf{D} \mathbf{y} \quad \text{s.t.} \quad \bar{\mathbf{y}} = \mathbf{C} \mathbf{y},$$

where \mathbf{V}_d is the $n \times n$ covariance matrix of \mathbf{d} that will depend on the parameters of the assumed ARIMA model for \mathbf{y} . Denton (1971) extends the Boot et al. (1967) smoothing method by incorporating a single high-frequency indicator series, \mathbf{x} , that is believed to approximate the short-term dynamics of $\bar{\mathbf{y}}$. Their approach seeks to obtain a disaggregate series \mathbf{y} that maximally preserves the movements in the indicator series

\mathbf{x} while being temporally consistent with $\bar{\mathbf{y}}$. Formally, they find solutions to

$$\min_{\mathbf{y}} (\mathbf{y} - \mathbf{x})^\top \mathbf{B} (\mathbf{y} - \mathbf{x}) \quad \text{s.t.} \quad \bar{\mathbf{y}} = \mathbf{C} \mathbf{y},$$

where \mathbf{B} is an $n \times n$ symmetric, non-singular weighting matrix. Denton (1971) specify various forms of \mathbf{B} including $\mathbf{B} = \mathbf{D}^\top \mathbf{D}$ to preserve period-to-period level changes between \mathbf{y} and \mathbf{x} , and $\mathbf{B} = \tilde{\mathbf{X}}^{-1} \mathbf{D}^\top \mathbf{D} \tilde{\mathbf{X}}$ with $\tilde{\mathbf{X}} = \text{diag}(\mathbf{x})$ to preserve proportional period-to-period change. The corresponding solution is given by

$$\mathbf{y} = \mathbf{x} + \mathbf{B}^{-1} \mathbf{C}^\top (\mathbf{C} \mathbf{B}^{-1} \mathbf{C}^\top)^{-1} (\bar{\mathbf{y}} - \mathbf{C} \mathbf{x}),$$

that involves adding a distribution of aggregated residuals to the indicator series \mathbf{x} to ensure temporal consistency.

Denton (1971) is a popular method amongst national statistical organisations to benchmark unbalanced data. For example, it is used within the X13-ARIMA-SEATS seasonal adjustment program (U.S. Census Bureau, 2022) to ensure aggregate totals of seasonally adjusted data matches aggregate totals of non-seasonally adjusted data. There has been modifications to the original Denton method that include preserving the percentage growth rate (Causey and Trager, 1981), improving the treatment of the initial conditions (Cholette, 1984), and accounting for bias and/or errors in the observed aggregate benchmark series (Cholette and Dagum, 1994).

2.1.4 Regression-based Methods with Indicator Series

Pioneered by the work of Chow and Lin (1971), regression-based methods for temporal disaggregation allow the practitioner to model more than one related high-frequency indicator series in the disaggregation of low-frequency data. The premise of the Chow-Lin framework is to assume that a linear relationship exists between the unobserved disaggregate target series, \mathbf{y} , and the set of high-frequency indicator series, \mathbf{X} , through

the expression

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}, \quad (2.1.4)$$

where $\boldsymbol{\beta} \in \mathbb{R}^{p \times 1}$ is a vector of unknown parameters and $\mathbf{u} \in \mathbb{R}^{n \times 1}$ is a residual vector. Note, it is possible for the matrix \mathbf{X} to contain deterministic terms such as constants and trends. The residual vector \mathbf{u} is mean-zero and has covariance matrix \mathbf{V} . Chow and Lin (1971) assume the data generating process of \mathbf{u} follows a first-order autoregressive (AR(1)) process, $u_t = \rho u_{t-1} + \epsilon_t$, with $\epsilon_t \sim N(0, \sigma_\epsilon^2)$ and $|\rho| < 1$. This assumption of a stationary residual process allows a cointegrating relationship between \mathbf{y} and \mathbf{X} when they are non-stationary, a likely scenario with economic time series. Thus, the regression coefficient $\boldsymbol{\beta}$ measures both the long- and short-run effect of \mathbf{X} on \mathbf{y} . The resulting covariance matrix \mathbf{V} has the well-known Toeplitz form

$$\mathbf{V} = \frac{\sigma_\epsilon^2}{1 - \rho^2} \begin{pmatrix} 1 & \rho & \dots & \rho^{n-1} \\ \rho & 1 & \dots & \rho^{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ \rho^{n-1} & \rho^{n-2} & \dots & 1 \end{pmatrix}, \quad (2.1.5)$$

containing two unknown parameters ρ and σ_ϵ^2 that must be estimated. Alternatives to the Chow-Lin AR(1) assumption include Fernandez (1981) who assume that residuals follow a random walk process³ ($u_t = u_{t-1} + \epsilon_t$), and Litterman (1983) who assume that residuals follow an ARIMA(1,1,0) process ($u_t = u_{t-1} + \epsilon_t$ and $\epsilon_t \sim AR(1)$).

As the dependent variable in (2.1.4) is unobserved, the regression is pre-multiplied by the $n_l \times n$ aggregation matrix \mathbf{C} to obtain the observable low-frequency counterpart given by

$$\bar{\mathbf{y}} = \bar{\mathbf{X}}\boldsymbol{\beta} + \bar{\mathbf{u}}, \quad (2.1.6)$$

³The random walk assumption of Fernandez (1981) can be viewed as analogous to the Denton (1971) additive first-difference method, where $\mathbf{B} = \mathbf{D}^\top \mathbf{D}$, when a preliminary estimate \mathbf{x} is used in the place of $\mathbf{X}\boldsymbol{\beta}$.

where $\bar{\mathbf{y}} = \mathbf{C}\mathbf{y}$, $\bar{\mathbf{X}} = \mathbf{C}\mathbf{X}$ and $\bar{\mathbf{u}} = \mathbf{C}\mathbf{u}$ with covariance matrix $\bar{\mathbf{V}} = \mathbf{C}\mathbf{V}\mathbf{C}^\top$. The regression equation in (2.1.6) is now fully observable and can be solved using standard techniques. As autocorrelation is present in the Chow-Lin assumption of an AR(1) residual process, this is dealt with by using the generalised least-squares (GLS) estimator:

$$\begin{aligned}\hat{\boldsymbol{\beta}} &= \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ \left\| \bar{\mathbf{V}}^{-1/2}(\bar{\mathbf{y}} - \bar{\mathbf{X}}\boldsymbol{\beta}) \right\|_2^2 \right\} \\ &= (\bar{\mathbf{X}}^\top \bar{\mathbf{V}}^{-1} \bar{\mathbf{X}})^{-1} \bar{\mathbf{X}}^\top \bar{\mathbf{V}}^{-1} \bar{\mathbf{y}}.\end{aligned}$$

The estimator for $\boldsymbol{\beta}$ is conditional on the covariance matrix $\bar{\mathbf{V}}$ which contains the unknown parameters ρ and σ_ϵ^2 . In fact, the σ_ϵ^2 parameter can be factored out of the covariance matrix $\bar{\mathbf{V}}$ and will cancel itself out in the estimator $\hat{\boldsymbol{\beta}}$. Thus, only ρ has to be estimated prior to the estimate for $\boldsymbol{\beta}$. It follows that $\hat{\boldsymbol{\beta}}$, conditional on $\hat{\rho}$, is a feasible GLS estimator of $\boldsymbol{\beta}$ (Kariya and Kurata, 2004). Originally, Chow and Lin (1971) proposed an iterative procedure to infer the ρ parameter from the observed autocorrelation of the aggregated residuals by identifying a functional relationship between annual and quarterly AR(1) processes, however, it has since been shown this method is not very reliable and suffers when sharp movements are present in the series (Chen, 2007). Bournay and Laroque (1979) provide a more intuitive method by first estimating $\hat{\boldsymbol{\beta}}$ and $\hat{\bar{\mathbf{V}}}$ via profile-likelihood maximisation, before searching for the autoregressive parameter over the stationary range of $\rho \in (-1, 1)$. I.e. the profile estimator of $\boldsymbol{\beta}$ is a function of ρ .

The feasible GLS estimator $\hat{\boldsymbol{\beta}}$ (i.e. $\hat{\boldsymbol{\beta}}_{\hat{\rho}}$) conditioned on the maximum likelihood estimate of ρ is used to construct the target high-frequency unobserved series \mathbf{y} . Chow and Lin (1971) show following the standard GLS argument (Kariya and Kurata, 2004) that the optimal (best linear unbiased) solution is given by

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{V}\mathbf{C}\bar{\mathbf{V}}^{-1}(\bar{\mathbf{y}} - \bar{\mathbf{X}}\hat{\boldsymbol{\beta}}).$$

The first part of this expression $\mathbf{X}\hat{\boldsymbol{\beta}}$ is an estimate for the conditional expectation of \mathbf{y} given \mathbf{X} . The second is an estimate of the high-frequency residual \mathbf{u} , obtained by adjusting the observed low-frequency residuals $\hat{\mathbf{u}} = \bar{\mathbf{y}} - \bar{\mathbf{X}}\hat{\boldsymbol{\beta}}$ with the smoothing matrix $\mathbf{L} = \mathbf{V}\mathbf{C}\bar{\mathbf{V}}^{-1}$ to ensure temporal consistency between the estimates $\hat{\mathbf{y}}$ and observations $\bar{\mathbf{y}}$. This solution follows the same structure as the derivations in Appendix A.1.

A benefit of a regression approach is the possibility to obtain confidence intervals and quantify any uncertainty in point estimates. Confidence intervals for the high-frequency estimates can be calculated from the covariance matrix of estimation errors given by

$$\text{Cov}(\hat{\mathbf{y}} - \mathbf{y}|\mathbf{X}) = (\mathbf{I}_m - \mathbf{L}\mathbf{C})\mathbf{V} + (\mathbf{X} - \mathbf{L}\bar{\mathbf{X}})(\bar{\mathbf{X}}\bar{\mathbf{V}}^{-1}\bar{\mathbf{X}}^\top)(\mathbf{X} - \mathbf{L}\bar{\mathbf{X}})^\top.$$

The regression model also provides a natural way to produce forecasts of the high-frequency target series outside the sample time window by simple extrapolation using estimated model parameters.

The Chow-Lin family (Chow and Lin, 1971; Fernandez, 1981; Litterman, 1983) of regression-based temporal disaggregation methods are vital for the production of official statistics and compilation of national accounts across many national statistical institutes (Eurostat, 2018). Several authors have considered extensions to the well-established regression-based framework. Silva and Cardoso (2001) and Proietti (2006) include dynamic features to the model by adding lagged values of the variable of interest and the indicator series to the regression equation. Proietti (2006) cast the lagged model into a state-space form and incorporate a time-varying cumulator variable to systematically track temporal aggregation constraints. The state-space framework allows inference to be conducted via the Kalman filter (Harvey, 1990). Moauro and Savio (2005) also consider the state-space framework by specifying a seemingly unrelated time series equations (SUTSE) model to take advantages of structural components of related time series. Labonne and Weale (2020) use the SUTSE model for temporal disaggregation of low-frequency data that is noisy and overlapping. Mitchell et al. (2005) derive an

approximation for temporal disaggregation when data are in logarithms. Note, this is an approximation since logarithms of three monthly estimates do not add up to the logarithm of the quarterly estimate. Proietti (2006) show how to obtain an exact disaggregation using a non-linear filtering algorithm in this setting. This is useful when aggregation constraints are non-linear. Angelini et al. (2006) propose a method to perform temporal disaggregation using a large, potentially high-dimensional number of indicator series. They perform principal components analysis on the large indicator set and use a small number of principal components explaining the majority of variance of the indicators as a new, low-dimensional indicator set. Chapter 3 introduces a novel method to perform temporal disaggregation in the presence of a high-dimensional set of indicator series that is based on a regularised M-estimation framework.

2.1.5 Multivariate Methods with Contemporaneous Constraints

Temporal disaggregation methods discussed so far are essentially univariate, aiming to obtain a single disaggregate time series, $\mathbf{y} \in \mathbb{R}^{n \times 1}$. These methods can be readily extended to a multivariate setting aimed at obtaining a matrix of disaggregate series $\mathbf{Y} \in \mathbb{R}^{n \times m}$. The m target series in question are likely to not only have to meet temporal aggregation constraints from the observed low-frequency counterpart of \mathbf{Y} , denoted by $\bar{\mathbf{Y}} \in \mathbb{R}^{n_l \times m}$, but also contemporaneous aggregation constraints from a cross-sectional total aggregate series $\mathbf{z} \in \mathbb{R}^{n \times 1}$. For example, multivariate temporal disaggregation methods can be used when producing more frequent statistics at the regional level of a nation, where timely regional estimates must reconcile temporally with observed low-frequency regional figures and contemporaneously with the high-frequency national figure. In recent years, reports (Pfeffermann, 2015; Bean, 2016; Koop et al., 2020b) have expressed the importance in understanding the economic disparity between regions and suggest that, by working out how regions are correlated with each other and deviate from the national average at a more disaggregate level will significantly help when evaluating

a nations economy and when making policy decisions. Since 2019, the ONS have been publishing a quarterly index of GDP for the countries of the UK and nine regions in England (classified by the International Territorial Level [ITL] 1). Quarterly estimates of the regions are based on the industry action taking place in that region coming from Value Added Tax (VAT) returns for almost two million business units to HM Revenue and Customs (HMRC). See ONS (2019b) for full details of the methodology.

Formally, the multivariate temporal disaggregation problem consists of an aggregation constraint given by

$$\mathbf{y}_c = \tilde{\mathbf{C}}\mathbf{y}, \quad (2.1.7)$$

where $\mathbf{y}_c = (\bar{\mathbf{Y}}_1, \dots, \bar{\mathbf{Y}}_m, \mathbf{z})^\top \in \mathbb{R}^{n(m+1) \times 1}$ stack each low-frequency observed series in $\bar{\mathbf{Y}}$ and the cross-section aggregate \mathbf{z} on top of each other, $\mathbf{y} \in \mathbb{R}^{nm \times 1}$ stack each high-frequency target series in \mathbf{Y} on top of each other, and $\tilde{\mathbf{C}} = [\mathbf{C}_1, \mathbf{C}_2]^\top \in \mathbb{R}^{n(m+1) \times nm}$ is the full aggregation matrix with $\mathbf{C}_1 = \mathbf{I}_m \otimes \mathbf{C}$ the temporal constraint and $\mathbf{C}_2 = \mathbf{1}_m \otimes \mathbf{I}_n$ the contemporaneous constraint with $\mathbf{1}_m$ the m -dimensional vector of ones to sum cross-sectionally across \mathbf{Y} . With the complete aggregation constraint (2.1.7) defined, the univariate regression-based framework (Chow and Lin, 1971) can be adapted to a multivariate framework:

$$\begin{aligned} \mathbf{y} &= \mathbf{X}\boldsymbol{\beta} + \mathbf{u}, \\ \text{s.t. } \mathbf{y}_c &= \tilde{\mathbf{C}}\mathbf{y}, \end{aligned}$$

where $\mathbf{X} = \text{diag}(\mathbf{X}_1, \dots, \mathbf{X}_m)$ is an $nm \times p$ block-diagonal matrix of indicator series for each region $j = 1, \dots, m$, with p_j indicators in each block and $p = p_1 + \dots + p_m$. The coefficient vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_m)^\top$ is a $p \times 1$ vector of unknown parameters and $\mathbf{u} = (\mathbf{u}_1, \dots, \mathbf{u}_m)^\top$ is the $nm \times 1$ stacked vector of random disturbances which has mean-zero and covariance matrix \mathbf{V} .

Assuming full knowledge of \mathbf{V} , the best linear unbiased estimator (BLUE) for \mathbf{y} can

be obtained similarly to before by the equations:

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{V}\tilde{\mathbf{C}}^\top \mathbf{V}_c^+ (\mathbf{y}_c - \mathbf{X}_c\hat{\boldsymbol{\beta}}) \\ \hat{\boldsymbol{\beta}} &= (\mathbf{X}_c^\top \mathbf{V}_c^+ \mathbf{X}_c)^{-1} \mathbf{X}_c^\top \mathbf{V}_c^+ \mathbf{y}_c,\end{aligned}$$

where $\mathbf{X}_c = \tilde{\mathbf{C}}\mathbf{X}$, $\mathbf{V}_c = \tilde{\mathbf{C}}\mathbf{V}\tilde{\mathbf{C}}^\top$ and $(\cdot)^+$ represents the Moore-Penrose pseudo inverse which is used as the aggregation matrix $\tilde{\mathbf{C}}$ becomes singular with the contemporaneous constraint involved. Furthermore, the error covariance matrix is found to be

$$\text{Cov}(\hat{\mathbf{y}} - \mathbf{y} | \mathbf{X}) = (\mathbf{I}_{nR} - \mathbf{V}\tilde{\mathbf{C}}^\top \mathbf{V}_c^+ \tilde{\mathbf{C}})\mathbf{V} + (\mathbf{X} - \mathbf{V}\tilde{\mathbf{C}}^\top \mathbf{V}_c^+ \mathbf{X}_c)(\mathbf{X}_c^\top \mathbf{V}_c^+ \mathbf{X}_c)^{-1}(\mathbf{X} - \mathbf{V}\tilde{\mathbf{C}}^\top \mathbf{V}_c^+ \mathbf{X}_c)^\top.$$

The solution is dependent on the covariance matrix \mathbf{V} . Rossi et al. (1982) and Di Fonzo (1990) originally proposed the disturbance series \mathbf{u} to follow a multivariate white noise process, $\mathbf{V} = E(\mathbf{u}\mathbf{u}^\top) = \boldsymbol{\Sigma} \otimes \mathbf{I}_n$. The $m \times m$ matrix $\boldsymbol{\Sigma}$ can be estimated using the OLS residuals of separate temporally aggregated regressions $\mathbf{Y}_j = \mathbf{C}\mathbf{X}_j + \mathbf{C}\mathbf{u}_j$, for $j = 1, \dots, m$. Pavía-Miralles and Cabrer-Borrás (2007) extended this restrictive assumption to model \mathbf{u} as either a multivariate set of independent AR(1) or random walk processes. Under such assumptions, the AR(1) parameter can be derived for each $\mathbf{u}_j \in \mathbb{R}^{n \times 1}$ separately using the univariate Chow-Lin procedure. They note that in some applications the non-stationary random walk hypothesis leads to better estimates. Therefore, they allow residuals to follow a random walk process if they fail to reject a unit-root, determined via an augmented Dickey-Fuller (ADF) test. The full covariance matrix \mathbf{V} can be constructed as $\mathbf{V}_{i,j} = \boldsymbol{\Sigma}_{i,j} \otimes (\boldsymbol{\Phi}_j^\top \boldsymbol{\Phi}_i)^{-1}$ for $i, j = 1, \dots, m$, where $\boldsymbol{\Phi}_i$ is a $n \times n$ bidiagonal matrix with ones on the main diagonal and $-\hat{\phi}_i$ (the estimated AR(1) parameter for region i) on the sub-diagonal, and $\boldsymbol{\Sigma}_{i,j}$ is the covariance of AR(1) shocks $\hat{\epsilon}_{i,t} = \hat{u}_{i,t} - \hat{\phi}_i \hat{u}_{i,t-1}$. If it assumed that $\phi_i = \phi$ for all $i = 1, \dots, m$, then we have a simpler $\mathbf{V} = \boldsymbol{\Sigma} \otimes \boldsymbol{\Phi}$, where $\boldsymbol{\Phi}_{s,t} = \phi^{|s-t|} / (1 - \phi^2)$ for times $s, t = 1, \dots, n$. In this case, $\boldsymbol{\Sigma}$ can be concentrated out of the the model's likelihood function and profile-likelihood

maximisation can be utilised over the stationary range $\phi \in (-1, 1)$ by grid search.

Guerrero and Nieto (1999) propose a model based approach that extends the multivariate BLUE procedure to the case where the series follow a general stationary VAR process. The added flexibility of the VAR enables the data itself suggests the VAR model as opposed to assuming the error structure beforehand as in the previous methods. The VAR model allows more complex inter-temporal correlations to be captured among regions and is estimated using a straightforward procedure (Guerrero and Nieto, 1999). Authors (Di Fonzo and Marini, 2011, 2015) extend the classic univariate benchmarking procedure of Denton (1971) to the multivariate setting to adjust multiple time series at once based on minimising additive and proportional first-differences, respectively. Proietti (2011b) propose a state-space representation of the multivariate BLUE models discussed that allows exact treatment of initial conditions when the series are non-stationary (i.e. via diffuse priors) and the use of the Kalman filter to handle missing data and perform out-of-sample extrapolation. Recent work (Koop et al., 2020a,b) also use a state-space framework for the task of producing quarterly estimates of Gross Value Added (GVA) growth for the ITL 1 regions of the UK satisfying both temporal and contemporaneous constraints. They adopt a Bayesian mixed-frequency VAR model treating unknown quarterly GVA growth as latent states and perform inference using Bayesian Markov chain Monte Carlo (MCMC) algorithms with a hierarchical Dirichlet–Laplace prior (Bhattacharya et al., 2015) to induce parsimony in the overparameterised VAR model.

2.2 Nowcasting

Nowcasting can be thought of as ‘forecasting’ the current or very recent state of the economy. This section hopes to provide a detailed literature review of the main techniques used for the task of nowcasting, going from conventional regression-based approaches

to more complex state-space solutions. For alternative reviews of the literature see Bańbura et al. (2013), Bok et al. (2018) and Fuleky (2019).

In this review, Section 2.2.1 explains what the nowcasting task involves using a visual representation that also highlights the challenges we face. Section 2.2.2 reviews the conventional regression-based approach involving aggregation to the low-frequency, while Section 2.2.3 introduces Mixed Data Sampling (MIDAS) models that work at the high-frequency. The rest of the section puts focus on perhaps the most popular nowcasting technique – Dynamic Factor models, detailing estimation methods in Section 2.2.4 and how they are adopted for mixed-frequency nowcasting in 2.2.5.

2.2.1 Introduction

Headline economic indicators are often released with a lag, which means that policy makers and econometricians usually have to wait an undesirably long time for information on economic activity. For example, the Quarterly National Accounts (QNA) release of UK GDP is around 13 weeks after the end of the reference quarter. Nowcasting is defined as the prediction of the present, the very near future and the very recent past (Bańbura et al., 2011) and has become a very popular technique in recent years to quickly estimate current economic conditions. Rather than wait for an indicators official release, nowcasts can be made in the weeks leading up to this release using available information related to the indicator of interest. Take as an example the quarterly GDP publication from the QNA in Figure 2.0.1. While the goal of temporal disaggregation was to convert this entire series to a monthly frequency (Figure 2.1.1), nowcasting puts focus on the very end of the time series and attempts to predict the next quarterly observation not published yet, e.g. in this case the figure for Q1 2023. Figure 2.2.1 is a visual representation of the nowcasting task we might face in this scenario. It assumes we are currently in March 2023, where the Q4 2022 figure for GDP has just been released from the QNA. The current quarter figure (Q1 2023) will not be released

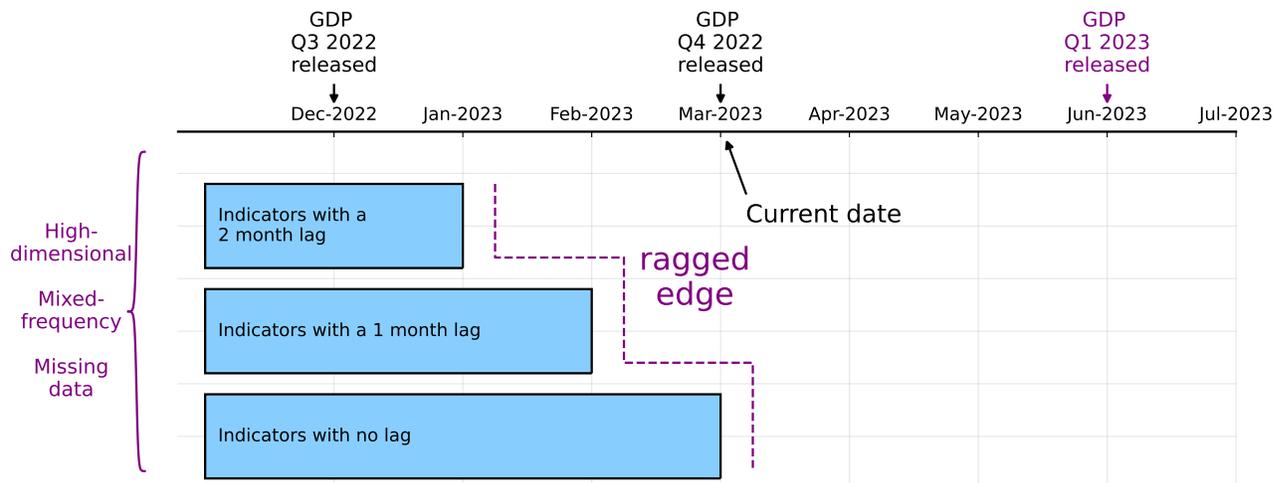


Figure 2.2.1: Visual representation for the task of nowcasting Q1 2023 GDP.

until June and therefore to avoid this delay, we can use related information (indicators) available to us in this quarter (e.g. business surveys, inflation indices, unemployment rate, etc...) along with the previous published quarters of GDP to construct a nowcast of Q1 2023. The idea is, as we move forward month by month (or week by week) to June, we obtain more information on Q1 GDP, whether it be from new data releases or data revisions of indicators. This way we can iteratively update our nowcast, hopefully becoming more accurate each time, and also track the so called ‘news’ component of indicator information as we learn more from one revision to the next.

Figure 2.2.1 also highlights the challenges one might face when performing the nowcasting task. The first problem is to deal with the “curse of dimensionality” when in situations with relatively more indicators than observations of GDP. As discussed, practitioners now have an abundance of potentially useful information from all different areas of the economy they may wish to model and hence this high dimensional situation is a likely one. The second problem is how one deals with mixed-frequency data. In the QNA GDP example, the response is quarterly yet the indicators are likely to be at a monthly frequency (or higher) to incorporate as much information as possible when constructing nowcasts. In the rest of this section we see methods to deal with mixed-frequency data, whether it be from simple aggregation to the observed low-frequency or

more complex state-space frameworks modelling everything at the highest frequency. A final problem which is very common with economic time series applications is the so-called ‘ragged edge’ problem – how to deal with missing data at the end of the sample caused by indicator series having different publication lags. As we see in Figure 2.2.1, the example gives a mixture of indicators with up to 2 months publication lag and hence a ragged edge is formed of information that are observed in Q1. A nowcasting model should efficiently be able to deal with this pattern of missing data, as well as potentially arbitrary patterns of missing data throughout the sample.

Notation

To introduce notation to the nowcasting task, without loss of generality we can assume we observe a collection of quarterly variables denoted by $\bar{\mathbf{y}}_{i,\tau}$ for variables $i = 1, \dots, m$ and quarters $\tau = 1, \dots, n_l$, and a collection of monthly indicators series $\mathbf{X}_{j,t}$ for variables $j = 1, \dots, p_x$ and months $t = 1, \dots, n$, and wish to obtain a nowcast for the next quarter of the targets $\bar{\mathbf{y}}_{i,n_l+1}$. For the UK GDP example in Figure 2.2.1, we have $m = 1$ and $n_l + 1 = \text{Q1 2023}$. Let us assume we update our nowcasts for the quarter of interest every month. In practice, it might be that a practitioner updates their nowcasts weekly, or even daily, as new information on indicators are released in real-time. This new set of data release is known as a ‘vintage’ of data occurring at vintage time v . For the explanation of nowcasting and the models discussed, let us just assume the vintages occur monthly, i.e. v represents a month.

Formally, the nowcast at a month n aims to obtain a projection

$$\mathbb{E}[\bar{\mathbf{y}}_{i\tau} \mid \Omega_n],$$

of the quarterly target onto the available information set Ω_n defined by

$$\Omega_n = \left\{ \mathbf{X}_{jt} : t = 1, \dots, n - l_j, j = 1, \dots, p_x ; \right. \\ \left. \bar{\mathbf{y}}_{i\tau} : \tau = 1, \dots, n_l, i = 1, \dots, m \right\},$$

where l_j is the lag (in months) for indicator j . The number of low frequency observations will be $n_l = \lfloor (n - l_y)/3 \rfloor$ for a quarterly target with monthly indicators with l_y being the lag (in months) of the quarterly target, e.g. for UK GDP from QNA we have $l_y = 3$.

Of course, the least demanding approach to the nowcasting task is to nowcast $\bar{\mathbf{y}}_{i,n_l+1}$ using only its own past, $\{\bar{\mathbf{y}}_{i,\tau}\}$ for $\tau = 1, \dots, n_l$, e.g. via ARIMA models or exponential smoothing. This approach can only capture a limited amount of information at the low-frequency and it is not possible to update nowcasts between published low-frequency periods (i.e. between two instances of GDP publication from the QNA). The following sections provide a review of popular nowcasting methods that make full use of all information in Ω_n , allowing intra-quarter updates of nowcasts.

2.2.2 Bridge Equations

Traditionally, applied econometricians at central banks and government agencies would use regression-based models, referred to as ‘bridge equations’ (Baffigi et al., 2004), for the task of nowcasting. The nowcasting variable(s) of interest (e.g. quarterly GDP) is linked, or bridged, in some way to a temporal aggregation of the higher frequency indicators. Therefore, this can be classed as a two-step approach where in the first step indicators, \mathbf{X}_t , are aggregated to the low-frequency, $\bar{\mathbf{X}}_\tau$. This could be a standard aggregation as discussed in Section 2.1.2 of data recorded at their original level, e.g. monthly flow data summed to be quarterly:

$$\bar{\mathbf{X}}_\tau = \sum_{i=0}^2 \mathbf{X}_t,$$

or it could be an aggregation of transformed data, usually made to ensure stationarity of the data fitted in models that assume stationarity, e.g. the monthly-to-quarterly sum of first-differenced data following:

$$\begin{aligned}
\Delta \bar{\mathbf{X}}_\tau &= \bar{\mathbf{X}}_\tau - \bar{\mathbf{X}}_{\tau-1} \\
&= (\mathbf{X}_t + \mathbf{X}_{t-1} + \mathbf{X}_{t-2}) - (\mathbf{X}_{t-3} + \mathbf{X}_{t-4} + \mathbf{X}_{t-5}) \\
&= (\mathbf{X}_t - \mathbf{X}_{t-1}) \\
&\quad + (\mathbf{X}_{t-1} - \mathbf{X}_{t-2}) + (\mathbf{X}_{t-1} - \mathbf{X}_{t-2}) \\
&\quad + (\mathbf{X}_{t-2} - \mathbf{X}_{t-3}) + (\mathbf{X}_{t-2} - \mathbf{X}_{t-3}) + (\mathbf{X}_{t-2} - \mathbf{X}_{t-3}) \\
&\quad + (\mathbf{X}_{t-3} - \mathbf{X}_{t-4}) + (\mathbf{X}_{t-3} - \mathbf{X}_{t-4}) \\
&\quad + (\mathbf{X}_{t-4} - \mathbf{X}_{t-5}) \\
&= \Delta \mathbf{X}_t + 2\Delta \mathbf{X}_{t-1} + 3\Delta \mathbf{X}_{t-2} + 2\Delta \mathbf{X}_{t-3} + \Delta \mathbf{X}_{t-4},
\end{aligned}$$

with $t = 3\tau$ for $\tau = 1, \dots, n_l$ and first-difference operator Δ defined as $\Delta x_t = x_t - x_{t-1}$. For many nowcasting applications of GDP, it is the growth rate being tracked involving a log-difference transform, $\Delta \log(\bar{\mathbf{y}}_\tau)$. In this case authors (Aruoba et al., 2009; Bańbura et al., 2011; Bańbura and Rünstler, 2011) have used the approximation by Mariano and Murasawa (2003) that roughly equates the arithmetic and geometric means:

$$\frac{1}{3} (\mathbf{y}_t + \mathbf{y}_{t-1} + \mathbf{y}_{t-2}) \approx \sqrt[3]{\mathbf{y}_t \mathbf{y}_{t-1} \mathbf{y}_{t-2}},$$

while others (Proietti, 2006; Proietti and Moauro, 2006; Proietti, 2011b) have imposed exact aggregation using a non-linear smoothing algorithm.

Once indicators are aggregated to the low-frequency, the second step involves bridging the low-frequency variable(s), $\bar{\mathbf{y}}_\tau$ to the aggregated indicators using a bridge equation of the form:

$$\bar{\mathbf{y}}_\tau = g(\bar{\mathbf{X}}_\tau) + \epsilon_\tau,$$

where ϵ_τ is an error process and $g(\cdot)$ is a link function that may involve a distribution of indicator lags, e.g.

$$g(\bar{\mathbf{X}}_\tau) = \sum_{l=0}^L \beta_l \bar{\mathbf{X}}_{\tau-l},$$

with β_l the coefficients to be estimated for lags $l = 0, 1, \dots, L$. It is also possible for the bridge equation to contain lags of the response $\bar{\mathbf{y}}_\tau$. In Section 2.1.4 we saw a bridge equation with $L = 0$ to obtain GLS estimates of β in Chow-Lin temporal disaggregation (Chow and Lin, 1971).

Drawbacks

While the bridge model approach is popular among NSIs and central banks due to its simplicity and ease of computation (Baffigi et al., 2004; Diron, 2008; Eurostat, 2017), it is not without its drawbacks. Bridge equations assume that the measurements are complete (fully observed) at the point of aggregation, for example, in temporal disaggregation we assumed observed indicators over $t = 1, \dots, n = 3n_l$ with $\bar{\mathbf{y}}_{n_l}$ observed. This assumption does not implicitly deal with the ragged edge problem typically faced in nowcasting applications when tracking a target in real-time. The missing data in the ragged edge could be initially predicted using univariate forecasting models (e.g. ARIMA) on each indicator series beforehand, however, this would of course lead to a large number of parameters needing to be estimated when the number of indicators is large. In addition to not being able to implicitly deal with the ragged edge, bridge equation models working at the low-frequency by temporal aggregation may potentially lead to a loss of information as the dynamics of the aggregate model can be rather different to that of the high-frequency model (Marcellino, 1999). The remaining methods model everything at the high-frequency by linking $\bar{\mathbf{y}}_\tau$ directly to \mathbf{X}_t , thus avoiding the potential loss of information from temporal aggregation.

2.2.3 Mixed Data Sampling

Pioneered by Ghysels et al. (2004), and reviewed by Ghysels et al. (2007) and Andreou et al. (2011), the Mixed Data Sampling (MIDAS) model is a nowcasting approach directly linking the low-frequency target $\bar{\mathbf{y}}_\tau$ to the high-frequency indicators \mathbf{X}_t via a single equation given by

$$\bar{\mathbf{y}}_\tau = \sum_{j=1}^{p_x} \beta_j \sum_{i=0}^{k-1} \phi_j(i; \boldsymbol{\theta}_j) L^i X_{j,k\tau} + \boldsymbol{\epsilon}_\tau, \quad (2.2.1)$$

where k is the number of high-frequency observations per low-frequency period (e.g. $k = 3$ when t are months and τ are quarters) and L^i is the lag operator such that $L^i X_{j,k\tau} = X_{j,k\tau-i}$. Note the use of $k\tau$ instead of t to ensure the correct months are summed. MIDAS is essentially taking a weighted sum (weighted via $\phi_j(i; \boldsymbol{\theta}_j)$) of the high-frequency indicators over every low-frequency period of $\bar{\mathbf{y}}_\tau$ and linking them with a single coefficient β_j for each indicator. The role of β_j is to capture the overall effect of an indicator $X_{j,\cdot}$ via $\bar{\mathbf{y}}_\tau$. Identification of β_j is ensured by the normalisation constraint $\sum_{i=0}^{k-1} \phi_j(i; \boldsymbol{\theta}_j) = 1$. To avoid the proliferation of parameters, the weighting function $\phi_j(i; \boldsymbol{\theta}_j)$ tends to take simple, parameterised forms such as the exponential Almon function and the Beta function given by:

$$\text{Exponential Almon Function:} \quad \phi_j(i; \boldsymbol{\theta}_j) = \frac{\exp(\theta_{1j}i + \theta_{2j}i^2)}{\sum_{h=0}^{k-1} \exp(\theta_{1j}h + \theta_{2j}h^2)}.$$

$$\text{Beta Function:} \quad \phi_j(i; \boldsymbol{\theta}_j) = \frac{f(\frac{i}{k}; \theta_{1j}, \theta_{2j})}{\sum_{h=0}^{k-1} f(\frac{h}{k}; \theta_{1j}, \theta_{2j})},$$

where

$$f(x; \theta_{1j}, \theta_{2j}) = \frac{x^{\theta_{1j}-1} (1-x)^{\theta_{2j}-1} \Gamma(\theta_{1j} + \theta_{2j})}{\Gamma(\theta_{1j}) \Gamma(\theta_{2j})},$$

and

$$\Gamma(\theta_q) = \int_0^\infty e^{-x} x^{\theta_q-1} .dx$$

is the standard Gamma function. In both functions $\boldsymbol{\theta}_j = (\theta_{1j}, \theta_{2j})$ are hyperparameters governing the shape of the weighting function. They are very flexible allowing lags that can decline slowly or fast, or even have a hump shape.

The parameters of the MIDAS regression (2.2.1) can be estimated using non-linear least squares (Ghysels et al., 2004) and then used to nowcast the current quarter:

$$\hat{\mathbf{y}}_{\tau+1} = \sum_{j=1}^{p_x} \hat{\beta}_j \sum_{i=0}^{k-1} \phi_j(i; \hat{\boldsymbol{\theta}}_j) L^i X_{j, k\tau+k-l_j},$$

that utilises information in the ragged edge of the final quarter for the indicators with lags l_j . It is possible to add lags of the indicators to the MIDAS regression (2.2.1), however, this would quickly increase the number of parameters to be estimated. For example, if l lags are added, then the sum from 0 to $k - 1$ would need to be done l times for l different versions of $\boldsymbol{\theta}_j$.

Several authors have considered extensions to the classic MIDAS. Clements and Galvão (2009) introduce autoregressive terms of the response and indicators using a common factor (AR-MIDAS). Foroni et al. (2015) avoid the use of the exponential weighting function and instead consider an unrestricted lag polynomial of order $k - 1$ that can be simply estimated via OLS (U-MIDAS). Ghysels et al. (2007) generalise the non-linear weighting function with an exponential generalized autoregressive conditional heteroscedastic (EGARCH) model useful for financial applications. Galvão (2013) combine the MIDAS with a smooth transition regression to allow a time-varying structure that can make changes in the predictive power of the indicators. Guérin and Marcellino (2013) incorporate regime changes in the parameters of the MIDAS model by incorporating an ergodic Markov-chain with a finite number of states.

Drawbacks

The obvious drawback of the MIDAS approach is the amount of parameters that need to be estimated – for p_x indicator series considered, this leads to p_x MIDAS models, which each may already have a substantial amount of parameters if the temporal aggregation ratio is large and/or several lags are used. The MIDAS approach is therefore not practical for the desired scenario of high-dimensional p_x . Other concerns are that the MIDAS (2.2.1) is set up for a univariate $\bar{\mathbf{y}}_\tau$ – multivariate output again adds to the proliferation of parameters. Also, missing data in the ragged edge is handled by just shifting the end-point of each indicator series to the most recent observed value and no consideration of the cross-sectional dynamics between indicators is taken into account that can potentially lead to accurate estimation of this missing data.

Due to these aforementioned shortcomings, econometricians in recent years have turned to state-space modelling approaches and in particular the use of dynamic factor models (DFMs). The state-space framework is very flexible and provides a basis for Kalman filtering and smoothing techniques to be used that can compactly estimate any missing data present. This naturally allows any low-frequency variables to be modelled at the high-frequency by writing the low-frequency variable as a partially observed high-frequency series, for example, a quarterly series can be written as a monthly series with every 3rd position observed and everywhere else missing. Latent states of the multivariate framework can capture dynamics of a large number of variables and exploit the cross-section when nowcasting in the ragged edge.

The next section provides an overview of DFMs and popular estimation strategies, before Section 2.2.5 presents how DFMs are used for nowcasting purposes. For other reviews of DFMs see Stock and Watson (2011) and Poncela et al. (2021). Alternative popular state-space approaches in large-scale macroeconomic problems not considered in this thesis include the use of mixed-frequency VAR models (Kuzin et al., 2011; Schorfheide and Song, 2015; Koop et al., 2020b) and dynamic stochastic general equilibrium models

(Kremer et al., 2006; Forni and Marcellino, 2014b).

2.2.4 Dynamic Factor Models

Originally formalised by Geweke (1977), the premise of the dynamic factor model (DFM) is to assume that the common dynamics of a large number of stationary zero-mean time series $\mathbf{X}_t = (X_{1,t}, \dots, X_{p,t})^\top$ stem from a relatively small number of unobserved (latent) factors $\mathbf{F}_t = (F_{1,t}, \dots, F_{r,t})^\top$ where $r \ll p$ through the linear system

$$\mathbf{X}_t = \mathbf{\Lambda} \mathbf{F}_t + \boldsymbol{\epsilon}_t, \quad (2.2.2)$$

for observations $t = 1, \dots, n$. The $p \times r$ matrix $\mathbf{\Lambda}$ provides a direct link between each factor in \mathbf{F}_t and each variable in \mathbf{X}_t . The larger the loading $|\Lambda_{i,j}|$ for variable i and factor j , the more correlated this variable is with the factor. The loading parameter is pivotal when trying to understand factor structure and is the main topic of discussion in Chapter 4. The common component $\boldsymbol{\chi}_t = \mathbf{\Lambda} \mathbf{F}_t$ captures the variability in the time series variables that is due to the common factors, while the idiosyncratic errors $\boldsymbol{\epsilon}_t = (\epsilon_{1,t}, \dots, \epsilon_{p,t})^\top$ capture the features that are specific to individual series, such as measurement error. Interpreting the common and idiosyncratic components is important when trying to understand data. For example, in an econometric application, it might be that the common component represents an underlying business cycle, while the idiosyncratic component may represent shocks specific to individual industries or regions (Kose et al., 2008; Forni and Gambetti, 2021).

It is generally assumed the idiosyncratic errors $\boldsymbol{\epsilon}_t$ are zero-mean and cross-sectionally uncorrelated, meaning their covariance matrix, denoted by $\boldsymbol{\Sigma}_\epsilon$, is diagonal. Models with these assumptions are termed *exact*. Under this assumption, the correlation of one series with another occurs only through the latent factors \mathbf{F}_t . When this assumption is relaxed, and the idiosyncratic errors are allowed to be (weakly) cross-correlated,

termed an *approximate* DFM, consistent estimation of the factors is still possible as $(n, p) \rightarrow \infty$ (Doz et al., 2011). Therefore, the ‘curse of dimensionality’ problem, often a burden for analysing time series models, is beneficial when estimating approximate DFMs (Barigozzi and Luciani, 2022).

Canonically, the dynamics of the latent factors in the DFM are specified as a stationary VAR(1) model:

$$\mathbf{F}_t = \mathbf{A}\mathbf{F}_{t-1} + \mathbf{u}_t, \quad (2.2.3)$$

where \mathbf{u}_t is a zero-mean series of disturbances with covariance matrix Σ_u . The measurement equation (2.2.2) along with the state equation (2.2.3) form a state space model described as a DFM. Three popular estimation techniques are now presented.

Principal Components Analysis (PCA)

A simple approach to estimate factor loadings, $\mathbf{\Lambda}$, is to consider the first r eigenvectors of the sample covariance matrix of \mathbf{X} associated with the largest r eigenvalues, essentially applying PCA to the time series. The factors are the principal components described by $\mathbf{F}_t = p^{-1}\mathbf{\Lambda}^\top \mathbf{X}_t$. This approach has been extensively reviewed in the literature (Stock and Watson, 2002; Bai, 2003; Doz and Fuleky, 2020). When mild conditions are placed on the correlation structure of idiosyncratic errors, such that they are assumed independent and exhibit no contemporaneous or serial correlation, the PCA estimator is the optimal non-parametric⁴ estimator for a large approximate DFM. With even tighter conditions of spherical idiosyncratic components, i.e. they are white noise Gaussian, then the PCA estimator is equivalent to the maximum likelihood estimator (Doz and Fuleky, 2020). The problem with using non-parametric PCA methods to estimate DFMs is that there is no consideration of the dynamics of the factors or idiosyncratic components. In particular, there is no feedback from the estimation of the state equation (2.2.3) to the measurement equation (2.2.2). For this reason, it is preferable to use parametric

⁴In the sense that temporal dependence is not restricted to that encoded via a parametric model.

methods that are able to account for temporal dependencies in the system.

Two-step Framework: PCA + Kalman filter and smoother

Giannone et al. (2008) proposed a two-step framework for estimating DFMs which has since been theoretically studied by Doz et al. (2011) and successfully applied to the field of nowcasting. This method involves the following two steps:

- Step 1: Preliminary estimates of the loadings $\mathbf{\Lambda}$ and the factors \mathbf{F}_t are found via PCA as described above. Σ_ϵ is estimated as the empirical covariance of $\hat{\epsilon}_t = \mathbf{X}_t - \hat{\mathbf{\Lambda}}^\top \hat{\mathbf{F}}_t$. The remaining parameters are estimated by fitting a VAR(1) model on $\hat{\mathbf{F}}_t$.
- Step 2: The model is cast into state space form (2.2.2)-(2.2.3) and the factors are re-estimated using the Kalman filter and smoother (Shumway and Stoffer, 1982; Koopman and Durbin, 2000).

The Kalman filter enables the factor estimates to be continually updated as new observations become available. This way, we are able to assess the impact new data releases of certain variables have in the nowcasting process, i.e. track the ‘news’. The Kalman filter provides the basis for smoothing, where the factor at a time t can be estimated based on all data in the sample. This is very helpful for handling missing data, whether it be backcasting missing at the start of the sample, forecasting missing data at the end of the sample in the ragged edge or interpolating arbitrary patterns of missing data throughout the sample.

Appendix A.2 provides the classic multivariate Kalman filter and smoother (KFS) equations of Shumway and Stoffer (1982), often used in DFM estimation. Also provided is the univariate treatment (sequential processing) of the multivariate equations for fast Kalman filter and smoother seen in Koopman and Durbin (2000) that can lead to substantial computational gains for exact DFMs. As it is assumed in exact DFMs that Σ_ϵ is diagonal, it becomes possible to filter the observations \mathbf{X}_t one element at a

time, as opposed to altogether as in the classic multivariate approach. This way, matrix inversions become scalar divisions and thus significant computational gains are possible.

Expectation Maximisation (EM)

Bañbura and Modugno (2014) build on the DFM representation of Watson and Engle (1983) and adopt an expectation maximisation (EM) approach to estimate the system (2.2.2)-(2.2.3) by quasi-maximum likelihood estimation (QMLE)⁵. Assuming the error processes $\{\epsilon_t\}$ and $\{u_t\}$ are Gaussian, and collecting all parameters of the DFM in $\theta = (\Lambda, \mathbf{A}, \Sigma_\epsilon, \Sigma_u)$, the idea is to write the joint log-likelihood of \mathbf{X}_t and \mathbf{F}_t for all $t = 1, \dots, n$ in terms of both the observed and unobserved data as:

$$\begin{aligned} \log \mathcal{L}(\mathbf{X}, \mathbf{F}; \theta) &= -\frac{1}{2} \log |\mathbf{P}_0| - \frac{1}{2} (\mathbf{F}_0 - \alpha_0)^\top \mathbf{P}_0^{-1} (\mathbf{F}_0 - \alpha_0) \\ &\quad - \frac{n}{2} \log |\Sigma_u| - \frac{1}{2} \sum_{t=1}^n \mathbf{u}_t^\top \Sigma_u^{-1} \mathbf{u}_t \\ &\quad - \frac{n}{2} \log |\Sigma_\epsilon| - \frac{1}{2} \sum_{t=1}^n \epsilon_t^\top \Sigma_\epsilon^{-1} \epsilon_t, \end{aligned} \quad (2.2.4)$$

where $\epsilon_t = \mathbf{X}_t - \Lambda \mathbf{F}_t$, $\mathbf{u}_t = \mathbf{F}_t - \mathbf{A} \mathbf{F}_{t-1}$, and the factors are assumed to have an initial distribution at $t = 0$ of $\mathbf{F}_0 \sim \mathcal{N}(\alpha_0, \mathbf{P}_0)$. The EM algorithm then involves iterating between two steps:

E-Step: Compute the expectation of the joint log-likelihood (2.2.4) conditional on the available information up to n , denoted by Ω_n , using the parameters estimated in the previous iteration, j :

$$\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \theta^{(j)}) | \Omega_n] ;$$

M-Step: Re-estimate the parameters through the maximisation of the conditional expected

⁵It is described as quasi-maximum likelihood as consistency results are robust to cross-sectional misspecification, serial correlation of idiosyncratic components, and non-Gaussianity.

joint log-likelihood:

$$\boldsymbol{\theta}^{(j+1)} = \arg \max_{\boldsymbol{\theta}} \mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}^{(j)}) | \Omega_n] ,$$

until convergence. A popular EM convergence criteria presented in Doz et al. (2012) is to say the algorithm has converged when the value

$$M_j = \frac{\log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j)}) - \log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j-1)})}{\left(\log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j)}) + \log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j-1)}) \right) / 2} ,$$

becomes less than some chosen threshold. The log-likelihood $\mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j)})$ is presented in Appendix A.2.

The E-step, involving computation of the conditional expected joint log-likelihood can be directly obtained from the KFS equations discussed in the two-step framework. The QMLE estimators in the M-step are shown to be constructed using state expectations and covariances again obtained from the KFS equations. Details of this M-step is discussed in Chapter 4 where an extension to the EM framework is presented to allow for sparse DFMs, i.e. DFMs with the loading matrix $\mathbf{\Lambda}$ assumed sparse. This chapter also details methods to tune for the number of factors r in the DFM.

Doz et al. (2012) and Barigozzi and Luciani (2022) provide consistency results of the EM approach as $(n, p) \rightarrow \infty$. It is proven under these settings that the estimated loadings, $\hat{\mathbf{\Lambda}}$, are \sqrt{n} -consistent and asymptotically normal if $\sqrt{n}/p \rightarrow 0$, the estimated factors $\hat{\mathbf{F}}$ are \sqrt{p} -consistent and asymptotically normal if $\sqrt{p}/n \rightarrow 0$, and the estimated common component, $\hat{\mathcal{X}} = \hat{\mathbf{\Lambda}}\hat{\mathbf{F}}^\top$, is $\min(\sqrt{n}, \sqrt{p})$ -consistent and asymptotically normal regardless of the relative rate of divergence of n and p (Barigozzi and Luciani, 2022). In terms of macroeconomic forecasting, the EM algorithm approach has many gains over the PCA and two-step procedures. It is much more efficient in small samples, as shown by simulation studies of Doz et al. (2012). It is a very flexible framework able to

deal with arbitrary patterns of missing data in \mathbf{X}_t , as shown in Bańbura and Modugno (2014). It enables one to impose restrictions on parameters when updating them in the EM algorithm. For example, Bańbura et al. (2011) impose restrictions on the loadings to reflect temporal aggregation and introduce factor specific groups of variables in their nowcasting application. Furthermore, it is a computationally efficient algorithm allowing fast forecasts for policy makers/analysts, as opposed to computationally demanding Markov Chain Monte Carlo (MCMC) based alternatives.

2.2.5 Nowcasting with Dynamic Factor Models

The ability to handle high-dimensions, interpolate arbitrary patterns of missing data throughout the sample and make the most of available data in the ‘ragged edge’ at the end of the sample, has made DFMs very popular amongst the nowcasting literature. Giannone et al. (2008) first applied DFMs for the nowcasting task when nowcasting GDP growth at the Board of Governors of the Federal Reserve. Other central banks and statistical institutes have since followed (Matheson, 2011; Schumacher, 2011; Rünstler, 2016; Castle et al., 2017). Giannone et al. (2008) take a two-step approach whereby in the first step, common factors, \mathbf{F}_t , are extracted from a large monthly indicator set using the system (2.2.2)-(2.2.3), and in the second step, these factors are aggregated to a quarterly frequency and regressed on quarterly GDP to obtain current-quarter nowcasts of GDP growth:

$$\hat{\mathbf{y}}_\tau = \hat{\boldsymbol{\beta}}\hat{\mathbf{F}}_\tau,$$

where $\bar{\mathbf{y}}_\tau$ is quarterly GDP growth, $\hat{\mathbf{F}}_\tau$ are the quarterly aggregated correspondent of $\hat{\mathbf{F}}_t$ and $\boldsymbol{\beta} \in \mathbb{R}^{m \times r}$ are regression coefficients. This two-step approach can be viewed as a bridge approach, the quarterly target is bridged onto a temporal aggregation of the factors. Similarly, one could simply bridge estimated factors at the monthly frequency in a MIDAS regression as in Marcellino and Schumacher (2010).

Alternatively, instead of nowcasting low-frequency targets after estimation of factors,

one could jointly model low-frequency targets with high-frequency indicators together in a single, coherent state-space (DFM) framework. This allows factor estimates and nowcasts of the target series to be obtained simultaneously, bringing in the dynamics of the entire cross-section of data available and also the ability to perform temporal disaggregation. The idea is to assume that the completely unobserved monthly version of the target (e.g. monthly GDP), denoted by \mathbf{y}_t , $t = 1, \dots, n$, admits the same factor model representation as the DFM assumed for the monthly indicators (2.2.2)-(2.2.3):

$$\mathbf{y}_t = \mathbf{\Lambda}_y \mathbf{F}_t + \boldsymbol{\epsilon}_t^{(y)}, \quad \boldsymbol{\epsilon}_t^{(y)} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}^{(y)}}), \quad (2.2.5)$$

where $\mathbf{\Lambda}_y \in \mathbb{R}^{m \times r}$ and $\boldsymbol{\epsilon}_t^y \in \mathbb{R}^{m \times 1}$ are factor loadings and idiosyncratic errors specific to variables in \mathbf{y} , while $\mathbf{\Lambda} \in \mathbb{R}^{p_x \times r}$ and $\boldsymbol{\epsilon}_t \in \mathbb{R}^{p_x \times 1}$ from (2.2.2) are factor loadings and idiosyncratic errors specific to the monthly indicators \mathbf{X} . To make this assumption, we require \mathbf{y}_t to be (weakly) stationary and hence \mathbf{y}_t is usually the first-differenced or log-first-differenced (for growth rates) transformation of the monthly target at its original level, i.e. $\mathbf{y}_t = \Delta \mathbf{Y}_t$ or $\mathbf{y}_t = \Delta \log(\mathbf{Y}_t)$, where \mathbf{Y}_t is the (non-stationary) monthly target at its original level.

Writing $\bar{\mathbf{y}}_t$, $t = 1, \dots, n$, as the partially observed monthly series which has the quarterly observations of $\bar{\mathbf{y}}_\tau$ assigned to every 3rd month, $t = 3\tau$, and is missing everywhere else, this can be expressed as the sum of its unobserved monthly contributions:

$$\bar{\mathbf{y}}_t = \begin{cases} \mathbf{y}_t + \mathbf{y}_{t-1} + \mathbf{y}_{t-2}; & \text{for } t = 3, 6, 9, \dots \\ \text{unobserved;} & \text{otherwise.} \end{cases}$$

The stationary version of this partially observed monthly series can be found by taking first-differences or log-first-differences, i.e. $\bar{\mathbf{Y}}_t = \Delta \bar{\mathbf{y}}_t$ or $\bar{\mathbf{Y}}_t = \Delta \log(\bar{\mathbf{y}}_t)$, where $\bar{\mathbf{Y}}_t$ is

the stationary version, and is linked to the differenced latent monthly data, \mathbf{y}_t , via

$$\bar{\mathbf{Y}}_t = \mathbf{y}_t + 2\mathbf{y}_{t-1} + 3\mathbf{y}_{t-2} + 2\mathbf{y}_{t-3} + \mathbf{y}_{t-4}, \quad (2.2.6)$$

derived in Section 2.2.2 and shown to be an approximation if log-differences are used (Mariano and Murasawa, 2003).

Bringing equations (2.2.5) and (2.2.6) together, it is possible to write:

$$\bar{\mathbf{Y}}_t = [\Lambda_y \ 2\Lambda_y \ 3\Lambda_y \ 2\Lambda_y \ \Lambda_y][\mathbf{F}_t \ \cdots \ \mathbf{F}_{t-4}]^\top + [1 \ 2 \ 3 \ 2 \ 1][\boldsymbol{\epsilon}_t^{(y)} \ \cdots \ \boldsymbol{\epsilon}_{t-4}^{(y)}]^\top. \quad (2.2.7)$$

The seminal paper by Bańbura et al. (2011), that has become the industry standard for nowcasting across the Euro area, use the measurement (2.2.7) along with the DFM of the indicators (2.2.2)-(2.2.3) to construct a new state space framework with the joint measurement $(\mathbf{X}_t, \bar{\mathbf{Y}}_t)$ and the latent state $(\mathbf{F}_t, \dots, \mathbf{F}_{t-4}, \boldsymbol{\epsilon}_t, \boldsymbol{\epsilon}_{t-4}^{(y)}, \dots, \boldsymbol{\epsilon}_{t-4}^{(y)})$. See Bańbura et al. (2011) for full details of this state space framework.

This state space framework of Bańbura et al. (2011) is set up with the objective to nowcast quarterly GDP growth rate, i.e. predict the latest quarterly value of $\bar{\mathbf{Y}}_t$. Other authors (Angelini et al., 2010; Bańbura and Rünstler, 2011; Proietti, 2011a) demonstrate how the nowcasting framework can also be adapted for the purpose of temporal disaggregation, i.e. the estimation of \mathbf{y}_t for $t = 1, \dots, n$, by treating the disaggregate series as latent states in the state vector. When \mathbf{y}_t represents growth rates, the approximate temporal aggregation (2.2.6) is converted into a systematic sampling problem by constructing a cumulator variable defined by the recursive formula:

$$\bar{\mathbf{Y}}_t^c = \rho_t \bar{\mathbf{Y}}_{t-1} + \frac{1}{3} \mathbf{y}_t$$

where $\rho_t = 0$ when t is the first month of a quarter and $\rho_t = 1$ for all other months (Harvey, 1990). See Bańbura and Rünstler (2011) for full details of the state space

framework to obtain \mathbf{y}_t . Proietti (2011a) consider a framework obtaining the disaggregate monthly series in its original level, i.e. they obtain \mathbf{Y}_t for $t = 1, \dots, n$. This is achieved by iterating between an EM algorithm for DFM estimation and a non-linear smoothing algorithm for estimation of \mathbf{Y}_t . The non-linear smoothing algorithm implements exact aggregation constraints when data in logarithms are used, rather than the Mariano and Murasawa (2003) approximation in equation (2.2.6). Chapter 5 proposes a joint state space framework incorporating a sparse dynamic factor model with an application to disaggregate quarterly UK Trade in Services data and nowcast it at the monthly level.

Chapter 3

Sparse Temporal Disaggregation

Temporal disaggregation is a method commonly used in official statistics to enable high-frequency estimates of key economic variables, such as GDP. The generalised least-squares regression approach (Chow and Lin, 1971), introduced in Chapter 2.1.4, have proven to work well when only a couple of indicator series are used for the disaggregation task. However, with the integration of alternative and administrative datasets, the collection of indicator series that can potentially provide important information on the output is vast and we find ourselves operating in a so-called *high-dimensional* setting. In this case, classical methods for temporal disaggregation demonstrably fail. It is precisely this gap in the temporal disaggregation literature that this chapter seeks to address: how to build a robust and interpretable methodology that can accurately disaggregate key survey based statistics by utilizing large numbers of alternative and administrative data sources. The motivation for high-dimensional temporal disaggregation methods from the perspective of performing high-frequency disaggregation for UK national GDP are laid out in Section 3.1, ending with an overview of the chapter's structure.

3.1 Introduction

The methodology presented in this chapter is motivated by the challenging task of performing high frequency disaggregation for UK national GDP, moving from a quarterly to a monthly resolution. A monthly estimate for GDP should accurately capture the monthly dynamics between quarters while remain temporally consistent with published quarterly figures. For this task, there are a considerable number of indicator series that one may wish to use to help indicate the monthly dynamics of GDP. The application in Section 3.5 considers a mixture of traditional, survey-based indicators, including the Monthly Business Survey (MBS) for turnover in the production and service industries, alongside Value Added Tax (VAT) returns data, retail sales indices, and several novel indicators such as traffic flows at ports and on roads. In total, 97 indicators are considered, all of which are collected at a monthly frequency.

Given the significant interest in fast measurements of economic activity, the ONS has developed a monthly GDP index, and published this statistic since May 2018. This index is constructed using monthly information from the Index of Services, Index of Production and Index of Construction. For details see ONS (2022b). Even though a monthly index exists in this case, there is still great interest in performing temporal disaggregation, the reasons are threefold. Firstly, the monthly index is an output based measure, however economists may also be interested in both expenditure and income based estimates. Since, temporal disaggregation can be applied to any output stream, either expenditure or income based measures could be used. The resulting high-frequency estimate can thus complement the existing output based index. Secondly, due to the construction of the index, publication lags the period of measurement, an issue common to most economic statistics. However, temporal disaggregation can be used to find indicators that are relevant and updated more frequently, potentially enabling the estimation of the output statistic at a more frequent rate than is traditionally reported. This way, we are able to evaluate higher frequency estimates of the economic activity itself, rather

than an index, and obtain detailed short-term information at the end of the sample. National Statistical Institutes (NSIs) are actively developing so-called fast-indicators for exactly this purpose and in this application several of these are considered in the form of traffic data. Finally, one of the key issues surrounding the fast release of data is in understanding the associated short-term movements. To this end, temporal disaggregation using interpretable indicator series can provide insight by highlighting which indicators are driving movement.

Traditionally, NSIs have relied on the well-established regression-based approach of Chow and Lin (1971) and its extensions (Fernandez, 1981; Litterman, 1983; Cholette and Dagum, 1994; Mitchell et al., 2005; Proietti, 2006) to perform the task of temporal disaggregation. This approach is centred around a generalised least-squares (GLS) regression that regresses the unobserved high-frequency target series (e.g. monthly GDP) onto a set of high-frequency indicator series. These methods have relied on only a couple of high-frequency indicator series to produce estimates. However, the prevalence of large, and increasing, volumes of administrative and alternative data-sources motivates the need for such methods to be adapted for high-dimensional settings. In fact, we naturally find ourselves in this setting when we consider the GDP disaggregation challenge previously outlined (97 series over 50 quarters).

One way to deal with a high-dimensional set of indicator series would be to just select a limited set of indicators that are commonly known to have a strong relationship with the target based on economic sense. For example, just using a total aggregate for each of the Index of Production, Services and Construction as a small set of indicators. However, solely relying on economic sense may overlook complex interactions economic variables possess and leads to subjective judgements rather than selection based on empirical predictive performance. A second option is to make a selection based on standard indexes of correlations between the target and indicators or other classical approaches concerning information criteria as suggested in Ng (2013). As the number

of indicator series to consider becomes large, these classic approaches quickly become computationally insufficient and will likely overfit the data. A third option is to extract a few factors from the large indicator set by means of statistical methods like principal components analysis (PCA). The idea is that this new, low-dimensional set of factors are able to explain the majority of variance in the full indicator set and not contain idiosyncratic noise that arises from features that are specific to an individual indicator variable such as measurement error. Standard temporal disaggregation procedures can then be performed using these estimated common factors as predictors (Angelini et al., 2006; Labonne and Weale, 2020). Although such approaches can be effective, the factor model approach possesses difficulties regarding specification and interpretability. There exists uncertainty in how the appropriate factor estimation method is selected, and in the number of factors used (Kuzin et al., 2013). A more considerable issue is the loss of interpretation into the individual effects of indicators, as it is the factors or principal components, that are retained instead of original predictors.

Methodologically, this chapter tackles the high-dimensional temporal disaggregation problem from the variable selection viewpoint. In particular, the key contribution is to establish a regularised M-estimation framework, referred to as *Sparse Temporal Disaggregation* (spTD), that extends the Chow-Lin approach for high-dimensional indicator sets and is able to simultaneously select the relevant indicators and estimate their impact. This approach is the first work to tackle the disaggregation problem in this way. A key aspect of the method is the incorporation of a penalty function on the regression parameters which operates alongside the usual GLS cost function. These regularisation functions are able to take a variety of forms, and help ensure the estimator is stable in high-dimensional settings (Bühlmann and Van De Geer, 2011). To illustrate the effectiveness of the methodology, the focus is put on the popular ℓ_1 penalty (LASSO) (Tibshirani, 1996). This has the twin benefits of computational robustness due to convexity, and interpretation due to its ability to produce sparse

sets of regression coefficients. In addition, ℓ_1 penalisation has rapid, robust algorithms available to compute parameter estimates over a range of sparsity levels (Efron et al., 2004).

The remainder of the chapter is structured as follows. Section 3.2 recaps the well-established regression-based framework of Chow and Lin (1971) and explores shortcomings when working in high-dimensions. Section 3.3 proposes a sparse modelling framework for high-dimensional temporal disaggregation and provides details of an estimation strategy and tuning procedure for ℓ_1 penalisation. Section 3.4 highlights the advantages of the proposed methodology through a simulation study. In Section 3.5, the method is implemented on real quarterly UK GDP data to construct a monthly estimate and inform us on the most relevant indicators for this task. Section 3.6 extends the sparse modelling framework to allow for a stable selection of indicators over time and this is analysed in a second case study attempting to disaggregate the quarterly index of service turnover for transportation in Italy¹.

3.2 Shortcomings of the Classic Temporal Disaggregation Framework

In Chapter 2.1.4 the well-established regression-based temporal disaggregation framework of Chow and Lin (1971) was presented. This framework seeks to obtain a high-frequency unobserved time series, $\mathbf{y} \in \mathbb{R}^n$, from its observed low-frequency counterpart, $\bar{\mathbf{y}} \in \mathbb{R}^{n_l}$, using a collection of p observed high-frequency indicator series, $\mathbf{X} \in \mathbb{R}^{n \times p}$, such that the unobserved target series \mathbf{y} is temporally consistent with $\bar{\mathbf{y}}$ for every low-frequency period observed. It is assumed that $n = kn_l$, where k is the disaggregation ratio, for example, $k = 3$ for quarterly-to-monthly disaggregation and $k = 4$ for annual-to-

¹This second case study was undertaken as an independent research project subsequent to the publication of the remainder of Chapter 3. Consequently, the possibility of inconsistencies arising is acknowledged.

quarterly disaggregation. Suppose that the following linear regression model holds at the high-frequency:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u},$$

where $\boldsymbol{\beta} \in \mathbb{R}^{p \times 1}$ is a vector of unknown parameters and $\mathbf{u} \in \mathbb{R}^{n \times 1}$ is a residual vector with zero mean and covariance matrix \mathbf{V} . Chow and Lin (1971) derive that the best linear unbiased estimator of \mathbf{y} that is consistent with the temporal aggregation (defined in Section 2.1.2) constraint $\bar{\mathbf{y}} = \mathbf{C}\mathbf{y}$, is given by

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{V}\mathbf{C}\bar{\mathbf{V}}^{-1}(\bar{\mathbf{y}} - \bar{\mathbf{X}}\hat{\boldsymbol{\beta}}) \quad (3.2.1)$$

where $\bar{\mathbf{X}} = \mathbf{C}\mathbf{X}$ and $\bar{\mathbf{V}} = \mathbf{C}\mathbf{V}\mathbf{C}^\top$. It is noticed that:

1. $\hat{\boldsymbol{\beta}}$ is the GLS estimator of $\boldsymbol{\beta}$ for the fully observed aggregated model: $\bar{\mathbf{y}} = \bar{\mathbf{X}}\boldsymbol{\beta} + \bar{\mathbf{u}}$, where $\bar{\mathbf{u}}$ is the aggregated residual process with mean zero and covariance $\bar{\mathbf{V}}$. This GLS estimator is given by

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ \left\| \bar{\mathbf{V}}^{-1/2}(\bar{\mathbf{y}} - \bar{\mathbf{X}}\boldsymbol{\beta}) \right\|_2^2 \right\} \\ &= (\bar{\mathbf{X}}^\top \bar{\mathbf{V}}^{-1} \bar{\mathbf{X}})^{-1} \bar{\mathbf{X}}^\top \bar{\mathbf{V}}^{-1} \bar{\mathbf{y}}. \end{aligned} \quad (3.2.2)$$

2. The estimator in (3.2.1) depends on the assumed distribution of high-frequency residuals, \mathbf{u} . The variations regularly considered are:

- AR(1) model (Chow and Lin, 1971): $u_t = \rho u_{t-1} + \epsilon_t$, $|\rho| < 1$ and $\epsilon_t \sim N(0, \sigma_\epsilon^2)$;
- Random walk model (Fernandez, 1981): $u_t = u_{t-1} + \epsilon_t$, $\epsilon_t \sim N(0, \sigma_\epsilon^2)$;
- Random walk-Markov model (Litterman, 1983): $u_t = u_{t-1} + \epsilon_t$, $\epsilon_t = \rho \epsilon_{t-1} + e_t$, $|\rho| < 1$ and $e_t \sim N(0, \sigma_e^2)$.

Despite the popularity of Chow and Lin (1971) to compile national accounts across Europe (Eurostat, 2018), the method runs into several shortcomings when operating

in data-rich environments NSIs now find themselves in. Several of these are outlined below, prior to introducing the novel regularised temporal disaggregation approach in Section 3.3.

In moderate and high-dimensions, the behaviour of the Chow-Lin procedure faces several statistical challenges: a) excessive variance in $\hat{\boldsymbol{\beta}}$ impacts interpretation in the weight given to indicator variables; b) unreliable estimation of the AR(1) parameter and variance (ρ, σ_ϵ^2) leads to poor performance in identifying \mathbf{V} , and thus the high frequency series; c) interpretation into which indicator series are most relevant is hampered since all indicators will be included in the model by default. Many of these shortcomings will be identified in the simulation study of Section 3.4. To date, there has been limited research to answer this collection of shortcomings in the temporal disaggregation literature.

Delving into the details of these challenges a little further, we see the GLS estimator of (3.2.2) takes the form:

$$\hat{\boldsymbol{\beta}} = \underbrace{(\bar{\mathbf{X}}^\top \bar{\mathbf{V}}^{-1} \bar{\mathbf{X}})}_{\mathbf{M}}^{-1} \bar{\mathbf{X}}^\top \bar{\mathbf{V}}^{-1} \bar{\mathbf{y}}. \quad (3.2.3)$$

This solution is only uniquely identified if the matrix \mathbf{M} highlighted in (3.2.3) is invertible. Using well-known linear algebra theory (Strang, 1993), this fails to be the case in high-dimensional ($n_l < p$) scenarios as $\text{rank}(\mathbf{M}) = n_l < \text{dim}(\mathbf{M}) = p$. In moderate-dimensional scenarios, \mathbf{M} may have many eigenvalues close to zero, leading to estimates $\hat{\boldsymbol{\beta}}$ with high variance and inevitably results in overfitting. In such settings, Ciammola et al. (2005) noted the poor performance of Chow-Lin in estimating the AR(1) parameter ρ . Further, a reliable estimate of σ_ϵ^2 is important for quantifying the uncertainty associated with the estimate $\hat{\boldsymbol{\beta}}$. Chow and Lin (1971) use the standard estimator: $\hat{\sigma}_\epsilon^2 = \|\bar{\mathbf{V}}^{-1/2}(\bar{\mathbf{y}} - \bar{\mathbf{X}}\hat{\boldsymbol{\beta}})\|_2^2/n_l$, however, it has been shown that even in the classical $n_l > p$ setting such an estimator is biased downwards (Yu and Bien, 2019).

Finally, in addition to the challenges of dimensionality, the Chow and Lin (1971)

approach offers limited model interpretation as no model selection is performed. Since it is sometimes difficult to collect high-frequency indicator series, it can be beneficial to identify which are of most importance when monitoring the economic phenomenon of interest. This way, when future estimates are made, irrelevant indicators can be avoided, reducing model complexity and the cost of creating outputs by a great deal.

3.3 Sparse Temporal Disaggregation

This section introduces a novel sparse temporal disaggregation (spTD) method. This method seeks to provide *robust* and *reliable* solutions, with a view to resolving the aforementioned shortcomings of the classic temporal disaggregation framework. Section 3.3.1 provides a general regularised M-estimation framework that allows us to encompass a variety of penalty functions in the Chow-Lin regression framework to accomplish temporal disaggregation in moderate and high dimensions. Section 3.3.2 draws attention to a specific penalty of the general framework, namely the ℓ_1 penalty and provides a detailed description of the estimation strategy and tuning procedure. Section 3.3.3 concludes with a discussion on how to extend the model when the indicator series are highly correlated. Let us assume throughout that the indicator set \mathbf{X} is fully observed up to the latest low-frequency observation. Note that, it would be possible to extrapolate indicator variables forward if they contain end-of-sample missingness by assuming the indicators follow a certain time series model.

3.3.1 A General Regularised M-estimation Framework for Temporal Disaggregation

With large scale datasets now becoming popular, developing parsimonious models contains numerous advantages. By imposing the assumption that only a relatively small subset $K = |\{\beta_j \neq 0\}|$ of the p possible indicators may actually be active in the model,

this gives us scope to achieve good performance in increasing the accuracy of estimators by discarding noisy information and helps towards revealing underlying characteristics in the data. The empirical performance of sparse modelling has been examined in various settings by Meinshausen and Bühlmann (2010) and Bühlmann and Mandozzi (2014).

Mirroring the established Chow and Lin (1971) temporal disaggregation approach to construct the high-frequency estimator $\hat{\mathbf{y}}$, we propose to study estimators of the form:

$$\hat{\boldsymbol{\beta}}_\rho = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ \underbrace{\left\| \bar{\mathbf{V}}^{-1/2} (\bar{\mathbf{y}} - \bar{\mathbf{X}} \boldsymbol{\beta}) \right\|_2^2}_{\text{Chow-Lin Cost Function}} + \underbrace{P_\lambda(\boldsymbol{\beta})}_{\text{Regulariser}} \right\}, \quad (3.3.1)$$

where we explicitly index the solution as a function of ρ to highlight dependence on this parameter. This estimator incorporates a regularising penalty function in conjunction with the classic Chow-Lin cost function to encode the assumption of sparsity. It does this by shrinking coefficients of indicator series, $\boldsymbol{\beta}$, towards zero that cause a large least-squares score in the Chow-Lin cost function. By doing so, we simultaneously select important indicator series and estimate their regression coefficients with sparse estimates. This will significantly reduce the variance in moderate dimensions and enable accurate estimators in high-dimensions. Not only is this novel in the area of temporal disaggregation, working with a GLS cost function in the high dimensional setting has to-date received little attention in the statistics literature².

The regulariser function, $P_\lambda(\boldsymbol{\beta})$, is indexed by the regularisation parameter $\lambda \geq 0$ that controls the degree of shrinkage. This function may take various forms depending on the assumptions required by the user. As indicated by Hastie et al. (2015) and Bühlmann and Van De Geer (2011), common choices include ℓ_q norm penalties $P_\lambda(\boldsymbol{\beta}) = \lambda \|\boldsymbol{\beta}\|_q^q = \lambda \sum_{j=1}^p |\boldsymbol{\beta}_j|^q$ for $q > 0$, and the ℓ_0 pseudo-norm penalty $\lambda I(\boldsymbol{\beta} \neq 0)$ for $q = 0$, where $I(\boldsymbol{\beta} \neq 0)$ is the number of non-zero components of $\boldsymbol{\beta}$. Estimates have the parsimonious property with some components being shrunk exactly to zero when $q \leq 1$, and the

²More on this is discussed in Chapter 7.

minimisation problem is convex when $q \geq 1$.

3.3.2 The ℓ_1 -spTD method

Given our interest in performing accurate model selection for temporal disaggregation, we here focus on the LASSO (ℓ_1) penalty (Tibshirani, 1996; Hastie et al., 2015). This specification provides an example of our general M-estimation framework whilst enabling us to examine the benefits afforded by ℓ_1 shrinkage. Notably, out of the class of ℓ_q norms, only the ℓ_1 penalty has the properties of yielding sparse solutions while maintaining convexity. We refer to this method as ℓ_1 -spTD and utilise the regulariser function

$$P_\lambda(\boldsymbol{\beta}) = \lambda \|\boldsymbol{\beta}\|_1 := \lambda_\rho \sum_{j=1}^p |\beta_j| .$$

A further useful property of LASSO penalisation is that fast algorithms exist to find estimators for a range of λ values. These methods are known as *path algorithms*, with least angle regression (LARS) (Efron et al., 2004) perhaps being one of the more widely used approaches. In this work we employ LARS to provide an initial screening of indicator series by constructing unique solution paths $\hat{\boldsymbol{\beta}}_\rho(\lambda)$, with varying sparsity levels controlled by λ_ρ . We suggest an additional re-fitting procedure here to reduce the bias in the LARS solution paths.

Unlike GLS which generates only one set of coefficient estimates, the loss function in (3.3.1) will produce a different set of estimates for each value of the regulariser parameter. Correctly selecting the best λ is vital. If too small, then not enough shrinkage occurs and we overfit. Conversely, if it is too large then we shrink too much and underfit. Since the parameter λ , and consequently the coefficient estimate $\hat{\boldsymbol{\beta}}_\rho$, depend on the specification of the AR parameter ρ , the estimator in (3.3.1) can be seen as maximising a penalised profile likelihood as a function of ρ . A final optimisation step searching over the stationary domain of ρ is then used to obtain a final set of parameter estimates.

Our approach proceeds according to the following steps, which we now elaborate on:

1. Generating solution paths $\hat{\beta}_\rho(\lambda)$ for a range of λ using the LARS algorithm;
2. Reducing the bias of these solution paths using a re-fitting procedure;
3. Tuning the model for the best λ and optimising over ρ .

To simplify notation we use $\tilde{\mathbf{y}} = \bar{\mathbf{V}}^{-1/2}\bar{\mathbf{y}}$, $\tilde{\mathbf{X}} = \bar{\mathbf{V}}^{-1/2}\bar{\mathbf{X}}$ and $\tilde{\mathbf{u}} = \bar{\mathbf{V}}^{-1/2}\bar{\mathbf{u}}$. This tilde notation represents the GLS rotated aggregate data we find LASSO solutions for.

Generating Solution Paths

To compute solution paths $\hat{\beta}_\rho(\lambda)$ of the convex problem (3.3.1), we use the LARS algorithm (Efron et al., 2004). This combination of forward stagewise regression (Hastie et al., 2007) and LASSO (Tibshirani, 1996) is able to provide full piecewise linear solution paths for many values of λ . We use LARS as it is a computationally fast algorithm that offers interpretable models, with stable and accurate predictions. Furthermore, the solution path forms a useful graphical output that shows key trade-offs in model complexity (Hesterberg et al., 2008).

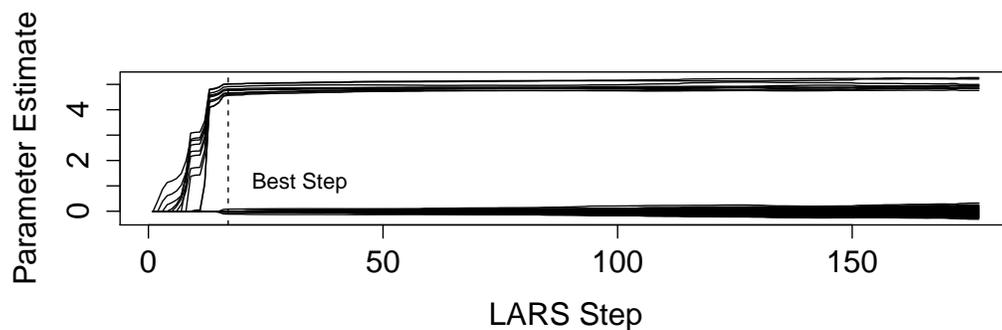
In our approach, we will need to solve Equation (3.3.1) multiple times, over a range of GLS rotations and regulariser parameters. For any fixed ρ , the LARS procedure begins with a large value of λ , and correspondingly sets all regression parameters $\beta = 0$. Let the i th indicator series be the one that is most correlated with the response $\tilde{\mathbf{y}}$, this indicator then enters the active set of relevant predictors. Rather than fit this predictor completely (i.e. set to OLS solution), the algorithm moves the coefficient β_i of this predictor continuously towards its least squares value, causing its correlation with the evolving residual to decrease in absolute value. The value of λ is then decreased until another predictor j has as much correlation with the current residual. At this point, the process is then paused and j enters the active set. The two corresponding predictor coefficients (β_i, β_j) now move in the direction of their joint least squares fit of

the current residual on the two predictors. The process continues adding predictors in this fashion by gradually decreasing λ , and is able to remove predictors from the active set (the LASSO step) if their coefficient hits zero. For this reason, the algorithm usually has more iterations than the number of predictors, p . Each iteration step $s = 1, \dots, S$ of the algorithm has a decreasing $\lambda^{(s)}$ value causing the solution paths to become less sparse as the algorithm moves forward. The final step occurs when the model becomes saturated with n_l non-zero coefficients. It is interesting to note that the sparsity of this final step contains no more than $\delta = \min(n_l, p)$ non-zero values. This means that the number of different LASSO estimated sub-models is typically $\mathcal{O}(\delta)$, which represents a huge reduction compared to all 2^p possible sub-models, especially in the case where $p \gg n_l$.

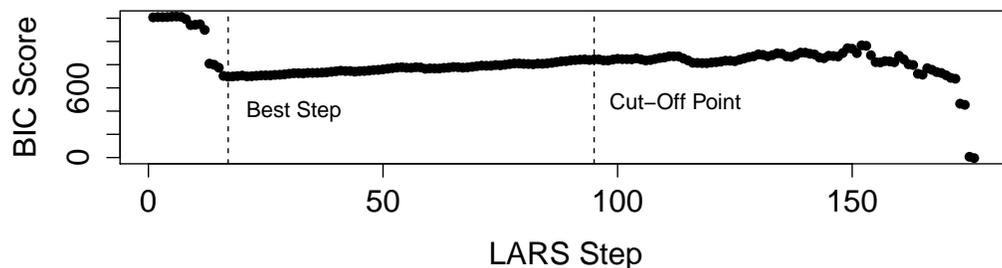
After running the LARS algorithm on the regression we get a set of solution paths $\hat{\beta}_\rho(\lambda^{(s)})$ for steps $s = 1, \dots, S$ that are of the form:

$$\left\{ \begin{array}{l} \text{Step 0:} \quad \hat{\beta}_\rho(\lambda^{(0)} : \lambda^{(0)} > 0) = (0, \dots, 0), \text{ with all values zero;} \\ \text{Step 1:} \quad \hat{\beta}_\rho(\lambda^{(1)} : 0 < \lambda^{(1)} < \lambda^{(0)}) = (--, \beta_\rho, --), \text{ one non-zero;} \\ \text{Step 2:} \quad \hat{\beta}_\rho(\lambda^{(2)} : 0 < \lambda^{(2)} < \lambda^{(1)}) = (--, \beta_{\rho;i}, \beta_{\rho;j}, --), \text{ two non-zero;} \\ \vdots \\ \text{Step } S: \quad \hat{\beta}_\rho(\lambda^{(S)}) = (--, \beta_{\rho;1}, \dots, \beta_{\rho;\delta}, --), \text{ with } \delta = \min(n_l, p) \text{ non-zero.} \end{array} \right.$$

Figure 3.3.1(a) shows the LARS algorithm applied to a synthetic data-set with non-stationary indicator series, $n_l = 100$, $p = 150$, $\rho = 0.5$ and the true β parameter having the first 10 values equal to 5 and the other 140 equal to 0. This demonstrates the algorithm beginning with the null model, then correctly activating the first 10 predictors towards 5. Then, as the algorithm moves forward, and λ decreases, the model becomes less sparse until the model is saturated.



(a)



(b)

Figure 3.3.1: Random walk synthetic data example using $n_l = 100$, $p = 150$, $\rho = 0.5$ and true predictor coefficient β with the first 10 values equal to 5 and the remaining 140 equal to 0. (a) Solution paths generated via LARS, plot shows $\hat{\beta}(\lambda^{(l)})$ as a function of step l ; (b) BIC plot for each step of the LARS algorithm indicating the best step and cut-off point.

Reducing the Bias in Solution Paths

Before tuning the above algorithm to select the optimal $\lambda^{(s)}$ for $s = 1, \dots, S$, we propose a preliminary step to reduce the bias in the solution paths. The significance of this preliminary step is assessed in the simulation study in Section 3.4. It is a recognised drawback of LASSO estimators that they include a small amount of bias, as their absolute value is typically smaller than that of the true parameter; $|\mathbb{E}(\hat{\beta}_\rho(\lambda)) - \beta_\rho| > 0$ for $\lambda > 0$. This behaviour can be seen in Figure 3.3.1(a) with the parameter estimates for the first 10 predictors being slightly below 5, whereas, the unbiased estimator in this

case would have expectation of exactly 5. Many authors (Bühlmann and Van De Geer, 2011; Zheng et al., 2014) have suggested a simple remedy to this issue is to treat LASSO as a variable screening procedure and to perform a second refitting step on the selected support. Belloni et al. (2013) provide a theoretically justified refitting technique which involves performing a least-squares re-estimation of the non-zero coefficients of the LASSO solution. We suggest adopting this least-squares re-estimation approach in our algorithm. With each solution path $\hat{\beta}_\rho(\lambda^{(s)})$ for $s = 1, \dots, S$ obtained from the LARS algorithm, a new design matrix $\tilde{\mathbf{X}}'$ is constructed containing only the predictors that have a non-zero coefficient in the corresponding solution path. We then perform usual least squares estimation on $(\tilde{\mathbf{X}}', \tilde{\mathbf{y}})$ and obtain estimates $\hat{\beta}_\rho$ without bias.

Tuning Model Parameters

The performance of our estimator relies heavily on the choice of tuning parameter λ to select the optimal estimate from $\hat{\beta}_\rho(\lambda^{(1)}), \dots, \hat{\beta}_\rho(\lambda^{(S)})$. We propose to use the BIC statistic (Schwarz, 1978) to achieve this and choose λ such that

$$\hat{\lambda}_\rho = \arg \min_{\lambda \in \{\lambda^{(1)}, \dots, \lambda^{(S)}\}} \left\{ -2\mathcal{L}(\hat{\beta}_\rho(\lambda), \hat{\sigma}_\epsilon^2) + \log(n_l)K_\lambda \right\}, \quad (3.3.2)$$

where $K_\lambda = |\{j : (\hat{\beta}_\rho(\lambda))_j \neq 0\}|$ is the degrees of freedom and $\mathcal{L}(\hat{\beta}_\rho(\lambda), \hat{\sigma}_\epsilon^2)$ is the log-likelihood of the GLS regression.

Since we are working with relatively small sample size of low frequency observations, BIC seems more appropriate than resampling procedures such as cross-validation or bootstrap and comes at a much lower computational expense (Friel et al., 2017). Furthermore, Efron et al. (2004) show how BIC offers substantially better accuracy than cross-validation and related non-parametric methods in selecting optimal λ in LARS. BIC is preferred to other information criteria as parsimony is a primary concern, and BIC generally places a heavier penalty on models with many variables due to the

$\log(n_l)K_\lambda$ term.

Assuming Gaussian errors, the log-likelihood of the GLS regression (3.3.2) is given by

$$\mathcal{L}(\boldsymbol{\beta}, \sigma_\epsilon^2) = -\frac{n_l}{2} \log(2\pi) - \frac{n_l}{2} \log(\sigma_\epsilon^2) - \frac{1}{2} \log(|\boldsymbol{\Sigma}|) - \frac{1}{2\sigma_\epsilon^2} (\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta})^T (\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta}),$$

where $|\boldsymbol{\Sigma}|$ is the determinant of the aggregated covariance matrix $\boldsymbol{\Sigma}$ such that $\bar{\mathbf{V}} = \sigma_\epsilon^2 \boldsymbol{\Sigma}$. We know $\boldsymbol{\Sigma}$ is of Toeplitz form (2.1.5) and depends solely on ρ . We can then maximise this log-likelihood at the reduced-bias estimator $\hat{\boldsymbol{\beta}}_\rho(\lambda)$ from the re-fitted LARS algorithm and using an estimator of the error variance $\hat{\sigma}_\epsilon^2$. Finding a good estimator of $\boldsymbol{\beta}$ has received considerably more attention in the literature than finding a good estimator of σ_ϵ^2 (Reid et al., 2016). However, constructing a reliable estimator of σ_ϵ^2 in finite samples is crucial as it enables one to understand the uncertainty in estimating $\boldsymbol{\beta}$ and be able to construct p -values and confidence intervals (Yu and Bien, 2019). We utilise the estimator

$$\hat{\sigma}_\epsilon^2 = \frac{1}{n_l - K_\lambda} \{\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}_\rho(\lambda)\}^\top \{\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}_\rho(\lambda)\},$$

which de-biases the residual sum of squares by the degrees of freedom. We use this form of estimator as it has been shown to have promising performance in an extensive simulation study done by Reid et al. (2016). After substituting the estimators into the log-likelihood to obtain $\mathcal{L}\{\hat{\boldsymbol{\beta}}_\rho(\lambda), \hat{\sigma}_\epsilon^2\}$, we simply search along the solution path to find the λ that minimises the BIC objective (3.3.2).

In high dimensional settings this BIC heuristic may not be optimal. Indeed, we find that after a certain number of steps of the LARS algorithm, when K_λ becomes close to n_l , the BIC starts to behave erratic and drop in value. Figure 3.3.1(b) plots the BIC score for each step of the LARS algorithm applied to the high dimensional synthetic non-stationary data-set. After around iteration 100, corresponding to K_λ being around 70, the BIC starts to exhibit this erratic behaviour. Such behaviour may

be due to the expected increase in variance in estimating $\hat{\sigma}_\epsilon^2$ with a small degrees of freedom. To accommodate this shortcoming, we adopt a heuristic approach and only considered solution paths $\hat{\beta}(\lambda)$ where K_λ is less than $n_t/2$; identified by the *cut-off* line shown. Constructing a well-behaved information criteria in high-dimensions is an active research area in the literature. For instance, the extended BIC formulas of [Chen and Chen \(2008\)](#), or alternative tuning strategies such as [Belloni et al. \(2011\)](#) form directions for future research.

Once we have found the LARS step minimising BIC, and thus $\hat{\lambda}$, we store the corresponding BIC score and repeat the process with a new initialisation of $\rho \in (-1, 1)$. The estimate $\hat{\rho}$ is then given by the ρ that has the lowest overall BIC score in our search. Using $\hat{\rho}$, we then have our final parameter estimate $\hat{\beta}_{\hat{\rho}}$. The pseudo code for this procedure can be found in Algorithm 1. Code for implementing the ℓ_1 -spTD algorithm, as well as the classic temporal disaggregation methods, can be found in the R package `DisaggregateTS` on CRAN. Chapter 6 provides details on the structure and functionality of this package.

3.3.3 Dealing with Correlated Indicator Series

It is likely that high-dimensional datasets of economic indicator series will exhibit correlation between series and multicollinearity. It has been shown ([Fan and Li, 2001](#); [Zou, 2006](#)) that in certain correlated design settings the choice of regularisation parameter λ is not guaranteed to satisfy the oracle properties and this can lead to inconsistent selection results in high-dimensions. Variable selection consistency is only guaranteed if the design matrix \mathbf{X} of indicators satisfies the so-called *irrepresentability condition* (see [Zou, 2006](#), for details). Intuitively, this condition says that relevant indicator series in the model are not allowed to be too correlated with irrelevant indicator series. If \mathbf{X} violates the irrepresentability condition then an irrelevant variable may enter the active set in the LARS algorithm before all relevant variables have entered, leading to

Algorithm 1 ℓ_1 -spTD

Input: $\tilde{\mathbf{y}} \in \mathbb{R}^{n_l}$ and $\mathbf{X} \in \mathbb{R}^{n \times p}$ **for** $\rho \in (-1, 1)$ **do****Initialise:** $\sigma_\epsilon^2 = 1$, $\mathbf{C} = \mathbf{I}_{n_l} \otimes \mathbf{1}_k$ and $\mathbf{V} := \text{Toeplitz}(\rho, \sigma_\epsilon^2) \in \mathbb{R}^{n \times n}$ **Aggregate:** $\bar{\mathbf{V}} = \sigma_\epsilon^2 \boldsymbol{\Sigma} = \mathbf{CVC}^\top \in \mathbb{R}^{n_l \times n_l}$ and $\bar{\mathbf{X}} = \mathbf{CX}$ **GLS Rotate:** $\tilde{\mathbf{y}} = \bar{\mathbf{V}}^{-1/2} \bar{\mathbf{y}}$ and $\tilde{\mathbf{X}} = \bar{\mathbf{V}}^{-1/2} \bar{\mathbf{X}}$ **LARS**($\tilde{\mathbf{y}} \sim \tilde{\mathbf{X}}$) $\implies \hat{\boldsymbol{\beta}}_\rho(\lambda^{(s)})$ **for** $s = 1$ **to** S **do**Find $K_\lambda = \{j : \hat{\boldsymbol{\beta}}_\rho(\lambda^{(i)})_j \neq 0\}$ **if** Perform refitting step to de-bias **then**Define $\tilde{\mathbf{X}}' = \tilde{\mathbf{X}}[K_\lambda]$ Derive $\hat{\boldsymbol{\beta}}_{GLS}(\lambda^{(s)})$ using $\tilde{\mathbf{y}} \sim \tilde{\mathbf{X}}'$ New non-zero values of $\hat{\boldsymbol{\beta}}(\lambda^{(s)})$ are $\hat{\boldsymbol{\beta}}_{GLS}(\lambda^{(s)})$ **end if****end for****for** j **in** $1, \dots, \max\{j : K_{\lambda^{(j)}} < \lfloor n_l/2 \rfloor\}$ **do** $\hat{\sigma}_{\epsilon;j}^2 = (\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}_\rho(\lambda^{(j)}))^\top (\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}_\rho(\lambda^{(j)})) / (n_l - K_\lambda)$ $\mathcal{L}(\hat{\boldsymbol{\beta}}_\rho(\lambda^{(j)}), \hat{\sigma}_\epsilon^2) = -n_l \log(2\pi)/2 - n_l \log(\hat{\sigma}_{\epsilon;j}^2)/2 - \log(\det(\boldsymbol{\Sigma}))/2 - (n_l - K_{\lambda^{(j)}})/2$ $\text{BIC}(\lambda^{(j)}) = -2\mathcal{L}\{\hat{\boldsymbol{\beta}}_\rho(\lambda^{(j)}), \hat{\sigma}_{\epsilon;j}^2\} + \log(n_l)K_{\lambda^{(j)}}$ **end for** $\hat{\lambda} = \arg \min_\lambda (\text{BIC})$ $\text{BIC}_\rho = \text{BIC}(\hat{\lambda})$ $\hat{\boldsymbol{\beta}}_\rho = \hat{\boldsymbol{\beta}}_\rho(\hat{\lambda})$ **end for** $\hat{\rho} = \arg \min_\rho (\text{BIC}_\rho)$ $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}_{\hat{\rho}}$ $\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{VC}^\top \bar{\mathbf{V}}^{-1}(\bar{\mathbf{y}} - \bar{\mathbf{X}}\hat{\boldsymbol{\beta}})$ using $\hat{\rho}$ for \mathbf{V}

incorrect support recovery. This will cause problems in our re-fitting strategy as an over-fitted support will be re-fit.

One approach to overcome this difficulty is to apply a different penalty to each predictor variable by re-weighting the usual ℓ_1 penalty. This is the *adaptive lasso* (Zou, 2006). For a linear model, it is defined as a two-stage procedure:

$$\hat{\boldsymbol{\beta}}_{\text{adapt}}(\lambda) = \arg \min_{\boldsymbol{\beta}} \left(\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^p \frac{|\beta_j|}{|\hat{\beta}_{\text{init},j}|} \right), \quad (3.3.3)$$

where $\hat{\boldsymbol{\beta}}_{\text{init}}$ is an initial estimator. Zou (2006) proposes to use the OLS estimate $\hat{\boldsymbol{\beta}}_{OLS}$ for $\hat{\boldsymbol{\beta}}_{\text{init}}$, however, in high-dimensional scenarios Bühlmann and Van De Geer (2011) propose to use the lasso estimate $\hat{\boldsymbol{\beta}}_{lasso}$ for $\hat{\boldsymbol{\beta}}_{\text{init}}$. We propose to use the estimate $\hat{\boldsymbol{\beta}}_{\hat{\rho}}$ from a run of our ℓ_1 -spTD algorithm for $\hat{\boldsymbol{\beta}}_{\text{init}}$. That is, using the aggregated indicator matrix $\tilde{\mathbf{X}}$ that has been GLS rotated by $\hat{\rho}$ obtained from the initial fit, we then re-scale this to get $\mathbf{X}_{\text{new}} = \tilde{\mathbf{X}}|\hat{\boldsymbol{\beta}}_{\hat{\rho}}|$. A second stage regression is then performed using LARS (with BIC tuning) on $(\mathbf{X}_{\text{new}}, \mathbf{y})$ to get an estimate $\hat{\boldsymbol{\beta}}^*$. Finally, we scale back by $\hat{\boldsymbol{\beta}}_{\text{adapt}} = \hat{\boldsymbol{\beta}}^*|\hat{\boldsymbol{\beta}}_{\hat{\rho}}|$ to obtain the adaptive lasso solution. In non-GLS settings, this has been shown (Zou, 2006; van de Geer et al., 2011) to have variable selection oracle properties even when the data violates the irrepresentability condition.

Bühlmann and Van De Geer (2011) outline scenarios where the irrepresentability condition holds despite the predictors in \mathbf{X} being correlated. These include scenarios when the covariance matrix of \mathbf{X} has equal positive correlation, Toeplitz structure or bounded pairwise correlation. In these settings, using LARS without adaptive weighting has been shown to be stable, Bach et al. (2011) have confirmed this good behaviour in simulations using different levels of correlation. Furthermore, they compare LARS against other lasso-solving algorithms such as coordinate-descent, re-weighted- ℓ_2 schemes, simple proximal methods, demonstrating that LARS both outperforms and is faster than the other methods for small and medium scale problems. LARS also performed

the best when predictors were highly correlated. This good behaviour is confirmed in the simulation study of Section 3.4.

3.4 Simulation Study

This section provides an in-depth simulation study to assess the performance of the ℓ_1 -spTD algorithm in performing annual-to-quarterly disaggregation in the high dimensional setting, and compares how the model does against the well-established Chow-Lin (CL) method (Chow and Lin, 1971) in moderate dimensions. The benefits of the *refit* step to reduce the bias of LASSO estimates are also examined. This consists of running Algorithm 1 and comparing the results with and without the refitting procedure, these methods are respectively referred to as spTD_RF and spTD.

To understand the behaviour of the methods under different covariate inputs, we consider two data generating processes for the indicator series in \mathbf{X} . Firstly, we consider an example with stationary series by simulating from the standard Normal distribution and secondly, we consider non-stationary series by simulating from a random walk process. That is, we consider (i) $x_{t,j} \sim N(0, 1)$ and (ii) $x_{t,j} = x_{t-1,j} + e_{t,j}$, $e_{t,j} \sim N(0, 1)$ respectively for $t = 1, \dots, n$ and $j = 1, \dots, p$, the residual series is given by an AR(1) process $u_j = \rho u_{j-1} + \epsilon_j$ with $|\rho| < 1$ and $\epsilon_j \sim \mathcal{N}(0, 1)$. We assume a sparse regression coefficient

$$\boldsymbol{\beta} = (\underbrace{5, \dots, 5}_{10}, \underbrace{0, \dots, 0}_{p-10}).$$

This allows us to generate the true ‘unobservable’ quarterly series using $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$ and we aggregate this using $\mathbf{C} = \mathbf{I}_{n_l} \otimes \mathbf{1}_4$ to get the ‘observable’ annual series $\bar{\mathbf{y}} = \mathbf{C}\mathbf{y}$.

To explore different situations we use the following scenarios of parameters. We fix the low-frequency sample size to be $n_l = 100$ and consider three scenarios for the number of indicator series p . These are $p = 30, 90$ and 150 to represent low, moderate and high dimensional settings respectively. We consider three values of the AR(1)

parameter $\rho = 0.2, 0.5$ and 0.8 to represent low, medium and high auto-correlation in the residuals respectively. Under each scenario we perform 1000 simulations and assess the performance on the following three outcomes:

1. The accuracy of the estimated quarterly disaggregated series $\hat{\mathbf{y}}$;
2. The recovery of the true regression coefficient parameters $\hat{\boldsymbol{\beta}} = (\hat{\boldsymbol{\beta}}_1, \dots, \hat{\boldsymbol{\beta}}_p)$;
3. The distribution of estimates for the true AR(1) parameter $\hat{\rho}$.

The study is similarly extended to include a correlated design setting for the indicators. First, a block structure with equal correlation that satisfies the irrepresentability condition (Bühlmann and Van De Geer, 2011). Second, a random covariance matrix design that does not satisfy this condition. We adopt the adaptive lasso weighting scheme in (3.3.3) to assess whether this improves spTD_RF in this second setting.

3.4.1 Estimating the True Quarterly Series

To assess how well we estimate \mathbf{y} , we compute the root mean squared error (RMSE) between the true series and our estimate. Figures 3.4.1 and 3.4.2 display the distribution of RMSE scores over 1000 simulations for $\hat{\mathbf{y}}$ in the stationary and non-stationary experiments respectively. The first column in each represents the low-dimensional setting with $p = 30$. In this setting both CL and the two versions of our method perform similarly well, with spTD_RF performing slightly better. It is interesting to note that RMSE score is only slightly worse for larger values of the true AR(1) parameter ρ , thus, the amount of auto-correlation present in the residuals does have much impact on temporal disaggregation results.

The middle columns represent the moderate dimensional setting, $p = 90$, and we start to notice the decrease in performance of CL. To make the comparison visible, we have taken a log-scale of RMSE scores (keeping y-axis values fixed) as the difference in performance is so large. Again with spTD_RF performing slightly better than spTD, the

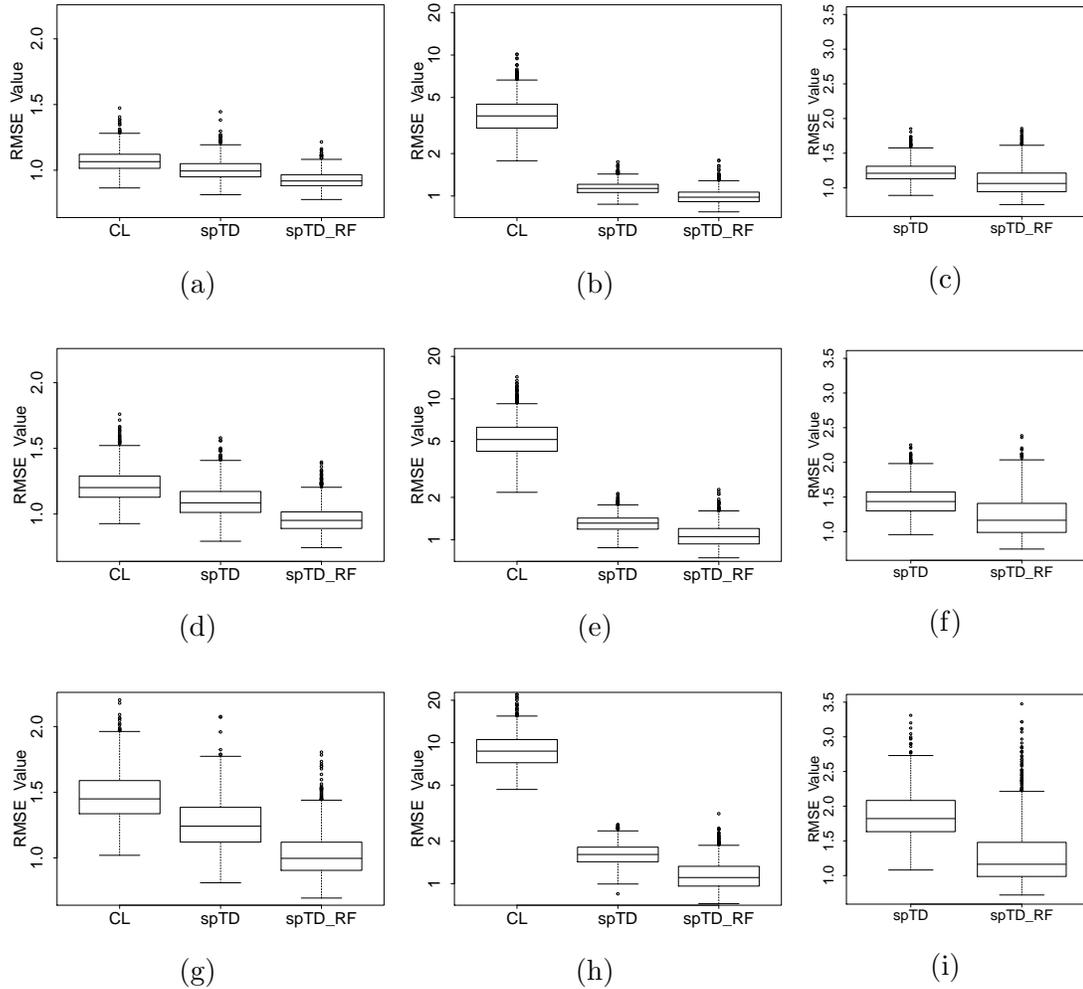


Figure 3.4.1: Boxplots comparing RMSE values of $\hat{\mathbf{y}}$ for CL, spTD and spTD_RF in the stationary i.i.d. standard Normal indicator series experiment. Left, middle and right columns represent $p = 30, 90$ and 150 respectively. Top, middle and bottom rows represent AR(1) parameter $\rho = 0.2, 0.5$ and 0.8 respectively.

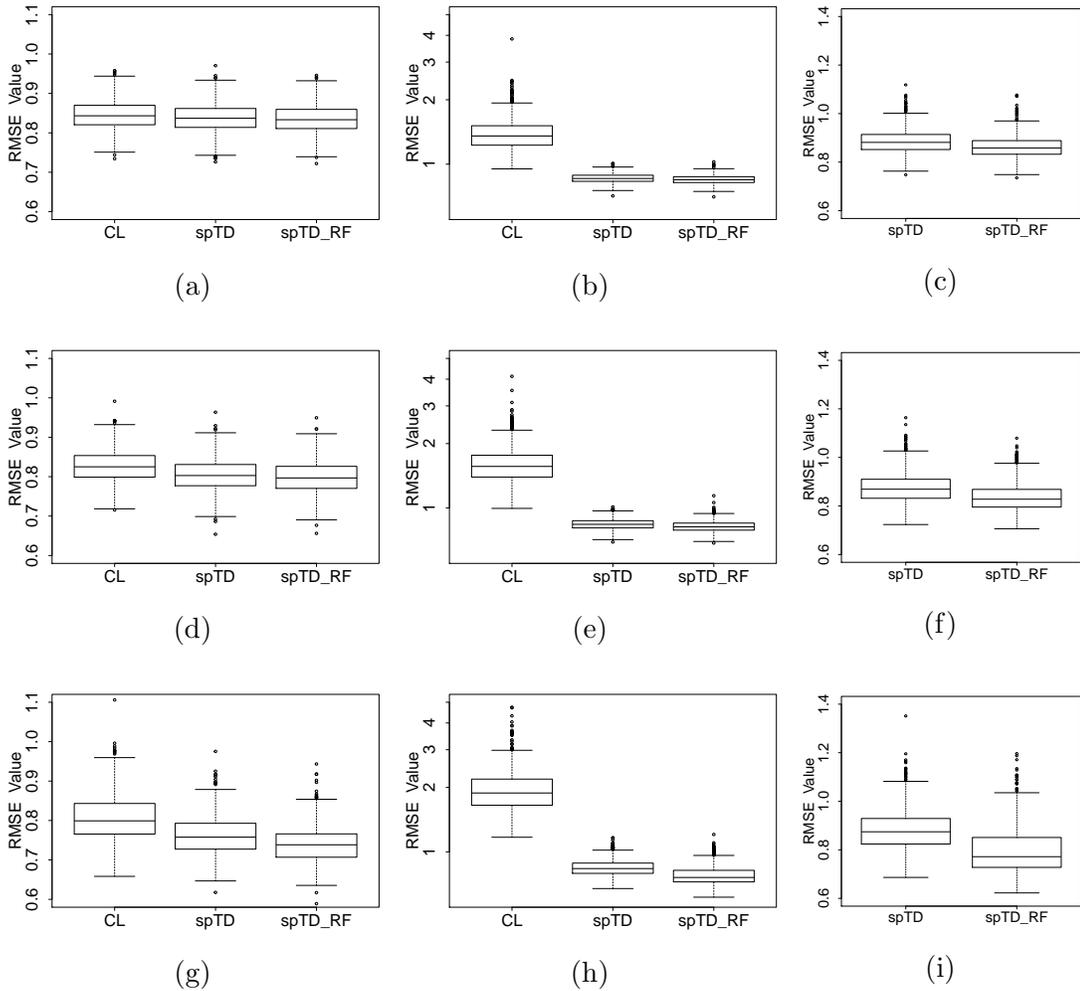


Figure 3.4.2: Boxplots comparing RMSE values of $\hat{\mathbf{y}}$ for CL, spTD and spTD_RF in the non-stationary random walk indicator series experiment. Left, middle and right columns represent $p = 30, 90$ and 150 respectively. Top, middle and bottom rows represent AR(1) parameter $\rho = 0.2, 0.5$ and 0.8 respectively.

proposed methods clearly perform a great deal better in moderate dimension scenarios in accurately disaggregating time series.

The columns on the right represent the high dimensional setting of $p = 150$. In this setting, CL can no longer be used, whereas, both our methods still perform very well with low RMSE scores across the 1000 simulations. Additionally, experiments with even larger dimension of $p = 400$ resulted in similar low RMSE scores for the methods we propose. A further interesting property of these results is that RMSE scores are generally lower in all scenarios for the non-stationary random walk indicator series experiment (Figure 3.4.2). This is a consequence of known results in the statistics literature showing least-squares estimators of the parameters for non-stationary but co-integrated series can converge to their true values faster than stationary series (Johansen, 1988).

3.4.2 Recovery of the True Regression Coefficient

In addition to the recovery of the true high-frequency series, a further motivation for temporal disaggregation is to understand potential drivers of short term change. To this end, we here assess the recovery of the regression coefficients. Table 3.4.1 provides the mean and standard deviation (in brackets) across 1000 iterations of our simulation assessing the recovery of the true β for the three methods CL, spTD and spTD-RF. It is quite clear from this table that spTD-RF provides the lowest score on all three metrics used across all scenarios. The improvement of our method on CL in the moderate dimension scenario can again be seen by the RMSE scores. Interestingly, as p increases, the RMSE for our methods decreases, meaning we are closer to recovering the true values of the support of β . The $\ell_\infty := \max_{i=1, \dots, p} |\hat{\beta}_i - \beta_i|$ metric has been included to measure the worst case error, again showing the improvements we make on CL. RMSE and ℓ_∞ are on average lower with non-stationary indicator series, however, more false positives (FP) are included as seen in the tables. The refitting step certainly reduces the number of FP and it is interesting to note that as the AR(1) parameter ρ increases,

the number of FP on average decreases.

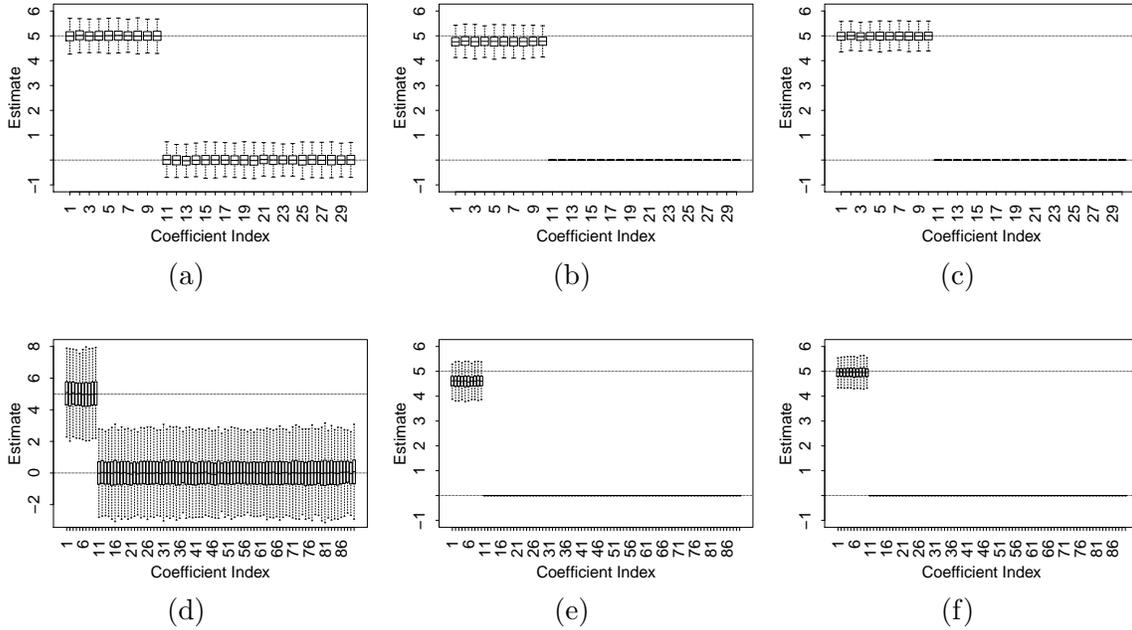


Figure 3.4.3: Boxplots for the estimates $\hat{\beta}_1, \dots, \hat{\beta}_p$ using CL, spTD and spTD_RF (left to right respectively) with $\rho = 0.5$ and $p = 30$ (top row) and $p = 90$ (bottom row) with stationary indicator series.

Figure 3.4.3 shows the distribution for each β parameter over 1000 iterations in the stationary indicator series experiment with CL, spTD and spTD_RF from left to right and $p = 30$ on top and $p = 90$ on the bottom. There is very large variance in the CL boxplots, with tails overlapping in the $p = 90$ setting (plot d). In comparison to our methods, the variance is very narrow around the true β values at 5 and 0. It is evident here to see the refitting strategy in action in plots (c) and (f), as the estimates in (b) and (e) have reduced in bias and shifted upwards towards the true support value of 5. Furthermore, the shrinkage effect for our methods are evident by there not being boxes for estimates $\hat{\beta}_6$ onwards. Except for outliers, which have not been shown on this plot, our method is able to correctly set coefficients to zero, which from a practical point of view means analysts/practitioners are able to identify irrelevant indicator series.

3.4.3 Distribution of AR(1) Parameter Estimate

We restrict our grid search for $\hat{\rho}$ to the positive stationary domain $[0.01, 0.99]$. Performing the search on the positive domain has been motivated previously in Chow-Lin related studies (Sax and Steiner, 2013; Ciammola et al., 2005; Miralles et al., 2003).

Figure 3.4.4 displays the distribution of $\hat{\rho}$ over 1000 iterations for the stationary indicator series experiment. A horizontal line is drawn at the true value of ρ to compare performance. In the first column representing $p = 30$ we observe how all three methods tend to over-estimate the true value of ρ , with CL over-estimating the most. There is also a large variance associated with the estimator when $\rho = 0.2$ and this is again the case when $p = 90$ (middle column). In this moderate dimensional setting CL fails to identify the true ρ completely with it seeming to only select ρ at the boundaries of the search space $(0, 1)$. The proposed methods seem to perform similarly well with the centre of mass surrounding the correct values. In the high dimensional setting we are again centred around the true values with a little more variance. We believe these estimates will be a lot more accurate with a better way of defining BIC in high dimensions.

Figure 3.4.5 shows the distribution of $\hat{\rho}$ with non-stationary indicator series. It is evident from these plots that in this non-stationary setting, correctly identifying the true ρ becomes a lot more challenging. In moderate dimensions ($p = 90$) CL always selected 0 as the estimate, whereas, our methods spread the search space, tending to under-estimate the truth. This large variance in performance may be the result of the algorithm including a lot more false positives in the non-stationary setting. It is worth noting that even though estimating the true AR parameter is a lot less accurate in the non-stationary setting, the recovery of β and \mathbf{y} is more accurate in this setting. Thus, one may conclude that correctly identifying ρ does not play that large of a factor in the quality of regression estimates.

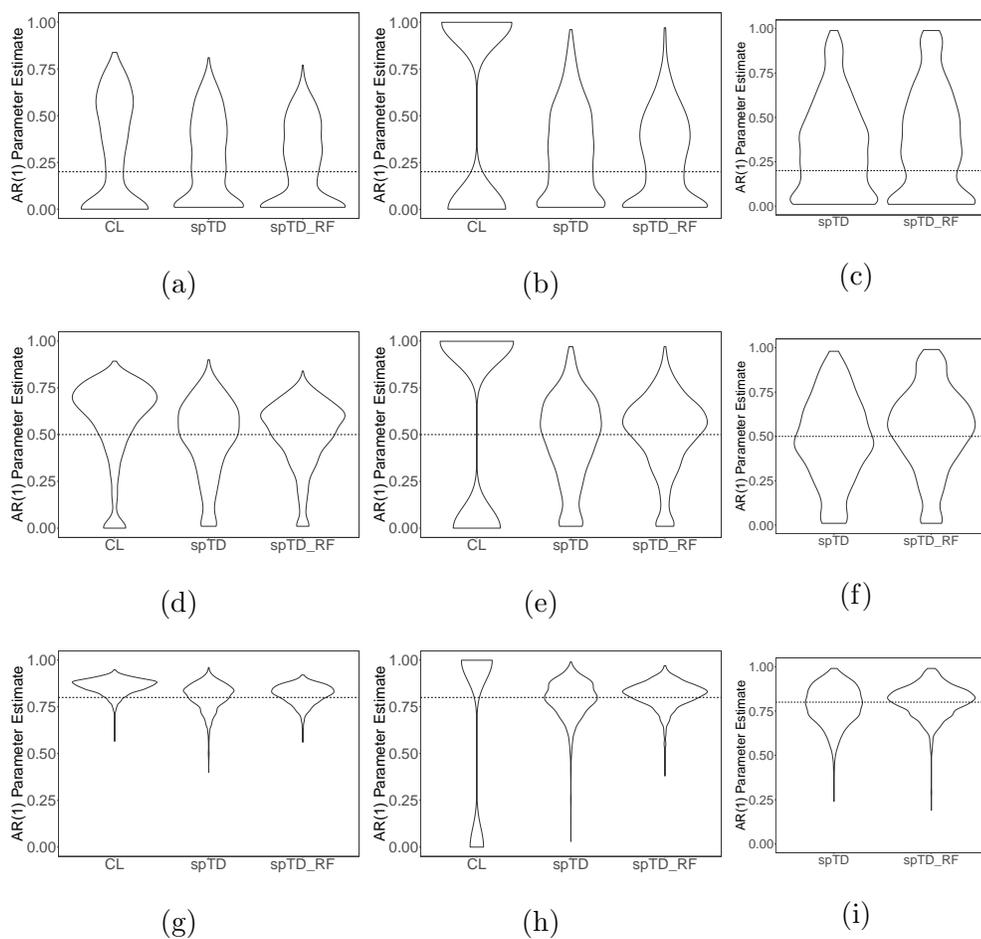


Figure 3.4.4: Violin plots showing the distribution of $\hat{\rho}$ for CL, spTD and spTD_RF using stationary indicator series. Dimension $p = 30, 90$ and 150 from left to right and true AR parameter ρ indicated by horizontal dashed line.

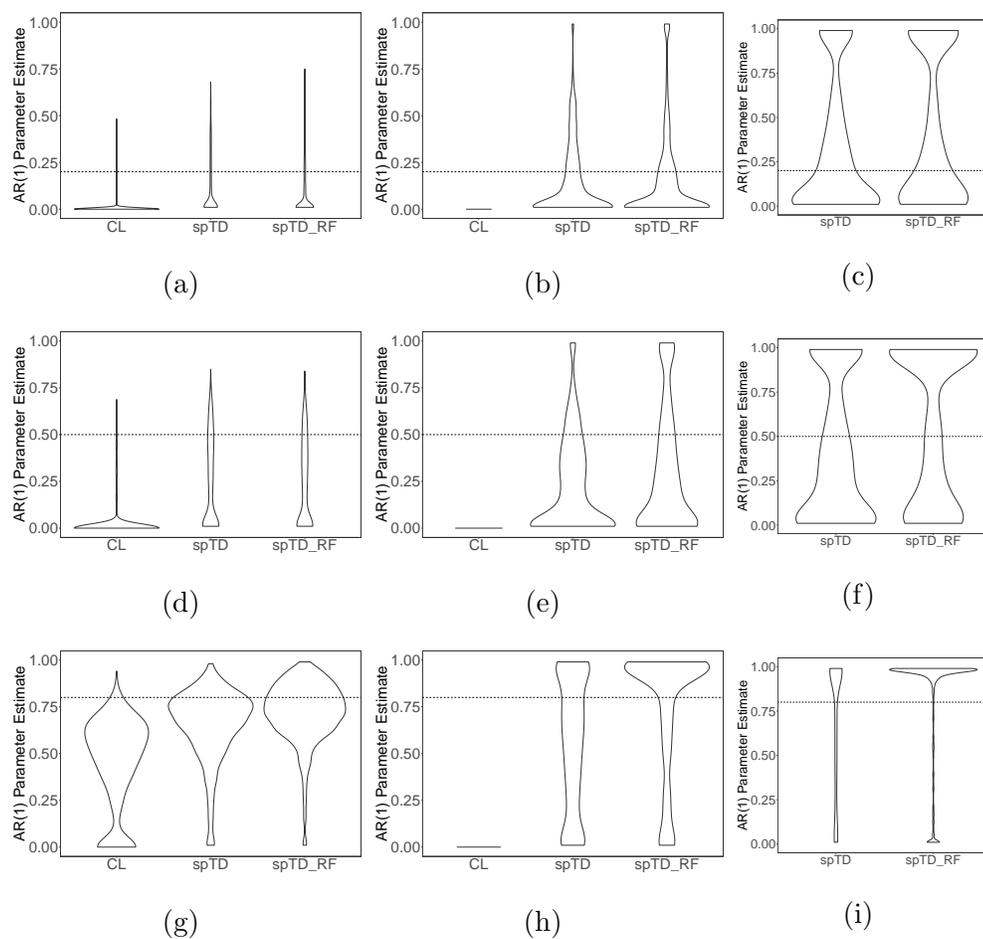


Figure 3.4.5: Violin plots showing the distribution of $\hat{\rho}$ for CL, spTD and spTD_RF using non-stationary indicator series. Dimension $p = 30, 90$ and 150 from left to right and true AR parameter ρ indicated by horizontal dashed line.

3.4.4 Correlated Design Setting

To investigate the effects of correlation present between indicator series, we extend the above i.i.d. setting for \mathbf{X} to now be generated from two correlated designs. The first has a block structure with 10 indicators in each block, where each block has a covariance matrix Σ with $\Sigma_{ii} = 1$ and $\Sigma_{ij} = \theta$ for $i \neq j$. We consider $\theta = \{0.2, 0.6, 0.9\}$ and let

$$\beta = (\underbrace{5, \dots, 5}_5, \underbrace{0, \dots, 0}_5)$$

in the first 3 blocks and 0 everywhere after. This equicorrelated design does not violate the irrepresentability condition. Therefore, in order to assess an even more challenging situation we further consider a random covariance design setting for \mathbf{X} that does break this condition.

We found the spTD_RF method handles the block correlation design setting very well, presenting accurate predictions for \mathbf{y} and recovery of the correct support for β in all scenarios. Figure 3.4.6 shows the distribution of each β_j for the most difficult case of $\theta = 0.9$. The variance using Chow-Lin is very large in this setting as shown in 3.4.6(d). We notice some variance around 0 in spTD without refit, whereas spTD_RF does an excellent job in recovering the true support.

In the second experiment considering the design that breaks the irrepresentability conditions, we use the coefficients

$$\beta = (\underbrace{-2, 2, -2, \dots, 2}_{10}, \underbrace{0, \dots, 0}_{80})$$

to allow a change of sign and a lower signal-to-noise ratio than previously. In this design setting we observe spTD_RF is still able to provide accurate estimates for the high-frequency series $\hat{\mathbf{y}}$ but struggles to obtain the correct support for β . Out of the 1000 simulations, spTD_RF produced on average 4.26 false positives with a large standard

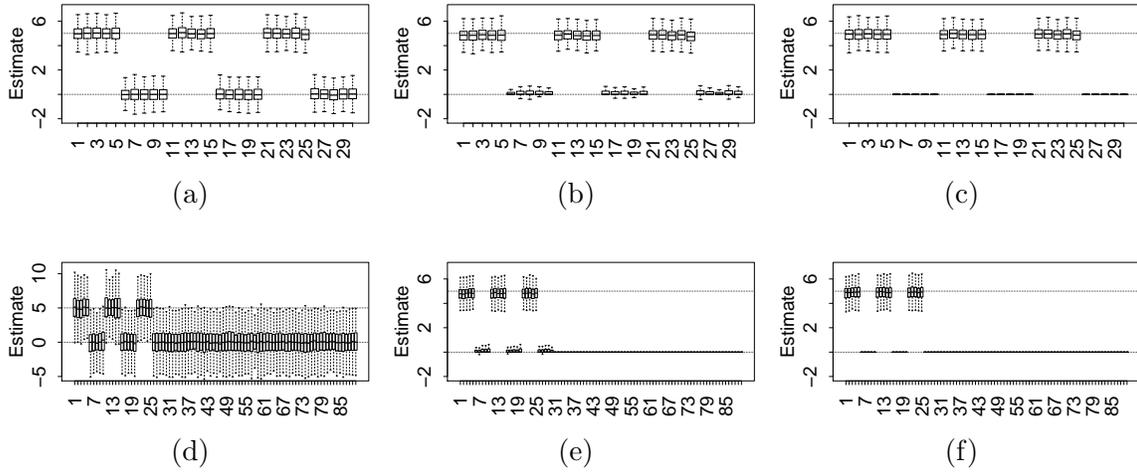


Figure 3.4.6: Boxplots for the estimates $\hat{\beta}_1, \dots, \hat{\beta}_p$ using CL, spTD and spTD_RF (left to right respectively) with $\rho = 0.5$ and $p = 30$ (top row) and $p = 90$ (bottom row) with correlated (0.9) indicator series.

deviation of 4.03. By re-fitting the obtained estimates using the adaptive re-weighting scheme outlined in Section 3.3.3, we see significant improvements in the support recovery performance with 91% of the 1000 iterations achieving exact support recovery and the remainder having at most 3 false positives.

3.5 Case Study 1 - Temporal Disaggregation of UK Gross Domestic Product

There are many measures we can use to assess the state of the economy. These are often disaggregated to different resolution levels, temporally, geographically, and across business sectors. To illustrate the methods developed, we consider the most popular measure of a nation's production, its GDP as the output series of interest, and a range of high frequency indicator series as inputs. In our analysis here we deliberately include indicators which are direct inputs into the deterministic calculation of GDP, for instance, reports from the monthly business survey (MBS). However, to supplement this information, we also include a set of relatively novel *fast indicators* produced by the

Indicator (No. of series)	Description
MBS Turnover in Production (38)	Total turnover of production industries in £million
MBS Turnover in Services (29)	Total turnover of service industries in £million
Retail Sales Index (4)	Value of retail sales by commodity at current prices as an index
VAT Diffusion Index (18)	Diffusion index tracking industry turnover from VAT returns
Traffic Flow on Roads (4)	Mean 15 minute traffic count for vehicles on roads across England
Traffic Flow at Ports (4)	Mean 15 minute traffic count for vehicles within 10km of a port in England

Table 3.5.1: Summary of indicators used in GDP analysis

ONS, based on VAT receipts, and traffic flows. A summary of the indicators we use in our analysis is given in Table 3.5.1. The use of sensor data for traffic flows is an indicator of particular interest here as this aligns with the ONS' endeavour to incorporate and gain insight from alternative data sources into the production of official statistics. This data provides mean 15 minute counts, aggregated to the monthly level, of the number of vehicles on roads across England and within 10km of ports with details on the size of the vehicle observed. This has potential to be a key indicator of supply and demand across England and trade activity at ports.

While it may seem counter-intuitive to try and recover the GDP series from quantities that directly drive this, there is still significant interest in how temporal disaggregated estimates align simply using monthly level data. In general, much more information goes into the estimation of the quarterly GDP results, and these are often considered the gold standard in terms of calibrated measurements of economic activity. Thus, disaggregating to a monthly level whilst maintaining consistency with the calibrated quarterly output is of significant interest. In this example, the output series we aim to disaggregate is the published quarterly (seasonally adjusted) GDP at chained volume measure from 2008 Q1 to 2020 Q2. To illustrate the behaviour of our monthly disaggregated estimates we make comparisons with the published monthly GDP index that represents a percentage growth-

rate based on the average of 2016 being 100%. To make these comparison meaningful we adjust the published monthly index to be a value in £million by benchmarking it onto the quarterly output using the Denton adjustment method discussed in Chapter 2.1.3.

As we are using a total count of 97 monthly indicator series and output data from 2008 Q1 to 2020 Q2 ($n_t = 50$ quarters) we are firmly in the high dimensional setting, and thus hope to capitalise on the benefits of sparse temporal disaggregation to produce a monthly estimate. Additionally, for comparison in the standard dimensional setting we produce an estimate using only 10 indicator series by aggregating our indicator set such that we have a single series each for MBS Turnover in Production , and MBS Turnover in Services, alongside four each for Retail Sales, and the VAT Diffusion Indices. This allows us to make a comparison with the traditional Chow and Lin temporal disaggregation procedure.

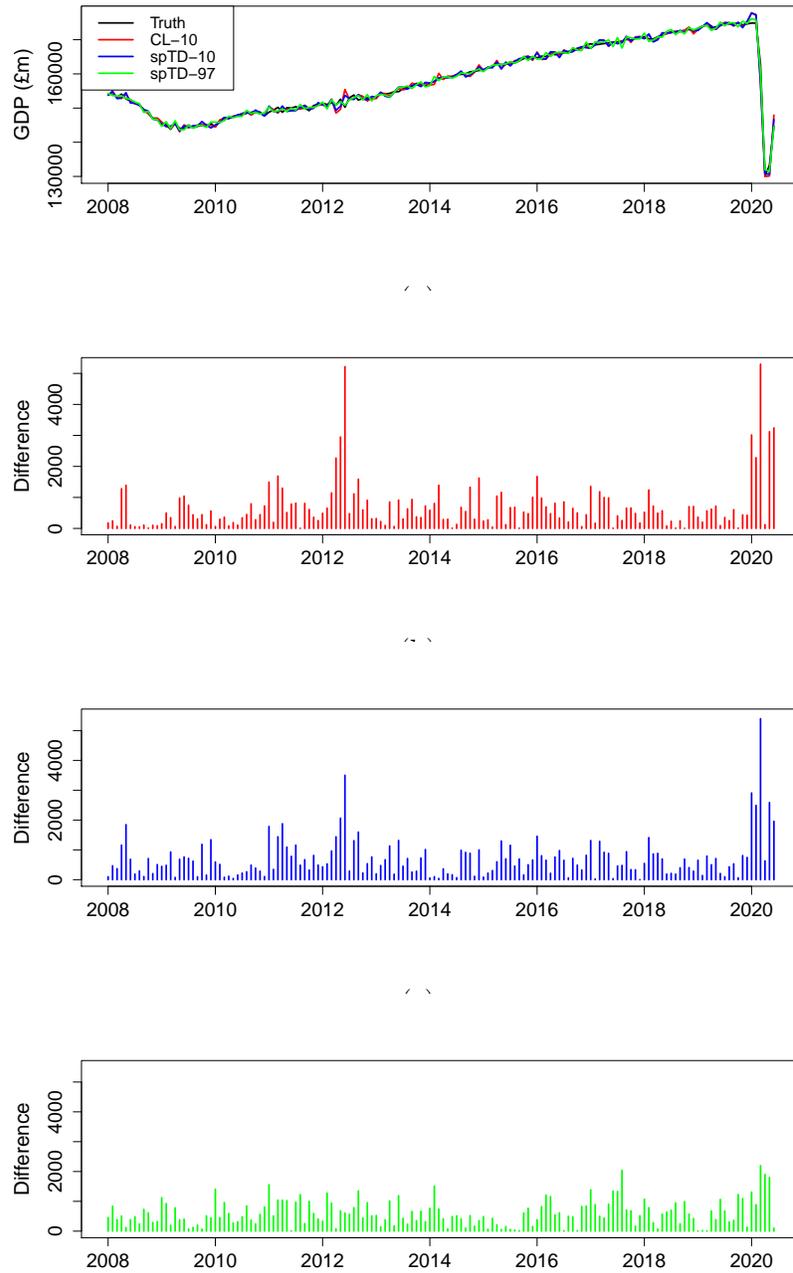
Before discussing our results, it is worth considering our data-quality checks and pre-processing routines. Generally, releases from the ONS are of high-quality and based on robust production pipelines, however, in this case there are a couple of missing observations from January to March 2015 for the road traffic indicator data. This is due to a pause in the data being published throughout this period, as it is only 3 months in total we simply apply imputation based on linearly interpolation for these series. Beyond issues of data-quality, we need to ensure indicator series are of comparable form, for instance, if we want to benchmark with respect to a seasonally adjusted output, it would be prudent to use seasonally adjusted input series. To this end, and to enable better comparison we therefore seasonally adjust each of the indicator series before performing temporal disaggregation by making use of the *seasonal* package in R that calls the automatic procedures of X13-ARIMA-SEATS to perform seasonal adjustment (Sax and Eddelbuettel, 2018). The final pre-processing step we perform is to re-scale our indicator series such that they are comparable in the eyes of the penalised regression problem. For example, the MBS results and road traffic counts are flow data, whereas VAT diffusion and

retail sales are indexes. We therefore scale both the inputs and output to have zero mean and unit variance in order to obtain more interpretable results. To re-scale the monthly estimate back to the original £million scale we use the following: Final Estimate = Model Estimate \times SD(Quarterly Output) + mean(Quarterly Output)/3.

3.5.1 Results

Given the automatic tuning procedure we have developed both ℓ_1 -spTD and the Chow-Lin method can be straightforwardly applied to the data. In the synthetic experiments, the refitting procedure improved performance over the pure lasso approach, as such we generally recommend the refitting strategy, and focus on this method in this application. Figure 3.5.1 shows the monthly estimates of disaggregating quarterly seasonally adjusted GDP at chained volume measures using Chow-Lin and ℓ_1 -spTD with 10 indicators (red and blue respectively) and also using ℓ_1 -spTD with 97 indicators (green line). Further it shows the index based monthly GDP series described above to compare against (black). For the purposes of this discussion, we will refer to this index based series as the true series. In general, all methods seem to be doing a good job of estimating the true trend in Fig. 3.5.1(a) with ℓ_1 -spTD using 97 indicators being the smoothest and most accurate. This advantage is demonstrated in panels (Fig. 3.5.1b-d) where the absolute error is shown to be both lower in magnitude, and more consistent over time. For instance, with the high-dimensional example we do not see large errors in the periods around early 2012, and 2020. This behaviour is further captured by the RMSE errors between the true monthly series and the estimates. For the Chow-Lin and ℓ_1 -spTD methods with 10 indicator series we obtain respectively RMSE scores of 1055.74 and 985.13. Given its access to a wider array of indicator series, the ℓ_1 -spTD method applied to 97 series gives significantly improved performance on average with an RMSE of 749.63.

Even though it may seem the performance of the estimates between Chow-Lin and ℓ_1 -spTD with 10 indicators is close, the real advantage of ℓ_1 -spTD stems from the fact



(d)

Figure 3.5.1: Comparison of estimated sequences and the ONS' monthly GDP index. a) Estimated high-frequency (monthly) series for 2008-2020. b-d) Absolute difference between the index and the estimate for each month (millions of pounds), b) CL with 10 indicators, c) ℓ_1 -spTD with 10 possible indicators, d) ℓ_1 -spTD with 97 possible indicators.

that our method selects a sparse subset of the most relevant indicators. This informs practitioners and users of this data about the main factors driving the economic variable of interest; assisting with understanding data revisions and analysis. Table 3.5.2 displays the coefficient estimates of the 10 indicator series used in the GDP analysis. ℓ_1 -spTD selects 3 out of the 10. These are: the aggregate series for MBS Turnover in Production, the VAT Diffusion Index for Production and the Retail Sales Index for Clothing and Footwear. Unsurprisingly, MBS for Production appears as one of the main components that goes into GDP. Perhaps somewhat more surprisingly is the large weight given to the clothing and footwear sales index. Indeed, if one plots the published monthly GDP index alongside this indicator it is clear they share very similar dynamics. Table 3.5.3 displays the indicator selected out of 97 in the high-dimensional task and their corresponding coefficient estimate. They are, 4 MBS Production industries, 7 MBS Services industries, Vehicles over 11.66m on roads, VAT Diffusion Index for Services, and the Retail sales Index for Clothing and Footwear. The benefits of using this many indicators is that we get information on exactly which industries in production and services are most informative across MBS surveys, alongside the inclusion of faster, alternative indicators. Interestingly, the fact that vehicles over 11.66m on roads is selected gives evidence that these relatively novel traffic indicators can be of use for rapid economic measurement. More large vehicles/lorries on the motorways could be a sign of increased supply and demand of goods, positively impacting the economy. Cars and aircraft drive manufacturing movements so it is no surprise they are selected. Similarly, mining and quarrying are significant in wider production.

Since many of these indicators may be correlated, as a final check we also run the adaptive lasso extension outlined in Section 3.3.3 to assess the impact on the variables selected. Table 3.5.2 and 3.5.3 shows the weightings from this adaptive extension. In the case of 10 indicators the method selects exactly the same variables, whereas in the 97 application, a subset of the indicators is returned as active. We can see in this case, the

Indicator	CL	ℓ_1 -spTD	Adaptive
MBS Turnover in Production Aggregate	0.0728	0.121	0.121
MBS Turnover in Services Aggregate	0.0505	0	0
VAT Diffusion Index - Agriculture	0.00327	0	0
VAT Diffusion Index - Production	-0.0395	-0.0206	-0.0206
VAT Diffusion Index - Services	0.00235	0	0
VAT Diffusion Index - Construction	0.0138	0	0
Retail Sales Index - Clothing	0.201	0.239	0.239
Retail Sales Index - Food	-0.0772	0	0
Retail Sales Index - Household	-0.00325	0	0
Retail Sales Index - Non-foods	0.0587	0	0

Table 3.5.2: Coefficient estimates of GDP analysis with 10 indicator series using Chow-Lin (CL), ℓ_1 -spTD and adaptive extension

Indicator	ℓ_1 -spTD	Adaptive
MBS Production - Mining and Quarrying	0.0232	0.0366
MBS Production - Fabricated Metal Products (excluding weapons)	0.00546	0
MBS Production - Motor Vehicles	0.0305	0.0234
MBS Production - Air and Spacecraft	0.00355	0
MBS Services -Wholesale Retail Trade Repair of Motor Vehicles	0.0599	0.0591
MBS Services - Rail and Land Transport	0.0140	0
MBS Services - Food and Beverage Serving	0.107	0.124
MBS Services - Legal Activities	0.0231	0.0281
MBS Services - Rental and Leasing	0.0344	0.0426
MBS Services - Employment	0.0198	0
MBS Services - Creative Arts and Entertainment	0.0187	0.0168
VAT Diffusion Index - 'S' Other Service Activities	-0.0201	-0.0216
Retail Sales Index - Clothing	0.00273	0
Vehicles Over 11.66m on Roads Across England	0.0263	0.0397

Table 3.5.3: Non-zero coefficient estimates of GDP analysis with 97 indicator series using ℓ_1 -spTD and adaptive lasso extension

adaptive lasso acts to further prune out potential false positives, however, the RMSE score of 799.11 is very similar to that achieved without using adaptive weighting.

3.6 Case Study 2 - Indirect Estimation of a Monthly Service Turnover Indicator in Italy

A second case study is now considered that applies the ℓ_1 -spTD method to perform quarterly-to-monthly disaggregation of the quarterly index of service turnover for transportation in Italy. The purpose of this second case study is to go into further detail on two important questions: (a) How well does the ℓ_1 -spTD model predict months in the current quarter that is not observed yet (i.e. nowcasting) compared to other popular models? (b) How stable are the indicators selected by ℓ_1 -spTD overtime as you make future forecasts? For this second question, an extension to the original ℓ_1 -spTD algorithm is proposed that fixes the number of variables to be selected to ensure more stable results.

The Italian turnover index is released monthly for industry and quarterly for market services by the Italian National Institute of Statistics (ISTAT) and is crucial for the estimation of GDP components from the production side in the Quarterly National Accounts (QNA). The publication delay is around 55 days after the end of the reference quarter and therefore a preliminary estimate based on an ARIMA forecast is used for input into the QNA which is published around 30 days after the reference quarter. For this reason there is great interest in producing an accurate prediction (nowcast) of the service turnover index before its quarterly release and also the desire to estimate it monthly.

The application refers to the turnover service index in transportation, particularly challenging since strongly affected in 2020 by the dramatic movements due to the COVID-19 pandemic and the resurgence of inflation at end of 2021. The series over the

sample 2010 Q1 to 2021 Q4 is shown in Figure 3.6.1. Potential monthly indicators are taken from a total of 143 time series:

- 56 series from Business Surveys (BS) in transports which provide information on manager's current assessments and expectations of the main economic variables in their company;
- 18 series from Assaeroporti (AA) (the association of Italian airports, representing 28 airport management companies for 37 airports at Italian and European institutions) referring to the number of commercial, cargo and passenger movements in both national and international flights;
- 69 series of monthly Turnover in Industry (IT) split by both sector of economic activity and reference market measuring evolution of sales by industrial enterprises at current prices.

Full details of the indicator data is provided in Appendix B.1.

The estimation sample for the quarterly response series representing service turnover index in transportation target spans over the quarters 2010 Q1 to 2021 Q4, while the monthly indicators span the months January 2010 to December 2021. Using this data, a pseudo-real time nowcasting exercise is set up as follows. The first round of the exercise involves using data over the sample 2010 Q1 - 2016 Q4 and indicators over January 2010 - March 2017. This represents a scenario where 2017 Q1 has not been released yet for the response series but we observe indicator information for the months in this quarter. The indicator information in these months is assumed to be fully observed, i.e. no ragged edge is considered. From this first round we are able to obtain a first monthly estimate of the transportation index for the months January 2010 to March 2017. At this point we can assess two results: (i) What indicators have been selected? (ii) What is the nowcast error if we aggregate the three monthly estimates in 2017 Q1 and compare this to the actual observed 2017 Q1 figure? After this, we can continue

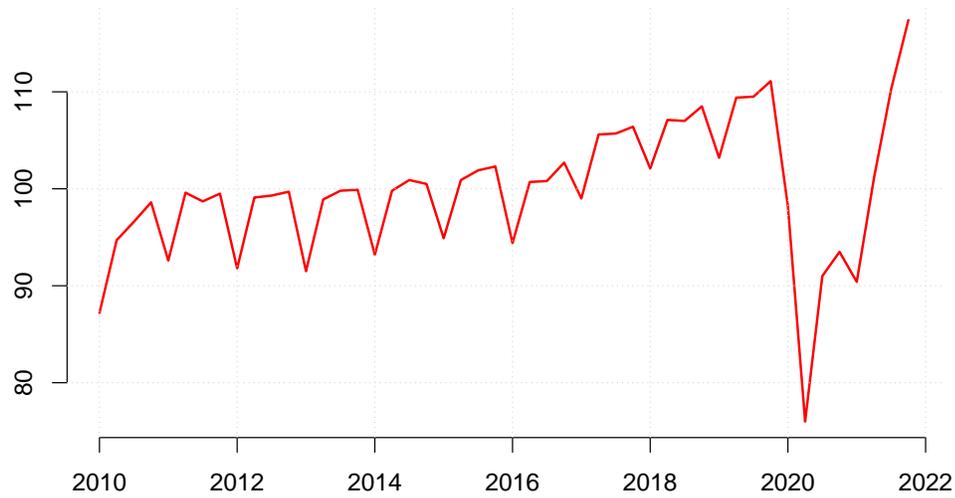


Figure 3.6.1: Italian quarterly turnover index in transports

to the second round where the data set is augmented moving the sample period one quarter ahead and the model is re-estimated to obtain estimates over January 2010 to June 2017. This experiment is iterated until 2021 Q4, a total of 21 steps. Note the final estimate (step 21) uses all information available and is not involved in calculating the over all nowcasting error.

Given the low number of available observations in the full sample, we opt to use an expanding window exercise, letting all available data be part of the in-sample regression whose parameters are used to generate the forecasts. As benchmark comparisons to the ℓ_1 -spTD model, we implement the following. A basic ARIMA model with parameters automatically selected using the R package ‘RJDemetra’. The classic temporal disaggregation methods of Chow-Lin (Chow and Lin, 1971) and Fernandez (Fernandez, 1981) also using ‘RJDemetra’. We apply the Chow-Lin and Fernandez models for each one of the 143 indicator series separately using the full sample of data (January 2010 - December 2021) and use the single indicator providing the best performance as

the indicator of choice in the nowcasting exercise, i.e. the best indicator is used as a consistent indicator throughout. Principal components analysis (PCA) is also used as a final benchmark model where 3 principal components (PCs) are used to summarise the indicator information set which are then used in a Fernandez temporal disaggregation. To measure nowcasting error we use the mean absolute error: $MAE = \sum_{t=1}^n |\hat{\mathbf{y}} - \bar{\mathbf{y}}|/n_t$, between the true quarterly figure, $\bar{\mathbf{y}}$, and the aggregate monthly estimates, $\hat{\mathbf{y}}$ and then average this for all 20 steps in the pseudo-real time exercise.

3.6.1 Results

Let us first identify what indicators are being selected for the nowcasting exercise. When applying the Fernandez method to each of the 143 indicator series in separate temporal disaggregations using the full sample, we find the number one ranked indicator, in terms of MAE, is `fid_ene` (domestic industrial turnover [IT] in energy) with an MAE of 2.90. It turns out the best 4 indicators all represent the energy sector and come from IT. Unsurprising since transport is a sector strongly dependent on energy in terms of costs and efficiency. In the PCA, 3 principle components were used to capture the majority of the variance in the indicator data. Applying PCA to the whole data sample we find that the first PC mainly consisted of IT indicators, while the second PC consisted of AA and the third, BS.

For sparse temporal disaggregation, we first apply ℓ_1 -spTD at each time point in the expanding window letting all the 143 potential indicators be selected. The result has been that just 68 of them appears almost once in the selection set over the 21 waves. A view of this subset is provided in Figure 3.6.2 where the y-axis displays the final quarter observed and the x-axis represents the index of the indicators out of the 68 unique indicators. Table 3.6.1 provides the corresponding selected indicator indexes of indicators appearing at least 5 times. The colours are representative of the membership data set: red = BS, yellow = AA, orange = IT.

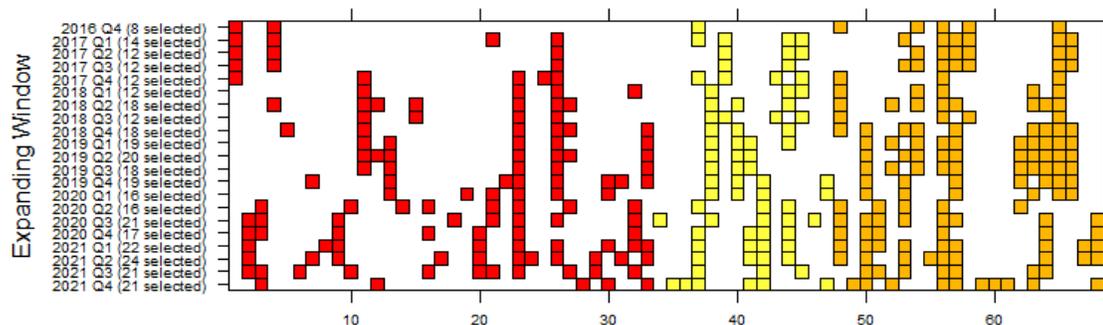


Figure 3.6.2: Chart of the indicators selected at each extension of the sample window. The y-axis displays the date observed from 2016:Q4 up to 2021:Q4. Refer to Table 3.6.1 for the indicator indexed on the x-axis. The colours are representative of the membership data-set: red = BS, yellow = AA, orange = IT.

Indicator	Dataset	N. of times selected	Indicator	Dataset	N. of times selected
fit_CA (56)	IT	18	bh49_i (33)	BS	8
ah49_o (23)	BS	16	air_ptr (40)	AA	7
ah49_z (26)	BS	16	air_pt (41)	AA	7
fie_CA (57)	IT	15	nh49_i (32)	BS	6
fie_CI (65)	IT	14	air_cmt (45)	AA	6
air_cms (44)	AA	12	sh_a (1)	BS	5
fid_ene (50)	IT	12	nh_a (2)	BS	5
fie_str (48)	IT	11	bh_a (3)	BS	5
fid_B (53)	IT	11	nh_d (4)	BS	5
fit_CI (64)	IT	11	ah_i (13)	BS	5
air_pin (38)	AA	10	sh49_p (21)	BS	5
air_pnz (37)	AA	9	nh49_z (27)	BS	5
air_pav (42)	AA	9	air_pue (39)	AA	5
fie_B (54)	IT	9	fie_ind (51)	IT	5
fie_CH (63)	IT	9	fid_CB (58)	IT	5
fie_CK (66)	IT	9	fit_CH (62)	IT	5
sh_g (11)	BS	8	fie_CM (68)	IT	5

Table 3.6.1: Selected indicators across the 21 sparse disaggregations. Number in brackets refer to the x-axis index in Figure 3.6.2.

At first view, Figure 3.6.2 shows the presence in the selected set is stable only for a handful of indicators throughout all 21 steps in the expanding window. While other indicators seem representative of specific events or shocks. In terms of data-sources distribution, we see that 78% of AA are selected, 59% of BS are selected and 30% of IT. It means that with respect to the original distribution, AA and BS are more represented at detriment of IT. The dynamic selection moves in a range that goes from a minimum of 8 indicators chosen in 2016:Q4 to a maximum of 24 indicators in 2021:Q2. Interestingly, the number of indicators increases overtime. In particular, it is 8 at the first run, it jumps to 14 at the second, then stabilises to 12 until the 8th run (18 at the 7th). The window of these runs corresponds to the period 2017-18. Then the number moves in the range 16-18 until the 15th run, period 2019-2020:Q2. Afterwords, it increases again towards the range 21-24 (apart 17 at the 17th run) in the last runs, when the windows looks at the rebounding of the economy after the big fall in 2020:Q3 and the most recent quarters.

Extending ℓ_1 -spTD for a stable number of indicators

To make the number of indicators selected stable in each nowcasting period, we adjust the ℓ_1 -spTD algorithm to produce β estimates of a desired support. We fix the desired support to be 8, the minimum number of indicators selected, and 24, the maximum number of indicators selected in the free selection exercise (Figure 3.6.2).

At each step of the LARS algorithm in ℓ_1 -spTD (Section 3.3.2), for a given tuning parameter value λ , an indicator enters or leaves the selected support set and therefore there may be multiple values of λ that lead to the same support size in $\hat{\beta}$. The best λ , and therefore best support size, is found by minimising BIC. This process can be readily adjusted by collecting all values of λ in the LARS algorithm that generate a solution of the desired support (8 or 24) and minimise the BIC over these λ values. This gives the best coefficient estimate of size 8 or 24 for that particular nowcasting window.

Model	Forecasting accuracy	
	MAE	Relative MAE
ARIMA	4.613	1.000
Chow-Lin	3.390	0.735
Fernandez fid_end (best indicator)	2.896	0.628
PCA	4.240	0.919
Sparse TD: free selection	2.399	0.520
Sparse TD: 8 free	2.223	0.482
Sparse TD: 24 free	2.187	0.474
Sparse TD & Fernandez with 8 fixed	2.098	0.455
Sparse TD & Fernandez with 24 fixed	1.657	0.359

Table 3.6.2: Pseudo-real time nowcasting results

When the support size was fixed to be 8, 29 unique indicators were present in the nowcasting exercise. When the support was fixed to be 24, 83 unique indicators were present. More detailed results are presented in Appendix B.2. As the number of unique indicators used does vary, a final approach was taken that uses the 8 indicators selected in Figure 3.6.2 in 2016:Q4 and the 24 selected in 2021:Q2 as a fixed set of variables for Fernandez temporal disaggregation. This is made with the purpose of avoiding much revisions when new data are released. We call this method: *Sparse TD & Fernandez with 8 or 24 fixed*. The method of using ℓ_1 -spTD with a fixed indicator support set is called *Sparse TD: 8 or 24 free*, while usual ℓ_1 -spTD is called *Sparse TD: free selection*. The results of all models applied to the pseudo-real time nowcasting exercise is displayed in Table 3.6.2.

The results suggest that an accurate selection of the indicators using ℓ_1 -spTD leads to substantial improvements in nowcasting performance. Indeed, the free selection method of Sparse TD provides a reduction in MAE of around 50% with respect to ARIMA and of around 17% with respect to the best Fernandez approach. Further reduction is achieved in this case when performing the selection strategy using the sparse TD approach under a fixed support set of 8 or 24 variables. Interestingly, the best approach for this data was Sparse TD & Fernandez with 24 fixed indicator, with an MAE reduction of around 30% with respect to the usual ℓ_1 -spTD method and of around 64% with respect to ARIMA.

3.7 Conclusion

The process of using high-frequency indicators to construct high-frequency renditions of low frequency information, known as temporal disaggregation, is widely used in official statistics. However, traditional methods (Chow and Lin, 1971; Dagum and Cholette, 2006) have become somewhat outdated and may not be compatible with the large-scale, high-frequency administrative and alternative data sources NSIs increasingly seek to utilise.

This chapter has introduced a novel Sparse Temporal Disaggregation (spTD) framework that can operate in settings where the number of indicators used can exceed the low-frequency sample size. The method builds estimators that incorporate regulariser penalty functions into the well-established Chow-Lin cost function to promote more parsimonious modelling. By focusing on the ℓ_1 -norm (LASSO) penalty (Tibshirani, 1996), methodology produced is able to simultaneously select important indicators and estimate their regression coefficients. Through extensive simulation studies, the ℓ_1 -spTD has been shown to outperform Chow-Lin temporal disaggregation in all moderate dimensional scenarios and provide accurate and interpretable estimates in high-dimensions. While recovery of the true high-frequency series and regression parameters becomes slightly more challenging with a smaller sample size ($n_t < 100$) and smaller regression coefficient value ($\beta_j < 5$) than ones considered, extra simulation studies not presented in this thesis do demonstrate an improved performance over Chow-Lin in these scenarios. As might be expected with a LASSO related method, when indicators are highly correlated and break the irrepresentability condition, recovery of the correct support of β is difficult. A further limitation of the proposed method is in scenarios when the true support of the indicator set considered is actually dense. As the ℓ_1 variant of our method relies on the assumption of sparsity, it may perform poorly in this setting. Factor based approaches may be a better option in this setting, or utilising alternative penalisation strategies, i.e. ridge regression.

Interestingly, the synthetic analysis also verified the difficulty of estimating ρ in the temporal disaggregation setting. This was an area of particular weakness for the classical Chow-Lin method (Ciammola et al., 2005). A similar issue is encountered within Proietti (2006) where the author suggests using a diffuse prior on the regression coefficients may help. Sparse TD appears to significantly outperform Chow-Lin in the estimation of ρ , however, can still struggle in moderate and high-dimensional settings. Interpreting the ℓ_1 penalised solution as a maximum a-posteriori estimator with a Laplacian prior on the regression coefficients, evidence has been provided that weakly informative priors may help in both the recovery of the regression and autocorrelation coefficients. Whilst in the experiments estimation of the autocorrelation structure (via ρ) does not have much impact on the high-frequency estimates themselves, it could significantly impact the confidence intervals associated with inference on the regression parameters and their coverage properties. Technical verification of this claim requires further work, and is complicated by the nature of the ℓ_1 regularisation. There is however a considerable literature on post-selection inference for the lasso (c.f. van de Geer et al., 2014), and it is feasible that this can also be adapted to the temporal disaggregation setting, an exciting direction for future work.

Applying this method to the disaggregation of quarterly GDP data also highlighted ℓ_1 -spTD's ability to achieve an accurate monthly estimate whilst informing us of the most relevant indicators. Unlike previous approaches there are no restrictions here on the number of indicators we can input to the model. This application can be easily extended to produce monthly (or more frequent) estimates of numerous headline variables and accurate disaggregated estimates can act as cost-efficient alternatives to sending out expensive surveys frequently. The second case study displays ℓ_1 -spTD's ability to outperform competing methods in a pseudo-real time nowcasting exercise of quarterly service turnover index of transportation in Italy. This study identified that working with a fixed support of indicators can lead to more stable and accurate

estimates.

Whilst developed in the context of official statistics, the methods proposed in this chapter have application far beyond this domain. For instance, they may be used to help companies produce more granular estimates of consumer behaviour, or to improve the resolution at which researchers can assess complex environmental processes.

Chapter 4

The Sparse Dynamic Factor Model: A Regularised Quasi-Maximum Likelihood Approach

Dynamic factor models (DFMs) have become popular in the macroeconomic nowcasting literature due to their ability to parsimoniously handle high-dimensional time series, deal with arbitrary patterns of missing data and produce predictions in real time. Many of these qualities and estimation approaches to the DFM were introduced in Section 2.2.4 of Chapter 2. Despite their many qualities, one may argue they suffer from a weakness - a lack of interpretability. In the classic DFM, with a dense loadings matrix, it can be difficult to determine which latent factors are driving the behavior of which observed variables. This chapter formally introduces a class of sparse DFMs whereby the loading matrices are constrained to have few non-zero entries, thus increasing interpretability of factors. A regularised M-estimator for the model parameters are presented, along with an efficient expectation maximisation algorithm to enable estimation. Synthetic experiments demonstrate consistency in terms of estimating the loading structure, and superior predictive performance where a low-rank factor structure may be appropriate.

The utility of the method is illustrated with two case studies in distinct fields to show the model’s versatility¹. The first case study consists of an economic nowcasting problem that attempts to nowcast 9 of the main trade commodities the UK exports worldwide using a large collection of related series. The second case study is an energy application forecasting electricity consumption across a large set of smart meters across a University campus. In both case studies the importance of interpreting factor structure is highlighted. Section 4.1 introduces the main contribution of the chapter, highlighting the interpretability shortcomings classic DFMs possess. The section ends with an overview of the chapter’s structure.

4.1 Introduction

Dynamic factor models (DFMs) are one of, if not the most, popular tool for summarising the sources of variation among large collections of time series variables. Originally formalised by Geweke (1977), it was perhaps the application of Sargent et al. (1977) showing how just two dynamic factors were able to explain the majority of variance in headline US macroeconomic variables that has led to DFMs popularity in modern day forecasting and structural analysis with ‘big data’. Their natural state-space representation makes it possible to conduct inference based on Kalman filtering and smoothing techniques, allowing a compact framework to handle mixed-frequency and missing data issues common in economic data sets and the ability to update forecasts in real-time.

The DFM is nowadays ubiquitous within the economic statistics community, with applications in nowcasting/forecasting (Giannone et al., 2008; Bańbura et al., 2011; Forni and Marcellino, 2014a), constructing economic indicators (Mariano and Murasawa, 2010; Grassi et al., 2015), and counterfactual analysis (Harvey, 1996; Luciani, 2015).

¹The case studies presented are based on two distinct projects undertaken both utilising the sparse DFM. While inconsistencies may arise, both case studies are presented in this chapter to demonstrate interesting applications.

Examples in other domains include psychology (Molenaar, 1985; Fisher, 2015), the energy sector (Wu et al., 2013; Lee and Baldick, 2016) and many more, see Stock and Watson (2011) and Poncela et al. (2021) for detailed surveys of the literature.

Formally introduced in Section 2.2.4 of Chapter 2, the state-space framework of a DFM consists of a measurement equation:

$$\mathbf{X}_t = \mathbf{\Lambda}\mathbf{F}_t + \boldsymbol{\epsilon}_t,$$

along with a state equation:

$$\mathbf{F}_t = \mathbf{A}\mathbf{F}_{t-1} + \mathbf{u}_t,$$

for observations $t = 1, \dots, n$, which together assume that the common dynamics of a large number of stationary time series variables $\mathbf{X}_t = (X_{1,t}, \dots, X_{p,t})^\top$ are driven by a small number $r \ll p$ of latent factors $\mathbf{F}_t = (F_{1,t}, \dots, F_{r,t})^\top$ which evolve dynamically as a VAR(1) process over time. The $p \times r$ matrix $\mathbf{\Lambda}$ contains factor loadings representing the relationship between the measurements and the underlying latent factors that drive their behaviour. The factor loadings play a critical role in the analysis of DFMs, providing an insight into the underlying structure of the data and the strength of correlation between factors and variables.

Unfortunately, in the classic DFM, every variable in \mathbf{X}_t is loaded onto the factors \mathbf{F}_t , i.e. $\mathbf{\Lambda}$ is a dense matrix. Often these loading weights are quite large and it is very difficult, if not impossible, to understand how individual series, or groups of series, are influencing the dynamics of the estimated latent factors (Despois and Doz, 2023). If instead, we could obtain factor estimates that are linear combinations of smaller, distinct groups of variables, where we could associate a given factor estimate with an underlying feature of the observed data, this would lead to tremendous improvements in interpretability.

A natural way one could impose this kind of interpretability into the factor structure

would be through the factor loadings $\mathbf{\Lambda}$. A classic approach to accomplish this is based on factor rotations, which has been widely used to minimise the complexity in the factor loadings to make the structure simpler to interpret. See Kaiser (1958) for the well-established varimax rotation and see Carroll (1953) and Jennrich and Sampson (1966) for the well-established quartmin rotation. For a recent paper on varimax rotation see Rohe and Zeng (2020). The well-established nowcasting paper of Bańbura et al. (2011) use prior knowledge of the variables to partition the factors into mutually independent global, real and nominal factors, assuming that the global factor is loaded by all the variables, while real and nominal factors are specific to real and nominal variables. They show this is helpful when calculating a global factor. Andreou et al. (2019) is a similar more recent application for grouped factor structure.

An alternative approach is to utilise modern tools from sparse modelling and regularised estimation to constrain the loading matrix to only have a few non-zero loadings for each factor, i.e. $\mathbf{\Lambda}$ is sparse. A DFM where the loadings matrix is assumed to be sparse can be described as a *sparse DFM*. One way to impose sparse restrictions on $\mathbf{\Lambda}$ is to estimate it using sparse principal components analysis (SPCA) (Zou et al., 2006). SPCA is a variation of traditional PCA that incorporates an elastic net penalty (i.e. a mix of ℓ_1 and ℓ_2 norms) to induce sparsity. The interpretable principal components can be viewed as static factors in a factor model and used alongside the Kalman smoother for prediction purposes. See Croux and Exterkate (2011) for a typical macroeconomic forecasting setting where they consider a robustified version. Kristensen (2017) use SPCA to estimate diffusion indexes with sparse loadings. Despois and Doz (2023) prove that SPCA consistently estimates the factors in an approximate factor model if the ℓ_1 penalty is of $\mathcal{O}(1/\sqrt{p})$. They compare SPCA with factor rotation methods and show an improved performance when the true loadings structure is sparse.

This main contribution of this chapter is to introduce a new class of sparse DFMs. Unlike previous research, the methodology presented implements regularisation within

an EM algorithm framework, allowing us to robustly handle arbitrary patterns of missing data, model temporal dependence in the processes, and impose weakly informative (sparse) prior knowledge on the factor loadings. It is argued that in settings where autocorrelation is moderately persistent, that the feedback provided through our EM procedure is important in aiding recovery of the factor loadings, as well as producing accurate forecasts.

The analysis within this paper is empirical in nature, considering three aspects: i) how our EM algorithm performs in recovering the true sparsity pattern in the factor loadings; ii) how the model contrasts with alternative models in a predictive setting, e.g. where we want to forecast either all the p time series, or just a subset of these; and iii) how the model and estimation routine can be used in practice to extract insights from complex real-world datasets.

The rest of the chapter is structured as follows. Section 4.2 formalises our sparse DFM and the sparsity assumptions placed on the loading matrices. Section 4.3 presents a regularised likelihood estimator for the model parameters, and introduces an EM algorithm to enable finding feasible estimates. Tuning procedures and implementation of the algorithm using the `sparseDFM` package (Chapter 6) is also discussed. Synthetic experiments comparing the EM methodology with competing methods are presented in Section 4.4. Real data examples in economics and the energy sector are presented in Sections 4.5 and 4.6 respectively. The paper concludes with a discussion of the results, and how the models and estimators can be further generalised to provide flexibility to users.

4.2 The Sparse DFM

Consider the p -variate time series $\{\mathbf{X}_t\}$ and r factors $\{\mathbf{F}_t\}$ related according to the model

$$\begin{aligned}\mathbf{X}_t &= \mathbf{\Lambda}\mathbf{F}_t + \boldsymbol{\epsilon}_t \\ \mathbf{F}_t &= \mathbf{A}\mathbf{F}_{t-1} + \mathbf{u}_t ,\end{aligned}\tag{4.2.1}$$

where $\{\boldsymbol{\epsilon}_t\}$ and $\{\mathbf{u}_t\}$ are multivariate white noise processes². For simplicity we assume $\mathbb{E}[\boldsymbol{\epsilon}_t\boldsymbol{\epsilon}_t^\top] = \boldsymbol{\Sigma}_\epsilon = \text{diag}(\boldsymbol{\sigma}_\epsilon^2)$ and $\boldsymbol{\sigma}_\epsilon^2 \in \mathbb{R}_+^p$ is a vector of idiosyncratic variances. Similarly, let $\mathbb{E}[\mathbf{u}_t\mathbf{u}_t^\top] = \boldsymbol{\Sigma}_u$ and assume the eigenvalues of the VAR matrix are bounded $\|\mathbf{A}\| < 1$, thus the latent process is assumed stationary. This model corresponds to an exact DFM, where all the temporal dependence is modelled via the latent factors.

In this context, our notion of sparsity relates to the assumption that many of the entries in the true loadings matrix, $\mathbf{\Lambda}$, will be zero. For instance, let the support of the k th column of the loading matrix be denoted

$$\mathcal{S}_k := \text{supp}(\mathbf{\Lambda}_{\cdot,k}) \subseteq \{1, \dots, p\} ,$$

such that $s_k = |\mathcal{S}_k|$. We refer to a DFM as being sparse if $s_k < p$ for some or all of the $k = 1, \dots, r$ factors. In practice, this is an assumption that many of the observed series are driven by only a few (r) latent factors, and that for many series only a subset of the factors will be relevant.

²Bañbura and Modugno (2014) show how the DFM can be easily adapted for serially correlated idiosyncratic errors, $\boldsymbol{\epsilon}_t$, following an AR(1) process. The `sparseDFM` package presented in Chapter 6 allows for both white noise and AR(1) $\boldsymbol{\epsilon}_t$. Appendix D.1 demonstrates why the adaption for AR(1) errors is computationally slower than white noise errors.

4.2.1 Consistency and Pervasiveness

In the sparse situation, whereby $s_k < p$, we will be able to model only a subset of the observations with each factor. To enable us to model all p variables and gain information relating to the r factors as n, p increase, we assume a couple of conditions on the specification. First, that the support of the observations, and the union of factor supports is equal, i.e. $\cup_{k=1}^r \mathcal{S}_k = \{1, \dots, p\}$, thus all observations are related to at least one of the factors. Second, that the support for each factor grows with the number of observed variables, in that $\{s_k\}$ is a non-decreasing sequence in p . Assumptions of this form would allow us, in principle, to assess the consistency of factor estimation as p grows.

This asymptotic analysis in p (and n) contrasts with the traditional setting with a fixed p – for which the factors cannot be consistently recovered and can only be approximated, with error that depends on the signal-to-noise ratio $\|\mathbf{\Lambda}\mathbf{\Sigma}_F\mathbf{\Lambda}^\top\|/\|\mathbf{\Sigma}_\epsilon\|$, where $\mathbf{\Sigma}_F = \mathbb{E}[\mathbf{F}_t\mathbf{F}_t^\top]$ (Bai and Li, 2016). Intuitively, this is due to the fact that if p is fixed, then we cannot learn anything more about the factor at a specific time t , as we do not get more information on the factors as n increases, instead we just get more samples (at different time points) relating to the series $\{\mathbf{F}_t\}$. When we go to the doubly asymptotic, or just $p \rightarrow \infty$ setting, then if the number of factors r is fixed or restricted to slowly grow in n then we can not only recover structures relating to $\{\mathbf{F}_t\}$, e.g. the specification of \mathbf{A} , but we can also get more information relating to the factor at the specific time t (Bai and Li, 2016; Barigozzi and Luciani, 2022). One way to ensure this growing information about the factors is to assume that they are in some sense pervasive – the more variables p we sample, the more this tells us about the r factors. We note, that for a more formal analysis of the DFM, a usual pervasiveness assumption placed on the loading conditions is given by Doz et al. (2011), whereby $\lim_{p \rightarrow \infty} p^{-1} \lambda_{\min}(\mathbf{\Lambda}^\top \mathbf{\Lambda}) > 0$, i.e. the average loading onto the least-influential factor is bounded away from zero.

In this work, we choose to focus on the empirical performance of our estimator, thus

we do not formalise the sparsity assumptions further. However, it is worth noting our empirical studies meet the pervasiveness assumptions regarding the support of the factor loadings.

4.2.2 Identifiability

In the following section, we will consider a QMLE estimator for the factor model based on assuming Gaussian errors $\boldsymbol{\epsilon}_t$ and \boldsymbol{u}_t . It is thus of interest to consider how the associated likelihood relates to the factors and their loadings. Adopting a Gaussian error structure and taking expectations over the factors, the likelihood for (4.2.1) is given by

$$\mathcal{L}(\boldsymbol{\Lambda}) \propto \log \det(\boldsymbol{\Lambda}^\top \boldsymbol{\Sigma}_F \boldsymbol{\Lambda} + \boldsymbol{\Sigma}_\epsilon) - \frac{1}{2} \text{tr} \left[(\boldsymbol{\Lambda}^\top \boldsymbol{\Sigma}_F \boldsymbol{\Lambda} + \boldsymbol{\Sigma}_\epsilon)^{-1} \frac{1}{n} \sum_{t=1}^n \boldsymbol{X}_t \boldsymbol{X}_t^\top \right].$$

An obvious identifiability issue arises here, such that if $\tilde{\boldsymbol{\Lambda}} = \boldsymbol{\Lambda} \boldsymbol{Q}$, $\tilde{\boldsymbol{F}}_t = \boldsymbol{Q} \boldsymbol{F}_t$, for any unitary matrix \boldsymbol{Q} , i.e. $\boldsymbol{Q}^\top = \boldsymbol{Q}^{-1}$, we have $\mathcal{L}(\tilde{\boldsymbol{\Lambda}}) = \mathcal{L}(\boldsymbol{\Lambda})$. Now consider the case of performing a rotation on the true loadings matrix, denoted by $\boldsymbol{\Lambda}_0$. Denote the set of all possible equivalent loadings as

$$\mathcal{E} := \{ \boldsymbol{\Lambda}_0^* \boldsymbol{Q} \mid \boldsymbol{Q}^\top = \boldsymbol{Q}^{-1}, \boldsymbol{Q} \in \mathbb{R}^{r \times r} \}.$$

The invariance of the likelihood to elements of this set mandates that theoretical analysis of the DFM is typically constructed in a specific frame of reference, c.f. Doz et al. (2011, 2012); Bai and Li (2016). For unique solutions one needs to impose normalisation constraints or a set of restrictions on $\boldsymbol{\Lambda}$, for example, authors (Geweke and Singleton, 1981; Proietti, 2011a) impose r^2 restrictions enforced by setting $\boldsymbol{\Lambda} = [\boldsymbol{I}_r; \boldsymbol{\Lambda}^*]$ where $\boldsymbol{\Lambda}^*$ is the $(p - r) \times r$ unrestricted loadings matrix. Interestingly, our sparsity assumptions restrict the nature of this equivalence class considerably, in that only loading matrices with sparse structure are permitted. In general, there will still be multiple sparse

representations that are allowed, and the issue of the scale invariance remains, however, the latter can be fixed by imposing a further constraint on the norms of the loading matrices. In this work, we demonstrate empirically that it is possible to construct estimators that are consistent up to rotations that maintain an optimal level of sparsity, in the sense that the true loading matrix is given by

$$\mathbf{\Lambda}_0 \in \arg \min_{\mathbf{\Lambda} \in \mathcal{E}} \sum_{k=1}^r \|\mathbf{\Lambda}_{\cdot,k}\|_0, \quad (4.2.2)$$

where $\|\mathbf{\Lambda}_{\cdot,k}\|_0 := |\text{supp}(\mathbf{\Lambda}_{\cdot,k})|$ counts the number of non-zero loadings. More generally (see Remark 1) we could consider selecting on the basis of the ℓ_q norm, $\|\mathbf{\Lambda}\|_q := (\sum_{ik} \Lambda_{i,k}^q)^{1/q}$. As Figure 4.2.1 shows, the ℓ_1 norm may still provide selection, however, the ℓ_2 norm provides no selection as it maintains the rotational invariance of the likelihood. In this paper, we restrict our equivalence set on the basis of the ℓ_0 norm, as above, that is, we specify the true loading matrices as those that maintain the highest number of zero values after consideration for all unitary linear transformations.

In practice, these issues mean we are unable to recover the correct sign of the factor loadings, whilst columns in the loading matrix may also be permuted, e.g. factor k can be swapped (under permutation of the columns in the loading matrix) with factor l , for any $k, l \in \{1, \dots, r\}$. These are the same identifiability issues which we face in PCA, whereby the eigenvectors can be exchanged in terms of order and direction. Note that, if the focus is solely on forecasting ability, then these identifiability issues are not a concern as forecasts depend on the space spanned by the factors, i.e. $\mathbf{\Lambda} \mathbf{F}_t^\top$, and not the factors themselves.

Remark 1 (Sparsity and invariance): To illustrate how the sparsity constraint (4.2.2) breaks the more general invariance that regular DFMs suffer, we can consider the quantity $\|\mathbf{\Lambda}_0^* \mathbf{Q}_{\text{rot}}(\theta)\|_q$, where $\mathbf{Q}_{\text{rot}}(\theta) \in \mathbb{R}^{2 \times 2}$ is a rotation matrix with argument $\theta \in (-\pi, \pi)$, and $\mathbf{\Lambda}_0^* \in \mathbb{R}^{10 \times 2}$ has the first column half filled with ones, and the rest zero, the second

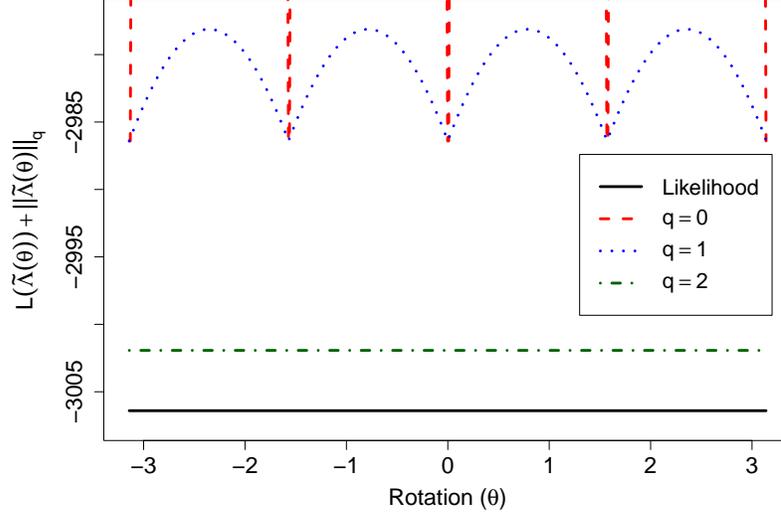


Figure 4.2.1: The impact of rotation on the function $\mathcal{L}(\tilde{\Lambda}(\theta)) + \|\tilde{\Lambda}(\theta)\|_q$ for $q = 0, 1, 2$. In the case of $q = 0, 1$, the set of feasible Λ_0 from 4.2.2 is restricted to the points $\theta \in \{0, \pm\frac{1}{2}\pi, \pm\pi\}$ corresponding to either swapping columns, or flipping signs.

column is set to be one minus the first. As we see from Fig. 4.2.1, without the additional restriction on our specification of Λ_0 , via Eq. (4.2.2), we would not be able to determine a preference for any particular element from the set $\mathcal{E} := \{\Lambda_0^* Q_{\text{rot}}(\theta) : \theta \in \{-\pi, \pi\}\}$. I.e. the likelihood alone $\mathcal{L}(\tilde{\theta})$ gives the same value for every $\theta \in \{-\pi, \pi\}$.

4.3 Estimation

Under the Gaussian error assumption, and collecting all parameters of the DFM (4.2.1) in $\theta = (\Lambda, \mathbf{A}, \Sigma_\epsilon, \Sigma_u)$, we are able to write the joint log-likelihood of the data \mathbf{X}_t and the factors \mathbf{F}_t as:

$$\begin{aligned}
 \log \mathcal{L}(\mathbf{X}, \mathbf{F}; \theta) &= -\frac{1}{2} \log |\mathbf{P}_0| - \frac{1}{2} (\mathbf{F}_0 - \alpha_0)^\top \mathbf{P}_0^{-1} (\mathbf{F}_0 - \alpha_0) \\
 &\quad - \frac{n}{2} \log |\Sigma_u| - \frac{1}{2} \sum_{t=1}^n \mathbf{u}_t^\top \Sigma_u^{-1} \mathbf{u}_t \\
 &\quad - \frac{n}{2} \log |\Sigma_\epsilon| - \frac{1}{2} \sum_{t=1}^n \epsilon_t^\top \Sigma_\epsilon^{-1} \epsilon_t,
 \end{aligned} \tag{4.3.1}$$

where $\boldsymbol{\epsilon}_t = \mathbf{X}_t - \boldsymbol{\Lambda}\mathbf{F}_t$, $\mathbf{u}_t = \mathbf{F}_t - \mathbf{A}\mathbf{F}_{t-1}$, and we have assumed an initial distribution at $t = 0$ of the factors as $\mathbf{F}_0 \sim \mathcal{N}(\boldsymbol{\alpha}_0, \mathbf{P}_0)$.

We propose to induce sparsity in our estimates using the familiar ℓ_1 penalty, with motivation similar to that of the LASSO (Tibshirani, 1996). Alternative penalty functions are available, however, the convexity of the ℓ_1 penalty is appealing. Even though the overall objective for the parameters is non-convex, due to the rotational invariance of the log-likelihood, the convexity of the penalty ensures we can quickly and reliably apply the sparsity constraints. We will make use of this structure in the algorithms we construct to find estimates in practice. It is worth noting that our focus here is on the factor loadings, and thus this is the object we regularise, possible extensions could consider additional/alternative constraints, for instance on the latent VAR matrix.

Our proposed estimator attempts to minimise a penalised negative log-likelihood, as follows

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} -\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) + \alpha R(\boldsymbol{\Lambda}), \quad (4.3.2)$$

where $\alpha \geq 0$. A larger α corresponds to a higher degree of shrinkage on the loadings, e.g. for a larger α we would expect more zero values in the loading matrices³.

4.3.1 A Regularised Expectation Maximisation Algorithm

The regularised likelihood (4.3.2) is incomplete, as whilst we have observations, we do not observe the factors. To solve this problem, we propose to construct an EM framework where we take expectations over the factors (fixing the parameters), then conditional on the expected factors we maximise the log-likelihood with respect to the parameters $\boldsymbol{\theta}$, we iterate this process until our estimates converge.

³In Chapter 3, λ was used to represent the ℓ_1 -norm penalty parameter. In Chapter 4 and 5 this notation is changed to be α to not clash with the factor loadings notation.

The EM algorithm involves calculating and maximising the expected log-likelihood of the DFM conditional on the available information Ω_n . Given the log-likelihood in (4.3.1), the conditional expected log-likelihood is

$$\begin{aligned} \mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \Omega_n] &= -\frac{1}{2} \log |\mathbf{P}_0| - \text{tr} \left\{ \mathbf{P}_0^{-1} \mathbb{E} [(\mathbf{F}_0 - \boldsymbol{\alpha}_0)(\mathbf{F}_0 - \boldsymbol{\alpha}_0)^\top | \Omega_n] \right\} \\ &\quad - \frac{n}{2} \log |\boldsymbol{\Sigma}_u| - \frac{1}{2} \sum_{t=1}^n \text{tr} \left\{ \boldsymbol{\Sigma}_u^{-1} \mathbb{E} [\mathbf{u}_t^\top \mathbf{u}_t | \Omega_n] \right\} \\ &\quad - \frac{n}{2} \log |\boldsymbol{\Sigma}_\epsilon| - \frac{1}{2} \sum_{t=1}^n \text{tr} \left\{ \boldsymbol{\Sigma}_\epsilon^{-1} \mathbb{E} [\boldsymbol{\epsilon}_t^\top \boldsymbol{\epsilon}_t | \Omega_n] \right\} . \end{aligned} \quad (4.3.3)$$

Ultimately, we wish to impose our regularisation on the expected log-likelihood, with our feasible estimator being given by

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} [-\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \Omega_n] + \alpha \|\boldsymbol{\Lambda}\|_1] . \quad (4.3.4)$$

We now provide details on the maximisation step and expectation step of our EM algorithm. Full derivations of the parameters obtained in the maximisation step can be found in Appendix C1. The expectation step involves univariate Kalman filter and smoother equations given in Appendix A.2.

The maximisation step

We use the following notation for the conditional mean and covariances of the state:

$$\begin{aligned} \mathbf{a}_{t|s} &= \mathbb{E} [\mathbf{F}_t | \Omega_s] , \\ \mathbf{P}_{t|s} &= \text{Cov}[\mathbf{F}_t | \Omega_s] , \\ \mathbf{P}_{t,t-1|s} &= \text{Cov}[\mathbf{F}_t, \mathbf{F}_{t-1} | \Omega_s] . \end{aligned}$$

conditional on all information we have observed up to a time s , denoted by Ω_s .

As shown in Bańbura and Modugno (2014), and fully derived in Appendix C1, the

maximisation of (4.3.3) results in the following expressions for the parameter estimates:

$$\hat{\boldsymbol{\alpha}}_0 = \mathbf{a}_{0|n} \quad ; \quad \hat{\mathbf{P}}_0 = \mathbf{P}_{0|n}$$

and letting $\mathbf{S}_{t|n} = \mathbf{a}_{t|n}\mathbf{a}_{t|n}^\top + \mathbf{P}_{t|n}$, and $\mathbf{S}_{t,t-1|n} = \mathbf{a}_{t|n}\mathbf{a}_{t-1|n}^\top + \mathbf{P}_{t,t-1|n}$ we have

$$\hat{\mathbf{A}} = \left(\sum_{t=1}^n \mathbf{S}_{t-1|n} \right)^{-1} \left(\sum_{t=1}^n \mathbf{S}_{t,t-1|n} \right), \quad (4.3.5)$$

$$\hat{\boldsymbol{\Sigma}}_{\mathbf{u}} = \frac{1}{n} \sum_{t=1}^n \left[\mathbf{S}_{t|n} - \hat{\mathbf{A}}(\mathbf{S}_{t-1,t|n}) \right]. \quad (4.3.6)$$

To minimise (4.3.4) for parameters $\boldsymbol{\Lambda}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$, we should also consider there might be missing data in \mathbf{X}_t . Let us define a selection matrix \mathbf{W}_t to be a diagonal matrix such that

$$W_{t,ii} = \begin{cases} 1 & \text{if } X_{i,t} \text{ observed} \\ 0 & \text{if } X_{i,t} \text{ missing} \end{cases}$$

and note that $\mathbf{X}_t = \mathbf{W}_t \mathbf{X}_t + (\mathbf{I} - \mathbf{W}_t) \mathbf{X}_t$. The update for the idiosyncratic error covariance is then given by

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\epsilon}} = \frac{1}{n} \sum_{t=1}^n \text{diag} \left[\mathbf{W}_t \left(\mathbf{X}_t \mathbf{X}_t^\top - 2 \mathbf{X}_t \mathbf{a}_{t|n}^\top \hat{\boldsymbol{\Lambda}}^\top + \hat{\boldsymbol{\Lambda}} \mathbf{S}_{t|n} \hat{\boldsymbol{\Lambda}}^\top \right) + (\mathbf{I} - \mathbf{W}_t) \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\epsilon}}^* (\mathbf{I} - \mathbf{W}_t) \right], \quad (4.3.7)$$

where $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\epsilon}}^*$ is obtained from the previous EM iteration. As noted in Algorithm 1, in practice we update $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\epsilon}}$ after estimating $\hat{\boldsymbol{\Lambda}}$, as the former is based on the difference between the observations and the estimated common component. The following section details precisely how we practically obtain sparse estimates for the factor loadings, the estimates can then be used in (4.3.7) and thus complete the M-step of the algorithm.

Incorporating sparsity

In this work, we propose to update $\hat{\mathbf{A}}$ by constructing an Alternative Directed Method of Moments (ADMM) algorithm (Boyd et al., 2011) to solve (4.3.4) with the parameters $(\hat{\mathbf{A}}, \hat{\Sigma}_u, \hat{\alpha}_0, \hat{\mathbf{P}}_0)$ fixed. We choose to use the ADMM approach as it provides a way to split up the fairly complex objective function into separate, simpler optimisation problems, making the variable updates in every iteration much easier. The algorithm proceeds by sequentially minimising the augmented Lagrangian

$$\mathcal{C}(\mathbf{\Lambda}, \mathbf{Z}, \mathbf{U}) := -\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \Omega_n] + \alpha \|\mathbf{Z}\|_1 + \frac{\nu}{2} \|\mathbf{\Lambda} - \mathbf{Z} + \mathbf{U}\|_F^2, \quad (4.3.8)$$

where $\mathbf{Z} \in \mathbb{R}^{p \times r}$ is an auxiliary variable, $\mathbf{U} \in \mathbb{R}^{p \times r}$ are the (scaled) Lagrange multipliers and ν is the scaling term. Under equality conditions relating the auxiliary (\mathbf{Z}) to the primal ($\mathbf{\Lambda}$) variables, this is equivalent to minimising (4.3.4), e.g.

$$\begin{aligned} & \arg \min_{\mathbf{Z}=\mathbf{\Lambda}} \max_{\mathbf{U}} \mathcal{C}(\mathbf{\Lambda}, \mathbf{Z}, \mathbf{U}) \\ & = \arg \min_{\mathbf{\Lambda}} [-\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \Omega_n] + \alpha \|\mathbf{\Lambda}\|_1] \end{aligned}$$

as (4.3.4) is convex in the argument $\mathbf{\Lambda}$ with all other parameters fixed, this argument holds for any $\nu > 0$ (Boyd et al., 2011; Lin et al., 2015).

The augmented Lagrangian (4.3.8) can be sequentially minimised via the following updates

$$\begin{aligned} \mathbf{\Lambda}^{(k+1)} &= \arg \min_{\mathbf{\Lambda}} \mathcal{C}(\mathbf{\Lambda}, \mathbf{Z}^{(k)}, \mathbf{U}^{(k)}) \\ \mathbf{Z}^{(k+1)} &= \arg \min_{\mathbf{Z}} \mathcal{C}(\mathbf{\Lambda}^{(k+1)}, \mathbf{Z}, \mathbf{U}^{(k)}) \\ &= \text{soft}(\mathbf{\Lambda}^{(k+1)} + \mathbf{U}^{(k)}; \alpha/\nu) \\ \mathbf{U}^{(k+1)} &= \mathbf{U}^{(k)} + \mathbf{\Lambda}^{(k+1)} - \mathbf{Z}^{(k+1)}. \end{aligned}$$

for $k = 0, 1, 2, \dots$, until convergence. Here, $\text{soft}(\cdot)$ refers to the soft thresholding operator defined by

$$\text{soft}_\kappa(x) = \begin{cases} x - \kappa & x > \kappa, \\ 0 & |x| \leq \kappa, \\ x + \kappa & x < -\kappa. \end{cases}$$

The first (primal) update is simply a least-squares type problem, whereby on vectorising $\mathbf{\Lambda}$ one finds

$$\begin{aligned} \text{vec}(\mathbf{\Lambda}^{(k+1)}) &= \left(\sum_{t=1}^n \mathbf{S}_{t|n} \otimes \mathbf{W}_t \mathbf{\Sigma}_\epsilon^{-1} \mathbf{W}_t + \nu \mathbf{I}_{pr} \right)^{-1} \times \\ &\quad \text{vec} \left[\sum_{t=1}^n \mathbf{W}_t \mathbf{\Sigma}_\epsilon^{-1} \mathbf{W}_t \mathbf{X}_t \mathbf{a}_{t|n}^\top + \nu (\mathbf{Z}^{(k)} - \mathbf{U}^{(k)}) \right]. \end{aligned} \quad (4.3.9)$$

Derivations of the updates can be found in Appendix C.2.

Remark 2 (Exploiting dimensionality reduction): For the $\mathbf{\Lambda}^{(k+1)}$ update, the dimensionality of the problem is quite large, leading to a naïve per-iteration cost of order $\mathcal{O}(r^3 p^3)$. A more efficient method for this step can be sought by looking at the specific structure of the matrix to be inverted. Define $\mathcal{A}_t = \mathbf{S}_{t|n}$, $\mathcal{B}_t = \mathbf{W}_t \mathbf{\Sigma}_\epsilon^{-1} \mathbf{W}_t$, and $\mathcal{C} = \sum_{t=1}^n \mathbf{W}_t \mathbf{\Sigma}_\epsilon^{-1} \mathbf{W}_t \mathbf{X}_t \mathbf{a}_{t|n}^\top + \nu (\mathbf{Z}^{(k)} - \mathbf{U}^{(k)})$, then the solution (4.3.9) can be written as

$$\begin{aligned} \text{vec}(\mathbf{\Lambda}) &= \left(\sum_{t=1}^n \mathcal{A}_t \otimes \mathcal{B}_t + \nu \mathbf{I}_{pr} \right)^{-1} \text{vec}(\mathcal{C}) \\ &= \mathcal{D}^{-1} \text{vec}(\mathcal{C}). \end{aligned}$$

Since $\mathbf{\Sigma}_\epsilon$ is diagonal in an exact DFM, \mathcal{B}_t is also diagonal and thus \mathcal{D} is made up of r^2 blocks such that each $(i, j)^{th}$ block is a diagonal matrix of length p for $i, j = 1, \dots, r$. To speed up the computation, we note that $\nu \mathbf{I}_{pr} = \nu \mathbf{I}_r \otimes \mathbf{I}_p$ and use the properties of

commutation matrices (Magnus and Neudecker, 2019, p. 54), denoted by \mathbf{K}_{rp} , to write

$$\begin{aligned}
& \left(\sum_{t=1}^n \mathcal{A}_t \otimes \mathcal{B}_t + \nu \mathbf{I}_r \otimes \mathbf{I}_p \right)^{-1} \\
&= \left[\sum_{t=1}^n \mathbf{K}_{rp} (\mathcal{B}_t \otimes \mathcal{A}_t) \mathbf{K}_{pr} + \mathbf{K}_{rp} (\mathbf{I}_p \otimes \nu \mathbf{I}_r) \mathbf{K}_{pr} \right]^{-1} \\
&= \mathbf{K}_{rp} \left(\sum_{t=1}^n (\mathcal{B}_t \otimes \mathcal{A}_t) + (\mathbf{I}_p \otimes \nu \mathbf{I}_r) \right)^{-1} \mathbf{K}_{pr}. \tag{4.3.10}
\end{aligned}$$

The matrix needing to be inverted in the final line of equation (4.3.10) is now a block diagonal matrix. We can extract each of the $1, \dots, p$ blocks separately and invert them one-by-one. The final result from (4.3.10) has the expected block structure with a diagonal matrix in each block but we can stack them into a cube to save storage. Overall, the operations can be completed with cost $\mathcal{O}(r^3p)$. Given that this needs to be performed for every iteration of the EM algorithm, our commutation trick results in significant computational gains.

Whilst other optimisation routines could be used to estimate the sparse loadings, the ADMM approach is appealing as it allows us to split (4.3.4) into sub-problems that can easily be solved. If one wished to incorporate more specific/structured prior knowledge, this approach can easily be altered to impose these assumptions, for instance, future work could consider group-structured regularisation allowing for more informative prior knowledge on the factor loadings to be incorporated. Hard constraints, e.g. where we require a loading to be exactly zero can also be incorporated at the \mathbf{Z} update stage by explicitly setting some entries to be zero. More discussion on future work ideas can be found in Chapter 7.

The expectation step

So far, we have discussed how to update the parameters conditional on the quantities $\mathbb{E}[\mathbf{F}_t | \boldsymbol{\Omega}_n]$, $\text{Cov}[\mathbf{F}_t | \boldsymbol{\Omega}_n]$, and $\text{Cov}[\mathbf{F}_t, \mathbf{F}_{t-1} | \boldsymbol{\Omega}_n]$. In our application, under the Gaussian

error assumption, these expectations can be easily calculated via the Kalman smoother.

More formally, given a set of parameter estimates $\hat{\boldsymbol{\theta}}$ obtained from the maximisation step, we can set up the state-space framework of the DFM as in 4.2.1 and run the univariate Kalman filter and smoothing equations of Koopman and Durbin (2000) to obtain quantities $\mathbb{E}[\mathbf{F}_t|\boldsymbol{\Omega}_n]$, $\text{Cov}[\mathbf{F}_t|\boldsymbol{\Omega}_n]$, and $\text{Cov}[\mathbf{F}_t, \mathbf{F}_{t-1}|\boldsymbol{\Omega}_n]$. As discussed in Appendix A.2, the univariate treatment (sequential processing) of the classical multivariate Kalman filter and smoother equations can lead to substantial computational gains when p is large, as processing observations \mathbf{X}_t one element at a time, as opposed to updating all p of them together, allows slow matrix inversions to become fast scalar divisions.

Remark 3 (Calculating the cross-covariance matrix): It is not obvious from Koopman and Durbin (2000) how the cross-covariance matrix, $\text{Cov}[\mathbf{F}_t, \mathbf{F}_{t-1}|\boldsymbol{\Omega}_n]$, is obtained. To calculate it, we make use of De Jong and Mackinnon (1988)'s theorem:

$$\mathbf{P}_{t-1,t|n} = \mathbf{P}_{t-1|t-1} \mathbf{A}^\top (\mathbf{P}_{t|t-1})^{-1} \mathbf{P}_{t|n},$$

which, after transposition, gives

$$\mathbf{P}_{t,t-1|n} = \mathbf{P}_{t|n} (\mathbf{P}_{t|t-1})^{-1} \mathbf{A} \mathbf{P}_{t-1|t-1}. \quad (4.3.11)$$

This is directly obtainable from the Kalman filter and smoother equations.

The expectation and maximisation steps are iterated until convergence which we define happens when

$$\frac{\log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j)}) - \log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j-1)})}{\left(\log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j)}) + \log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j-1)}) \right) / 2} < 10^{-4}, \quad (4.3.12)$$

for iteration j . The log-likelihood $\log \mathcal{L}(\mathbf{X}; \hat{\boldsymbol{\theta}}^{(j)})$ is given in equation (A.2.5) of Appendix A.2.

4.3.2 Parameter Tuning

There are two key parameters that need to be set for the sparse DFM. The first is to select the number of factors, and the second is to select an appropriate level of sparsity. One may argue that these quantities should be selected jointly, however, in the interests of computational feasibility, we here propose to use heuristics, first selecting the number of factors, and then deciding on the level of sparsity. This mirrors how practitioners would typically apply the DFM model, where there is often a prior for the number of relevant factors (or more usually an upper bound). Both the number of factors, and the structure of the factor loadings impact the practical interpretation of the estimated factors.

Choosing the number of factors

To calculate the number of factors to use in the model we opt to take the information criteria approach of Bai and Ng (2002). There are several criteria that are discussed in the literature, for example, the paper of Bai and Ng (2002) suggests three forms⁴. We use the criteria of the following form:

$$IC(r) = \log V_r(\bar{\mathbf{F}}, \bar{\mathbf{\Lambda}}) + r \left(\frac{n+p}{np} \right) \log \min(n, p), \quad (4.3.13)$$

where

$$V_r(\bar{\mathbf{F}}, \bar{\mathbf{\Lambda}}) = \frac{1}{np} \sum_{i=1}^p \sum_{t=1}^n \mathbb{E} [\bar{\epsilon}_{i,t}^2]$$

and $\bar{\epsilon}_{i,t} = X_{t,i} - \bar{\mathbf{\Lambda}}_{i,\cdot} \bar{\mathbf{F}}_t$ is found using PCA when applied to the standardised data. The preliminary factors $\bar{\mathbf{F}}$ correspond to the principal components, and the estimated loadings $\bar{\mathbf{\Lambda}}$ corresponding to the eigenvectors. Should the data contain missing values, they can be interpolated using the median of the series and then smoothed with a simple

⁴In our experiments and applications, we compared all criteria and they typically give similar results within ± 1 of each other, for simplicity, only one IC is presented here. Other IC are available in the `sparseDFM` package discussed in Chapter 6.

moving window. This is taken as the initialisation step in the `sparseDFM` package.

Remark 4: We note that ideally one may wish to apply the EM procedure to get more refined estimates of both the factors and loadings, however, in the interests of computational cost and in-line with current practice we propose to use the quick (preliminary) estimates above, denoted with $\bar{\Lambda}$ rather than $\hat{\Lambda}$.

Tuning the regulariser

Once a number of factors r has been decided, we tune α by performing a simple search over a logarithmically spaced grid and minimise a Bayesian Information Criteria defined as

$$BIC(\alpha) = \log \left(V_{\alpha}(\hat{\mathbf{F}}, \hat{\Lambda}) \right) + \frac{\log(np)}{np} \sum_{k=1}^r \hat{s}_k, \quad (4.3.14)$$

where \hat{s}_k is the number of non-zero entries in the k th column of the estimated loading matrix. In this case, we run the EM algorithm until convergence and then evaluate the BIC using the resulting $\hat{\mathbf{F}}$ and $\hat{\Lambda}$, this procedure is repeated for each α in the grid. An example of the resulting curve can be seen in the empirical applications of Sections 4.5 and 4.6. To limit searching over non-optimal values, an upper limit for α is set whereby, if the loadings for a particular factor are all set to zero, then we terminate the search.

Remark 5: Tuning both the number of factors, and the regulariser for these models is a topic of open research and discussion. Indeed, whilst the criteria of Bai and Ng (2002) are well used, there is still lively debate about what is an appropriate number of factors, and this usually determined by a mix of domain (prior) knowledge and heuristics such as those presented above. The heuristics provided here seem reasonable in the applications and experiments we consider, however, we do not claim they are optimal for all scenarios.

4.3.3 Implementation

The estimation routine is implemented as part of the R package `sparseDFM` available via CRAN and presented in Chapter 6. In brief, the EM routine and ADMM updates are implemented in C++ using the Armadillo library. Initialisation of the ADMM iterates utilises a warm start procedure whereby the solution at the previous iteration of the EM algorithm initialises the next solution. Furthermore, warm-starts are utilised when searching over an α tuning grid. As noted in other applications (Hu et al., 2016) starting the ADMM procedure can lead to considerable speed-ups. With regards to the augmentation parameter ν in the ADMM algorithm, this is simply set to 1 for the experiments run in this chapter, however, it is possible that tuning this parameter could lead to further speedups.

On the first iteration of the algorithm, the EM procedure is initialised by a simple application of PCA to the standardised data, analogously to how the preliminary factors $\bar{\mathbf{F}}$ and loadings $\bar{\mathbf{\Lambda}}$ were found in Section 4.3.2. A VAR(1) model is then fitted to the preliminary factors to obtain estimates for \mathbf{A} and $\mathbf{\Sigma}_u$. The initial state distribution $\mathbf{F}_0 \sim N(\boldsymbol{\alpha}_0, \mathbf{P}_0)$ is such that $\boldsymbol{\alpha}_0 = \mathbf{0}$ and $\text{Vec}(\mathbf{P}_0) = (\mathbf{I}_{rp} - \mathbf{A} \otimes \mathbf{A})^{-1} \text{Vec}(\mathbf{\Sigma}_u)$. A summary of the EM algorithm as a whole is given in Algorithm 2.

4.4 Simulation Study

We provide a Monte-Carlo numerical study to show the performance of our QMLE estimator in terms of recovery of sparse loadings and the ability of the sparse DFM to forecast missing data at the end of the sample. In particular, we simulate from a ground-truth model according to:

$$\begin{aligned} \mathbf{X}_t &= \mathbf{\Lambda} \mathbf{F}_t + \boldsymbol{\epsilon}_t, & \boldsymbol{\epsilon}_t &\sim N(\mathbf{0}, \mathbf{\Sigma}_\epsilon), \\ \mathbf{F}_t &= \mathbf{A} \mathbf{F}_{t-1} + \mathbf{u}_t, & \mathbf{u}_t &\sim N(\mathbf{0}, \mathbf{\Sigma}_u), \end{aligned}$$

Algorithm 2 EM algorithm for sparse DFM

Require: \mathbf{X} , α

Ensure: Λ , \mathbf{A} , Σ_ϵ , Σ_u

- 1: Initialize $\boldsymbol{\theta} = (\Lambda, \mathbf{A}, \Sigma_\epsilon, \Sigma_u)$ via cubic spline fitting (for missing value imputation) followed by PCA and a VAR fit
 - 2: **repeat**
 - 3: • **E-step:**
 - 4: Obtain $\mathbf{a}_{t|n}$ and $\mathbf{P}_{t|n}$ via univariate Kalman filtering and smoothing
 - 5: Calculate $\mathbf{P}_{t,t-1|n}$ via Eq. (4.3.11)
 - 6: • **M-step:**
 - 7: Update \mathbf{A} and Σ_u via Eqs. (4.3.5) and (4.3.6)
 - 8: Initialize $\Lambda^{(0)} = \mathbf{Z}^{(0)} = \mathbf{U}^{(0)} = 0$
 - 9: **for** $k = 0, \dots$, until convergence **do**
 - 10: $\Lambda^{(k+1)} = \arg \min_{\Lambda} \mathcal{C}(\Lambda, \mathbf{Z}^{(k)}, \mathbf{U}^{(k)})$ via Eqs. (4.3.9) and (4.3.10)
 - 11: $\mathbf{Z}^{(k+1)} = \text{soft}(\Lambda^{(k+1)} + \mathbf{U}^{(k)}; \alpha/\nu)$
 - 12: $\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + \Lambda^{(k+1)} - \mathbf{Z}^{(k+1)}$
 - 13: **end for**
 - 14: Update Σ_ϵ via Eq. (4.3.7)
 - 15: **until** convergence
-

for $t = 1, \dots, n$ and \mathbf{X}_t having p variables. We set the number of factors to be $r = 2$ and consider true model parameters of the form:

$$\mathbf{\Lambda} = \mathbf{I}_2 \otimes \mathbf{1}_{p/2} = \begin{bmatrix} \mathbf{1}_{p/2} & \mathbf{0}_{p/2} \\ \mathbf{0}_{p/2} & \mathbf{1}_{p/2} \end{bmatrix}, \quad \mathbf{\Sigma}_\epsilon = \mathbf{I}_p, \quad \mathbf{A} = \begin{bmatrix} a & 0 \\ \rho & 0 \end{bmatrix}, \quad \mathbf{\Sigma}_u = \begin{bmatrix} 1 - a^2 & 0 \\ 0 & 1 - \rho^2 \end{bmatrix}.$$

The loadings matrix $\mathbf{\Lambda}$ is a block-diagonal matrix which is $1/2$ sparse with $p/2$ ones in each block. We set up the VAR(1) process of the factors in this way such that we can adjust the cross-correlation parameter ρ between the factors while having factors that always have variance one. This allows us to understand how important a cross-correlation at non-zero lags structure is when assessing model performance. We vary the ρ parameter between $\rho = \{0, 0.6, 0.9\}$, going from no cross-correlation to strong cross-correlation between the factors. We set the covariance of the idiosyncratic errors to be \mathbf{I}_p in order to have a signal-to-noise ratio between the common component $\mathbf{\Lambda}\mathbf{F}_t$ and the errors ϵ_t equal to one.

4.4.1 Recovery of Sparse Loadings

We apply our sparse DFM (SDFM) to simulated data from the data generating process above to assess how well we can recover the true loadings matrix $\mathbf{\Lambda}$. We compare our method to sparse principal component analysis⁵ (SPCA) applied to \mathbf{X}_t to test which settings we are performing better in. We tune for the best ℓ_1 -norm parameter in both SDFM and SPCA using the BIC function (4.3.14) by searching over a large grid of logspaced values from 10^{-3} to 10^2 . We also make comparisons to the regular DFM approach of Bańbura and Modugno (2014) to test the importance of using regularisation when the true loading structure is sparse.

The estimation accuracy is assessed with mean absolute error (MAE) between the estimates and true loadings according to $(rp)^{-1} \|\hat{\mathbf{\Lambda}} - \mathbf{\Lambda}\|_1$. We also provide results for

⁵The SPCA algorithm is implemented using the `elasticnet` R package available on CRAN.

the F1 score for the sparsity inducing methods of SDFM and SPCA to measure how well the methods capture the true sparse structure. Due to invariance issues discussed in Section 4.2.2, the estimated loadings may not be on the same scale as the true loadings, we thus first re-scale the estimated loadings such that their norm is equal to that of the simulated loadings, i.e. $\|\hat{\mathbf{\Lambda}}\|_2 = \|\mathbf{\Lambda}\|_2$. The estimated loadings from each model are identified up to column permutations and therefore we permute the columns of $\hat{\mathbf{\Lambda}}$ to match the true order of $\mathbf{\Lambda}$. We do this by measuring the ℓ_2 -norm distance between the columns of $\hat{\mathbf{\Lambda}}$ and $\mathbf{\Lambda}$ and iteratively swapping to match the smallest distances.

Figure 4.4.1 displays the results for the loadings recovery where we have fixed the number of observations to be $n = 100$ and vary the number of variables between $p = \{18, 60, 120, 180\}$ along the x-axis and the cross-correlation parameter in the VAR(1) process between $\rho = \{0, 0.6, 0.9\}$ going from the left to middle to right plot respectively. The top panel shows the median MAE score (in logarithms) over 100 experiments while the bottom panel shows the F1 scores. We provide confidence bands for both representing the 25th and 75th percentiles.

It is clear from the plots that the sparsity inducing methods of SDFM and SPCA are dominating a regular DFM when the true loadings structure is in fact sparse. It is also clear that SPCA performs poorly, compared with SDFM, when the cross-section of the data increases for a fixed n . This is even more noticeable from the F1 score when ρ increases. This highlights the importance of the SDFM's ability to capture correlations between factors at non-zero lags. Unlike SPCA, the EM algorithm of SDFM allows feedback from the estimated factors when updating model parameters, allowing it to capture these factor dependencies. We see improved scores in MAE as the cross-section increases for SDFM. This follows the intuition of the EM algorithm framework as we learn more about the factors as the dimension $p \rightarrow \infty$. We should remark that for most scenarios the F1 score for SDFM is almost one, however, when $p = 18$ and ρ is high, the score does drop. In this setting, a low value for α minimises BIC, meaning almost no

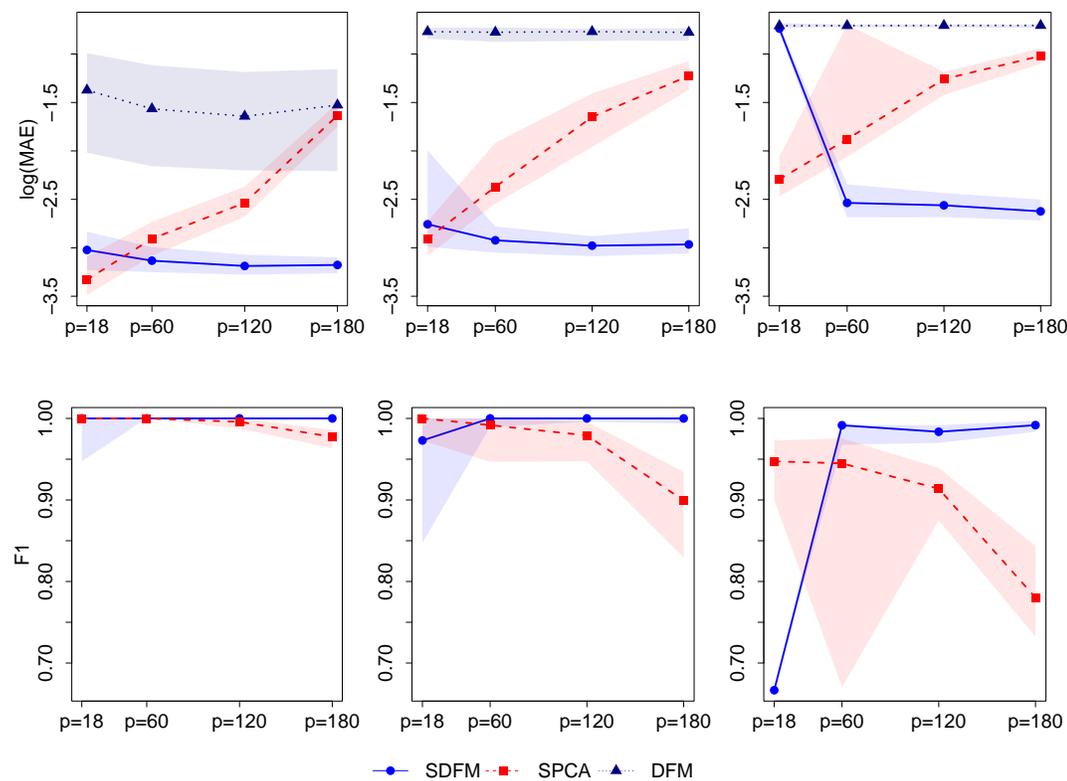


Figure 4.4.1: Median log-MAE score (top panel) and median F1 score (bottom panel) for recovering factor loadings across 100 experiments with a shaded confidence band of the 25th and 75th percentile. The plots represent a setting with a fixed $n = 100$ and varying number of variables p and where the cross-correlation parameter in the VAR(1) process is set to $\rho = 0$ (left plot), $\rho = 0.6$ (middle plot) and $\rho = 0.9$ (right plot).

sparsity is applied (a very similar result to a regular DFM fit). Here, the two factors are highly correlated and it appears the cross-sectional is not large enough to accurately separate the factors. In practice it is likely that cross-section will be large and hence this result is not too concerning.

4.4.2 Forecasting Performance

To evaluate our ability to forecast missing data at the end of the sample, we simulate data from the data generating process above with $n = 200$, $p = 64$ and consider $\rho = \{0, 0.6, 0.9\}$, and assume different patterns of missing data at the end of the sample. We consider a 1-step ahead forecast case where we set 25%, 50%, 75% and then 100% of variables to be missing in the final row of \mathbf{X} . When allocating variables to be missing we split the data up into the two loading blocks and set the first 25%, 50%, 75% and 100% of each loading block to be missing. For example, the variables 1 to 8 and 33 to 40 are missing in the 25% missing scenario. We are interested in forecasting the missing data in the final row of \mathbf{X} and we calculate the average MAE over 100 experiments.

We make comparisons with a sparse vector-autoregression (SVAR) model⁶ as this is a very popular alternative forecasting strategy for high-dimensional time series that is based on sparse assumptions. As our factors are generated using a VAR(1) process with a sparse auto-regression matrix, we are interested to see whether SVAR will be able to capture the cross-factor auto-correlation when producing forecasts. We also apply a standard AR(1) process to each of the variables needing to be forecasted as a benchmark comparison.

Figure 4.4.2 displays the results of the simulations plotting MAE for each of the 3 methods and each simulation setting. In all settings we find SDFM to outperform both SVAR and AR(1). When ρ is set to be 0.9, we find SVAR does improve its forecasting

⁶The SVAR algorithm is implemented using the `BigVAR` R package available on CRAN. This has a built-in cross-validation mechanism to tune for the best ℓ_1 -penalty parameter which we use in our simulations.

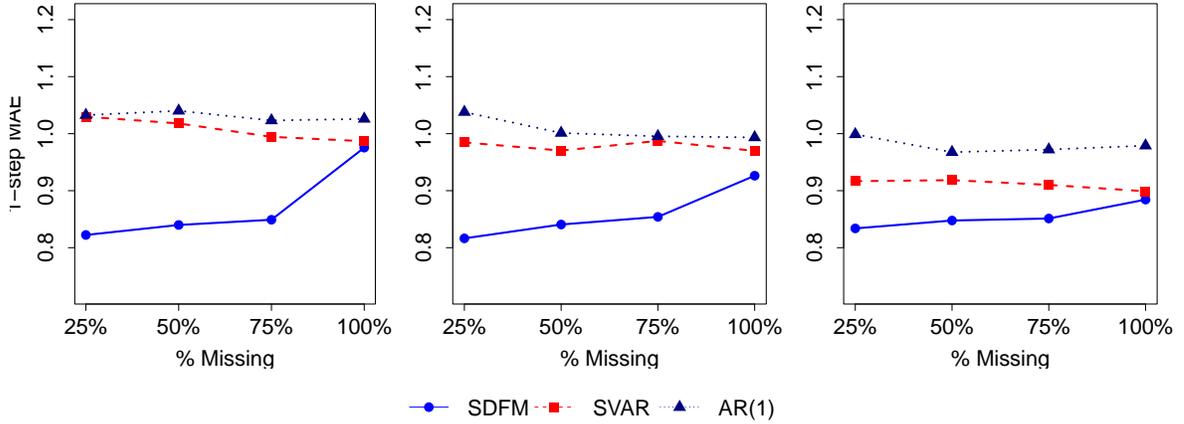


Figure 4.4.2: Average MAE score forecasting, as a function of the level of missing data in the last sample. From left-right: $\rho = 0$, $\rho = 0.6$, $\rho = 0.9$. Plot indicates the 50th percentile of performance across 100 experiments with $n = 100$, $p = 64$.

performance as opposed to when $\rho = 0$ as the VAR(1) process driving the factors becomes more prominent. The results confirm SDFM’s ability to make use of variables that are present at the end of the sample when forecasting the missing variables. We see this by the rise in MAE when 100% of the variables are missing at the end of the sample and the model can no longer utilise available data in this final row. The MAE remains fairly flat as the amount of missingness rises from 25% to 75% showing SDFM’s ability to forecast correctly even when there is small amount of data available.

4.4.3 Computational Efficiency

To assess the computational scalability, we simulate from a sparse DFM where $\mathbf{\Lambda} = \mathbf{I}_r \otimes \mathbf{1}_{p/r}$ and $\mathbf{\Sigma}_\epsilon = \mathbf{I}_p$, and the factors are a VAR(1) with $\mathbf{A} = 0.8 \times \mathbf{I}_r$ and $\mathbf{\Sigma}_u = (1 - 0.8^2) \times \mathbf{I}_r$. We record the number of EM iterations and the time they take for each ℓ_1 -norm parameter α up to the optimal ℓ_1 -norm parameter $\hat{\alpha}$ and then take the average time of a single EM iteration. We repeat the experiment ten times for each experimental configuration.

The results are presented in Figure 4.4.3, which demonstrates scalability as a function

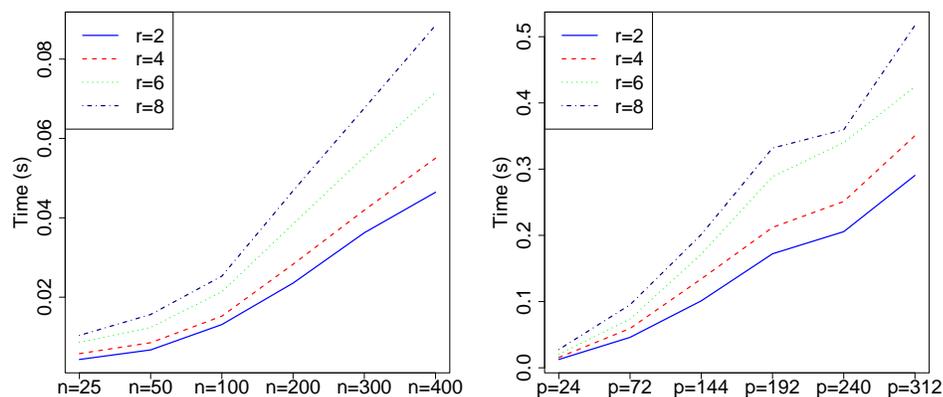


Figure 4.4.3: Summary of computational cost. Left: as a function of n , with fixed $p = 24$. Right: as a function of p , with fixed $n = 100$. Average performance across 10 experiments.

of n , and p , under different assumptions on the number of factors $r = 2, 4, 6, 8$. As expected, the cost is approximately linear in n and p , with increasing cost as a function of the number of factors r . The results demonstrate the utility of using the univariate smoothing approach as well as the matrix decomposition when calculating required inversions.

4.5 Case Study 1 - Nowcasting UK Trade in Goods

This first case study attempts to explore the benefits of our sparse DFM versus the classic DFM in a typical macroeconomic nowcasting exercise on monthly UK Trade in Goods exports data with a large set of monthly indicator series. The second case study in Section 4.6 moves away from the field of economics and uses the sparse DFM to interpret energy consumption around our university campus and compares its forecasting performance with competing methods. Together, both case studies hopefully demonstrate how practitioners can use the sparse DFM to draw conclusions from the induced sparse factor structure to interpret key data sets and accurately forecast.

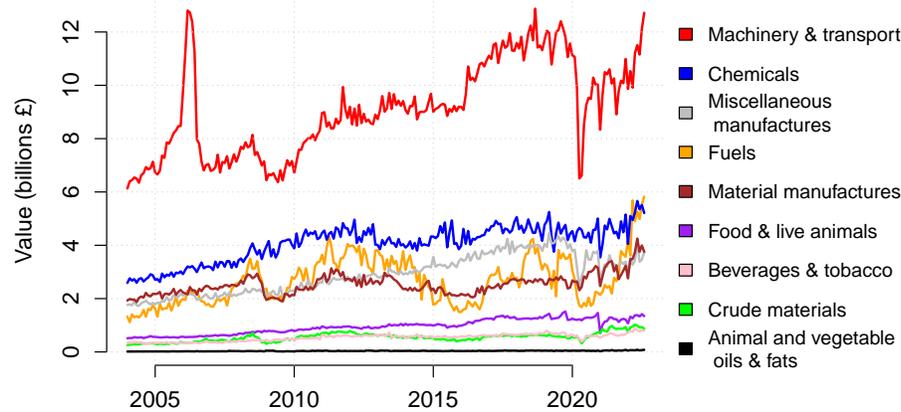


Figure 4.5.1: Breakdown of the 9 trade in goods target series.

4.5.1 Data

The UK's economy is heavily dependent on international trade, with exports and imports of goods and services making up a significant proportion of its GDP. In 2022, goods trade totalled £815 billion, while service trade totalled £902 billion (UK Parliament, 2023). Currently, the Office for National Statistics (ONS) publish UK trade in goods data monthly with a 2 month publication lag. This includes a breakdown of all commodities traded to and from countries worldwide. Trade in services data on the other hand is only available monthly for an aggregate worldwide figure for total imports and exports, information on the breakdown of services and country trade is quarterly with a 3 month publication lag. Given key events in recent years that have imposed restrictions on international trade for the UK, such as voting to leave the European Union (EU) in 2016 and the coronavirus pandemic reaching UK shores early 2020, it is of significant interest to produce timely estimates of trade data avoiding the long 2-3 month publication delays.

In this case study we focus on nowcasting trade in goods data by utilising a large set of monthly indicators available within the 2 month publication lag window. In

Label	No.	Data	Description	Delay
TiG	9	Trade in Goods	Value of UK exports of the 9 main commodities by current price.	2
IoP	89	Index of Production	Movements in the volume of production for the UK production industries.	2
CPI	166	Consumer Price Inflation	The rate at which the prices of goods and services bought by households rise or fall.	1
PPI	153	Producer Price Inflation	Changes in the prices of goods bought and sold by UK manufacturers.	1
Exch	12	Exchange Rates	Sterling exchange rates with 12 popular currencies.	1
Conf	2	Confidence Indices	Business and consumer confidence indices from opinion surveys.	1
GT	14	Google Trends	Popularity scores of 14 google search queries related to trade in goods	0

Table 4.5.1: Description of data used in the model including goods target series and monthly indicator series.

particular, we look to nowcast 9 of the main commodities the UK exports worldwide. A breakdown of these 9 trade commodities are available in Figure 4.5.1. The nowcasting task of trade in services data is left for Chapter 5, where a large state-space model is introduced combing the sparse DFM with a multivariate temporal disaggregation framework to allow quarterly-to-monthly disaggregation of service data.

As trade data accounts for a large proportion of the UK's GDP, there are many potential informative economic indicator series recorded at the monthly frequency (or higher) available for us to use. It is important to consider the availability of indicators in the nowcasting months of interest as we wish to extract as much observed information as possible when projecting the latent targets onto this space. With this in mind, for the task of nowcasting monthly goods trade data, we use a collection of 434 monthly indicator variables that are displayed in Table 4.5.1, along with the 9 monthly goods target series the UK exports worldwide. The second column, *No.*, gives the number of series used for the data set, while the final column, *Delay*, gives the publication delay in months of each variable.

The index of production is used as an important macroeconomic indicator providing insight into the overall growth of the economy by tracking changes in the output of the

manufacturing sector. Indices on consumer and producer inflation aim to capture price changes which may contain information about aggregate demand. They are available with a 1 month delay and hence we can exploit more information at the end of the sample. Exchange rates can have significant impact on demand and transaction costs of goods trade and can affect the price competitiveness of goods in international markets. Similarly, opinion surveys on business and consumer confidence are soft indicators that provide economic agents' estimations on supply and demand of goods. Finally, we use Google trends data with 14 search queries related to trade in goods. Google trends is a tool that tracks the frequency and popularity of search terms on Google. It has become a popular indicator for macroeconomic forecasting purposes in recent years as it allows the analysis of timely, comprehensive and cost-effective data on consumer behavior and trends (Choi and Varian, 2012; Ferrara and Simoni, 2022). It is available with no delay, providing us with real-time information for our nowcasts.

4.5.2 Interpreting Factor Structure

The full data sample we collect is January 2004 to October 2022, with missing data only present in the ragged edge at the end of the sample, e.g. the months of September and October 2022 are missing for the goods target series. The data is first made sufficiently stationary by taking first-differences of all variables. A first task involves getting a sense of the sources of variation amongst the entire high-dimensional data set. Can we interpret the latent factor structure that capture the variation? To estimate factors we first choose the number of factors to use according to the Bai and Ng (2002) criteria given in (4.3.13). According to this criteria, the best number of factors to use is 7 as shown in Figure 4.5.2 (top left). However, the screeplot (Fig. 4.5.2 [top right]) seems to suggest that after 4 factors, the addition of more factors does not add that much in terms of explaining the variance of the data. For this reason, we choose to use 4 factors when modelling.

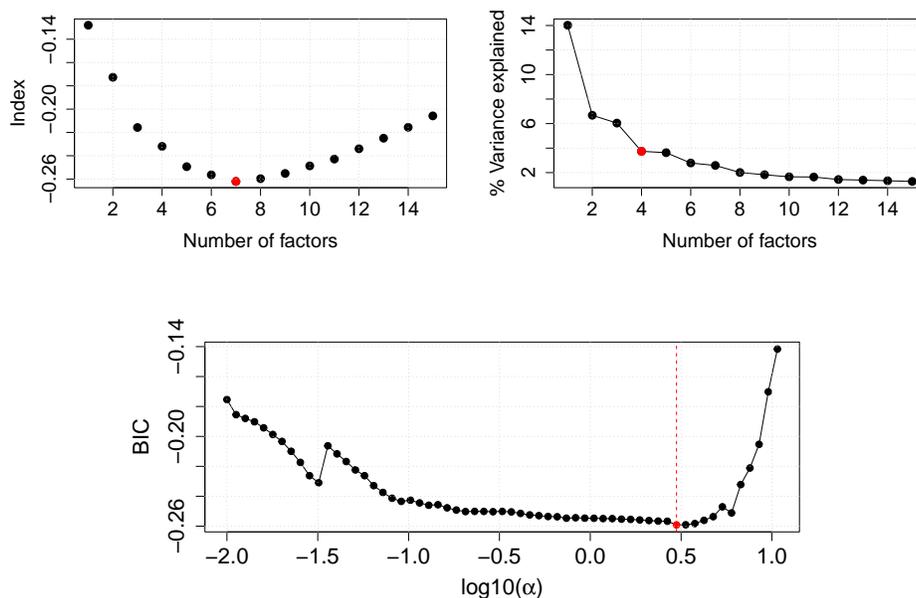


Figure 4.5.2: Top left: Information Criteria (4.3.13) as a function of number of retained factors r . Top right: Proportion of variance explained, based on PCA applied to the scaled and pre-processed data set. Bottom: BIC as a function of α , vertical line indicates minimiser and the α used in the subsequent analysis.

We now fit the sparse DFM to the entire data set (both monthly targets and indicators) via the EM procedure in Algorithm 2 and tune for a range of α parameters in a log-space grid between 10^{-2} and 10^3 . We restrict α to be 0 (i.e. unregularised OLS loadings) for the 9 monthly target series as we do not wish to impose sparsity on the series we wish to nowcast. Figure 4.5.2 (bottom) displays the entire BIC path from 4.3.14 with the realised minimiser at $\hat{\alpha} \approx 2.984$. The BIC quickly starts to rise after this value as $\hat{\mathbf{\Lambda}}$ becomes more sparse until the cut-off constraint at $\alpha \approx 10.723$ when an entire column of $\mathbf{\Lambda}$ is set to zero.

The estimated factor loadings using $\alpha \approx 2.984$ are presented in Figure 4.5.3 adjacent to the estimated loadings in a regular DFM fit ($\alpha = 0$ for all variables). For the regular DFM, the estimated factor structure is very difficult to interpret as the loadings are present in each with a high value (in absolute value). Factor 1 could be interpreted as a measure of inflation (CPI and PPI heavily present), although exchange rates also have

quite a strong loading. Factor 2 is more heavily weighted with the production index, however, all other indicators are also heavily present and could be important. Factors 3 and 4 are almost impossible to interpret with every indicator seeming important. The loading plots from sparse DFM are much easier to interpret with many indicator groups becoming sparse. Factor 1 is clearly a measure of producer price inflation. We can observe the dynamics of factors in Figure 4.5.4, and see the top figure, corresponding to Factor 1, follows very similar dynamics to an overall measure of PPI, with the incline post 2020. Factor 2 in Figure 4.5.4 shows a clear shock for the COVID-19 pandemic matching the dynamics of the production index for the UK. We find the loadings of Factor 2 to be almost entirely weighted by the production index along with certain google search words to do with trading oil. Factor 3 is almost entirely loaded by CPI along with google searches to do with manufacturing, cars and animals. Factor 4 is negatively loaded with a mixture of CPI and PPI and exchange rates data and we see the dynamics of this factor in Fig. 4.5.4 has rises during the recessions of 2009 and 2020.

4.5.3 Pseudo-Real Time Nowcasting Exercise

The primary objective of the sparse DFM is to make the factor structure for a complex data set much easier to interpret than a regular DFM – this is apparent in Figure 4.5.3. It is also of interest to explore if the sparse DFM also produces more accurate nowcasts than the regular DFM with this data. We can set up a pseudo-real time nowcasting exercise where we try and predict the missing ragged edge in an expanding window from August 2018 to July 2022. As it would be difficult to store all previous vintages of data back to 2018, we just use the complete data set (up to October 2022) and at each nowcasting window create a ragged edge following the 2 month lag for the targets and IoP and 1 month lag for everything else except google trends. We begin with the window Jan 2004 - Aug 2018, where the current available target exports data is June 2018. We produce nowcasts for July 2018 (representing a nowcast 1 month prior to

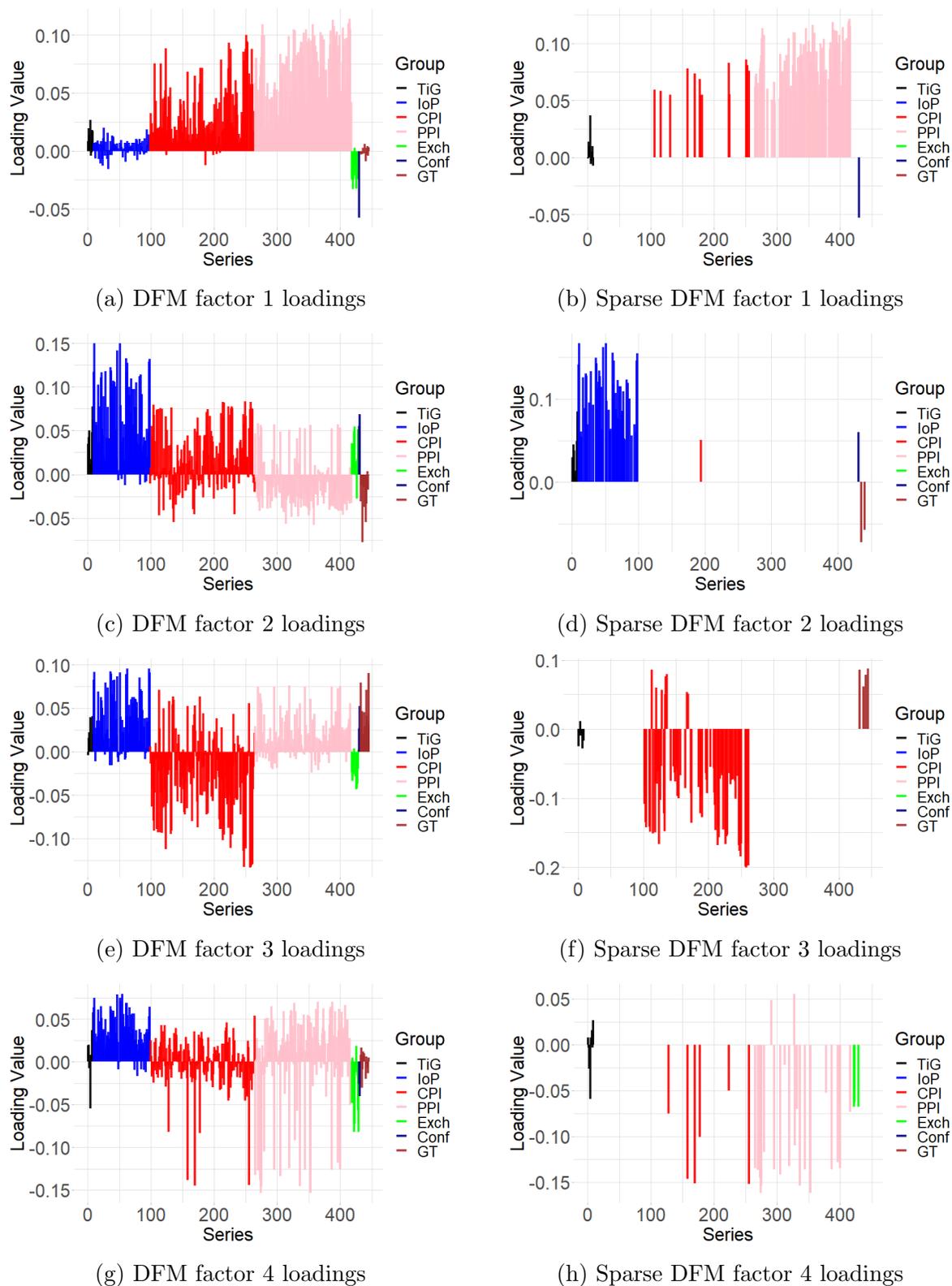


Figure 4.5.3: Factor loadings estimates for all 4 factors using a regular DFM and a sparse DFM. Colour coded by indicator series data group of Table 4.5.1.

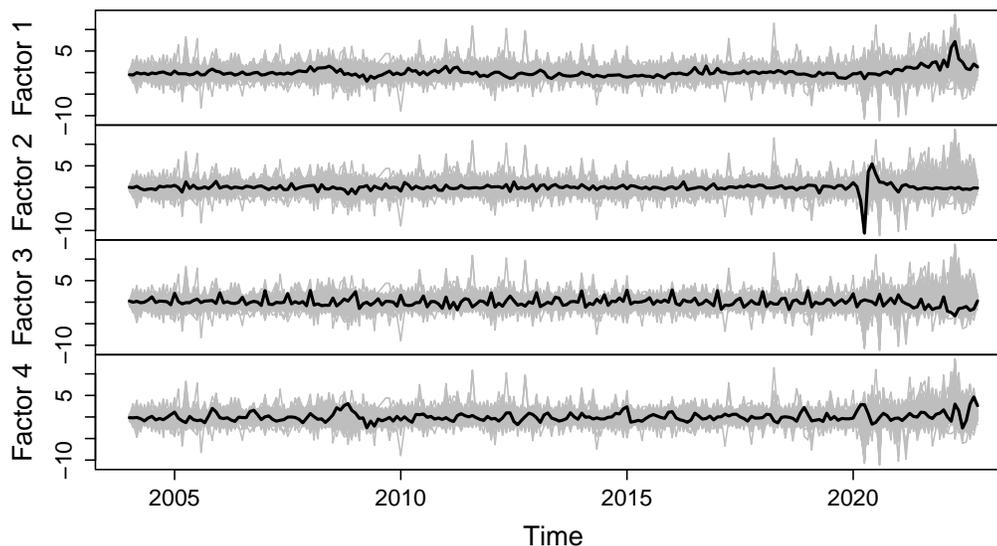


Figure 4.5.4: Estimated factors (black) with original data (grey) as a function of time using the optimal $\alpha \approx 2.984$ chosen according to BIC.

	Model	Mean	25%	50%	75%
Horizon 1	DFM	373.7	152.4	232.9	455.1
	SDFM	297.2	141.1	212.4	306.9
Horizon 2	DFM	437.2	201.8	302.7	515.1
	SDFM	357.6	161.9	233.8	366.4

Table 4.5.2: Pseudo real-time nowcasting exercise results.

release) and for August 2018 (representing a nowcast 2 months prior to release). We produce these nowcasts for horizon 1 and 2 for every month up to July 2022 - a total of 48 months.

Table 4.5.2 displays the average, 25th, 50th and 75th percentiles of the mean absolute error across the 48 month expanding window for horizon 1 and 2 in both models. It appears that Sparse DFM is performing better than a regular DFM, with lower average mean absolute error and tighter bands around the median. As expected, the error for horizon 1 is slightly lower than horizon 2 as it is able to exploit all indicators with a 1 month lag in its estimation.

4.6 Case Study 2 - The Dynamic Factors of Energy Consumption Around a University Campus

This section details application of the sparse DFM to a real-world problem, namely the forecasting and interpretation of energy consumption across our university campus. This second case study is provided with hope to demonstrate the sparse DFM's versatility on alternative data sets away from economics. Beyond forecasting consumption in the near-term future, the aim here is to also characterise the usage in terms of what may be considered typical consumption profiles. These are of specific interest to energy managers and practitioners, as understanding how energy is consumed in distinct buildings can help target interventions and strategy to reduce waste. Also demonstrated is how the sparse DFM, and in particular the EM algorithm, can be used to impute missing data and provide further insight.

4.6.1 Data and Preprocessing

In this application, the data consists of one month of electricity consumption data measured across $p = 42$ different buildings on our universities campus. This data is constructed based on a larger dataset, which monitors energy at different points throughout a building, in our case, we choose to aggregate the consumption so that one data stream represents the consumption of a single building. The data is gathered at 10 minute intervals (measuring consumption in kWh over that interval), resulting in $n = 3,456$ data points spanning 24 days worth of consumption in November 2021, we further hold out one day $n_{\text{test}} = 144$ data points to evaluate the out-of-sample performance of the DFM model. An example of time series from the data set is presented in Figure 4.6.1.

In the energy consumption forecasting literature, various alternative methods exist to model this data, such as generalized additive models (Wood, 2015) and multiple seasonal

exponential smoothing (Taylor and Short, 2003). For a comprehensive review, refer to Deb et al. (2017). However, our primary objective in this study is to understand how consumption in this diverse environment is *typically* structured i.e. we wish to extract typical patterns of consumption that can well represent how energy is used across the campus. To this end, forecasting is of secondary importance to inferring structure and we decide not to remove the relatively clear seasonal (daily) patterns in consumption prior to fitting the factor model, the hope being, that these patterns will somehow be pervasive in the derived factors.

Whilst we do have metadata associated with each of these buildings for sensitivity purposes we choose to omit this in our discussions here, the buildings are presented as being approximately categorised under the following headings:

Accommodation: Student residences, and buildings primarily concerned with accommodation/student living.

Admin: Office buildings, e.g. HR, administration, and central university activities.

Misc: Other student services, e.g. cinema, shopping, sports facilities.

Mixed: Buildings which mix teaching and accommodation. For instance, seminar rooms on one floor with accommodation on another.

Services: Management buildings, porter/security offices.

Teaching: Teaching spaces like lecture theatres, seminar rooms.

4.6.2 Factor Estimates and Interpretation

To estimate factors we first choose a number of factors according to criterion (4.3.13), which leads to 4 factors being specified. Next, we apply the sparse DFM model via the EM procedure in Algorithm 2. We run the algorithm to scan across a range of α

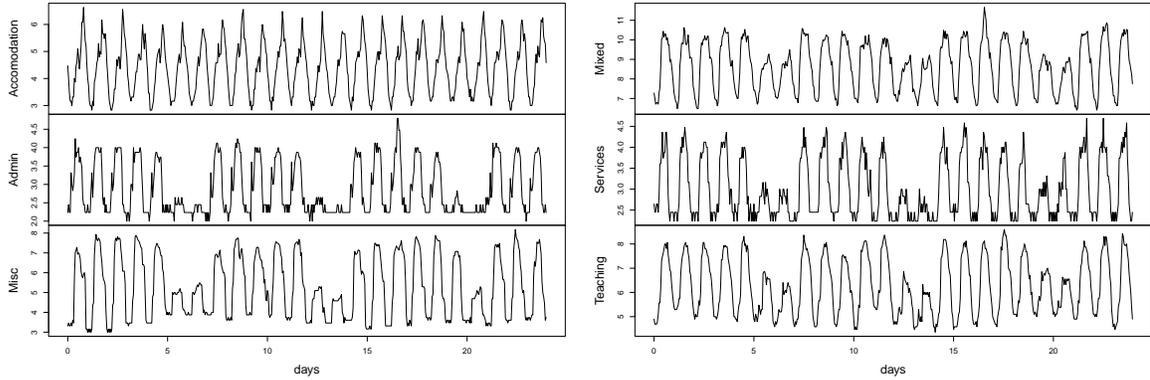


Figure 4.6.1: Example of time series readings for the 24 days under analysis. The figures present the square root of the consumption in each hour ($\sqrt{6\text{kWh}}$) for different types of building, and illustrate the diverse nature of consumption.

parameters, and in this case, the BIC criteria suggests to impose moderate sparsity corresponding to $\alpha \approx 0.01$. To give some intuition, the loadings $\hat{\mathbf{A}}$ for $\alpha = 0.01$ and $\alpha = 0$ (classic DFM) are visualised in Figure 4.6.2, a visualisation for the corresponding factors $\mathbf{a}_{t|n}$ are given in Figure 4.6.3.

For brevity, we focus on analysing the results of the sparse DFM model. Of particular interest for the energy manager is the interpretation of consumption that the sparse DFM model provides, and this is most obvious for the third and fourth factors in this case. A visualisation of the factor behaviour on a typical weekday is given in Figure 4.6.4 where there is a clear ordering in the uncertainty surrounding the factor behaviour, e.g. Factor one has small confidence intervals, whereas Factor 4 has more uncertain behaviour, especially during the working day. Interestingly, the sparse DFM only really differs from the regular DFM in these third and fourth factors, where the latter exhibits slightly greater variation in behaviour. The sparse DFM is able to isolate these further factors to specific buildings. For example, the building identified by the circle in Figure 4.6.2 is known to be active primarily throughout the night, and we see its factor loadings reflect this, e.g. the regular working day cycles for Factor 1 are not present, however, the evening and early morning features (Factors 3, and 4) are represented. For the teaching

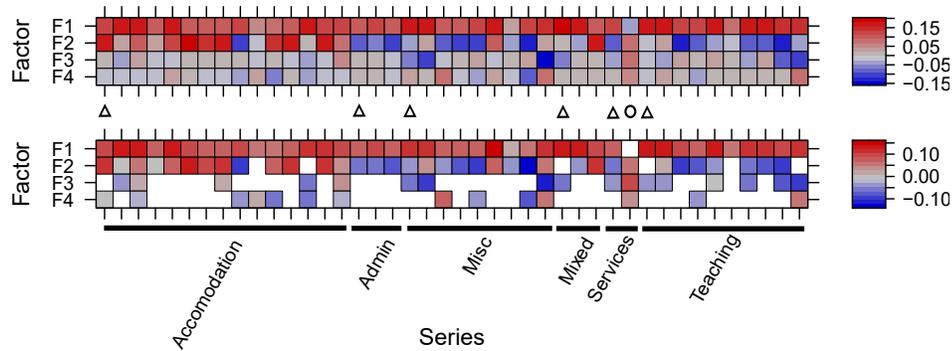


Figure 4.6.2: Estimated factor loadings for the regular DFM (top) and sparse DFM (bottom). Series are categorised according to one of six building types, triangles indicate the example series plotted in Fig 4.6.1.

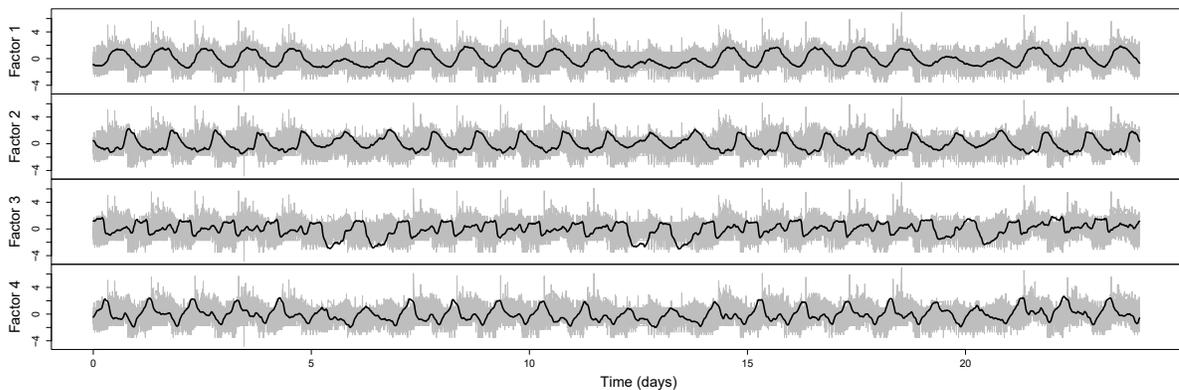


Figure 4.6.3: Estimated factors (black) with original data (grey) as a function of time using the optimal $\alpha = 0.01$ chosen according to BIC. When multiplied by the factor loadings (top) gives the estimated common component.

buildings, we see that the loading on Factor 2, and 3, are negative, indicating a sharp drop-off in energy consumption in the evening/overnight, again, this aligns with our expectations based on the usage of the facilities.

4.6.3 Forecasting Performance

The primary motivation for applying the sparse DFM in the context of this application is to aid in interpreting the consumption across campus. However, it is still of interest to examine how forecasts from the DFM compare with competitor methods. For consistency, we here provide comparison to the AR(1) and sparse VAR methods detailed earlier. These models all harness a simple autoregressive structure to model temporal

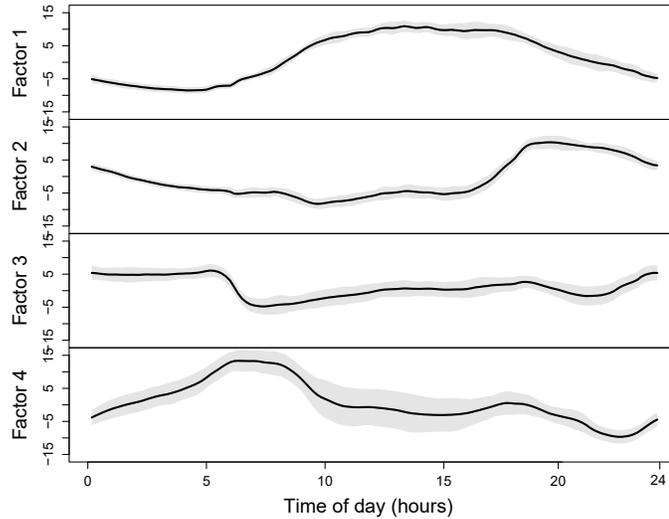


Figure 4.6.4: Average factor profile as a function of time-of-day, $t = 0$ corresponding to midnight. The solid line is a pointwise average of the factor $\hat{\mathbf{a}}_{t|n}$ across the 18 weekdays in the sample, confidence intervals are constructed as ± 1.96 the standard-deviation.

dependence, specifically regressing only onto the last set of observations (or factors), i.e. they are Markov order one. Our experiments assess performance of the models in forecasting out-of-sample data, either $h = 6$ steps ahead (1 hour), or $h = 36$ steps ahead (6 hours). The forecasts are updated in an expanding window manner, whereby the model parameters are estimated on the 24 days of data discussed previously, the forecasts are then generated sequentially based on $n + t = 1, \dots, n_{\text{test}} = 144 - h$ observations. An example of the forecasts generated (and compared to the realised consumption) is given in Figure 4.6.5. A striking feature of the DFM based model is its ability to (approximately) time the increases/decreases in consumption associated with the daily cycle. These features in the AR(1) and sparse VAR model are only highlighted after a period of h steps has passed, e.g. the models cannot anticipate the increase in consumption.

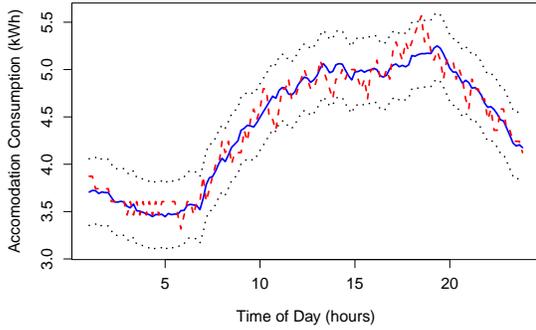
A more systematic evaluation of the forecast performance is presented in Figure 4.6.6, where the average error is calculated for each building, for each of the different models. We see that for the 1 hour ahead forecasts, all methods perform similarly, with the sparse

DFM winning marginally, and the AR(1) forecasts demonstrating more heterogeneity in the performance. There is no clear winner across all the buildings, for most (30) buildings the DFM forecasts prove most accurate, with the AR being best on 2, and the SVAR winning on the remaining 10. Moving to the 6 hour ahead forecasts, the dominance of the sparse DFM becomes clear, winning across 39 of the buildings, and the AR method winning on 3. Interestingly, the SVAR fails to win on any building, falling behind the simpler AR approach. This suggests, that in this application the activity of one building may not impact that of another across longer time-frames, however, the behaviour of the latent factors (common component) does provide predictive power.

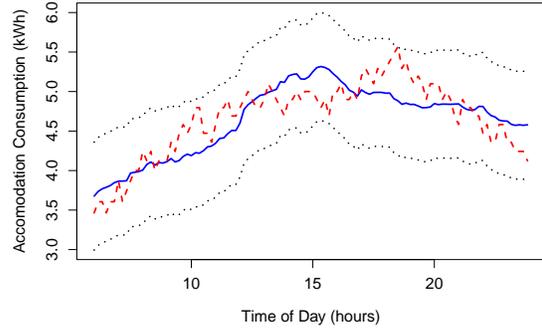
One could reasonably argue that we should not use these competitor models in this way for forecasting, e.g. we would likely look to add seasonal components corresponding to previous days/times, and/or potentially a deterministic (periodic) trend model. However, these extensions can also potentially be added to the DFM construction. Instead of absolutely providing the best forecasts possible, this case-study aims instead to highlight the differences in behaviour across the different classes of models (univariate, multivariate sparse VAR, and sparse DFM), and the fact that the sparse DFM can borrow information from across the series in a meaningful way, not only to aide interpretation of the consumption, but also to provide more accurate forecasts by harnessing the common component.

4.7 Conclusion

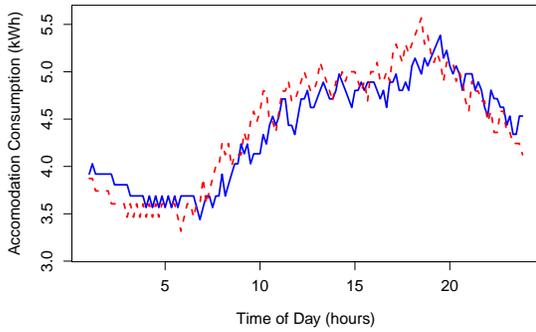
This chapter has presented a novel method for performing inference in sparse dynamic factor models via a regularised expectation maximisation algorithm. The analysis of the related QML estimator provides support for its ability to recover structure in the factor loadings, up to permutation of columns, and scaling. From the literature, it appears this is the first time the QMLE approach has been studied for the sparse DFM, with analysis



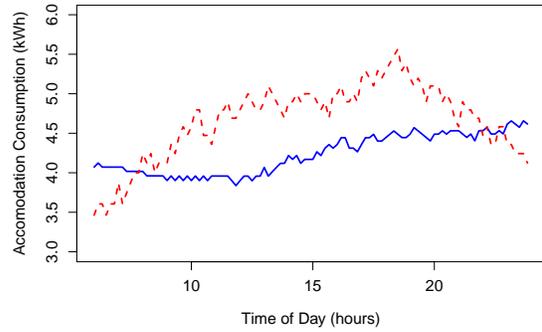
(a) 1 hour ahead forecast: Sparse DFM



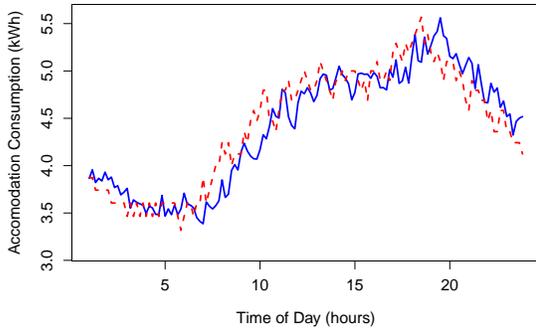
(b) 6 hour ahead forecast: Sparse DFM



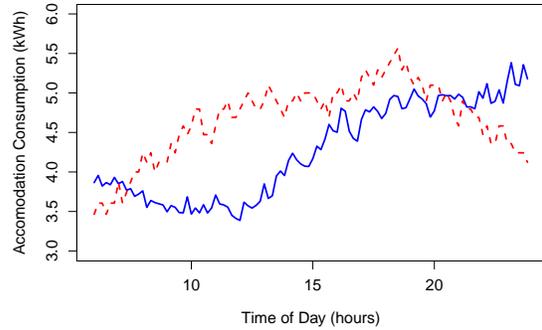
(c) 1 hour ahead forecast: AR(1)



(d) 6 hour ahead forecast: AR(1)

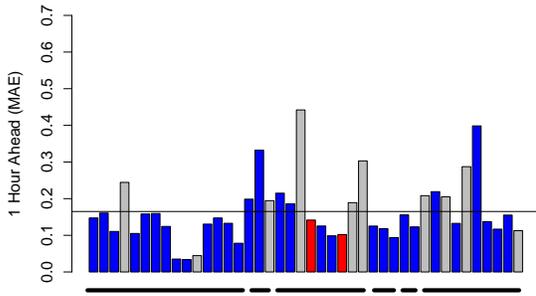


(e) 1 hour ahead forecast: Sparse VAR(1)

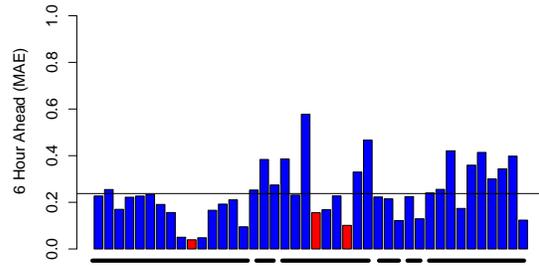


(f) 6 hour ahead forecast: Sparse VAR(1)

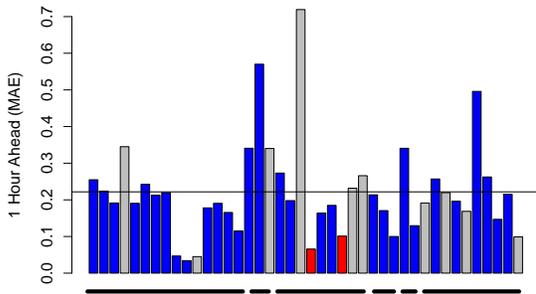
Figure 4.6.5: Example of predicted consumption ($\sqrt{\text{kWh}}$) in one (accommodation) building on the campus. Forecasts are in blue and true data in red. The left column represents 1 hour ahead forecasts based on an expanding window, whilst the right column represents 6 hour ahead forecasts. The SDFM and SVAR are tuned on the 24 days of data prior to that presented in the figure. Confidence intervals for the SDFM are based on $1.96 \times [\hat{\Lambda} P_{t|n} \hat{\Lambda}^\top + \hat{\Sigma}_\epsilon]^{1/2}$.



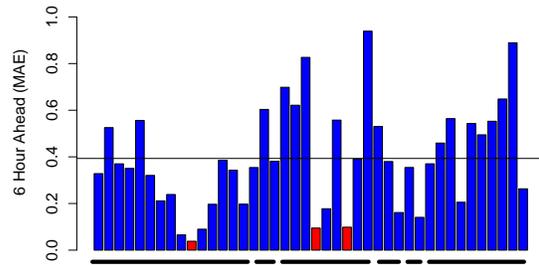
(a) 1 hour ahead error: Sparse DFM



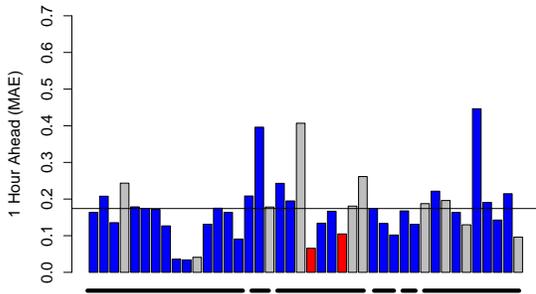
(b) 6 hour ahead error: Sparse DFM



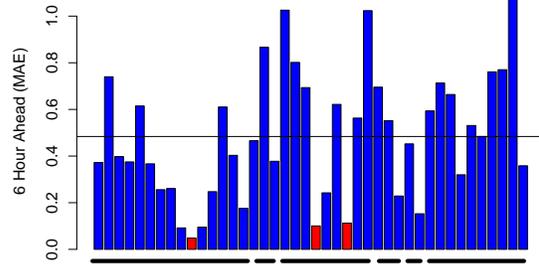
(c) 1 hour ahead error: AR(1)



(d) 6 hour ahead error: AR(1)



(e) 1 hour ahead error: Sparse VAR(1)



(f) 6 hour ahead error: Sparse VAR(1)

Figure 4.6.6: Forecast errors (MAE) for each building for (left column) 1 hour ahead forecast, and (right column) 6 hour ahead forecast. Performance evaluated on one hold out day ($144 - h$ data points). Each bar is colored according to which method performs best for that building. Blue: SDFM, red: AR(1), grey: SVAR. The solid black line indicates average performance across all buildings, the grouping of buildings is indicated via the dashed line under the plots.

extending recent investigations by Despois and Doz (2023) using more simplistic sparse PCA based approaches. When factors are thought to be dependent, e.g. modelled via a VAR(1) construction, the QMLE approach appears particularly beneficial relative to SPCA. Empirical exercises have also validated that simple BIC based hyper-parameter tuning strategies appear to be able to provide reasonable calibration of sparsity in the high-dimensional setting.

There is much further work that can be considered for the class of sparse DFMs proposed here, for example looking at developing theoretical arguments on consistency, of both factor loadings, and the factor estimates themselves. This work opted for an empirical analysis of the EM algorithm, which perhaps is more immediately useful for practitioners. Implementation of the methodology is provided in an easy-to-use R package described in Chapter 6. On a more methodological front, one could consider extending the ℓ_1 -norm regularisation strategy to look at different types of sparsity assumption, or indeed to encode other forms of prior. More detail on these two avenues of future work is provided in Chapter 7.

To conclude, adding sparsity in the DFM framework seems to be feasible in the sense that we can construct estimators that can reliably recover this structure. As shown in the synthetic experiments, the QMLE approach proposed here seems to be favourable relative to more simplistic sparse PCA approaches, especially in the setting where there is dependence between the factors and in the high-dimensional setting. The econometrics application, focused on nowcasting UK trade in exported goods, demonstrates clear advantages over a regular DFM in terms of interpreting the underlying factor structure within a complex, high-dimensional dataset. The application to modelling energy consumption shows the broad utility the DFM has, allowing us to identify distinctive profiles of buildings that qualitatively align with our intuition as well as forecasting consumption ahead of time. Overall the sparse DFM provides a useful alternative to other high-dimensional time series models, for both predictive and inferential tasks. An

additional benefit of the EM approach is its ability to readily handle arbitrary patterns of missing data, an issue often faced in the analysis of high-dimensional time series.

Chapter 5

Timely Estimates of UK Trade in Services: A Sparse Dynamic Factor Model Approach

Chapter 4 introduced a new class of Sparse Dynamic Factor Models (DFMs) using a regularised expectation maximisation algorithm that provided interpretation into the latent factor structure, the ability to handle missing data and demonstrated nowcasting accuracy. The sparse DFM was applied to a large panel of monthly UK trade in goods exports data and related indicators to produce nowcasts that avoid the 2 month publication delay goods data currently possess. This chapter extends the large monthly panel of goods data and related indicators to include unobserved monthly trade in services data corresponding to the 11 subdivisions of the service sector. As service data is only available quarterly with a 3 month publication delay, the sparse DFM is incorporated into a joint state-space framework that can perform multivariate quarterly-to-monthly temporal disaggregation of the latent service data. This joint state-space framework is then used to derive monthly time series estimates of UK service trade with the European Union (EU) and this statistic is tracked, alongside published UK-EU

monthly trade in goods, over the past 10 years using an interrupted time series model to explore possible impacts leaving the EU has had on UK trade. Section 5.1 motivates the work of this chapter and concludes with an overview of the chapter's structure.

5.1 Introduction

Trade in services is a key economic indicator for the UK economy, as the service industries account for around 80% of its GDP. While business services and financial services are the largest components, accounting for around half of UK exports, other service industries such as tourism and travel, transportation, and telecomms, contribute significantly to the economic output and generate a great deal of employment. A breakdown of the 11 main subdivisions of the service sector and their contribution to the UK economy in 2021 is presented in Figure 5.1.1. Understanding the structure and dynamics of service trade is crucial when making informed decisions about economic policies, whether it be promoting economic growth in service sectors or assessing the impact of tariffs and trade barriers during trade negotiations – an accurate and comprehensive data set on trade in services is vital.

Unlike data on trade in goods, which is published monthly by the Office for National Statistics (ONS), data on trade in services is limited to a quarterly frequency and published with a 3 month delay¹. This delay and lack of granularity for service trade data makes the task of tracking deviations in this statistic very difficult. A task which no less gained popularity in recent years as, in June 2016, the UK voted to leave the European Union (EU) - a significant destination of exports and source of imports for the UK. Following the 2016 referendum, the EU and UK began negotiations about the UK's withdrawal up until January 2020 when the EU-UK withdrawal agreement

¹The ONS do in fact publish a total trade in services series monthly representing total imports and exports to the whole world. This is constructed using ARIMA forecasts benchmarked to the quarterly measurement. However, data on the subdivisions of the service sector and data on individual countries or groups of countries (e.g. the EU) the UK trades with is only available quarterly. We compare our estimates to these published monthly total series in Section 5.5.

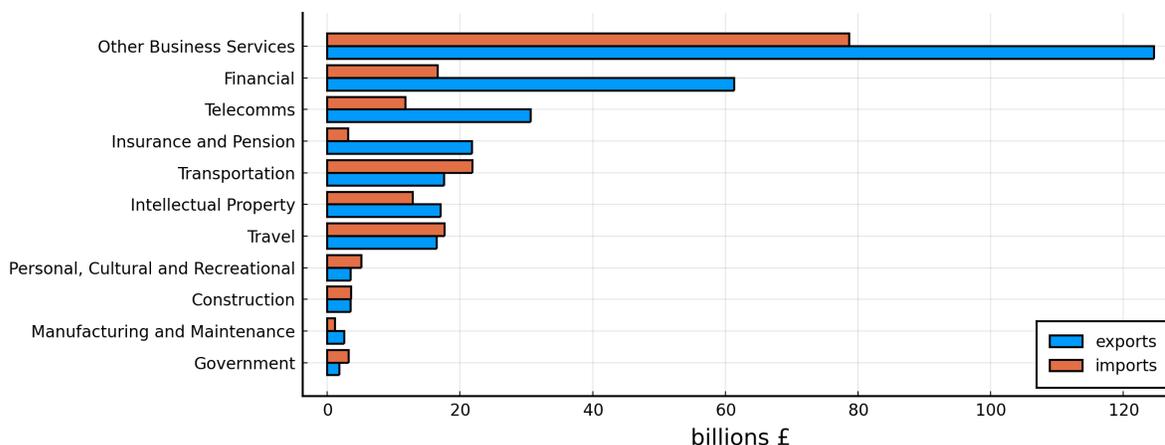


Figure 5.1.1: Contribution of the 11 subdivisions of the service sector in 2021.

came into force. This agreement set the terms of the withdrawal in accordance with Article 50 of the Treaty of the European Union. The UK then entered a transition period out of the EU which ended on the 30 December 2020 when the EU-UK Trade and Cooperation Agreement (TCA) was signed with the UK officially exiting the EU². While the TCA ensures there are no tariffs or quotas on trade in services, it does not provide the same level of access as the UK had when it was a member of the EU free market, as now service trade must adhere to regulations that vary across different countries of the EU. Figure 5.1.2 displays the quarterly time series of total imports and total exports worldwide of UK trade in services during the period of Q1 2015 to Q4 2021, along with some key dates highlighted.

The three month publication delay of service trade data means that policy makers and economists have to wait until June 2021 for information on Quarter 1 (Q1) of 2021 and thus assessing the immediate impact the UK's official transition out of the EU had on UK service trade becomes challenging. This motivates the need for timely estimates of service trade around the key date of when the TCA was signed. The work presented in this chapter attempts to achieve exactly this. Specifically, a quarterly-to-monthly multivariate temporal disaggregation of the observed quarterly service data for the 11 subdivisions of the service sector is performed in conjunction with a nowcasting

²See UK Parliament (2021) for a complete timeline of Brexit events.

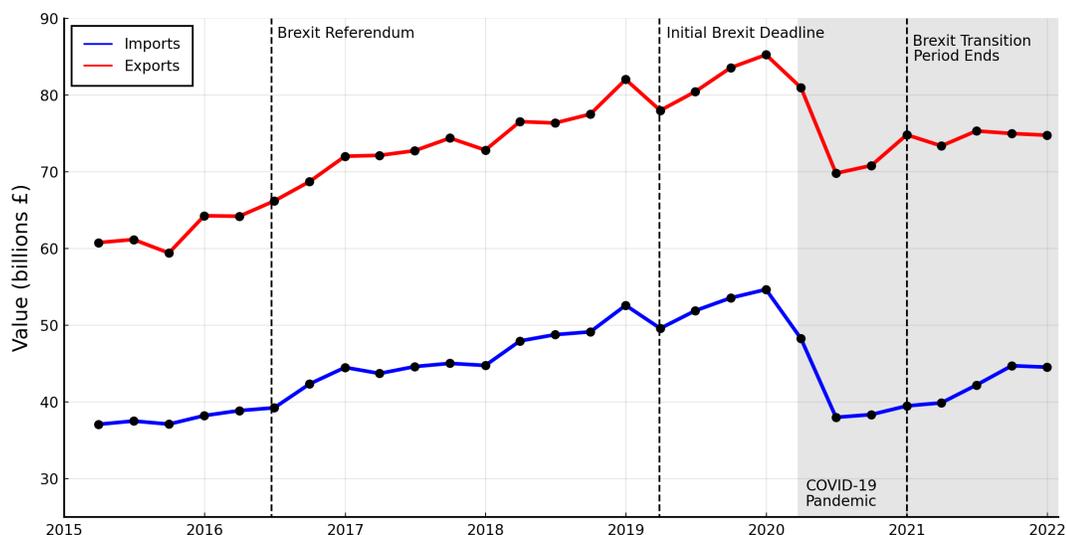


Figure 5.1.2: Quarterly time series of total imports and total exports of UK trade in services during the period Q1 2015 to Q4 2021.

exercise predicting the months in the current (unpublished) quarter. This hopes to provide monthly time series spanning the whole observed quarterly period, providing better visibility of the short-term dynamics of service trade, while also avoiding the long publication delay through the monthly out-of-sample predictions (nowcasts). The nowcasts, and disaggregate estimates, can be updated when new indicator information becomes available within the current quarter and all estimates are produced at their original current price level (as opposed to growth rates) to observe the value of service data at a given point in time.

A mixed-frequency state-space framework is constructed that jointly measures a large panel of monthly indicator series with quarterly observations of the imports and exports of the subdivisions of service trade. In particular, a sparse dynamic factor model (DFM), introduced in Chapter 4, is assumed with the unobserved monthly disaggregate series of service trade treated as latent states along with a few latent factors capturing the overall variance of the data. This way, the disaggregate targets can be directly extracted from the Kalman smoother and are ensured to be temporally consistent with the quarterly observations by augmenting a cumulator variable into the state equation,

Label	No.	Data	Description	Delay
TiS	24	Trade in Services (Quarterly)	Value of UK exports and imports of services by current price.	3
TiG	20	Trade in Goods	Value of UK exports and imports of goods by current price.	2
IoS	90	Index of Services	Movements in the volume of output for the UK services industries.	2
IoP	89	Index of Production	Movements in the volume of production for the UK production industries.	2
PPI	153	Producer Price Inflation	Changes in the prices of goods bought and sold by UK manufacturers.	1
CPI	166	Consumer Price Inflation	The rate at which the prices of goods and services bought by households rise or fall.	1
Exch	12	Exchange Rates	Sterling exchange rates with 12 popular currencies.	1
GT	17	Google Trends	Popularity scores of 17 google search queries related to trade in services.	0

Table 5.1.1: Description of data used in the model including quarterly target series and monthly indicator series. No. = number of indicators. Delay = publication delay in months.

similar to that seen in Harvey (1990), Proietti (2011a) and Frale et al. (2011). The large panel of monthly indicators considered includes most of those discussed in Case Study 1 of Chapter 4 (see Table 4.5.1) that are extended to include more relevant indicators for trade in services. Trade in goods is here used as an indicator as it is likely to be somewhat interdependent with trade in services. For example, it is common that industries tend to source goods and services from the same country and therefore if demand for a good rises this is likely to create demand for a service that supports the production and distribution of that good. Similarly, if countries impose tariffs or regulations on the import/export of goods, this may restrict service trade. The Index of Services is added as a new indicator to track overall growth of the service sector. Google trends search words are changed to be more relevant for service trade. A full description of the indicators (and targets) used are displayed in Table 5.1.1.

After the state-space model is fully defined and able to produce monthly estimates of trade in services data, it is then used to assess the potential impact leaving the EU (Brexit) has had on UK trade. This purely experimental study uses an interrupted time

series model (Linden, 2015) on 4 target series of interest to quantify potential Brexit effects on trade. These targets are the imports and exports of (estimated) monthly UK trade in services with EU countries and (published) monthly UK trade in goods with EU countries. January 2021 is used as the “intervention date” corresponding to the official Brexit date (when the TCA was signed) and the idea is to explore how the trend of observed UK trade differs to the so-called “counterfactual trend” that would have occurred if Brexit did not happen. As the intervention date overlaps with the coronavirus pandemic, it is difficult to distinguish between Brexit effects and coronavirus effects on trade and therefore a “control group” is used based on UK trade with non-EU countries to represent those not effected by Brexit. A comparison is made with an interrupted time series study using quarterly trade data to explore the advantages of working at the disaggregate (monthly) level. In the literature, numerous studies (Corcoran et al., 2015; Bernal et al., 2017; Silva et al., 2020) have looked to implement interrupted time series models within the domains of economics and health. However, to our knowledge, this study is the first to employ temporally disaggregated data to the model, particularly in the assessment of Brexit’s impact on UK trade.

The rest of the chapter is structured as follows. Section 5.2 sets up the problem we face before fully defining the joint state-space framework in Section 5.3 that augments the sparse DFM (Chapter 4) with the latent monthly targets of service trade. Section 5.4 compares the proposed state-space framework approach with Chow-Lin temporal disaggregation (Chow and Lin, 1971) and sparse temporal disaggregation (Chapter 3) in a simple simulation study to assess temporal disaggregation ability. Section 5.5 then applies this model to the entire data to estimate latent monthly series for the subdivisions of the service sector and conducts a pseudo-real time nowcasting exercise to evaluate accuracy. Section 5.6 concludes with an experimental study assessing the possible impact Brexit has had on UK trade using an interrupted time series analysis. Section 5.7 closes with a discussion.

5.2 Problem Set Up

Let Y_{it} denote the i^{th} unobservable monthly trade in services subdivision at time t , for $i = 1, \dots, p_y$, $t = 1, \dots, n$. In this application $p_y = 24$ corresponding to both imports and exports of total services plus 11 subdivisions of the service sector we are interested in. The task at hand is to make accurate estimates of Y_{it} for all $t = 1, \dots, n$ by making use of all relevant information we have available to us in month n , defined as Ω_n . This information set will consist of the observed quarterly trade in services data $\bar{Y}_{i\tau}$ for quarters $\tau = 1, \dots, n_l$ and monthly indicator series X_{jt} for $j = 1, \dots, p_x$ where $p_x = 547$. The available information in month n will depend on the publication delays of each variable and will therefore contain a ragged edge structure at the end of the sample. In month n , the available information set will be

$$\Omega_n = \{(\bar{Y}_{i\tau}, X_{jt}) : i = 1, \dots, p_y, j = 1, \dots, p_x, \tau = 1, \dots, \lfloor (n-3)/3 \rfloor, t = 1, \dots, n-l_j\},$$

where $\lfloor \cdot \rfloor$ denotes integer division. This takes into account the 3 month publication delay for the quarterly service data and the different publication delays for the indicators, where l_j is the delay (in months) of indicator $j \in \{1, \dots, p_x\}$.

Supposing we are in month n , we aim to estimate monthly trade in services up to n based on this available information set and hence our target can be denoted by the conditional expectation

$$\hat{Y}_{it} = \mathbb{E}[Y_{it} | \Omega_n],$$

for $i = 1, \dots, p_y$ and $t = 1, \dots, n$. This involves a temporal disaggregation problem up to the last available quarter $\tau = \lfloor (n-3)/3 \rfloor$, which obtains estimates \hat{Y}_{it} for $t = 1, \dots, 3 \times \lfloor (n-3)/3 \rfloor$ that are temporally consistent with the observed quarterly figures (i.e. the sum of three estimated months in a quarter sum to that quarters figure). It then involves a ‘nowcasting exercise’ for the months $t = 3 \times \lfloor (n-3)/3 \rfloor + 1, \dots, n$, using

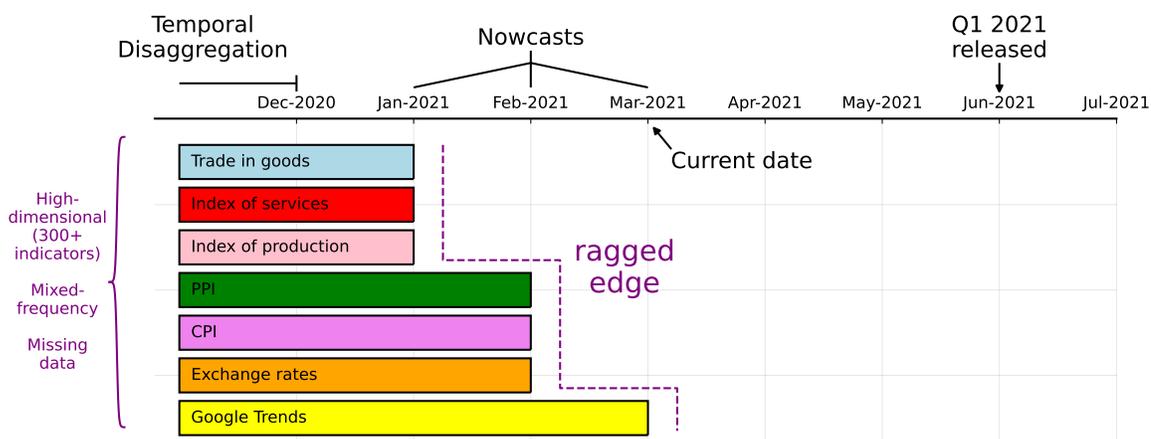


Figure 5.2.1: Overview of what our problem would look like if we are currently in March 2021. Release of Q1 Trade in Services data not available until June. Our model is able to produce monthly nowcasts in Q1 by using the monthly indicator information available - grouped into data category colours. It also produces a temporally consistent disaggregate series up to December 2020.

the available information in these last few months. Figure 5.2.1 is a visual interpretation of the task we face if we were currently in March 2021. In this month the Q4 2020 figure has been released and we can make a temporal disaggregation up to December 2020 and nowcast the months January to March 2021. In these last three months we have a ragged structure for the indicators and want to exploit the available information to make accurate nowcasts. We are able to continue this nowcasting process in April and May until the actual release of Q1 2021 data in June, with each month providing us with more information from the monthly indicator releases. This allows us to make early interpretations for Q1 of trade in services avoiding the long wait till June, while simultaneously providing us with a back series at a more timely frequency than we currently have. The left-side of the figure highlights three data challenges we face in this task that our model must be able to handle. These general nowcasting challenges were also discussed in Chapter 2.2.

5.3 Joint state-space Framework

Following the trade in goods application of Case Study 1 in Chapter 4, a sparse dynamic factor model (DFM) is assumed for the large panel of monthly indicator series given by:

$$\mathbf{X}_t = \mathbf{\Lambda}\mathbf{F}_t + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \text{N}(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon), \quad (5.3.1)$$

$$\mathbf{F}_t = \mathbf{A}\mathbf{F}_{t-1} + \mathbf{u}_t, \quad \mathbf{u}_t \sim \text{N}(\mathbf{0}, \boldsymbol{\Sigma}_u), \quad (5.3.2)$$

where \mathbf{X}_t is first made stationary by taking differences where appropriate and standardising the indicator series prior to modelling. The induced sparsity in $\mathbf{\Lambda}$ hopes to better explain the role indicator groups play in the dynamics of latent factors, possibly allowing us to make useful statements like: ‘this factor estimate is largely associated with data coming from trade in goods’ or ‘this factor consists of variables that were strongly impacted by Brexit’.

The trade in services data series exhibits stationarity after applying first-differences through an Augmented Dickey-Fuller test (Cheung and Lai, 1995). As a result, we define $\mathbf{y}_t = \Delta\mathbf{Y}_t$ and $\bar{\mathbf{y}}_\tau = \Delta\bar{\mathbf{Y}}_\tau$ as the stationary series utilized for the DFM. Here, Δ represents the first-difference operator, and \mathbf{Y}_t and $\bar{\mathbf{Y}}_\tau$ encompass all targets $i = 1, \dots, p_y$. It is worth noting that the model can be expanded to accommodate various stationary transformations, such as the log-difference transformation within a non-linear aggregation framework, as discussed in Proietti (2011a). To incorporate the latent trade in services monthly target series, \mathbf{Y}_t , into a joint state-space framework it is assumed their stationary counterpart, \mathbf{y}_t , when standardised, $\mathbf{y}_t^* = (\mathbf{y}_t - \boldsymbol{\mu})/\mathbf{D}$, admits the same factor model representation as (5.3.1)-(5.3.2):

$$\mathbf{y}_t^* = \mathbf{\Lambda}_y\mathbf{F}_t + \boldsymbol{\epsilon}_t^{(y)}, \quad \boldsymbol{\epsilon}_t^{(y)} \sim \text{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}^{(y)}}), \quad (5.3.3)$$

where $\boldsymbol{\mu}$ is the $p_y \times 1$ vector of means and \mathbf{D} is the $p_y \times p_y$ diagonal matrix of standard

deviations of \mathbf{y}_t . The parameter $\mathbf{\Lambda}_y$ explains how the targets are loaded onto the factors and $\boldsymbol{\epsilon}_t^{(y)}$ is the idiosyncratic noise with covariance $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}^{(y)}}$. The idiosyncratic error processes $\{\boldsymbol{\epsilon}_t\}$ and $\{\boldsymbol{\epsilon}_t^{(y)}\}$ are both assumed to be IID white noise, i.e. $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}^{(y)}}$ are both diagonal matrices. A straightforward extension to model these error processes as AR(1) can be conducted following methodology laid out in Appendix D.1, however, this approach will increase the number of states in the model by a considerable amount. Alternative approaches possibly following methodology laid out in Jungbacker et al. (2009) are left as future work and discussed in Chapter 7.

For temporal disaggregation purposes we require the latent monthly estimates, Y_{it} , in each quarter to be temporally consistent with the observed quarterly values, $\bar{Y}_{i\tau}$. Define \tilde{Y}_{it} for $t = 1, \dots, n$, to be the partially observed monthly series which has the quarterly observations of $\bar{Y}_{i\tau}$ assigned to every 3rd month and is missing everywhere else, i.e. $\tilde{Y}_{it} = \bar{Y}_{i\tau}$ for $t = 3\tau$, $\tau = 1, \dots, n_l$, otherwise unobserved. The aggregation constraint that must be satisfied is:

$$\tilde{Y}_{i,t} = \begin{cases} Y_{i,t} + Y_{i,t-1} + Y_{i,t-2}; & \text{for } t = 3, 6, \dots, \lfloor (n-3)/3 \rfloor, \\ \text{unobserved}; & \text{otherwise.} \end{cases}$$

Following Harvey (1990), a temporal aggregation constraint of this form can be incorporated into a state-space framework by constructing a cumulator:

$$\tilde{Y}_{i,t}^c = \rho_t \tilde{Y}_{i,t-1}^c + Y_{i,t} \quad (5.3.4)$$

which cumulates the additive aggregation at each quarter by using the cumulator coefficient:

$$\rho_t = \begin{cases} 0 & t = 3\tau - 2, \tau = 1, \dots, n_l, \\ 1 & \text{otherwise,} \end{cases}$$

which is equal to 0 in the first month of a quarter and 1 otherwise. This leads to a

cumulative sum following:

$$\begin{aligned}
\tilde{Y}_{i,1}^c &= Y_{i,1}, \\
\tilde{Y}_{i,2}^c &= Y_{i,1} + Y_{i,2}, \\
\tilde{Y}_{i,3}^c &= Y_{i,1} + Y_{i,2} + Y_{i,3}, \\
\tilde{Y}_{i,4}^c &= Y_{i,4}, \\
\tilde{Y}_{i,5}^c &= Y_{i,4} + Y_{i,5}, \\
&\vdots
\end{aligned}$$

where the measurement $\tilde{Y}_{i,t}$ is equal to the cumulator $\tilde{Y}_{i,t}^c$ at every $t = 3\tau$ for $\tau = 1, \dots, n_l$.

With the equations (5.3.1), (5.3.2), (5.3.3) and (5.3.4) fully defined, it is possible to bring them together in a single state-space framework. The state-space measurements will be $\mathbf{Z}_t = (\mathbf{X}_t \tilde{\mathbf{Y}}_t)^\top \in \mathbb{R}^{p \times 1}$ where $p = p_x + p_y$. The state-space state vector is defined as $\boldsymbol{\alpha}_t = (\mathbf{F}_t \mathbf{Y}_t \tilde{\mathbf{Y}}_t^c)^\top \in \mathbb{R}^{(p+p_y) \times 1}$ where the target is to estimate \mathbf{Y}_t for $t = 1, \dots, n$. To write the state-space model in traditional form, (5.3.3) is re-written as:

$$\begin{aligned}
\mathbf{Y}_t &= \mathbf{Y}_{t-1} + \boldsymbol{\mu} + \mathbf{D}\mathbf{y}_t^* \\
&= \mathbf{Y}_{t-1} + \boldsymbol{\mu} + \mathbf{D}(\boldsymbol{\Lambda}_y \mathbf{F}_t + \boldsymbol{\epsilon}_t^{(y)}) \\
&= \mathbf{Y}_{t-1} + \boldsymbol{\mu} + \mathbf{D}\boldsymbol{\Lambda}_y(\mathbf{A}\mathbf{F}_{t-1} + \mathbf{u}_t) + \mathbf{D}\boldsymbol{\epsilon}_t^{(y)} \\
&= \mathbf{Y}_{t-1} + \boldsymbol{\mu} + \mathbf{D}\boldsymbol{\Lambda}_y \mathbf{A}\mathbf{F}_{t-1} + \mathbf{D}\boldsymbol{\Lambda}_y \mathbf{u}_t + \mathbf{D}\boldsymbol{\epsilon}_t^{(y)},
\end{aligned}$$

since $\mathbf{y}_t^* = ((\mathbf{Y}_t - \mathbf{Y}_{t-1}) - \boldsymbol{\mu})/\mathbf{D}$. This is then used to re-write (5.3.4) as:

$$\tilde{\mathbf{Y}}_t^c = \rho_t \tilde{\mathbf{Y}}_{t-1}^c + \mathbf{Y}_{t-1} + \boldsymbol{\mu} + \mathbf{D}\boldsymbol{\Lambda}_y \mathbf{A}\mathbf{F}_{t-1} + \mathbf{D}\boldsymbol{\Lambda}_y \mathbf{u}_t + \mathbf{D}\boldsymbol{\epsilon}_t^{(y)}.$$

This allows the following state-space model to be written:

$$\begin{bmatrix} \mathbf{X}_t \\ \tilde{\mathbf{Y}}_t \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Lambda} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{p_y} \end{bmatrix} \begin{bmatrix} \mathbf{F}_t \\ \mathbf{Y}_t \\ \mathbf{Y}_t^c \end{bmatrix} + \begin{bmatrix} \boldsymbol{\epsilon}_t \\ \mathbf{0} \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{F}_t \\ \mathbf{Y}_t \\ \mathbf{Y}_t^c \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} & \mathbf{0} \\ D\boldsymbol{\Lambda}_y \mathbf{A} & \mathbf{I}_{p_y} & \mathbf{0} \\ D\boldsymbol{\Lambda}_y \mathbf{A} & \mathbf{I}_{p_y} & \rho_t \mathbf{I}_{p_y} \end{bmatrix} \begin{bmatrix} \mathbf{F}_{t-1} \\ \mathbf{Y}_{t-1} \\ \mathbf{Y}_{t-1}^c \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\mu} \\ \boldsymbol{\mu} \end{bmatrix} + \begin{bmatrix} \mathbf{I}_r & \mathbf{0} \\ D\boldsymbol{\Lambda}_y & D \\ D\boldsymbol{\Lambda}_y & D \end{bmatrix} \begin{bmatrix} \mathbf{u}_t \\ \boldsymbol{\epsilon}_t^{(y)} \end{bmatrix},$$

which is defined by the notation:

$$\mathbf{Z}_t = \mathbf{G}\boldsymbol{\alpha}_t + \mathbf{e}_t \quad (5.3.5)$$

$$\boldsymbol{\alpha}_t = \mathbf{H}_t \boldsymbol{\alpha}_{t-1} + \mathbf{c} + \mathbf{R}\boldsymbol{\eta}_t. \quad (5.3.6)$$

Note that the state transition matrix, \mathbf{H}_t , is time dependent as it depends on the cumulator coefficient ρ_t . The measurement error, \mathbf{e}_t , is assumed to be mean-zero with covariance $\boldsymbol{\Sigma}_e \in \mathbb{R}^{p \times p}$ that is a block-diagonal matrix with $\boldsymbol{\Sigma}_e$ in the first block and $\mathbf{0}_{p_y}$ in the second. The state noise, $\boldsymbol{\eta}_t$, is again mean-zero with block-diagonal covariance $\boldsymbol{\Sigma}_u \in \mathbb{R}^{(r+p_y) \times (r+p_y)}$ with $\boldsymbol{\Sigma}_u$ and $\boldsymbol{\Sigma}_{\epsilon^{(y)}}$ in each block.

5.3.1 Initialisation

Before the mixed-frequency state-space model from equations (5.3.5)-(5.3.6) is estimated using a regularised expectation maximisation (EM) algorithm, the model parameters $\boldsymbol{\theta} = (\boldsymbol{\Lambda}, \boldsymbol{\Lambda}_y, \mathbf{A}, \boldsymbol{\Sigma}_e, \boldsymbol{\Sigma}_u, \boldsymbol{\Sigma}_{\epsilon^{(y)}})$ and the distribution of the initial state vector given by $\boldsymbol{\alpha}_0 = (\mathbf{F}_0, \mathbf{Y}_0, \mathbf{Y}_0^c)^\top$ have to be estimated. The number of factors to use is chosen a priori by minimising Bai and Ng (2002) information criteria:

$$IC(r) = \log V_r(\bar{\mathbf{F}}, \bar{\boldsymbol{\Lambda}}) + r \left(\frac{n + p_x}{np_x} \right) \log \min(n, p_x), \quad (5.3.7)$$

where

$$V_r(\bar{\mathbf{F}}, \bar{\mathbf{\Lambda}}) = \frac{1}{np_x} \sum_{i=1}^{p_x} \sum_{t=1}^n \mathbb{E} [\bar{\boldsymbol{\epsilon}}_{i,t}^2]$$

and $\bar{\boldsymbol{\epsilon}}_{i,t} = X_{t,i} - \bar{\mathbf{\Lambda}}_{i,\cdot} \bar{\mathbf{F}}_t$ is found using PCA when applied to the balanced standardised indicator data set \mathbf{X} , with preliminary factors $\bar{\mathbf{F}}$ corresponding to the principal components, and the estimated loadings $\bar{\mathbf{\Lambda}}$ corresponding to the eigenvectors. The indicator matrix is balanced by interpolating missing data with cubic splines and a simple moving average.

An initial estimation of each latent series, Y_{it} , for $i = 1, \dots, p_y$, is computed by using the sparse temporal disaggregation (spTD) method of Chapter 3 applied to each quarterly series, $\bar{Y}_{i,\tau}$ and monthly indicators \mathbf{X}_t separately. Missing data within \mathbf{X}_t is balanced in the same way as above and the raw, undifferenced data is used assuming a cointegrating relationship between \mathbf{Y}_t and \mathbf{X}_t as discussed in Chapter 3. These preliminary estimates of \mathbf{Y}_t are then first-differenced and standardised to obtain preliminary \mathbf{y}_t^* series which are then used to construct \mathbf{Z}_t with which a sparse DFM model is fit with r factors following Algorithm 2 in Chapter 4. This initialisation approach allows all model parameters in (5.3.5)-(5.3.6) to be found, including the ℓ_1 -norm regularisation parameter for sparse factor loadings. Note, sparsity is not induced on the target series \mathbf{y}_t^* in \mathbf{Z}_t . To prioritise a comprehensive understanding of the problem at hand, we deliberately kept the tuned ℓ_1 -norm regularisation parameter fixed for the regularised EM algorithm discussed in the subsequent section.

The initial factor mean $\mathbb{E}[\mathbf{F}_0 | \boldsymbol{\Omega}_n]$ is set to zero and the covariance $\text{Cov}(\mathbf{F}_0 | \boldsymbol{\Omega}_n)$ is set to the $r \times r$ matrix of the stacked vector $(\mathbf{I}_{rp} - \mathbf{A} \otimes \mathbf{A})^{-1} \text{Vec}(\boldsymbol{\Sigma}_u)$. As the state variables \mathbf{Y}_t and \mathbf{Y}_t^c are non-stationary, their conditional mean and covariance are initialised using an *approximate diffuse* initialisation (Brave et al., 2022), i.e. zero-mean and a very large covariance matrix set to 10^7 on the diagonal. Although this is not as statistically sound as using a more complex exact treatment of diffuse states (De Jong, 1991), experiments show the approximation performs well. A future work direction is to

better explore the importance of an exact diffuse initialisation.

5.3.2 Estimation and Convergence

The measurement equation (5.3.5) along with state equation (5.3.6) can be estimated using a regularised EM algorithm as in Chapter 4. The E-step, which involves computing the conditional expectation and covariance of the state vector, can be done using the univariate Kalman filter and smoother equations (Koopman and Durbin, 2000) given in Appendix A.2. When variables in \mathbf{Z}_t are missing, as will be the case in the ragged edge for the indicators \mathbf{X}_t (or just general patterns of missing data) and in $\tilde{\mathbf{Y}}_t$ when $t \neq 3\tau$, the Kalman gain term for these variables are omitted from the update calculations. Similarly, the value of ρ_t in the matrix \mathbf{H}_t is adjusted depending on t in the filter and smoother equations, equal to 0 when t is the first month of a quarter and 1 otherwise. The M-step calculations for the sparse DFM assumed for \mathbf{X}_t are the same as those derived in Chapter 4 with the $\mathbf{\Lambda}$ parameter being updated using an ADMM algorithm for regularisation using the optimal ℓ_1 -norm penalty parameter found in the initial sparse DFM fit. The parameters $\mathbf{\Lambda}_y$ and $\mathbf{\Sigma}_{\epsilon(y)}$ can be obtained similarly within the EM framework, corresponding to the least-squares estimators of the conditional expectation of \mathbf{Y}_t from the Kalman filter and smoother equations. The E and M steps are repeated until convergence which is determined when the maximum absolute difference between the current and previous iteration of each target series becomes smaller than 10^4 .

5.4 Simulation Study

To investigate the estimation ability of the proposed joint state-space model, a simple simulation study is conducted comparing this model to classic Chow-Lin temporal disaggregation (Chow and Lin, 1971) and sparse temporal disaggregation (Chapter 3). A matrix of $p_x = 30$ indicators with $n = 120$ observations are simulated according to a

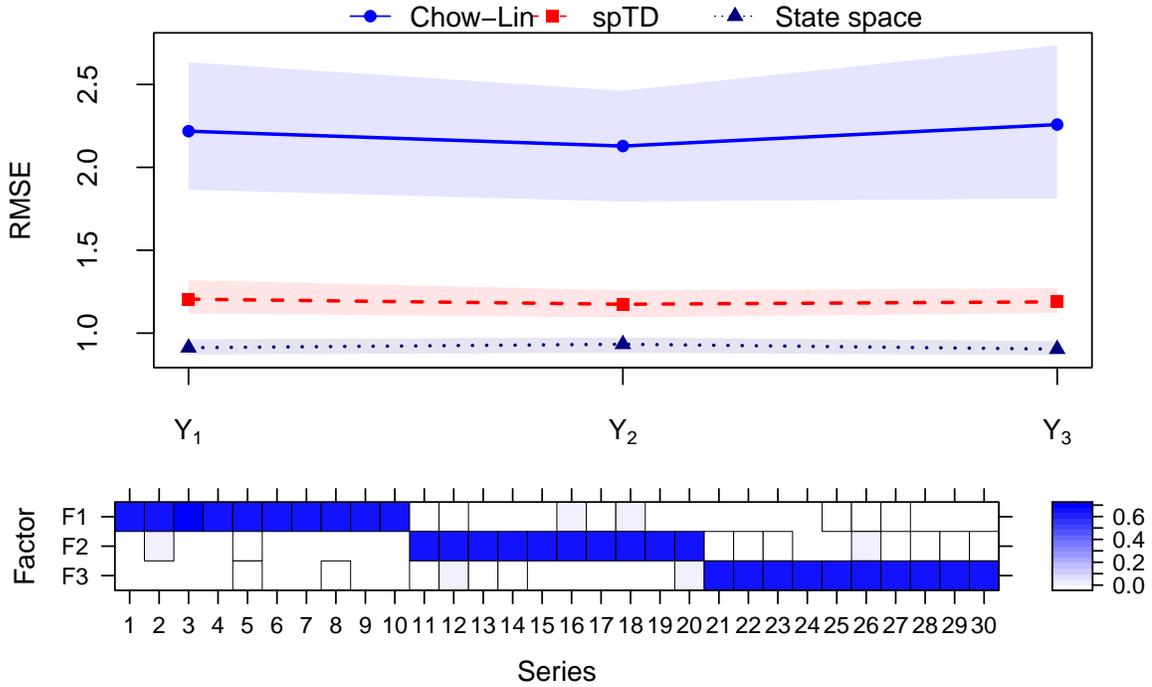


Figure 5.4.1: (Top) Average RMSE scores over 100 iterations for recovery of true disaggregate series Y_1 , Y_2 and Y_3 , with 25% and 75% confidence bands. (Bottom) Average loadings matrix estimate over 100 iterations for the state-space sparse DFM approach.

sparse DFM (5.3.1)-(5.3.2) with $r = 3$ factors, sparse loading matrix $\Lambda = \mathbf{I}_r \otimes \mathbf{1}_{10}$, factor transition matrix $\mathbf{A} = 0.8 \times \mathbf{I}_r$, and covariance matrices $\Sigma_\epsilon = \mathbf{I}_{p_x}$ and $\Sigma_u = (1 - 0.8^2) \times \mathbf{I}_r$. Next, $p_y = 3$ target disaggregate series are simulated using the 3 factors with coefficient Λ_y set to be the $p_y \times r$ matrix of ones in the regression (5.3.3) with $\Sigma_{\epsilon(y)} = \mathbf{I}_{p_y}$. The disaggregate target series are thought of as monthly series and are temporally aggregated to a quarterly frequency to obtain the observed low-frequency series \bar{Y}_τ , $\tau = 1, \dots, n_l$ with $n_l = 40$. A ragged edge structure is then created in the indicator matrix such that the first 10 variables have the last 2 data points missing, the next 10 have 1 missing and the final 10 have none missing. The last data point in \bar{Y} is also treated as missing.

This simulation set up is similar to the one we face in the trade in services application and other general econometric problems. While it is expected that the benchmark

temporal disaggregation methods will perform well in this small simplistic set up, they are not as practical as the joint state-space framework for this task as any missing data has to be first forecasted before fitting the models and as they are univariate methods, estimation of each disaggregate target has to be computed separately. The 3 models, Chow-Lin, spTD and state-space, are applied to 100 versions of the simulated data and the root mean squared error (RMSE) metric³ between the true disaggregate series, \mathbf{Y}_t , and the estimates, $\hat{\mathbf{Y}}_t$, are calculated. The top image of Figure 5.4.1 displays the average RMSE scores of all 3 models on all 3 target series (along the x-axis) along with the 25% and 75% confidence bands. The results reveal that the joint state-space model is performing the best, obtaining the most accurate temporal disaggregation estimates and having the narrowest confidence intervals. Chow-Lin is performing the worst as expected since $p_x = 30$. Sparse temporal disaggregation also performs well and uses the adaptive LASSO penalty to deal with correlated indicators induced by the VAR(1) factor structure. To test whether the true (sparse) loadings parameter is recovered accurately in the joint state-space framework, the average loading matrix estimate over 100 iterations is plotted in the bottom image of Figure 5.4.1. The true sparse structure of $\mathbf{\Lambda} = \mathbf{I}_r \otimes \mathbf{1}_{10}$ appears to be recovered well, with the largest average loading that should be sparse being 0.0065. In summary, the joint state-space model appears to perform temporal disaggregation accurately while at the same time exhibit the desirable features the sparse DFM possessed.

5.5 Monthly Estimation of the Subdivisions of the Service Sector

We now apply the state-space model (5.3.5)-(5.3.6) with regularised EM estimation discussed in Section 5.4 to the entire dataset (Table 5.1.1) to estimate monthly disag-

³Both the MAE and RMSE metric were calculated and led to the same conclusions.

gregate series for the 24 target series of interest representing imports and exports of total trade in services and the 11 subdivisions of the service sector. All data, except Google trends, is obtained from the ONS website using the December 2022 vintage and we take data going back to January 2004. This leads to observed quarterly trade in services data from Q1 2004 to Q3 2022 (a total of 75 quarters) and fully observed monthly indicator series from January 2004 to September 2022 (a total of 225 months). Following the initialisation outlined in 5.3.1, the number of factors chosen by the Bai and Ng (2002) information criteria (5.3.7) is chosen to be $r = 4$ factors. Using 4 factors, an initial sparse DFM is fitted to the first-differenced and balanced dataset to obtain preliminary parameter estimates and tuned over a large log-spaced grid of ℓ_1 -norm penalty parameters from 10^{-3} to 10^4 using BIC (4.3.14). This results in an optimal tuning parameter $\alpha^* \approx 2$ which is very similar to the optimal parameter found in the trade in goods case study of Chapter 4. This is expected since a large majority of the data used is the same in this study and hence we may also expect a similar factor structure capturing the overall variance in the data.

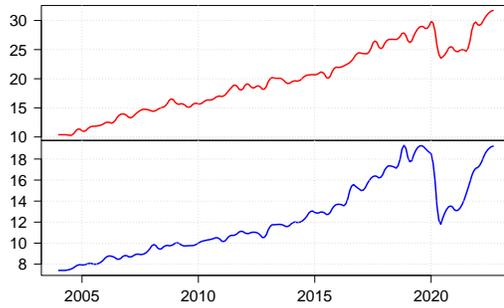
Using the preliminary parameter estimates and a fixed ℓ_1 -norm penalty parameter $\alpha^* \approx 2$, the regularised EM algorithm of Section 5.3.2 is applied to the data to obtain monthly estimates of trade in services from January 2004 to September 2022. These estimates are temporally consistent with the observed quarterly data in that 3 monthly estimates in a quarter sum exactly to that corresponding observed quarter. Figure 5.5.1 (a)-(1) displays these estimated series with exports on top in red and imports in blue on the bottom⁴. The series are ordered in terms of value with the y-axis representing £billion. Figure 5.5.2 compares our monthly estimates of total trade in services with those published by the ONS. We appear to be obtaining very similar smooth-looking results to the ONS monthly estimates with the only real difference due to the coronavirus shock in 2020-2021 as shown in the right plot displaying percentage difference between

⁴All code and data results are available upon request.

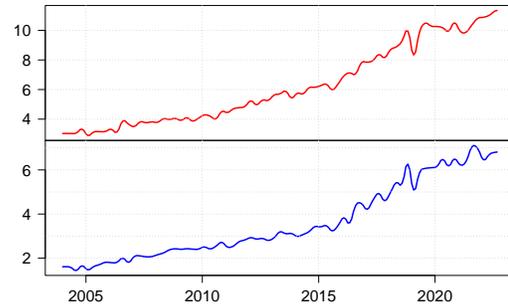
our estimates \mathbf{y}^{est} and the published figures \mathbf{y}^{pub} , i.e. $(\mathbf{y}^{\text{pub}} - \mathbf{y}^{\text{est}})/\mathbf{y}^{\text{est}} \times 100\%$. The smoothness of our series estimates is due to the estimated factor loadings for the first-differenced targets, i.e. $\mathbf{\Lambda}_y$, being small and thus not being influenced by the factors by a great deal. The results appear to look similar to that of a Denton smoothed estimate (Denton, 1971) that minimises first-difference change over time as described in Section 2.1.3 of Chapter 2. This is usually a favoured approach by National Statistical institutes when the consistency of smoothed estimates is a preference. For less-smooth estimates, extensions into alternative error processes for (5.3.3) can be conducted.

Figures 5.5.3 and 5.5.4 respectively display the factor loadings and factor series estimated in the joint state-space model (5.3.5)-(5.3.6). The factor structure and dynamics look very similar to the results obtained in the trade in goods case study (Sect. 4.5) with Factor 1 only weighted with inflation measures (both producer and consumer) and Factor 2 heavily weighted with the IoP, IoS and also TIG, capturing the overall state of the economy and the coronavirus shock in 2020. Factors 3 and 4 seem to be a mixture of inflation data and Google trends. On each figure we display the estimated coefficients for the monthly target series, i.e. $\mathbf{\Lambda}_y$, to show how the targets are loaded onto each factor.

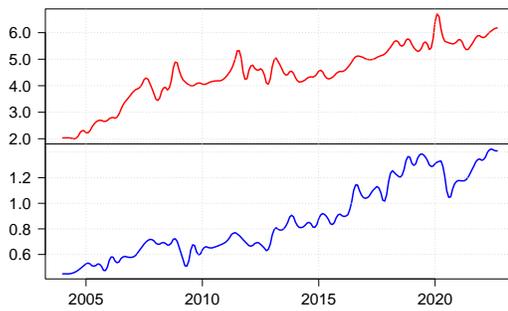
We now conduct a pseudo-real time nowcasting exercise to assess how accurately the joint state-space model is able to nowcast the current quarter using available information in the ragged edge. The nowcasting exercise is described as pseudo-real time as we use a fixed vintage of data (December 2020) and manually construct the ragged edge described by the indicator lags found in Table 5.1.1 at each nowcasting window. This nowcasting window is expanding every month from September 2018 to August 2022 (48 periods) and in each month a nowcast of the current unpublished quarter is calculated. That is, the first nowcasting window uses data from January 2004 to September 2018, where we wish to nowcast Q3 2018 using the sum of disaggregate estimates: July, August and September 2018. The window is then expanded to October 2018 where the Q3



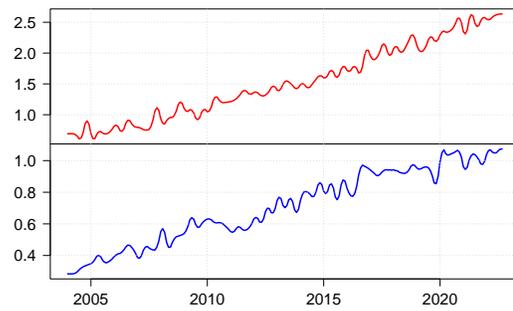
(a) Total Services



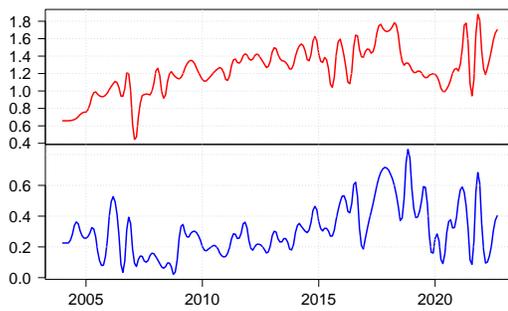
(b) Other Business Services



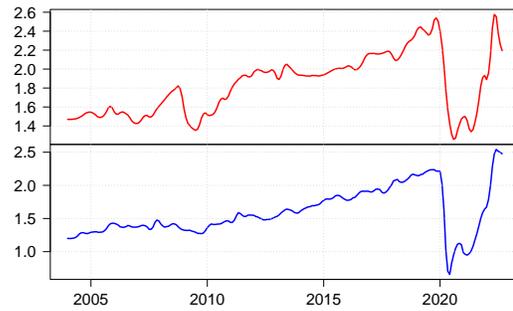
(c) Financial Services



(d) Telecomms

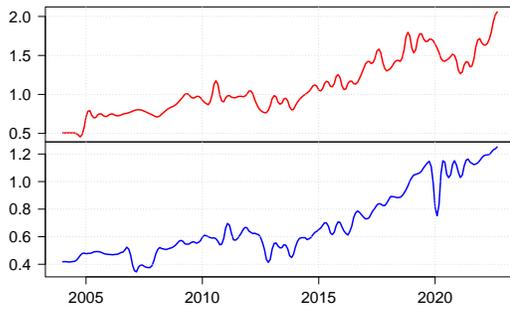


(e) Insurance and Pension Services

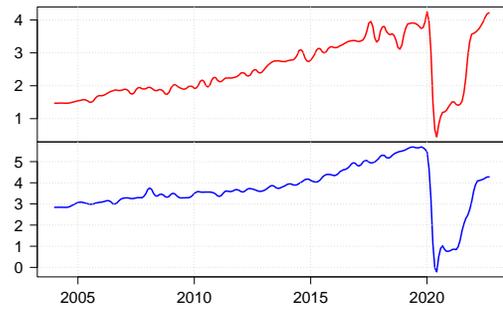


(f) Transport Services

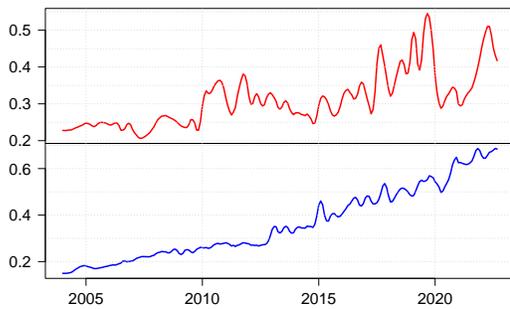
Figure 5.5.1: Estimated monthly disaggregate trade series of the subdivisions of the service sector (£billion). Exports trade in red on top and imports trade in blue at the bottom.



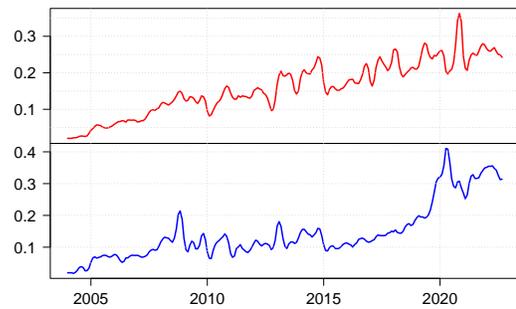
(g) Intellectual Property Services



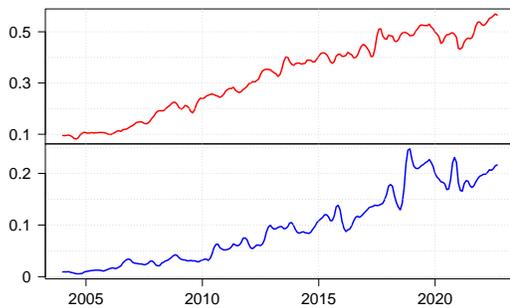
(h) Travel Services



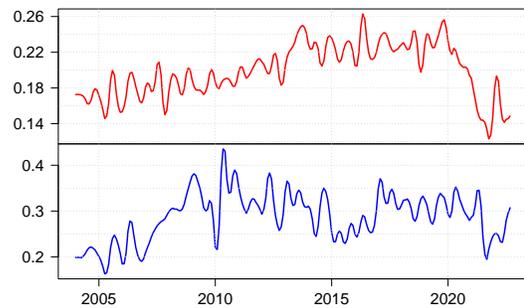
(i) Personal, Cultural and Rec. Services



(j) Construction Services



(k) Manufacturing and Maintenance Services



(l) Government Services

Figure 5.5.1: Estimated monthly disaggregate trade series of the subdivisions of the service sector (£billion). Exports trade in red on top and imports trade in blue at the bottom (cont.)

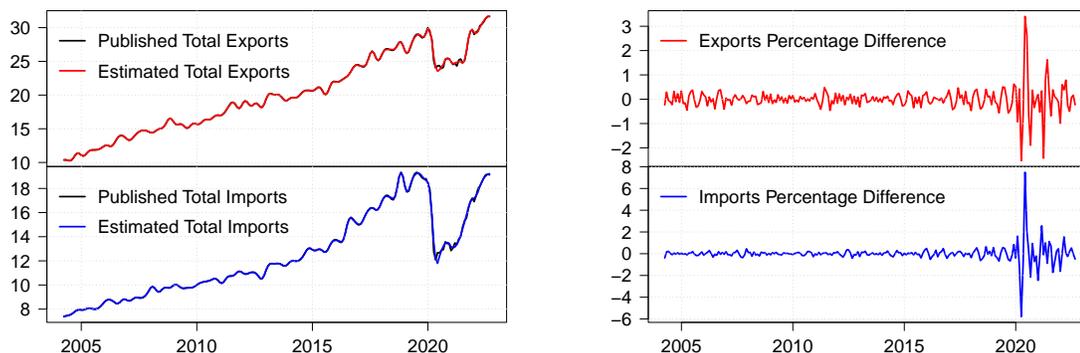


Figure 5.5.2: (Left - £billion) Published monthly total exports and imports time series by ONS with estimated series. (Right - %) Percentage difference between published series, \mathbf{y}^{pub} , and estimated series, \mathbf{y}^{est} , i.e. $(\mathbf{y}^{\text{pub}} - \mathbf{y}^{\text{est}})/\mathbf{y}^{\text{est}} \times 100\%$.

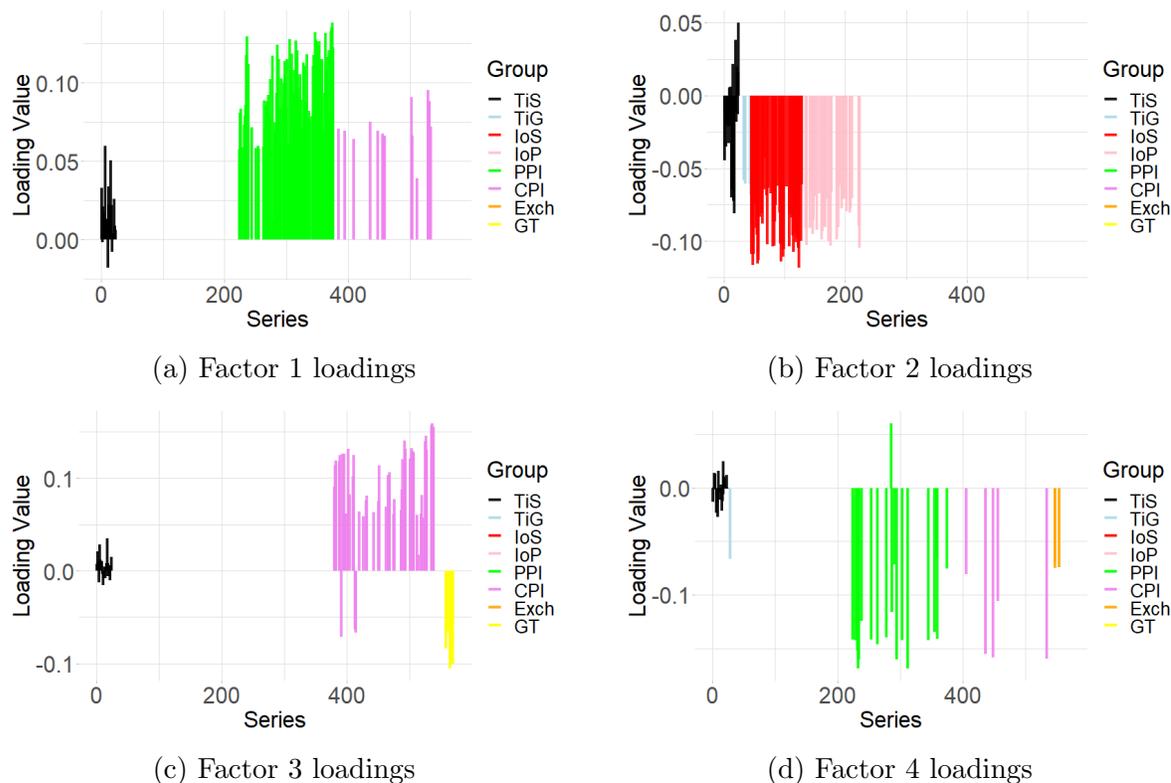


Figure 5.5.3: Estimated factor loadings for all 4 factors in the joint state-space framework trade in services application. Loadings colour coded by indicator group.

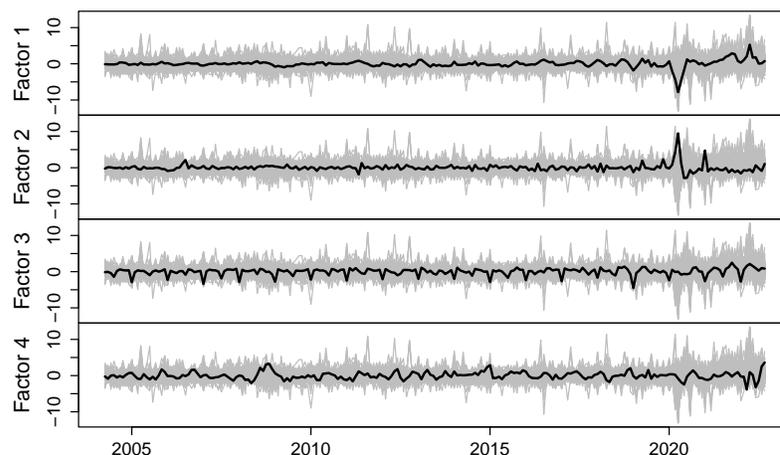


Figure 5.5.4: Estimated factors (black) with original data (grey) as a function of time from the joint state-space framework trade in services application.

2018 nowcast is re-calculated as the ragged edge moves along a month. This procedure continues with a new nowcasting quarter of interest every 3 months.

To compare the joint state-space framework to an alternative high-dimensional temporal disaggregation method we fit the sparse temporal disaggregation (spTD) procedure of Chapter 3 to the data and similarly aggregate months estimates in the latent current quarter to obtain a current quarter nowcast. As the spTD method is univariate and cannot handle missing data the procedure is applied to each target separately and any missing data is first forecasted using ARMA models with a possible order of up to 3. Figure 5.5.5 displays the results of the nowcasting exercise in terms of mean absolute error (MAE) over the 48 month expanding window between the observed quarterly figure and aggregated monthly estimates in that quarter with the proposed joint state-space model on the left and spTD on the right. Each bar represents the MAE for one of the 24 target series ordered from left to right by the service subdivisions: total; insurance & pension; construction; financial; telecomms; intellectual property; transport; manufacturing & maintenance; other business; personal, cultural &

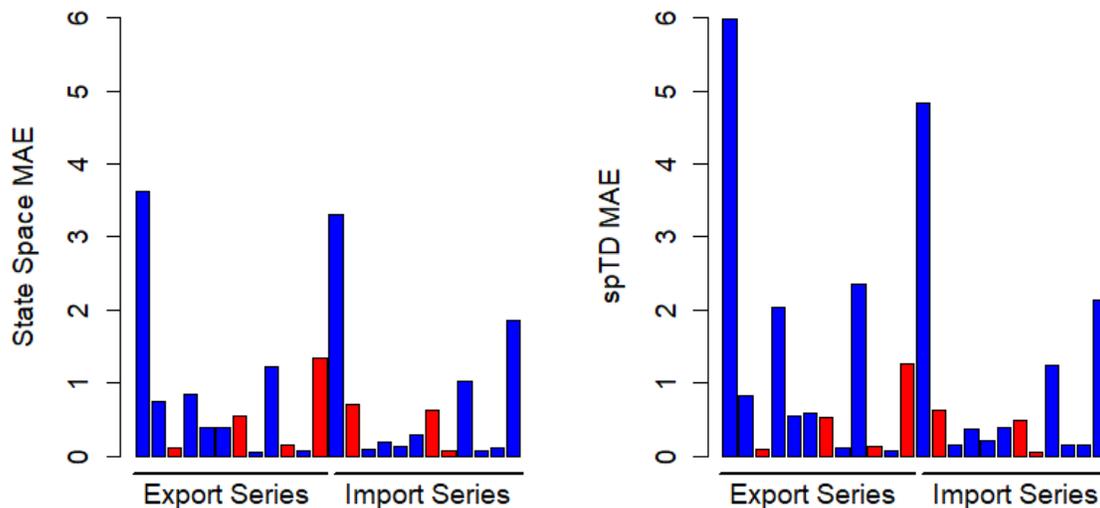


Figure 5.5.5: Pseudo-real time nowcasting errors (MAE) for each trade in services target series with bars ordered by the service subdivisions: total; insurance & pension; construction; financial; telecomms; intellectual property; transport; manufacturing & maintenance; other business; personal, cultural & recreational; government; and travel. Grouped by exports and imports. Performance evaluated by estimating observed quarters over the period September 2018 to August 2022. Left plot displays the joint state-space MAE results and the right plot displays spTD. Each bar is coloured according to which method performs best for that target series. Blue: Joint state-space, red: spTD.

recreational; government; and travel, which are grouped by exports and imports. To easily visualise which method performs best for each target series the winning method is coloured blue for the proposed state-space model and red for spTD. As we see, the proposed model beats spTD on 17 out of 24 target series.

5.6 Examining UK Trade Throughout the Brexit Transition Period⁵

5.6.1 Introduction

Producing information on UK trade at a monthly frequency, as opposed to quarterly, provides us with greater levels of interpretation into the short-term dynamics of the process, allowing us to study short-term fluctuations and trends that could be significant for policy makers when making informed decisions. In this experimental study, we explore these dynamics throughout the Brexit transition period and in particular, explore the effects (if any) the signing of the Trade and Cooperation Agreement (TCA) and officially transitioning out of the EU at the end of 2020 had on UK trade in goods and services. The 4 target series of interest that we wish to draw conclusions on are: monthly exports and imports of UK trade in goods (all commodities) to EU countries and monthly exports and imports of UK trade in services (all services) to EU countries over the period of January 2015 to September 2022. The monthly trade in goods data are published by the ONS at current market prices (£billion) and are seasonally adjusted. The monthly trade in services data are estimated using the joint state-space model described in Section 5.3 with the same monthly indicator series found in Table 5.1.1. Note, these estimates for services are themselves uncertain and so one must treat the overall estimated effects of the TCA with caution.

To visualise the monthly fluctuations of the 4 target series throughout the Brexit transition period, we plot each series between January 2015 and September 2022 in Figure 5.6.1 (blue line). The plots also display (dashed red line) the corresponding series' trade with non-EU member countries (i.e. the rest of the world [ROW]) to allow us to

⁵This section could potentially constitute a distinct chapter on its own, thereby possibly introducing a degree of inconsistency with the remainder of Chapter 5. However, it has been incorporated within this chapter for the sake of conciseness, considering that it relies on the findings and outcomes presented in Section 5.5.

compare how trade dynamics changed throughout this time⁶. The shaded grey region in each plot starts in January 2021 after the TCA was signed and the UK officially transitioned out of the EU. Firstly, looking at trade in goods (top panel) it appears with both exports and imports, the UK's trade with the EU and ROW followed similar trends and fluctuations before the TCA, both with a sharp drop in the first half of 2020 due to the start of the coronavirus pandemic and both with quick recoveries later in the year. It appears when the TCA came into force in January 2021 both exports and imports trade with the EU quickly fell in this month, a 46% decrease from December 2020 for exports and a 34% decrease for imports, but then after this month both series recovered. Goods exports to EU countries appears to recover very sharply and actually overtakes goods trade to non-EU countries, whereas goods imports from EU countries appears to recover slower and trade with non-EU countries becomes greater; possibly changing trend compared to pre-intervention. For trade in services (bottom panel) the immediate impact of the TCA seems less obvious than for trade in goods. While 2021 levels of service trade are less than the 2019 level, much of this decline is likely to be an effect of the coronavirus pandemic. In both exports and imports of services it appears the gap between trade with EU countries and the ROW has increased since the pandemic and has not yet (before September 2022) gone back to its previous difference – possibly an effect of Brexit.

To quantify the significance Brexit had on UK trade with EU countries we set up an interrupted time series analysis (Linden, 2015) with January 2021 as the so-called “intervention date”. Here we test the significance of this intervention by comparing trends before and after to what might have occurred if Brexit did not happen; known as the counterfactual trend. This research contributes to recent studies trying to analyse impacts Brexit has had on UK trade activity. Freeman et al. (2022) conduct a difference-in-difference (DID) event study comparing the evolution of UK-EU trade with UK-ROW

⁶The monthly series for trade in services with non-EU countries was also estimated using the joint state-space framework.

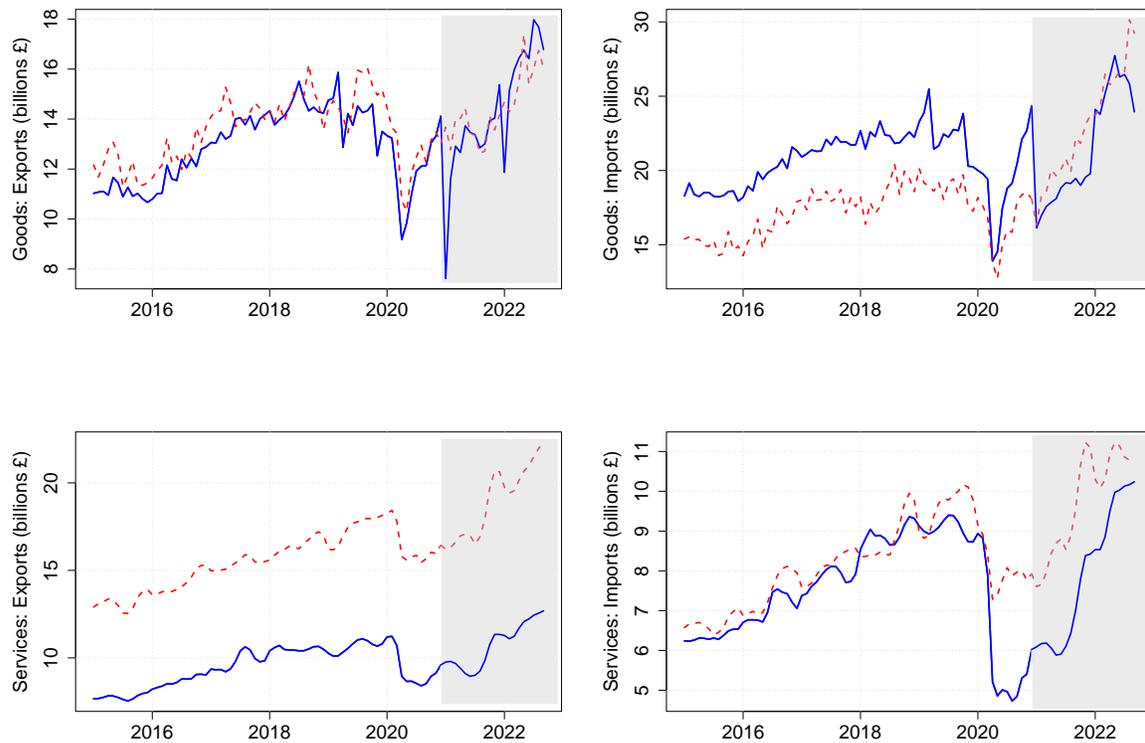


Figure 5.6.1: Monthly UK trade with EU countries (blue) and with non-EU countries (red). Top left: published trade in goods exports. Top right: published trade in goods imports. Bottom left: estimated trade in services exports from joint state-space model. Bottom right: estimated trade in services imports from joint state-space model. Grey shaded region starts in January 2021 when the UK officially transitioned out of the EU.

trade from the start of the referendum to the TCA. Their results find no statistically significant evidence that uncertainty and anticipation effects about Brexit led to a decline in the UK's trade with the EU trade before the TCA but do find the TCA led to an immediate decline in UK-EU trade relative to UK-ROW trade. Du et al. (2023) also perform a DID study and also find an immediate negative impact of the TCA on UK-EU trade, however, they find this decline to have largely disappeared by the beginning of 2022, with actual and counterfactual outcomes becoming approximately the same. Buigut and Kapar (2023) make use of a gravity model over three phases of the Brexit period and conclude the Brexit referendum phase led to a decline in UK-EU trade by around 10.5% and the transition phase by around 15% but do not find a significant effect due to the post transition (TCA) phase. Other literature (Dhingra et al., 2017; Fusacchia et al., 2022) simulate a range of Brexit outcomes using a general equilibrium approach and test the significance of each possible counterfactual outcome.

5.6.2 Interrupted Time Series Analysis

An interrupted time series model (ITSM) (Linden, 2015) is appropriate for the task at hand for two reasons, firstly, we want to understand how an outcome (UK trade with the EU) has changed after an intervention at a specific point in time (January 2021 - representing the end of the Brexit transition period), and secondly, we have a good amount of data before and after this intervention date as we are working with monthly time series up to September 2022. The general ITSM for a single-group under study is a linear model of the form⁷:

$$Y_t = \beta_0 + \beta_1 T_t + \beta_2 D_t + \beta_3 D_t T_t + \epsilon_t,$$

⁷Notation used is similar to that found in Linden (2015).

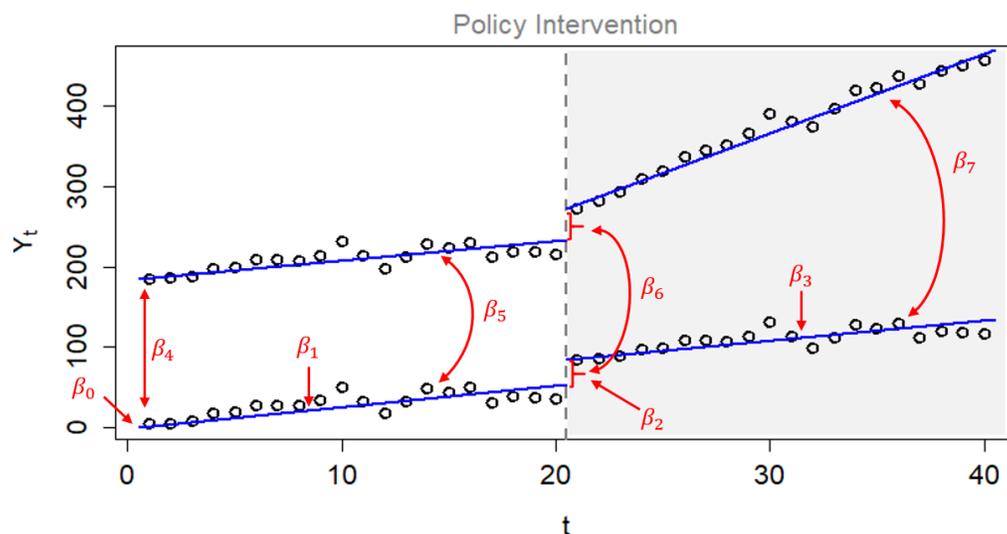


Figure 5.6.2: Visual interpretation of a multi-group ITSM with a treatment group (top line) and a control group (bottom line) using simulated data with a policy intervention at time $t = 20$. Coefficients $\beta_0 - \beta_7$ correspond to those in equation 5.6.1. A similar figure can be found in Linden (2015).

where Y_t is the outcome variable measured at each equally spaced time point t , T_t is the time since the start of the study, D_t is a dummy variable equal to 0 before the intervention and 1 after the intervention, $D_t T_t$ is an interaction term indicating time passed since the intervention, and ϵ_t is random error. These coefficients can be visualised by the bottom line of Figure 5.6.2. Assuming just a single-group ITSM where the bottom line is the outcome variable of interest, β_0 corresponds to the intercept of this outcome variable, β_1 tells us the general trend of the outcome before the intervention, β_2 represents how the outcome level has changed from the last observation before the intervention to the first observation after the intervention (immediate intervention impact), and β_3 is the difference between the slope of the outcome before and after the intervention, informing us on the long-term effects the intervention has had on the outcome variable of interest. We can study the statistical significance of these coefficients (i.e. find p-values) to get a sense of what might have happened if the intervention was not put in place – known as the counterfactual.

A problem with single-group ITSMs is the possibility that external variables not accounted for in the model have strong explanatory effects on the change in the outcome variable. The non-inclusion of these variables may lead to biased estimated coefficients in the model, leading to an over or under estimation of intervention impact. For example, it is very likely the coronavirus pandemic starting early 2020 is a confounding variable that has effected UK trade with the EU. To tackle this issue, a multi-group ITSM can be assumed with the introduction of a control group that is not subject to the intervention but share the same confounders as the treatment group (the outcome variable of interest that is impacted by the intervention). In this study we use UK trade with non-EU countries as a control group. The general ITSM for a multi-group study with a control group is a linear model of the form:

$$Y_t = \beta_0 + \beta_1 T_t + \beta_2 D_t + \beta_3 D_t T_t + \beta_4 Z + \beta_5 Z T_t + \beta_6 Z D_t + \beta_7 Z D_t T_t + \epsilon_t, \quad (5.6.1)$$

where Y_t is now a bivariate outcome variable with a treatment group (trade with EU countries) and a control group (trade with non-EU countries) and Z is a dummy variable which is equal to 0 for the control group and 1 for the treatment group. Figure 5.6.2 depicts a multi-group ITSM with the control group as the bottom line and the treatment group as the top line. The assumption is that the trend before and after the intervention in the control group remains unchanged, just as it would in the treatment group if the intervention did not happen (counterfactual). Coefficients β_1 to β_3 are as previously described and correspond to the control group. Coefficients β_4 to β_7 reflect the treatment group in the presence of the control group such that β_4 is the difference in the intercepts between control and treatments groups, β_5 is the difference in trend between the control and treatment before the intervention, β_6 represents the difference in level between the two groups immediately following the intervention, and β_7 is the difference between the difference of the trend between the control and treatment before the intervention and the difference of the trend between the control and treatment after the intervention,

i.e. a difference-in-difference informing us on the difference between the sustained effect of the control and treatment group after the intervention. Control groups are usually selected if they are comparable to the treatment group before the intervention, meaning we look for controls such that the coefficients β_4 and β_5 in (5.6.1) are not statistically significant (i.e. p-value > 0.05).

5.6.3 Results

We now apply the multi-group ITSA to each of the 4 target series, monthly imports and exports of trade in goods and services to EU countries, over the period of January 2015 to September 2022 with January 2021 as the intervention date and use control groups corresponding to the respective trade with non-EU countries. The analysis is repeated using the published quarterly version of the above trade in goods and services data to explore if using disaggregate data helps when determining the significance of Brexit effects. When fitting the ITSM we must deal with auto-correlation present in the residual process ϵ_t as not accounting for this can lead to an under-estimation of standard errors and an over-estimation of statistical significance. The approach we take is to first fit an ordinary least-squares (OLS) regression to (5.6.1) and then fit an ARMA(p, q) model to the estimated residuals $\hat{\epsilon}_t$. With estimated ARMA parameters from this fit we perform a generalised least-squares (GLS) rotation of the data and re-estimate coefficients. We do find this to somewhat change the p-values of the estimated coefficients.

Table 5.6.1 and 5.6.2 presents the estimated coefficients for the goods trade targets and service trade targets respectively, with their respective standard errors in parentheses and p-values in square brackets. Coefficients that are statistically significant (i.e. p-value < 0.05) are highlighted in bold. These results mostly agree with the interpretation of the plots presented in Figure 5.6.1 and conclusions drawn by authors Freeman et al. (2022) and Du et al. (2023) in their DID analysis. The control group of trade with non-EU countries (of both goods and services) does appear to follow the same level

and trend as the treatment group in the pre-intervention period given the coefficients β_4 and β_5 are not statistically significant in any table; the interest lies in studying the significance of coefficients post-intervention. Firstly, the TCA intervention does appear to have a significant impact on trade in goods of exports. The immediate impact in January 2021 leads to a decline in goods exports to EU countries by around £6.9b with a p-value very close to 0. The DID of trend pre and post intervention between goods exports to EU and non-EU countries is significant as well, with the difference increasing towards EU trade. This reveals that trade in goods of exports has been steadily growing since the intervention, thus recovering from the initial immediate losses. Interestingly, when conducting analysis on the quarterly data, the coefficient β_7 was not deemed significant with a p-value of 0.167 and the p-value for β_6 increased to 0.023. This could highlight the benefits of working with disaggregate data to capture the significance of these trends. In addition, it appears the standard errors for all the quarterly outputs are larger than those for monthly outputs, further substantiating the benefits derived from data disaggregation.

Trade in goods of imports from EU countries also has a significant immediate decline following the TCA of around £5.9b. It's DID, β_7 , suggests there has been a long-term decline post the TCA, however, this result is not deemed statistically significant. What is interesting is that trade in goods of imports from non-EU countries (i.e. the control group) appears to have a significant immediate decline of around £2.4b following the TCA and a significant long-term increase in trend in the post-TCA period. The results from the trade in services data seems to suggest the TCA intervention did not lead to any real significant impact on trade in services to EU countries for both exports and imports. Service exports to non-EU countries does appear to have a significant long-term incline in the post-TCA period given the β_3 estimate. Comparing these results to the aggregate quarterly data analysis we find the p-values for β_6 and β_7 to increase further with the quarterly data.

	Goods: Exports (Monthly)	Goods: Exports (Quarterly)	Goods: Imports (Monthly)	Goods: Exports (Quarterly)
β_1	0.008 (0.025) [0.751]	0.154 (0.299) [0.610]	0.028 (0.031) [0.368]	0.333 (0.325) [0.311]
β_2	0.316 (0.881) [0.720]	-0.879 (3.294) [0.791]	-2.351 (1.148) [0.042]	-7.362 (4.807) [0.132]
β_3	0.077 (0.112) [0.491]	1.036 (1.102) [0.352]	0.601 (0.140) [0.000]	5.047 (1.376) [0.001]
β_4	-1.885 (1.589) [0.237]	-3.506 (6.817) [0.609]	2.435 (1.944) [0.212]	9.469 (7.027) [0.183]
β_5	0.041 (0.036) [0.253]	0.171 (0.444) [0.703]	0.029 (0.044) [0.509]	0.097 (0.467) [0.837]
β_6	-6.913 (1.248) [0.000]	-10.900 (4.660) [0.023]	-5.926 (1.625) [0.000]	-15.189 (6.807) [0.030]
β_7	0.336 (0.139) [0.017]	2.048 (1.462) [0.167]	-0.151 (0.174) [0.387]	-0.746 (1.800) [0.680]

Table 5.6.1: Interrupted time series model coefficients (standard errors) [p-value] for the UK-EU trade in goods monthly and quarterly series. Significant coefficients (i.e. p-value < 0.05) in bold.

	Services: Exports (Monthly)	Services: Exports (Quarterly)	Services: Imports (Monthly)	Services: Imports (Quarterly)
β_1	0.049 (0.038) [0.204]	0.432 (0.257) [0.099]	0.019 (0.040) [0.639]	0.164 (0.469) [0.727]
β_2	-0.526 (0.369) [0.156]	-2.247 (2.548) [0.382]	-0.481 (0.349) [0.171]	-2.005 (2.430) [0.413]
β_3	0.255 (0.091) [0.005]	2.144 (0.898) [0.021]	0.140 (0.0863) [0.107]	1.369 (1.031) [0.190]
β_4	-5.248 (3.092) [0.092]	-15.808 (6.010) [0.111]	-0.312 (3.322) [0.925]	-0.938 (12.941) [0.943]
β_5	-0.022 (0.059) [0.713]	-0.204 (0.389) [0.601]	-0.022 (0.057) [0.700]	-0.251 (0.666) [-0.707]
β_6	0.541 (0.522) [0.302]	2.133 (3.603) [0.556]	0.339 (0.494) [0.494]	1.703 (3.437) [0.622]
β_7	-0.136 (0.128) [0.289]	-1.047 (1.204) [0.388]	0.071 (0.122) [0.563]	0.732 (1.459) [0.618]

Table 5.6.2: Interrupted time series model coefficients (standard errors) [p-value] for the UK-EU trade in services monthly and quarterly series. Significant coefficients (i.e. p-value < 0.05) in bold.

5.7 Conclusion

The first half of this chapter motivated the need for monthly estimates of UK trade in services and proposed a joint state-space framework that performs multivariate temporal disaggregation while incorporating the benefits of the sparse DFM (Chapter 4). Essentially, the joint model fuses a sparse DFM assumed for the high-dimensional indicator set with latent disaggregate target series treated as latent states assumed to have the same factor model representation as the sparse DFM, and these disaggregate targets are systematically sampled over each quarter using a cumulator variable to ensure temporal consistency with the observed quarterly data. Model parameters and the disaggregate series are estimated using a regularised EM algorithm that converges when disaggregate estimates do not change by much from one iteration to the next. The model appears to perform well in disaggregation ability in simulation studies and pseudo-real time data analysis with alternative temporal disaggregation methods. The model produced monthly estimates for total imports and exports trade in services that

are very similar to those published by the ONS.

The general structure of the joint state-space framework (5.3.5)-(5.3.6) can be adapted in several ways to model more complex data features and these open the door to future work ideas. It is possible to incorporate cross-sectional constraints to the disaggregate target matrix \mathbf{Y}_t by including another cumulator variable that aggregates subdivisions of the service sector contemporaneously such that in a month, t , the subdivisions total matches the published total imports or exports monthly series value for month t . Proietti (2011a) work with disaggregate data transformed by logarithms and propose a non-linear smoothing algorithm using a Taylor's series approximation to ensure exact aggregation in levels. A similar approach can be considered here if a log-transform is required.

The second half of the chapter conducted a study to assess how Brexit has affected the UK's trade with the EU relative to the rest of the world. For this task, an interrupted time series model (ITSM) was used to quantify how the level and trend of UK-EU trade changed following the signing of the TCA in January 2021. Analysis was conducted for all output variables of interest at the monthly frequency using published monthly trade in goods data by the ONS and estimated monthly trade in services using the proposed joint state-space model in the first half of the chapter. We find that working with disaggregate data does lead to a reduction in p-values in the ITSM fit and reveals more informative conclusions about Brexit effects.

Using UK trade with non-EU countries (rest of the world) as a control group to distinguish effects due to the coronavirus pandemic, we find evidence of a significant decline in UK-EU trade in goods immediately following the TCA intervention but no such evidence for UK-EU trade in services. Following this immediate decline of trade in goods, goods exported to EU countries shows evidence of a recovery in the post-transition period with a long-term increase in trend in goods trade with EU countries relative to non-EU countries. In contrast, the recovery in imported goods trade did not exhibit

the same pattern, as trade with non-EU countries witnessed an increase during this post-transition period. Regarding trade in services, our analysis using the ITSM fit did not yield conclusive evidence regarding the effects of the TCA. The simplistic ITSM leads to interpretable measures on Brexit effects on UK-EU trade and can be refined in several ways, which we explore in detail in Chapter 7.

Chapter 6

Temporal Disaggregation and Nowcasting with Sparsity Constraints in R

The focus of the thesis has been on creating new methodology for the important tasks of temporal disaggregation and nowcasting that can handle high-dimensional amounts of data in an interpretable way by exploiting sparsity constraints. Chapter 3 introduced Sparse Temporal Disaggregation (spTD), a new high-dimensional temporal disaggregation method that extends the well-established [Chow and Lin \(1971\)](#) family of methods to a regularised M-estimation framework allowing simultaneous selection of relevant indicators and estimation of their impact. Chapter 4 introduced the Sparse Dynamic Factor Model (sparse DFM), a new class of DFMs that impose sparsity constraints on the loadings matrix to allow interpretability of factor structure using an expectation maximisation (EM) algorithm with regularised quasi-maximum likelihood estimation of parameters. For reproducibility purposes, both of these core methodologies have been developed into R packages available on the Comprehensive R Archive Network (CRAN) allowing practitioners to easily implement them.

Section 6.1 introduces the first package called `DisaggregateTS` that implements the `spTD` methodology proposed in Chapter 3 along with traditional standard-dimensional temporal disaggregation methods introduced in Chapter 2.1. Section 6.2 then introduces the second package called `sparseDFM` that implements the sparse DFM methodology of Chapter 4 and alternative DFM estimation techniques discussed in Chapter 2.2. In both sections the structure of the packages are outlined and details are given on the input and outputs of core functions. The packages are applied to synthetic and real-data case studies as a walk-through guide on how to use them. Note, the `sparseDFM` package is more developed than the `DisaggregateTS` package providing several plotting features for visualisation and producing an S3 class object with the main function `fit`. The S3 class object allows generic R functions to be called such as `plot()`, `predict()`, `summary()`, `fitted()` and `residual()`.

6.1 DisaggregateTS: An R Package for High-Dimensional Temporal Disaggregation

This package was developed in collaboration with Dr Kaveh Salehzadeh Nobari.

6.1.1 Introduction

The aim of the package `DisaggregateTS` is to offer the tools for the R user to implement popular temporal disaggregation methods proposed in the literature, as well as their extension to high-dimensional settings. In particular, the package offers the classic temporal disaggregation methods: `Denton` (Denton, 1971), `Denton-Cholette` (Cholette, 1984), `Chow-Lin` (Chow and Lin, 1971), `Fernandez` (Fernandez, 1981), `Litterman` (Litterman, 1983), and the high-dimensional method(s) of Chapter 3: `spTD` and `adaptive-spTD`. The `spTD` methods accept the input of a high-dimensional indicator matrix, \mathbf{X} , and will output a sparse coefficients vector, β , that can be used to identify relevant indicators in

the temporal disaggregation task. All methods are suitable for disaggregation situations where the high-frequency is an integer multiple of the low-frequency (e.g. quarters to months) and the output of disaggregate estimates are constrained such that either their sum, average, first or last value is consistent with the low-frequency input. The package also allows the possibility for extrapolation (forecasting) of the disaggregate output series. This will be performed if the number of observations of the high-frequency indicator matrix, \mathbf{X} , is greater than the product of the number of low-frequency observations in \mathbf{Y} and the aggregation ratio supplied by the user (e.g. `aggRatio = 3` for quarter to month disaggregation).

The main contribution of this package stems from the `spTD` and `adaptive-spTD` options pertaining to sparse temporal disaggregation and adaptive sparse temporal disaggregation. There are alternative software packages that exist that perform classical (standard-dimensional) temporal disaggregation. The `tempdisagg` package by Sax and Steiner (2013) is a popular temporal disaggregation package in R allowing each of the classical methods discussed above. In MATLAB, the temporal disaggregation toolbox by Quilis (2018) also implements these classic methods as well as the mixed-frequency state-space modelling toolbox by Brave et al. (2022) for a state-space approach. The `JDemetra+` software developed by the National Bank of Belgium (NBB) in cooperation with the Deutsche Bundesbank and Eurostat offers tools for seasonal adjustment, outlier detection, nowcasting and temporal disaggregation. The main function of the `DisaggregateTS` package is now presented before a simulation exercise is conducted using the package's data generator function `TempDisaggDGP()`.

6.1.2 Package Structure

Installation

The `DisaggregateTS` package can be installed from CRAN as follows:

```
install.packages("DisaggregateTS")
```

The development version can be installed from GitHub using the `devtools` package (Wickham et al., 2022):

```
devtools::install_github("mosley1/DisaggregateTS")
```

To load the package into the R environment, type:

```
library(DisaggregateTS)
```

Core Function

The main function of the package which performs each temporal disaggregation method discussed is `disaggregate()`. This function is of the form:

```
disaggregate(Y, X, aggMat = "sum", aggRatio = 4, method = "Chow-Lin",
             Denton = "first")
```

It takes the following arguments:

- **Y**: An $n_l \times 1$ data matrix representing the low-frequency series that we wish to disaggregate.
- **X**: An $n \times p$ data matrix representing p high-frequency indicator series. It must be the case that $n \geq n_l \times \text{aggRatio}$. If $n > (n_l \times \text{aggRatio})$ then the last $n - (n_l \times \text{aggRatio})$ values of the disaggregate series output, y , are extrapolated. In the event that there is no input, the indicator matrix **X** is replaced with an $n \times 1$ vector of ones.
- **aggMat**: A character coinciding with with the aggregation matrix **C** defined in Chapter 2.1.1. It has been set to "sum" by default, rendering it suitable for flow data. Alternative options include "first", "last" and "average".

- **aggRatio**: An integer corresponding to the aggregation ratio between high and low frequency variables. The default value is 4, which for example represents the ratio of annual to quarterly data.
- **method**: A character representing the disaggregation method to implement. This argument has been set to "Chow-lin" method by default. Other classical low-dimensional options include "Denton", "Denton-Cholette", "Fernandez", and "Litterman", where these techniques have been extensively discussed in Chapter 2.1. For high-dimensional temporal disaggregation use the option "spTD" or "adaptive-spTD" extensively discussed in Chapter 3. "adaptive-spTD" implements the adaptive LASSO version of spTD relevant when, for instance, the columns of the design matrix \mathbf{X} exhibit multicollinearity, and the irrepresentability condition is violated.
- **denton**: A character representing the type of weighting matrix used in the Denton disaggregation methods discussed in Chapter 2.1.3. This is only relevant when `method = "Denton"` or `method = "Denton-Cholette"`. The default option is "first" representing first-difference changes. Other options include "absolute", "second" and "proportional".

The output of the `disaggregate()` function includes:

- **y_Est**: An $n \times 1$ data matrix corresponding to the estimated high-frequency disaggregate series.
- **beta_Est**: A $p \times 1$ data matrix corresponding to the estimated coefficient vector.
- **rho_Est**: A value in $(0, 1)$ corresponding to the estimated AR(1) residual parameter. This is only relevant for methods assuming AR(1) residuals.
- **ul_Est**: An $n_l \times 1$ data matrix corresponding to the estimated aggregate residual series. I.e. equal to $\mathbf{Y} - \mathbf{C}\mathbf{X}\hat{\boldsymbol{\beta}}$.


```

# Store synthetic data
X = SyntheticData$X_Gen
beta = SyntheticData$Beta_Gen
y = SyntheticData$y_Gen
Y = SyntheticData$Y_Gen
e = SyntheticData$e_Gen

```

This has simulated a synthetic annual-to-quarterly temporal disaggregation dataset alongside 50 ‘years’ of observations, with 30 indicator series with a coefficient vector,

```

as.numeric(beta)
[1] 3 0 0 0 0 0 3 -3 0 0 3 0 -3 0 -3 0 0 0 0 0 0 0
     3 3 0 0 0 0 -3 0

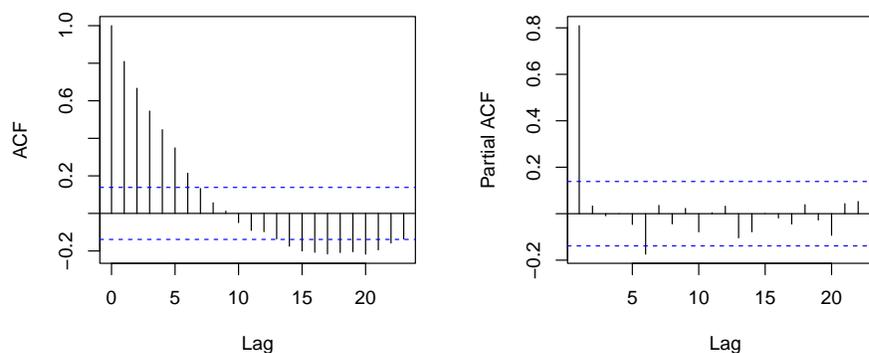
```

that is 70% sparse, and an AR(1) error process with $\rho = 0.8$. We can check this AR(1) residual process by plotting the autocorrelation function (acf) and the partial autocorrelation function (pacf):

```

acf(e, main = "")
pacf(e, main = "")

```



clearly showing an AR(1) process with parameter 0.8.

We can now fit the standard Chow-Lin temporal disaggregation method to the data, as well as the spTD and adaptive-spTD methods to compare performance:

```
fit.CL <- disaggregate(Y, X, aggMat = "sum",
                      aggRatio = 4, method = "Chow-Lin")
fit.sptd <- disaggregate(Y, X, aggMat = "sum",
                        aggRatio = 4, method = "spTD")
fit.adasptd <- disaggregate(Y, X, aggMat = "sum",
                            aggRatio = 4, method = "adaptive-spTD")
```

Firstly, we can compare estimates of the sparse β :

```
# Print beta estimates
betaEstimate = cbind(beta, fit.CL$beta_Est, fit.sptd$beta_Est,
                    fit.adasptd$beta_Est)
colnames(betaEstimate) = c("True", "Chow-Lin", "spTD", "ada-spTD")
round(betaEstimate,3)
```

	True	Chow-Lin	spTD	ada-spTD
[1,]	3	2.794	3.383	3.545
[2,]	0	0.338	0.000	0.000
[3,]	0	0.462	0.000	0.000
[4,]	0	-0.392	0.000	0.000
[5,]	0	-0.356	0.514	0.000
[6,]	0	-0.081	0.000	0.000
[7,]	3	2.020	2.661	2.962
[8,]	-3	-3.667	-2.990	-3.226
[9,]	0	0.580	0.000	0.000
[10,]	0	0.078	0.000	0.000

[11,]	3	2.928	2.364	2.494
[12,]	0	-0.941	0.000	0.000
[13,]	-3	-3.864	-2.752	-3.048
[14,]	0	-0.254	0.000	0.000
[15,]	-3	-3.474	-2.780	-2.804
[16,]	0	0.231	0.000	0.000
[17,]	0	-0.090	0.423	0.000
[18,]	0	-0.038	0.000	0.000
[19,]	0	0.230	0.000	0.000
[20,]	0	1.440	0.000	0.000
[21,]	0	0.366	0.000	0.000
[22,]	0	0.615	0.000	0.000
[23,]	3	4.356	3.102	3.283
[24,]	3	2.581	3.136	3.120
[25,]	0	-1.784	-0.384	0.000
[26,]	0	-0.512	0.000	0.000
[27,]	0	-0.341	-0.385	0.000
[28,]	0	0.409	0.000	0.000
[29,]	-3	-2.454	-3.211	-3.504
[30,]	0	0.692	0.000	0.000

As we see, spTD and adaptive-spTD both perform variable selection and induce sparsity.

The adaptive-spTD obtains the exact sparsity, while spTD possesses a few false positives.

We can also check the estimation of the AR(1) parameter which was set to be 0.8:

```
fit.CL$rho_Est
```

```
[1] 0.9351062
```

```
fit.sptd$rho_Est
```

```
[1] 0.8162954
```

```
fit.adasptd$rho_Est
[1] 0.8162954
```

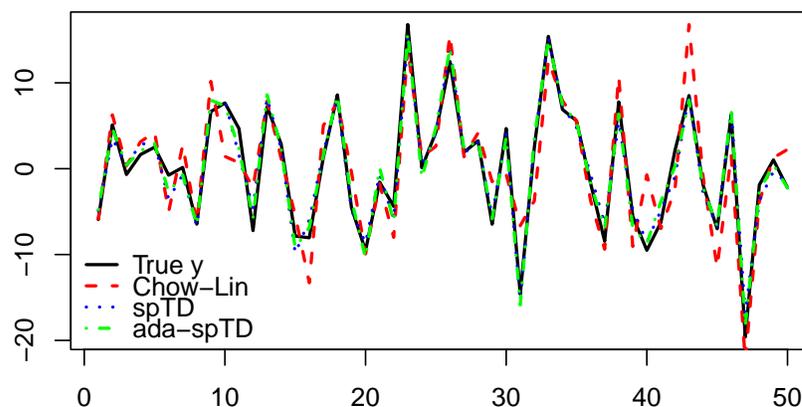
Both spTD and adaptive-spTD recover this parameter very well.

To compare the performance in recovering the true high-frequency series \mathbf{y} we first compute the root mean squared error (RMSE) between \mathbf{y} and the estimates and then plot the estimates on top of the truth for the final 50 quarters to make visualisation easier.

```
# Compute RMSE values
sqrt(mean((y - as.numeric(fit.CL$y_Est))^2))
[1] 3.262439
sqrt(mean((y - fit.sptd$y_Est)^2))
[1] 1.131614
sqrt(mean((y - fit.adasptd$y_Est)^2))
[1] 1.015399

# Plot estimates on top of truth (last 50 quarters)
plot(y[151:200], type = "l", lwd = 2, ylab = "", xlab = "")
lines(fit.CL$y_Est[151:200], col = "red", lty = 2, lwd = 2)
lines(fit.sptd$y_Est[151:200], col = "blue", lty = 3, lwd = 2)
lines(fit.adasptd$y_Est[151:200], col = "green", lty = 4, lwd = 2)
legend("bottomleft", legend = c("True y", "Chow-Lin", "spTD",
    "ada-spTD"), lty = c(1,2,3,4), lwd = 2,
    col = c("black", "red", "blue", "green"), bty = "n")
```

The results of the simulation clearly show the spTD methods are able to recover the true parameters and latent high-frequency series very accurately, with improved performance over Chow-Lin.



6.2 sparseDFM: An R Package to Estimate Dynamic Factor Models with Sparse Loadings

This package was developed in collaboration with Dr Tak-Shing Chan.

6.2.1 Introduction

The aim of the package `sparseDFM` is to offer the tools for the R user to implement dynamic factor models with the option to induce sparse loadings. In summary, the package offers three different popular DFM estimation techniques as well as the new sparse DFM estimation technique of Chapter 4. The estimation options are:

- `PCA` - principal components analysis (PCA) for static factors seen in Stock and Watson (2002).
- `2Stage` - the two-stage framework of PCA plus Kalman filter & smoother seen in Giannone et al. (2008) and Doz et al. (2011).
- `EM` - the quasi-maximum likelihood approach using the EM algorithm to handle

arbitrary patterns of missing data seen in Bańbura and Modugno (2014).

- `EM-sparse` - a regularised EM approach allowing for sparse factor loadings seen in Chapter 4.

The package allows the user the option of two different idiosyncratic error processes:

- `IID` - errors are IID white noise: $\epsilon_{i,t} \sim N(0, \sigma_i^2)$.
- `AR1` - errors follow an AR(1) process: $\epsilon_{i,t} = \phi_i \epsilon_{i,t-1} + e_{i,t}$ with $e_{i,t} \sim N(0, \sigma_i^2)$.

Idiosyncratic errors following an AR(1) process are augmented into the DFM framework by treating them as latent states as in Bańbura and Modugno (2014). This slows down computation time of the DFM as more states have to be estimated. It is therefore recommended to treat errors as IID and perform post-analysis of residuals where possible. Details on the AR(1) error case are provided in Appendix D.1.

The package also allows the option of two different approaches for estimating the Kalman filter and smoother equations:

- `multivariate` - classic Kalman filter and smoother equations seen in Shumway and Stoffer (1982).
- `univariate` - univariate treatment (sequential processing) of the multivariate equations for fast Kalman filter and smoother seen in Koopman and Durbin (2000).

Huge speed ups are possible using the univariate treatment approach when the panel of data used is large. Appendix D.1 shows the computational speed comparison between the univariate and multivariate methods. It is recommended to use the univariate approach when errors are IID and the multivariate approach when they are AR(1).

`sparseDFM` is the first R package able to implement sparse DFMs using an efficient EM algorithm. In the package, most algorithms are coded in C++ for computational

speed. Alternative software exists that can implement classic DFMs. For PCA estimation techniques the `princomp` and `prcomp` functions from the `stats` package in R are popular, as well as the `FactoMineR` package by Lê et al. (2008) offering more options for the structure of variables and the `elasticnet` package by Zou, Hui and Hastie, Trevor (2012) offering sparse PCA estimation. The `MARSS` R package of Holmes et al. (2012) allows maximum likelihood estimation of general state space structures that can be used for DFMs. The `dfms` R package of Krantz and Bagdziunas (2022) implements the three classic DFM estimation methods assuming IID errors and utilises C++ code for speed. The R packages `nowcasting` by De Valk et al. (2019) and `nowcastDFM` by Hopp (2021) implement the mixed-frequency time series nowcasting DFM methods of Giannone et al. (2008) and Bańbura et al. (2011)¹. In MATLAB the `MFSS` toolbox (Brave et al., 2022) provides a range of estimation methods for mixed-frequency state space models. In Python the `statsmodels` package provides classic DFM estimation. The core structure of the `sparseDFM` package is now presented.

6.2.2 Package Structure

Installation

The `sparseDFM` package can be installed from CRAN as follows:

```
install.packages("sparseDFM")
```

The development version can be installed from GitHub using the `devtools` package (Wickham et al., 2022):

```
devtools::install_github("mosley1/sparseDFM")
```

To load the package into the R environment, type:

```
library(sparseDFM)
```

¹These nowcasting packages have now been removed from the CRAN repository. However, they are still accessible via GitHub.

Core Function

The core function of the package is `sparseDFM()` allowing the different DFM estimation methods. The function can be implemented using:

```
sparseDFM(X, r, q = 0, alphas = logspace(-2,3,100), alg = "EM-sparse",
          err = "IID", kalman = "univariate", store.parameters = FALSE,
          standardize = TRUE, max_iter = 100, threshold = 1e-4)
```

It takes the following arguments:

- **X**: An $n \times p$ numeric data matrix or data frame representing p stationary time series variables with n observations. Missing data should be represented by `NA`.
- **r**: The number of factors the model should estimate. This can be selected using the function `tuneFactors()`. Usually less than 10 factors should be sufficient.
- **q**: The first q variables in **X** will not be regularised. This parameter is only relevant when `alg = "EM-sparse"`. It is useful when you want to ensure variables are always loaded onto all factors and hence regularisation is not applied to these variables. Set the first q series in **X** to be these variables. The default is `q = 0` where all variables are regularised.
- **alphas**: Numeric vector of ℓ_1 -norm penalty parameters. This parameter is only relevant when `alg = "EM-sparse"`. The default grid is set to be `alphas = logspace(-2, 3, 100)` to ensure a wide range of α values. Each iteration of the EM algorithm uses the previous α 's parameter estimates as a warm-start to the EM algorithm. The grid search will stop when all of the values in **alphas** are used for estimation or when a column of the estimated loadings $\hat{\mathbf{\Lambda}}$ becomes entirely 0. This will occur if α is too high. Single ℓ_1 -norm penalty parameters can be used as well as vectors.
- **alg**: The choice of estimation algorithm. Options include:

- "PCA": principal components analysis (PCA) for static factors (Stock and Watson, 2002);
 - "2Stage": the two-stage framework of PCA plus Kalman filter/smoothing (Giannone et al., 2008);
 - "EM": The quasi-maximum likelihood approach using the EM algorithm to handle arbitrary patterns of missing data (Bańbura and Modugno, 2014);
 - "EM-sparse": The novel sparsified EM approach allowing LASSO regularisation on factor loadings (Chapter 4). This is the default.
- **err**: The choice of distribution for idiosyncratic residuals. Options include:
 - "IID" for IID white noise residuals and this is the default.
 - "AR1" for AR(1) residuals. It is recommended to use IID errors for computational efficiency. See Appendix D.1 for details.
 - **kalman**: The choice of Kalman filter and smoother equations to use. Options include:
 - "multivariate": classic Kalman filter and smoother equations seen in Shumway and Stoffer (1982).
 - "univariate": univariate treatment (sequential processing) of the multivariate equations for fast Kalman filter and smoother seen in Koopman and Durbin (2000). This is the default.
 - **store.parameters**: A logical parameter indicating whether the estimation output of parameters and factors should be stored for every value of α considered in `alphas`. This parameter is only relevant when `alg = "EM-sparse"`. This may be useful when checking alternative outputs other than the one for the optimal α . The default is `store.parameters = FALSE`.

- `standardize`: A logical parameter indicating whether the data should be standardised or not before estimating the model. The default is `standardized = TRUE`.
- `max_iter`: An integer representing the maximum number of EM iterations. The default is `max_iter = 100`.
- `threshold`: Tolerance on EM iterates. See equation (4.3.12) in Chapter 4. Default is `threshold = 1e-4`.

The function `sparseDFM` returns an S3 object of class `'sparseDFM'`. This allows the user to pass the generic functions: `print()`, `summary()`, `plot()`, `residuals()`, `resids()`, `fitted()`, and `predict()` to the returned `sparseDFM` object to display generic outputs. The `predict()` function will itself return an S3 object of class `'sparseDFM_forecast'` to allow the user to `print()` forecast results. There are plenty of helpful types of plot included in the package to allow the user to visualise estimation results. For example, the coloured group line plots of Figure 4.5.3 are produced by using `type = "loading.grouplineplot"` in `plot()`. More detail on other plot types will be provided in the empirical application in the next section.

When called, the S3 object `sparseDFM` returns a list-of-lists containing the following:

- `data`: A list containing information on the data used for the model fit. Including the original data matrix, the initial balanced data matrix from `fillNA()`, the fitted values $\hat{\mathbf{A}}\hat{\mathbf{F}}$ and more.
- `params`: A list containing the estimated model parameters of $\hat{\mathbf{A}}$, $\hat{\mathbf{\Lambda}}$, $\hat{\Sigma}_u$ and $\hat{\Sigma}_e$.
- `state`: A list containing the estimated states and state covariances for each time point considered. I.e. the factors and factor covariances, along with AR(1) errors and covariances if `err = "AR1"`.

- `em`: A list containing information about the convergence of the EM algorithm and ℓ_1 -norm penalty parameter tuning.
- `alpha.output`: Parameter and state outputs for each ℓ_1 -norm penalty parameter in `alphas`. Only stored if `store.parameters = TRUE`.

The package also includes eight other functions the user can employ. A brief description of all functions are provided in Table D.2 of Appendix D.2.

6.2.3 Empirical Application: Exploring UK Inflation

This case study focuses on a small subset of quarterly CPI (consumer price inflation) index data for the UK taken from the Office for National Statistics' (ONS) website². The data contains 36 variables of different classes of the inflation index and 135 observations from 1989 Q1 to 2022 Q3. The purpose of this small, 36 variable example is to demonstrate the core functionality and the ability to graphically display results with the `sparseDFM` package. The data itself is provided with the package, as well as the Trade in Goods dataset used in Case Study 1 of Chapter 4 as a larger dataset example.

The data can be loaded directly into the R environment using:

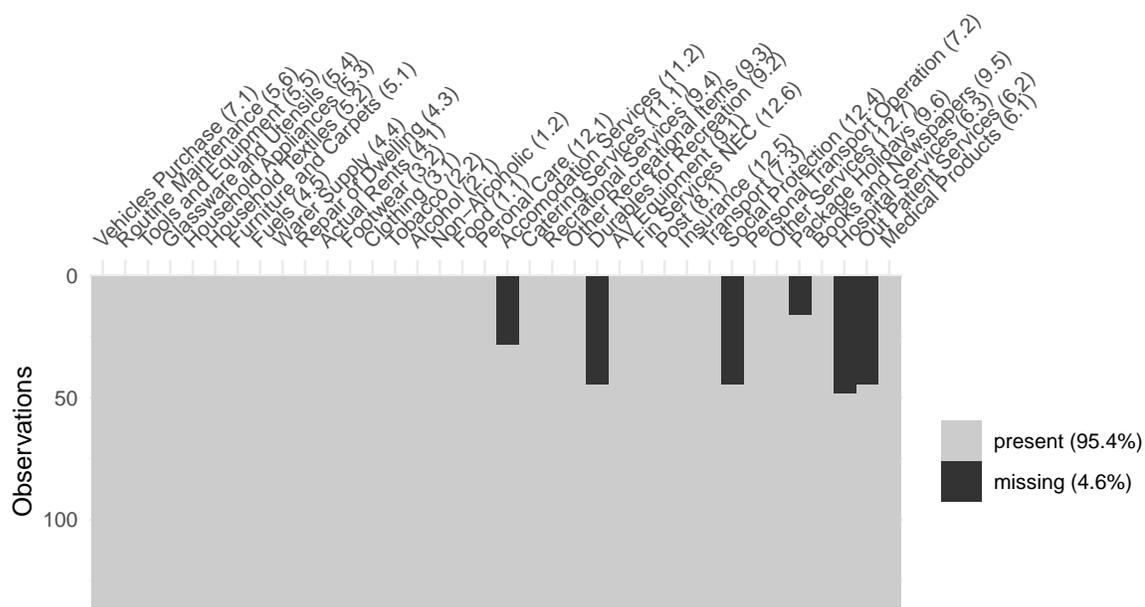
```
R> data <- inflation
```

Before we fit a model, it is worthwhile to first perform some exploratory data analysis. Two main properties to look out for when working with dynamic factor models are the amount of missing data present and if the data series look stationary or not. The function `missing_data_plot()` allows the user to visualise where missing data is present:

```
R> missing_data_plot(data)
```

We can see that 6 of the 36 variables have missing data towards the start of the sample.

²We use the Q4 2022 release (benchmarked to 2015=100) from <https://www.ons.gov.uk/economy/inflationandpriceindices/datasets/consumerpriceindices>



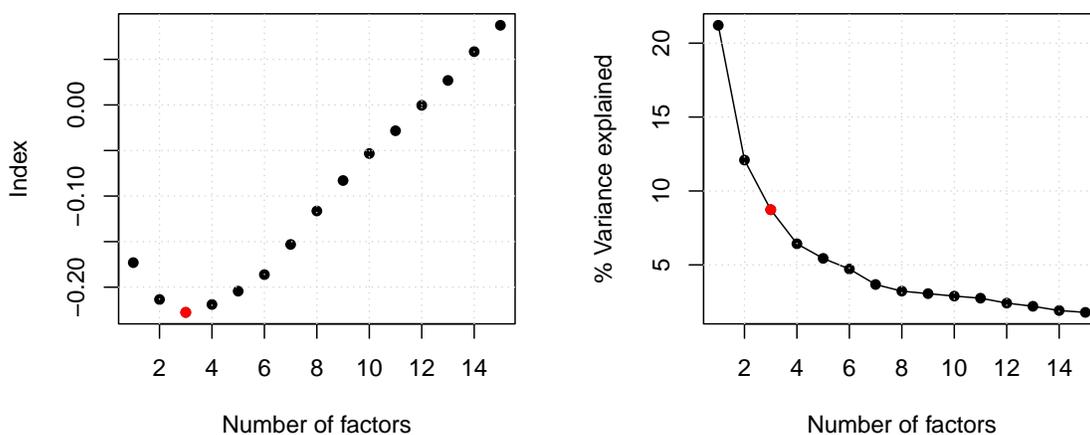
In a DFM, we assume the latent factors to follow a stationary VAR(1) process and hence we require the input data to be stationary. After exploration it appears the inflation data is stationary in first-differences. Hence, we can apply the function `transformData()` and set the `stationary_transform` parameter to represent a first-difference transform for all variables:

```
R> new_data <- transformData(data,
                             stationary_transform = rep(2, ncol(data)))
```

We store the stationary first-difference data as the variable `new_data`. The `stationary_transform` parameter is a vector of length matching the number of variables in the data that contains a stationary transform index for each variable with the options: 1 (no change), 2 (first-difference), 3 (second-difference), 4 (log-first-difference), 5 (log-second-difference), 6 (growth rate) or 7 (log growth rate). The returned transformed data set is of the same dimension as the original data with `NA` where missing data is, for example, the first row of `new_data` will all be `NA` as we do not have an observed 1988 Q4 value to difference with.

Now we have a stationary data set, the next step would be to determine the number of factors to use with our DFM. To do this we can apply the function `tuneFactors()` that selects the best number of factors to use based on Bai and Ng (2002) information criteria:

```
R> tuneFactors(new_data, type = 2, standardize = TRUE,
               r.max = min(15, ncol(X)-1), plot = TRUE)
```



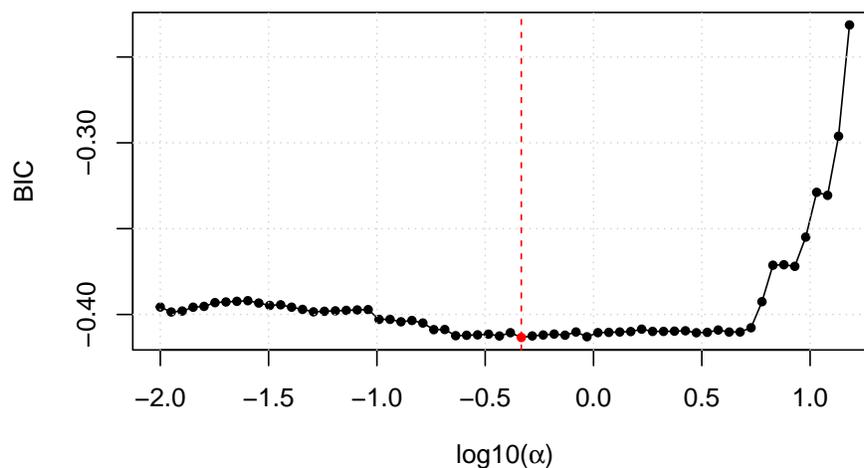
This function plots the information criteria (IC) values (left) and the proportion of variance explained based on PCA (right) with the best number of factors, $r = 3$, highlighted in red. The IC used by default is type 2. See Bai and Ng (2002) for types 1 and 3.

Using 3 factors, it is now time to fit a sparse DFM to the (stationary) data using the default grid of LASSO tuning parameters and the default (exact) idiosyncratic error assumption (IID) and Kalman technique (univariate):

```
fit.sdfm <- sparseDFM(new_data, r = 3, alphas = logspace(-2,3,100),
                     alg = "EM-sparse", err = "IID", kalman = "univariate")
```

To get a summary of the model fit type `summary(fit.sdfm)`. For information on the tuning of the LASSO parameter using BIC it is possible to do:

```
# The best alpha chosen
fit.sdfm$em$alpha_opt
#> [1] 0.4641589
# Plot the BIC values for each alpha
plot(fit.sdfm, type = "lasso.bic")
```



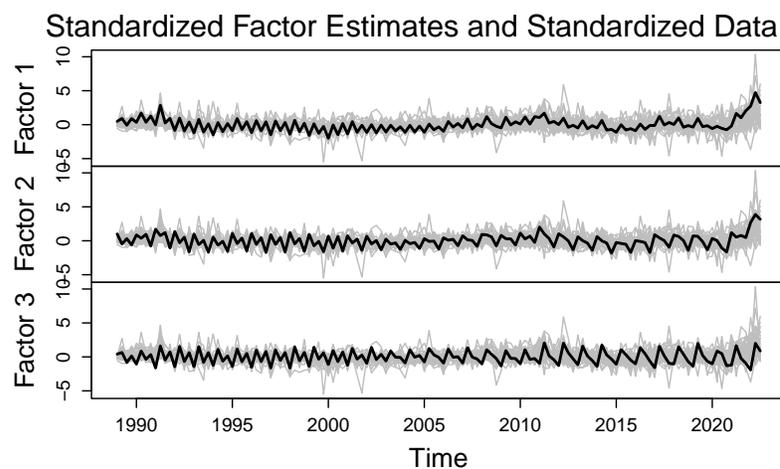
For the purposes of nowcasting or interpolation of missing data, an estimated series, $\hat{\Lambda}_t$, can be extracted from the sparse DFM fit using:

```
# Estimated series 1 from sparse DFM
series <- 1
estimate <- fit.sdfm$data$fitted.unscaled[,series]
```

This estimate is re-scaled back to the original series 1 scale. To get a plot of the estimated factors, use the following:

```
# Plot estimated factors
plot(fit.sdfm, type = "factor")
```

To visualise the estimated loadings it is possible to make line plots (as seen in Chapter 4) and also a heatmap using:



```
# Make a heatmap of the estimated loadings
plot(fit.sdfm, type = "loading.heatmap")
```

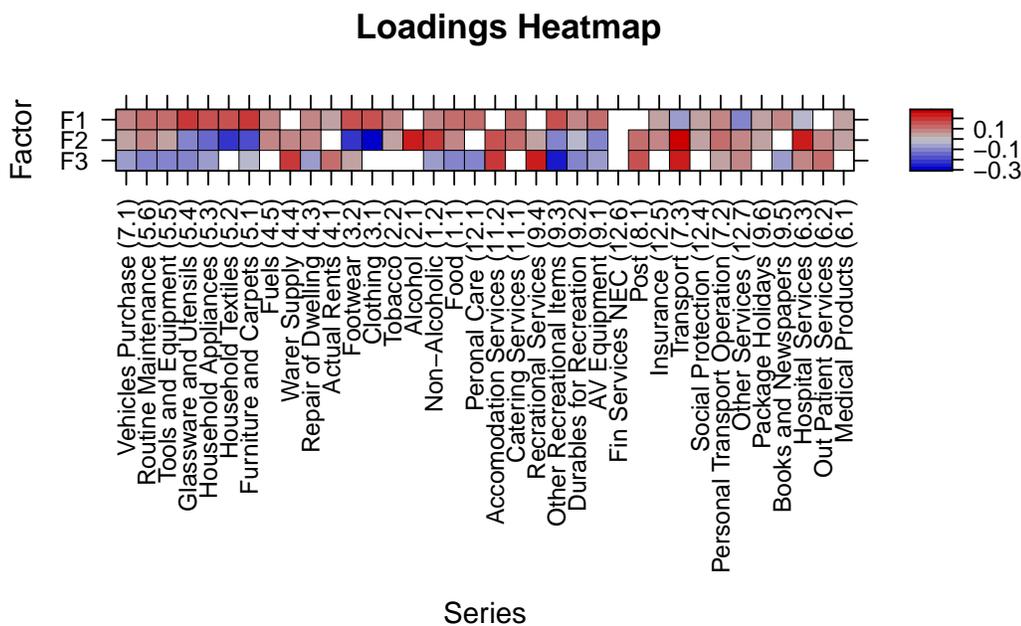


Figure 6.2.1: Heatmap of the estimated factor loadings in the inflation empirical example.

which shows the presence of sparse loadings with white squares (see Figure 6.2.1). It is possible to set `use.series.names = FALSE` in the plots to use numbers instead of long variable names.

Chapter 7

Conclusions and Further Work

Temporal disaggregation and nowcasting are both imperative techniques for the production and understanding of official statistics. In today's rapidly evolving economy, the abundance of administrative and alternative high-frequency data sources necessitates the extension of these techniques to handle high-dimensional data. This thesis has introduced innovative methodologies with the endeavour to achieve exactly this. In closing, a summary of the key findings from each chapter is provided, accompanied by a thoughtful discussion on the remaining open challenges and potential avenues for future research. While the work presented in this thesis is primarily empirical in nature, areas for theoretical exploration are also identified.

Chapter 3 introduced a novel Sparse Temporal Disaggregation (spTD) framework that extends the well-established Chow-Lin family (Chow and Lin, 1971) of temporal disaggregation methods to operate in settings where the number of indicators used can exceed the low-frequency sample size. A key aspect of the framework is the incorporation of a regulariser penalty function on the regression parameters which operates alongside the usual GLS cost function. Particular focus is put on the popular ℓ_1 (LASSO) penalty (Tibshirani, 1996) due to its benefits in terms of interpretability and computational robustness. Additionally, the adaptive LASSO (Zou, 2006) weighting

scheme is presented as an effective approach for handling correlated indicator series that violate the irrepresentability condition (Bühlmann and Van De Geer, 2011). To find estimates, the efficient LARS algorithm of Efron et al. (2004) is employed to provide an initial screening of indicator series by constructing unique solution paths with varying sparsity levels controlled by a BIC tuning routine (Schwarz, 1978; Reid et al., 2016). An additional re-fitting procedure is then used to reduce the bias in the LARS solution paths.

The benefits of the proposed method are explored through extensive simulation studies and empirical applications to real-data. The simulation studies reveal the superior performance of spTD compared to the traditional Chow-Lin method across multiple aspects, including accuracy in estimating the true disaggregate series, recovery of sparse regression coefficients, and recovery of AR(1) residual parameters. Additionally, the re-fit procedure, employed to mitigate bias in LASSO parameter estimates, further enhances the accuracy of spTD. Applying spTD to the quarterly-to-monthly disaggregation of UK GDP data permitted the inclusion of alternative indicator sources, facilitating the interpretation of their relevance in generating a monthly GDP series. Unlike previous approaches, no restriction is placed on the number we can include. Furthermore, an additional application involving the derivation of the quarterly service turnover index in Italy demonstrates the improved performance of spTD over competing methods typically used for this problem. This application highlights the potential stability and accuracy gains associated with working with a fixed support of indicators.

The generality of the proposed GLS penalisation framework allows regularisation functions that are able to take a variety of forms. Future work may look to extend the analysis to include more informative structured priors. For instance, the exploration of the group LASSO (Yuan and Lin, 2006) could allow for automatic selection of groups of related indicators, instead of individual indicators, based on pre-identified economic factors. Additionally, the fused LASSO (Tibshirani et al., 2005) could be leveraged to

promote smoothness between estimates by penalising the ℓ_1 -norm of both the coefficients and their successive differences. This regularisation approach may effectively preserve crucial patterns and structures within the data. The latter may help encourage more stable estimates, a property we found beneficial in the disaggregation of Italian turnover data in Section 3.6. Similarly, incorporating the stability selection subsampling approach by Meinshausen and Bühlmann (2010) could contribute to obtaining stable solutions for the selected indicators within the spTD framework.

While the chapter provided empirical motivations for combining a GLS cost function with the LASSO for the purpose of high-dimensional temporal disaggregation, future work could also explore the theoretical motivations. To date, theoretical work for GLS in the high-dimensional setting has received little attention. However, the recent paper of Chronopoulos et al. (2022) is the first to establish asymptotic normality results for their debiased GLS LASSO estimator. Additionally, Kock (2016) shows that the adaptive LASSO is oracle efficient in stationary and non-stationary autoregressions but uses more relaxed assumptions than Chronopoulos et al. (2022). This theoretical analysis of post-selection inference could be used to establish confidence intervals on the regression coefficients (under regularisation) and thus provide confidence intervals for the estimated disaggregate series. Currently, confidence intervals for the spTD methodology just consider variance of the AR(1) residual error and ignore the variance in regression coefficients. Mapping these GLS LASSO consistency results into the temporal disaggregation application is an interesting direction for future work.

Chapter 4 formally introduced a class of sparse dynamic factors models (DFMs) whereby the loading matrices in a regular DFM are constrained to have few non-zero entries. This parsimonious approach provides greater interpretability, unraveling the underlying latent factor structure. Unlike previous research, the methodology presented implements regularisation within an EM algorithm framework, enabling robust handling of missing data patterns, modelling of temporal dependence in the

processes, and imposition of weakly informative (sparse) prior knowledge on the factor loadings. Synthetic experiments showed that in settings where autocorrelation between factors is moderately persistent, the feedback mechanism inherent in the EM procedure plays a vital role in accurately recovering the factor loadings and producing precise forecasts.

To demonstrate the models versatility, the sparse DFM is applied to two large datasets from different domains. The first an econometrics application nowcasting UK trade in exported goods, offering a clear economic interpretation of the estimated factors that a regular DFM struggled to achieve. The second an application into modelling energy consumption around a University campus. Utilising the sparsity-inducing framework, it became possible to identify distinctive building profiles, such as those predominantly active during nighttime, aligning seamlessly with our qualitative intuition. Furthermore, a forecasting exercise on out-of-sample data, whether 1 or 6 hours ahead, revealed the superior accuracy of the sparse DFM over alternative methods.

One of the advantages of the ℓ_1 regularised EM algorithm is that it can be easily extended to impose alternative restrictions on parameters, or indeed to encode different types of sparsity assumptions. Two potential extensions could be to relax the assumption that the factor loadings remain constant over time, or adopt a group LASSO type regularisation on the loadings. The former strategy would employ time-varying factor loadings that have been shown to be more robust when the underlying latent structure changes over time (Del Negro and Otrok, 2008; Breitung and Eickmeier, 2011). The latter would enable users to associate factors with pre-defined sub-sets of the observed series, but still in a somewhat data-driven manner. For instance, in the context of trade in exported goods, grouping the series based on countries could provide a comprehensive breakdown of countries to which the UK exports. Similarly, the energy application of Case Study 2 could consider grouping the series via type of building and encouraging sparsity at this grouped level, rather than at the building level. This could be particularly

useful if we consider the application to smart meters at the sub-building, e.g. floor-by-floor, or room-by-room level. One of the benefits of the ADMM optimisation routine developed in this work is that it easily extended to these settings.

Ongoing research involves extending the proposed sparse DFM to incorporate weakly informative group-sparse priors on factor loadings in a neuroscience application modeling resting state fMRI (functional magnetic resonance imaging) activity across a set of subjects. The interest lies in whether using sparse groupings associated with anatomical labels from regions of interest around the brain improve the ability to predict and/or identify common components in resting state blood flow around the brain activity, and quantify how this evolves when changing tasks.

Theoretical analysis of the proposed estimation routine is challenging for several reasons. First, one would need to decide whether to analyse the theoretical minimiser (QMLE), or the feasible estimate provided by the EM algorithm. Second, we need to consider the performance as a function of both n and p . For example, Proposition 2 from Barigozzi and Luciani (2022) gives theoretical results for the consistency of factor loadings for the regular unregularised QMLE and for a dense DFM model. A further line of work would be to generalise these results to the sparse DFM setting, for instance, can we show a result analogous to Theorem 1 in Bai and Li (2016), that shows the QMLE estimator of the loadings is equivalent to the OLS estimator applied to the true factors? The recent paper of Despois and Doz (2023) investigates consistency results of the sparse PCA. It is of interest to make a formal theoretical comparison between the sparse PCA based approaches and the QMLE approach presented here.

Chapter 5 extends the regularised EM algorithm framework of the sparse DFM to address the modelling of mixed-frequency time series data. This extended framework enables exact temporal disaggregation of lower-frequency variables by incorporating a cumulator variable (Harvey, 1990). The cumulator variable systematically samples latent disaggregate targets over each low-frequency period to ensure temporal aggregation

constraints are met. Disaggregate targets are treated as latent states in the joint state space model and can be explicitly estimated by the univariate Kalman filter and smoother equations (Koopman and Durbin, 2000). A synthetic study revealed the new framework is able to compete in terms of temporal disaggregation with spTD while accurately capturing the underlying factor structure when the data is assumed to follow a sparse DFM. The model provided an efficient way for the monthly estimation of the 11 main subdivisions of the service sector the UK imports and exports worldwide. Practitioners can incorporate a wide range of indicators into the model to assess estimation robustness, and the induced sparse loadings provide an easily interpretable measure of indicator relevance. The model's monthly estimates of total trade in service are compared against the actual published figures from the ONS, revealing very similar dynamics.

The state-space framework proposed in Chapter 5 lays the foundations for a versatile way to jointly perform temporal disaggregation and nowcasting of high-dimensional mixed-frequency data. This flexible framework opens up numerous possibilities for future extensions and advancements. The assumption of white noise errors for the latent disaggregate series regression onto the factors (Equation 5.3.3) is fairly restrictive. This naive assumption is likely to be a reason why our monthly estimates for the service subdivisions are so smooth and align so well with the ONS published total figures (as they use a smoothing-based benchmark method for estimation). Errors could be modelled as AR(1) processes by adding them as latent states following methodology laid out in Bańbura and Modugno (2014), however, this approach would slow down computation of the Kalman filter and smoother equations significantly as highlighted in Appendix D.1. An appealing alternative, proposed by Jungbacker et al. (2009), is to employ a switching method that dynamically switches between the Bańbura and Modugno (2014) approach and the Reis and Watson (2010) approach. The latter models the autoregressive component by including lags of the variable in the measurement equation, but only if the variable is available at time t . When the amount of missing

data is small, this can lead to substantial computational gains.

As we are modelling a multivariate output of disaggregate targets, it is worth considering potential interdependencies among targets and the necessary constraints that the cross-section must satisfy. To address the interdependencies, one approach is to assume a stationary VAR(1) model for the error process. This means that each target residual may interact linearly with both its own past values as well as those of every other included series; allowing more complex inter-temporal correlations among targets. Given the number of parameters grows quadratically with the number of series included in the model, it may be beneficial to induce sparsity into the VAR(1) matrix. An insightful paper by Hecq et al. (2022) incorporates a regulariser that permits hierarchical sparsity patterns in the VAR(1) matrix. This regulariser prioritises the inclusion of coefficients based on the recency of the information they contain. Leveraging a similar strategy in the joint model of Chapter 5 could yield significant advantages.

To address the cross-sectional additive constraints among the targets, we can augment the state space framework by including an additional cumulator variable. This variable would sum contemporaneously across targets to match an aggregate figure. For example, in the context of subdivisions of services and commodities, we can constrain them to equal the total trade figure. Moreover, given recent interest in producing more disaggregate estimates for regional economies in the UK (Koop et al., 2020b), the framework proposed in this chapter with an extension for contemporaneous constraints offers an alternative approach to ensure that the regional GDP figures sum up to the national GDP figure.

The chapter concluded with an intervention study using an ITSM that provided interpretable measures on the Brexit effects on UK-EU trade. There is room for refinement and enhancement of the ITSM approach. For example, multi-group analysis (Linden, 2015) can be conducted on the different commodities the UK trades to see if certain goods have been effected more. Similarly, the multi-group analysis can be

conducted on the different subdivisions of the service sector. Also, multi-intervention analysis (Bernal et al., 2017) may be conducted using multiple dates to test significance of change at, e.g. the Brexit referendum in June 2016 or the start of the coronavirus pandemic in March 2020. Analysis should be conducted again in the next few years to get a more accurate assessment of long-run effects.

To conclude, worldwide events in the past few years has revealed the importance of possessing a timely and comprehensive dataset that encompasses all facets of our economy. The research into developing robust statistical models that can accurately estimate the current state of the economy is ever growing, with a vast amount of open problems still to consider. The methodologies and frameworks presented in this thesis aim to make significant contributions to this pivotal field. By integrating high-dimensional statistics into the domains of temporal disaggregation and nowcasting, with a focus on interpretability, this research endeavors to facilitate transparency, accuracy, and a better understanding of policy effects and shocks within the dynamic landscape of today's economy.

Appendix A

Appendix to Chapter 2

A.1 Quadratic Optimisation with a Linear Equality Constraint for Temporal Disaggregation

Define a general quadratic minimisation problem with a linear equality constraint as

$$\begin{aligned} \min_{\mathbf{y}} \quad & f(\mathbf{y}) = (\mathbf{y} - \mathbf{z})^\top \mathbf{B}(\mathbf{y} - \mathbf{z}) \\ \text{s.t.} \quad & \bar{\mathbf{y}} = \mathbf{C}\mathbf{y}, \end{aligned}$$

where it is assumed that

- $\mathbf{y} \in \mathbb{R}^{n \times 1}$ is unknown,
- $\mathbf{z} \in \mathbb{R}^{n \times 1}$ and $\bar{\mathbf{y}} \in \mathbb{R}^{n_i \times 1}$ are known,
- $\mathbf{B} \in \mathbb{R}^{n \times n}$ is a known symmetric, non-singular matrix to ensure the objective function is convex and has a unique minimum,
- $\mathbf{C} \in \mathbb{R}^{n_i \times n}$ is a known constraint matrix with full row rank to ensure the constraint function is well-defined with a unique solution for \mathbf{y} that satisfies the constraint.

The solution of the constrained quadratic minimisation problem can be solved by minimising the Lagrangian function defined by

$$\begin{aligned}\mathcal{L}(\mathbf{y}, \boldsymbol{\lambda}) &= f(\mathbf{y}) - 2\boldsymbol{\lambda}^\top(\bar{\mathbf{y}} - \mathbf{C}\mathbf{y}) \\ &= \mathbf{y}^\top \mathbf{B}\mathbf{y} + \mathbf{z}^\top \mathbf{B}\mathbf{z} - 2\mathbf{y}^\top \mathbf{B}\mathbf{z} - 2\boldsymbol{\lambda}^\top(\bar{\mathbf{y}} - \mathbf{C}\mathbf{y}),\end{aligned}$$

with respect to \mathbf{y} and the vector of Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^{n_i \times 1}$. Taking partial derivatives of $\mathcal{L} := \mathcal{L}(\mathbf{y}, \boldsymbol{\lambda})$ with respect to \mathbf{y} and $\boldsymbol{\lambda}$ and setting equal to zero, we obtain the equations:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{y}} = 2\mathbf{B}\mathbf{y} - 2\mathbf{B}\mathbf{z} + 2\mathbf{C}^\top \boldsymbol{\lambda} = 0; \quad (\text{A.1.1})$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} = 2\mathbf{C}\bar{\mathbf{y}} - 2\mathbf{y} = 0. \quad (\text{A.1.2})$$

Since \mathbf{B} is invertible, equation (A.1.1) can be re-arranged to obtain

$$\mathbf{y} = \mathbf{z} - \mathbf{B}^{-1}\mathbf{C}^\top \boldsymbol{\lambda}. \quad (\text{A.1.3})$$

This can be substituted into equation (A.1.2) and re-arranged to obtain the solution for $\boldsymbol{\lambda}$:

$$\boldsymbol{\lambda} = (\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top)^{-1}(\mathbf{C}\mathbf{z} - \bar{\mathbf{y}}),$$

where it has been assumed that $\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top$ is invertible. Substituting the solution for $\boldsymbol{\lambda}$ into (A.1.3) we obtain the solution for \mathbf{y} as

$$\mathbf{y} = \mathbf{z} + \mathbf{B}^{-1}\mathbf{C}^\top (\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top)^{-1}(\bar{\mathbf{y}} - \mathbf{C}\mathbf{z}).$$

This is the general solution for the estimate of the latent disaggregate series found in the temporal disaggregation methods discussed in Chapter 2.1. Specifically, the methods differ according to:

- Boot et al. (1967) smoothing assume $\mathbf{z} = 0$ (no indicators) and $\mathbf{B} = \mathbf{D}^\top \mathbf{D}$, where \mathbf{D} is the matrix of the first-differences defined in equation (2.1.2);
- Stram and Wei (1986) smoothing assume $\mathbf{z} = 0$ (no indicators) and $\mathbf{B} = \mathbf{D}^\top \mathbf{V}_d^{-1} \mathbf{D}$, where \mathbf{V}_d is the covariance matrix of first-difference process $\{y_t - y_{t-1}\}$ that is assumed to follow an ARIMA process;
- Denton (1971) smoothing assume $\mathbf{z} = \mathbf{x}$ (a single indicator) and $\mathbf{B} = \mathbf{D}^\top \mathbf{D}$ for the additive variant or $\mathbf{B} = \tilde{\mathbf{Z}}^{-1} \mathbf{D}^\top \mathbf{D} \tilde{\mathbf{Z}}^{-1}$ for the proportional variant where $\tilde{\mathbf{Z}} = \text{diag}(\mathbf{z})$;
- Chow and Lin (1971) family of regression methods assume $\mathbf{z} = \mathbf{X}\boldsymbol{\beta}$ (multiple indicators $\mathbf{X} \in \mathbb{R}^{m \times p}$ and generalised least-squares estimator $\boldsymbol{\beta} \in \mathbb{R}^{p \times 1}$) and $\mathbf{B} = \mathbf{V}^{-1}$, where \mathbf{V} is the covariance matrix of the residual process $\{y_t - \boldsymbol{\beta}^\top \mathbf{X}_t\}$ that is assumed to follow an AR(1) process in Chow and Lin (1971), a random walk process in Fernandez (1981), and an ARIMA(1,1,0) process in Litterman (1983).

A.2 Multivariate and Univariate Kalman Filter and Smoother Equations

This appendix provides the classic multivariate Kalman filter and smoother (KFS) equations of Shumway and Stoffer (1982) and the fast univariate treatment KFS equations of Koopman and Durbin (2000) that can be used to estimate the mean and covariance of the latent states in a state space framework based on observations of the system. The state space framework of focus in this thesis is the dynamic factor model (DFM) given

by equations

$$\mathbf{X}_t = \mathbf{\Lambda}\mathbf{F}_t + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon), \quad (\text{A.2.1})$$

$$\mathbf{F}_t = \mathbf{A}\mathbf{F}_t + \mathbf{u}_t, \quad \mathbf{u}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_u), \quad (\text{A.2.2})$$

as introduced in Chapter 2.2.4, where the latent states are the factors, \mathbf{F}_t . Define the following notation for the conditional mean and covariances of the state:

$$\mathbf{a}_{t|s} = \mathbb{E}[\mathbf{F}_t | \boldsymbol{\Omega}_s],$$

$$\mathbf{P}_{t|s} = \text{Var}(\mathbf{F}_t | \boldsymbol{\Omega}_s),$$

$$\mathbf{P}_{t,t-1|s} = \text{Cov}(\mathbf{F}_t, \mathbf{F}_{t-1} | \boldsymbol{\Omega}_s),$$

conditional on all information we have observed up to a time s , denoted by $\boldsymbol{\Omega}_s$.

Filtering aims to make recursive predictions of the mean and covariance of the state at t given information up to $t - 1$, i.e. finds $\mathbf{a}_{t|t-1}$ and $\mathbf{P}_{t|t-1}$ starting from $t = 1$ to $t = n$. Smoothing then works backwards in time to find smoothed estimates of the mean and covariance of the state given all information, i.e. finds $\mathbf{a}_{t|n}$ and $\mathbf{P}_{t|n}$ starting from $t = n$ to $t = 1$.

A.2.1 Multivariate KFS Equations - Shumway and Stoffer (1982)

The classic approach to filtering and smoothing found in Shumway and Stoffer (1982) and detailed in Harvey (1990) is based on considering the contribution of every variable in the observational vector at each successive time point, therefore this is described as a multivariate approach..

Assuming the set of all parameters $\boldsymbol{\theta} = (\mathbf{\Lambda}, \mathbf{A}, \boldsymbol{\Sigma}_\epsilon, \boldsymbol{\Sigma}_u)$ in (A.2.1)-(A.2.2) and initial

conditions $\mathbf{a}_{0|0}$ and $\mathbf{P}_{0|0}$ are known, the filtering equations for $t = 1, \dots, n$ are

$$\mathbf{a}_{t|t-1} = \mathbf{A}\mathbf{a}_{t-1|t-1} \quad (\text{predicted factor estimate})$$

$$\mathbf{P}_{t|t-1} = \mathbf{A}\mathbf{P}_{t-1|t-1}\mathbf{A}^\top + \Sigma_u \quad (\text{predicted factor covariance})$$

$$\mathbf{v}_t = \mathbf{X}_t - \Lambda\mathbf{a}_{t|t-1} \quad (\text{innovation error})$$

$$\mathbf{C}_t = \Lambda\mathbf{P}_{t|t-1}\Lambda^\top + \Sigma_\epsilon \quad (\text{innovation covariance})$$

$$\mathbf{K}_t = \mathbf{P}_{t|t-1}\Lambda^\top\mathbf{C}_t^{-1} \quad (\text{Kalman gain})$$

$$\mathbf{a}_{t|t} = \mathbf{a}_{t|t-1} + \mathbf{K}_t\mathbf{v}_t \quad (\text{updated factor estimate})$$

$$\mathbf{P}_{t|t} = \mathbf{P}_{t|t-1} - \mathbf{K}_t\Lambda\mathbf{P}_{t|t-1} \quad (\text{updated factor covariance})$$

and the smoothing equations for $t = n, \dots, 1$ are

$$\mathbf{J}_{t-1} = \mathbf{P}_{t-1|t-1}\mathbf{A}^\top(\mathbf{P}_{t|t-1})^{-1} \quad (\text{smoother gain})$$

$$\mathbf{a}_{t-1|n} = \mathbf{a}_{t-1|t-1} + \mathbf{J}_{t-1}(\mathbf{F}_{t|n} - \mathbf{A}\mathbf{a}_{t-1|t-1}) \quad (\text{smoothed factor estimate})$$

$$\mathbf{P}_{t-1|n} = \mathbf{P}_{t-1|t-1} + \mathbf{J}_{t-1}(\mathbf{P}_{t|n} - \mathbf{P}_{t|t-1})\mathbf{J}_{t-1}^\top \quad (\text{smoothed factor covariance})$$

where $\mathbf{a}_{n|n}$ and $\mathbf{P}_{n|n}$ are initialised based on the last filter estimates. At any stage where an observation $X_{t,i}$ is missing at time t , this variable is omitted from the calculation of the Kalman gain \mathbf{K}_t . The lagged-covariance matrix $\mathbf{P}_{t,t-1|n}$ is found using backwards recursions on:

$$\mathbf{P}_{t-1,t-2|n} = \mathbf{P}_{t-1|t-1}\mathbf{J}_{t-2}^\top + \mathbf{J}_{t-1}(\mathbf{P}_{t,t-1|n} - \mathbf{A}\mathbf{P}_{t-1|t-1})\mathbf{J}_{t-2}^\top,$$

for $t = n, \dots, 2$, where

$$\mathbf{P}_{n,n-1|n} = (\mathbf{I} - \mathbf{K}_n\Lambda)\mathbf{A}\mathbf{P}_{n-1|n-1}.$$

The filtering equations require the covariance of innovation errors \mathbf{C}_t to be inverted.

This can be difficult when p is large. Harvey (1990) show this inversion can be made easier by making use of the Woodbury identity and find:

$$\mathbf{C}_t^{-1} = \boldsymbol{\Sigma}_\epsilon^{-1} - \boldsymbol{\Sigma}_\epsilon^{-1} \boldsymbol{\Lambda} (\mathbf{P}_{t|t-1}^{-1} + \boldsymbol{\Lambda}^\top \boldsymbol{\Sigma}_\epsilon^{-1} \boldsymbol{\Lambda})^{-1} \boldsymbol{\Lambda}^\top \boldsymbol{\Sigma}_\epsilon^{-1}, \quad (\text{A.2.3})$$

which is valid when $\boldsymbol{\Sigma}_\epsilon$ and $\mathbf{P}_{t|t-1}$ are non-singular. This inversion is easy to evaluate if $\boldsymbol{\Sigma}_\epsilon$ is diagonal (as assumed in exact factors models). In cases when p is much larger than r , as is the case in the majority of DFM problems, the inversion of $\mathbf{P}_{t|t-1}$ is a much easier operation than the inversion of \mathbf{C}_t directly.

Harvey (1990) further show that the determinant of \mathbf{C}_t can be written as:

$$|\mathbf{C}_t| = |\boldsymbol{\Sigma}_\epsilon| \times |\mathbf{P}_{t|t-1}| \times |\mathbf{P}_{t|t-1}^{-1} + \boldsymbol{\Lambda}^\top \boldsymbol{\Sigma}_\epsilon^{-1} \boldsymbol{\Lambda}|. \quad (\text{A.2.4})$$

Both (A.2.3) and (A.2.4) are used in the calculation of the log-likelihood of the innovations:

$$\log \mathcal{L}(\mathbf{X}; \boldsymbol{\theta}) = -\frac{1}{2} \sum_{t=1}^n (p_t \log(2\pi) + \log |\mathbf{C}_t| + \mathbf{v}_t \mathbf{C}_t^{-1} \mathbf{v}_t) .$$

where p_t is the number of variables observed (not-missing) at t . Note, this log-likelihood is used to determine convergence of the EM algorithm approach of Bańbura and Modugno (2014).

A.2.2 Univariate KFS Equations - Koopman and Durbin (2000)

The classic KFS equations can be slow when p is large. Since it is assumed $\boldsymbol{\Sigma}_\epsilon$ is diagonal, we can equivalently filter the observations \mathbf{X}_t one element at a time, as opposed to all together as in the classic approach (Durbin and Koopman, 2012; Koopman and Durbin, 2000). As matrix inversion becomes scalar divisions, huge speedups are possible. This approach can be described as a univariate treatment or sequential processing of the classic multivariate equations.

Define the individual elements $\mathbf{X}_t = (X_{t,1}, \dots, X_{t,p})^\top$, $\mathbf{\Lambda} = (\mathbf{\Lambda}_1^\top, \dots, \mathbf{\Lambda}_p^\top)^\top$, $\mathbf{\Sigma}_\epsilon = \text{diag}(\sigma_{\epsilon_1}^2, \dots, \sigma_{\epsilon_p}^2)$ and define new notation similar to that found in Durbin and Koopman (2012):

$$\begin{aligned}\mathbf{a}_{t,i} &= \mathbb{E}[\mathbf{F}_{t,i} | \mathbf{\Omega}_{t-1}, X_{t,1}, \dots, X_{t,i-1}], \\ \mathbf{a}_{t,1} &= \mathbb{E}[\mathbf{F}_{t,1} | \mathbf{\Omega}_{t-1}], \\ \mathbf{P}_{t,i} &= \text{Var}[\mathbf{F}_{t,i} | \mathbf{\Omega}_{t-1}, X_{t,1}, \dots, X_{t,i-1}], \\ \mathbf{P}_{t,1} &= \text{Var}[\mathbf{F}_{t,1} | \mathbf{\Omega}_{t-1}],\end{aligned}$$

for $i = 1, \dots, p$ and $t = 1, \dots, n$.

The univariate treatment (Koopman and Durbin, 2000) iterates the below equations. These update equations are equivalent in form to the multivariate ones of the classic Shumway and Stoffer (1982) approach except that the t subscript now becomes a t, i subscript, and the $t|t$ subscript now becomes $t, i + 1$. This means that we are filtering spatiotemporally, processing one i after another in a sequential manner, instead of updating all p measurements at once:

$$\begin{aligned}v_{t,i} &= X_{t,i} - \mathbf{\Lambda}_i \mathbf{a}_{t,i}, \\ C_{t,i} &= \mathbf{\Lambda}_i \mathbf{P}_{t,i} \mathbf{\Lambda}_i^\top + \sigma_{\epsilon,i}^2, \\ \mathbf{K}_{t,i} &= \mathbf{P}_{t,i} \mathbf{\Lambda}_i^\top C_{t,i}^{-1}, \\ \mathbf{a}_{t,i+1} &= \mathbf{a}_{t,i} + \mathbf{K}_{t,i} v_{t,i}, \\ \mathbf{P}_{t,i+1} &= \mathbf{P}_{t,i} - \mathbf{K}_{t,i} C_{t,i} \mathbf{K}_{t,i}^\top,\end{aligned}$$

for $i = 1, \dots, p$ and $t = 1, \dots, n$. If $X_{t,i}$ is missing or $C_{t,i}$ is zero, omit the term containing

$\mathbf{K}_{t,i}$. The transition to $t + 1$ is given by the following prediction equations:

$$\begin{aligned}\mathbf{a}_{t+1,1} &= \mathbf{A}\mathbf{a}_{t,p+1}, \\ \mathbf{P}_{t+1,1} &= \mathbf{A}\mathbf{P}_{t,p+1}\mathbf{A}^\top + \Sigma_{\mathbf{u}}.\end{aligned}$$

These prediction equations are exactly the same as the multivariate ones (i.e., predictions are not treated sequentially but all at once). Univariate treatment is perhaps better called univariate updates plus multivariate predictions.

Unlike Shumway and Stoffer (1982), the measurement update comes before the transition; however, is it possible to revert to doing the transition first if the initial state means and covariances start from $t = 0$ instead of $t = 1$. Likewise, univariate smoothing is defined by:

$$\begin{aligned}\mathbf{L}_{t,i} &= \mathbf{I}_m - \mathbf{K}_{t,i}\mathbf{\Lambda}_i, \\ \mathbf{r}_{t,i-1} &= \mathbf{\Lambda}_i^\top C_{t,i}^{-1}v_{t,i} + \mathbf{L}_{t,i}^\top \mathbf{r}_{t,i}, \\ \mathbf{N}_{t,i-1} &= \mathbf{\Lambda}_i^\top C_{t,i}^{-1}\mathbf{\Lambda}_i + \mathbf{L}_{t,i}^\top \mathbf{N}_{t,i}\mathbf{L}_{t,i}, \\ \mathbf{r}_{t-1,p} &= \mathbf{A}^\top \mathbf{r}_{t,0}, \\ \mathbf{N}_{t-1,p} &= \mathbf{A}^\top \mathbf{N}_{t,0}\mathbf{A},\end{aligned}$$

for $i = p, \dots, 1$ and $t = n, \dots, 1$, with $\mathbf{r}_{n,p}$ and $\mathbf{N}_{n,p}$ initialised to 0. Again, if $X_{t,i}$ is missing or $C_{t,i}$ is zero, drop the terms containing $\mathbf{K}_{t,i}$. Finally, the equations for $\mathbf{a}_{t|n}$ and $\mathbf{P}_{t|n}$ are:

$$\begin{aligned}\mathbf{a}_{t|n} &= \mathbf{a}_{t,1} + \mathbf{P}_{t,1}\mathbf{r}_{t,0}, \\ \mathbf{P}_{t|n} &= \mathbf{P}_{t,1} - \mathbf{P}_{t,1}\mathbf{N}_{t,0}\mathbf{P}_{t,1}.\end{aligned}$$

These results will be equivalent to $\mathbf{a}_{t|n}$ and $\mathbf{P}_{t|n}$ from the classic multivariate approach,

yet obtained with substantial improvement in computational efficiency.

The log-likelihood for the above filter (Durbin and Koopman, 2012; Koopman and Durbin, 2000) is given below:

$$\begin{aligned} \log \mathcal{L}(\mathbf{X}; \boldsymbol{\theta}) &= -\frac{1}{2} \sum_{t=1}^n \sum_{i=1}^p \iota_{t,i} \left(\log(2\pi) + \log |C_{t,i}| + v_{t,i}^\top C_{t,i}^{-1} v_{t,i} \right) \\ &= -\frac{1}{2} \sum_{t=1}^n \sum_{i=1}^p \iota_{t,i} \left(\log(2\pi) + \log C_{t,i} + \frac{v_{t,i}^2}{C_{t,i}} \right), \end{aligned} \quad (\text{A.2.5})$$

as $C_{t,i}$ and $v_{t,i}$ are scalars. Here, $\iota_{t,i}$ is zero if $X_{t,i}$ is missing or $C_{t,i}$ is zero, or one otherwise. This log-likelihood (A.2.5) is used to determine convergence of the EM algorithm in the sparse DFM framework of Chapter 4.

Appendix B

Appendix to Chapter 3

B.1 Indicator Data Description for Section 3.6

This appendix presents a detail of the 143 predictors adopted for Case Study 2 of disaggregating the quarterly index of service turnover in Italy. The full data set is divided into three sets representative of the three main sources: *Monthly Business Survey (BS)*, *Asso-Aeroporti (AA)*, *Industrial Turnover Indices (IT)*. These are found in the subsequent tables: Table B.1.1, B.1.2 and B.1.3.

B.2 Indicators Selected in stable ℓ_1 -spTD for Section 3.6

This appendix shows the selection results for fixing the number of predictors equal to 8 and 24 in ℓ_1 -spTD. The two parameters are in line with the minimum and the maximum number of indicators selected in one time and occurred respectively in 2016:Q4 and 2021:Q2. Tables B.2.1 and B.2.2 respectively show the selected indicators across the 21 disaggregation periods and include those selected at least 5 times. Figures B.2.1 and B.2.2 then visualise this with red = BS, yellow = AA, orange = IT data sets.

Table B.1.1: Monthly Business Survey - The Table reports 56 series downloaded related to Business survey (BS) in transportation and storage (NACE H) and land transport and transport via pipelines (NACE H49): balance (s) and frequency of reply marked as improved/increased (a), remained unchanged (n) and deteriorated (b) on 7 questions.

sh_a/ah_a/nh_a/ bh_a	business situation development over the past 3 months
sh49_a/ah49_a/nh49_a/bh49_a	balance/ improved /remained unchanged/deteriorated
sh_d/ah_d/nh_d/bh_d	evolution of the demand over the past 3 months
sh49_d/ah49_d/nh49_d/bh49_d	balance/increased/remained unchanged/decreased
sh_p/ah_p/nh_p/bh_p	expectations of the demand over the next 3 months
sh49_p/ah49_p/nh49_p/bh49_p	balance/increased/remained unchanged/decreased
sh_o/ah_o/nh_o/bh_o	expectations of the employment over the next 3 months
sh49_o/ah49_o/nh49_o/bh49_o	balance/increased/remained unchanged/decreased
sh_z/ah_z/nh_z/bh_z	expectations of the prices over the next 3 months
sh49_z/ah49_z/nh49_z/bh49_z	balance/increased/remained unchanged/decreased
sh_g/ah_g/nh_g/bh_g	evolution of the employment over the past 3 months
sh49_g/ah49_g/nh49_g/bh49_g	balance/increased/remained unchanged/decreased
sh_i/ah_i/nh_i/bh_i	expectations of the Italian economic situation over the next 3 months
sh49_i/ah49_i/nh49_i/bh49_i	balance/increased/remained unchanged/decreased

Table B.1.2: Asso-Aeroporti - The Table reports 18 monthly series downloaded from Assaeroporti (AA).

air_mcnz	national commercial flight movements
air_mcin	international commercial flight movements
air_mcue	intra-EU (Switzerland and UK included) commercial flight movements
air_mct	total commercial flight movements (air_mcnz+ air_mcin)
air_mcav	general aviation and other movements
air_mctt	total commercial flight movements (air_mcnz+air_mcin+air_mcav)
air_pnz	national passengers
air_pin	international passengers
air_pue	intra-EU (Switzerland and UK included) passengers
air_ptr	transit passengers
air_pt	total commercial passengers (air_pnz+air_pin+air_pue)
air_pav	general aviation and other passengers
air_ptt	total passengers
air_cma	avio cargo goods
air_cms	surface cargo goods
air_cmt	total cargo goods
air_cpt	mail cargo
air_ctt	total cargo

Table B.1.3: Turnover in Industry The Table reports all 69 monthly series representative of the Industrial Turnover Indices (IT), The list includes the tern total/domestic/foreign turnover

fit_int/fid_int/fie_int	intermediate goods
fit_str/fid_str/fie_str	instrumental goods
fit_cdu/fid_cdu/fie_cdu	durable goods
fit_cnd/fid_cnd/fie_cnd	non-durable goods
fit_cto/fid_cto/fie_cto	consumer goods
fit_ene/fid_ene/fie_ene	energy goods
fit_ind/fid_ind/fie_ind	total industry (NACE B to E)
fit_man/fid_man/fie_man	total industry less energy (NACE B to E less energy)
fit_B/fid_B/fie_B	mining and quarrying (NACE B)
fit_C/fid_C/fie_C	manufacturing (NACE C)
fit_CA/fid_CA/fie_CA	food, beverages and tobacco (NACE CA)
fit_CB/fid_CB/fie_CB	textiles, apparel and leather (NACE CB)
fit_CC/fid_CC/fie_CC	wood, paper and printing (NACE CC)
fit_CD/fid_CD/fie_CD	coke, and refined petroleum (NACE CD)
fit_CE/fid_CE/fie_CE	chemicals (NACE CE)
fit_CF/fid_CF/fie_CF	pharmaceuticals (NACE CF)
fit_CG/fid_CG/fie_CG	plastics non-metallic (NACE CG)
fit_CH/fid_CH/fie_CH	metals (NACE CH)
fit_CI/fid_CI/fie_CI	computers and electronic (NACE CI)
fit_CJ/fid_CJ/fie_CJ	electrical products (NACE CJ)
fit_CK/fid_CK/fie_CK	machinery and equipment (NACE CK)
fit_CL/fid_CL/fie_CL	transport equipment (NACE CL)
fit_CM/fid_CM/fie_CM	other manufacturing (NACE CM)

Indicator	Dataset	N. of times selected	Indicator	Dataset	N. of times selected
fit_CA (23)	IT	21	fie_CK (29)	IT	8
fie_str (14)	IT	14	bh49_d (8)	BS	7
air_13 (21)	AA	14	air_35 (20)	AA	6
air_27 (18)	AA	12	sh_a (1)	BS	5
ah49_z (10)	BS	9	nh_a (2)	BS	5
fid_CB (25)	IT	9	ah49_o (9)	BS	5
fie_CI (28)	IT	9	nh49_i (11)	BS	5
fie_CA (24)	IT	8			

Table B.2.1: Selected indicators across the 21 sparse disaggregations. Number in brackets refer to the indicator on the x-axis of Figure B.2.1.

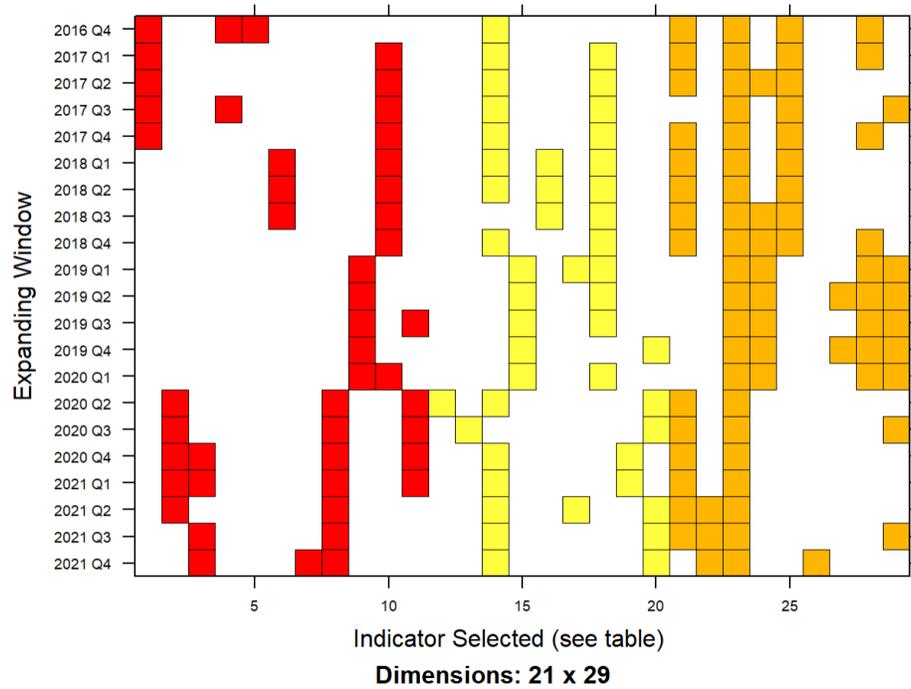


Figure B.2.1: Indicators selected when fixing the number of indicators selected to be 8. Refer to Table B.2.1 for the indicator on the x-axis.

Indicator	Dataset	N. of times selected	Indicator	Dataset	N. of times selected
fie_CA (69)	IT	21	bh_o (13)	BS	10
fit_B (62)	IT	20	air_15 (49)	AA	10
fid_ene (65)	IT	20	air_19 (51)	AA	10
ah49_z (37)	BS	17	air_33 (55)	AA	10
air_29 (54)	AA	16	fie_B (66)	IT	9
fit_CI (77)	IT	16	nh_p (9)	BS	8
bh49_p (34)	BS	15	bh_g (20)	BS	8
bh49_i (45)	BS	15	ah_a (2)	BS	7
fit_CH (74)	IT	14	ah49_o (35)	BS	7
nh49_z (38)	BS	13	fid_CB (70)	IT	7
air_23 (53)	AA	13	nh_a (3)	BS	6
fie_CI (79)	IT	13	sh_z (14)	BS	6
sh49_i (43)	BS	12	ah_i (21)	BS	6
nh49_p (33)	BS	11	air_13 (48)	AA	6
nh49_g (41)	BS	11	fie_CK (81)	IT	6
air_21 (52)	AA	11	fie_CM (83)	IT	6
fit_CA (68)	IT	11	nh_d (7)	BS	5
fie_CC (71)	IT	11	bh_d (8)	BS	5
fie_CH (76)	IT	11	fie_CL (82)	IT	5

Table B.2.2: Selected indicators across the 21 sparse disaggregations. Number in brackets refer to the indicator on the x-axis of Figure B.2.2.

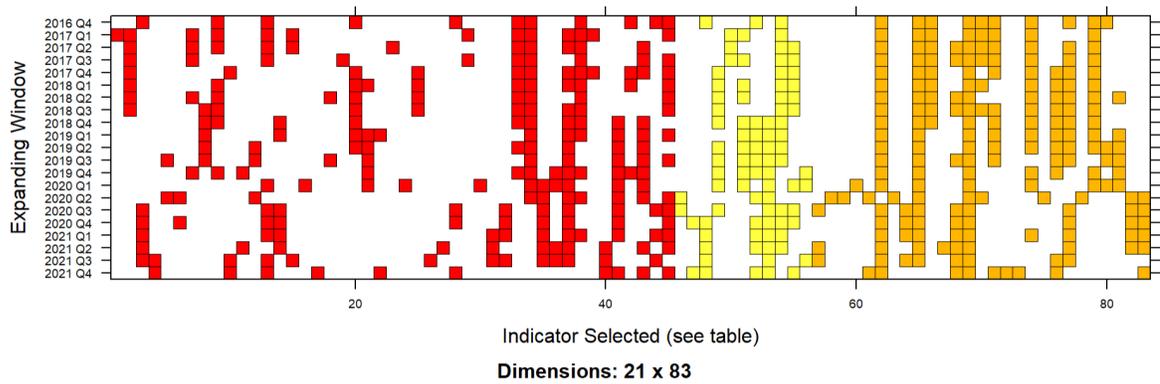


Figure B.2.2: Indicators selected when fixing the number of indicators selected to be 24. Refer to Table B.2.2 for the indicator on the x-axis.

Appendix C

Appendix to Chapter 4

C.1 Maximisation Step Derivations for Section 4.3.1

This appendix derives the parameter estimates found in the maximisation step of the EM algorithm in Section 4.3.1 of Chapter 4. Similar derivations can be found in the appendix of Bańbura and Modugno (2014). We wish to maximise the conditional expected log-likelihood:

$$\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \boldsymbol{\Omega}_n] = -\frac{1}{2} \log |\mathbf{P}_0| - \text{tr} \{ \mathbf{P}_0^{-1} \mathbb{E} [(\mathbf{F}_0 - \boldsymbol{\alpha}_0)(\mathbf{F}_0 - \boldsymbol{\alpha}_0)^\top | \boldsymbol{\Omega}_n] \} \quad (\text{C.1.1})$$

$$- \frac{n}{2} \log |\boldsymbol{\Sigma}_u| - \frac{1}{2} \sum_{t=1}^n \text{tr} \{ \boldsymbol{\Sigma}_u^{-1} \mathbb{E} [\mathbf{u}_t \mathbf{u}_t^\top | \boldsymbol{\Omega}_n] \} \quad (\text{C.1.2})$$

$$- \frac{n}{2} \log |\boldsymbol{\Sigma}_\epsilon| - \frac{1}{2} \sum_{t=1}^n \text{tr} \{ \boldsymbol{\Sigma}_\epsilon^{-1} \mathbb{E} [\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_t^\top | \boldsymbol{\Omega}_n] \} . \quad (\text{C.1.3})$$

We refer to $\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \boldsymbol{\Omega}_n]$ just as \mathbf{E} to make notation easier. In each derivation we use the linearity of conditional expectation: $\text{Cov}(X, Y | A) = \mathbb{E}(XY | A) - \mathbb{E}(X | A)\mathbb{E}(Y | A)$ for two random variables X, Y and non-zero event A . We also use the derivative property $\partial(\log |\mathbf{Z}|) / \partial(\mathbf{Z}^{-1}) = -\mathbf{Z}$ for a matrix \mathbf{Z} . Recall, we define the conditional mean and

covariances of the state as:

$$\begin{aligned}\mathbf{a}_{t|n} &= \mathbb{E}[\mathbf{F}_t | \boldsymbol{\Omega}_n], \\ \mathbf{P}_{t|n} &= \text{Var}[\mathbf{F}_t | \boldsymbol{\Omega}_n], \\ \mathbf{P}_{t,t-1|n} &= \text{Cov}[\mathbf{F}_t, \mathbf{F}_{t-1} | \boldsymbol{\Omega}_n],\end{aligned}$$

conditional on all information we have observed up to n , denoted by $\boldsymbol{\Omega}_n$, and let

$$\mathbf{S}_{t|n} = \mathbf{a}_{t|n} \mathbf{a}_{t|n}^\top + \mathbf{P}_{t|n}, \text{ and } \mathbf{S}_{t,t-1|n} = \mathbf{a}_{t|n} \mathbf{a}_{t-1|n}^\top + \mathbf{P}_{t,t-1|n}.$$

Expanding out the conditional expectation in the initial state distribution part (C.1.1) we obtain:

$$\begin{aligned}\mathbb{E}[(\mathbf{F}_0 - \boldsymbol{\alpha}_0)(\mathbf{F}_0 - \boldsymbol{\alpha}_0)^\top | \boldsymbol{\Omega}_n] &= \mathbb{E}[\mathbf{F}_0 \mathbf{F}_0^\top - 2\mathbf{F}_0 \boldsymbol{\alpha}_0^\top + \boldsymbol{\alpha}_0 \boldsymbol{\alpha}_0^\top | \boldsymbol{\Omega}_n] \\ &= \mathbb{E}[\mathbf{F}_0 \mathbf{F}_0^\top | \boldsymbol{\Omega}_n] - 2\mathbb{E}[\mathbf{F}_0 | \boldsymbol{\Omega}_n] \boldsymbol{\alpha}_0^\top + \boldsymbol{\alpha}_0 \boldsymbol{\alpha}_0^\top \\ &= \mathbb{E}[\mathbf{F}_0 | \boldsymbol{\Omega}_n] \mathbb{E}[\mathbf{F}_0^\top | \boldsymbol{\Omega}_n] + \text{Var}[\mathbf{F}_0 | \boldsymbol{\Omega}_n] - 2\mathbb{E}[\mathbf{F}_0 | \boldsymbol{\Omega}_n] \boldsymbol{\alpha}_0^\top + \boldsymbol{\alpha}_0 \boldsymbol{\alpha}_0^\top \\ &= \mathbf{a}_{0|n} \mathbf{a}_{0|n}^\top + \mathbf{P}_{0|n} - 2\mathbf{a}_{0|n} \boldsymbol{\alpha}_0^\top + \boldsymbol{\alpha}_0 \boldsymbol{\alpha}_0^\top \\ &= \mathbf{S}_{0|n} - 2\mathbf{a}_{0|n} \boldsymbol{\alpha}_0^\top + \boldsymbol{\alpha}_0 \boldsymbol{\alpha}_0^\top.\end{aligned}$$

Differentiating \mathbf{E} with respect to $\boldsymbol{\alpha}_0$ and \mathbf{P}_0^{-1} we obtain:

$$\begin{aligned}\frac{\partial \mathbf{E}}{\partial \boldsymbol{\alpha}_0} &= -\frac{1}{2} \mathbf{P}_0^{-1} \frac{\partial}{\partial \boldsymbol{\alpha}_0} \mathbb{E}[(\mathbf{F}_0 - \boldsymbol{\alpha}_0)(\mathbf{F}_0 - \boldsymbol{\alpha}_0)^\top | \boldsymbol{\Omega}_n] = -\frac{1}{2} \mathbf{P}_0^{-1} (-2\mathbf{a}_{0|n} + 2\boldsymbol{\alpha}_0) \\ \frac{\partial \mathbf{E}}{\partial \mathbf{P}_0^{-1}} &= \frac{1}{2} \mathbf{P}_0 - \frac{1}{2} \mathbb{E}[(\mathbf{F}_0 - \boldsymbol{\alpha}_0)(\mathbf{F}_0 - \boldsymbol{\alpha}_0)^\top | \boldsymbol{\Omega}_n] \\ &= \frac{1}{2} \mathbf{P}_0 - \frac{1}{2} (\mathbf{S}_{0|n} - 2\mathbf{a}_{0|n} \boldsymbol{\alpha}_0^\top + \boldsymbol{\alpha}_0 \boldsymbol{\alpha}_0^\top).\end{aligned}$$

Setting both derivatives equal to 0, we first get the result

$$\hat{\boldsymbol{\alpha}}_0 = \mathbf{a}_{0|n},$$

and use this to obtain

$$\hat{P}_0 = P_{0|n}.$$

Expanding out the expectation in the state transition part (C.1.2) we obtain:

$$\begin{aligned} \mathbb{E} [\mathbf{u}_t^\top \mathbf{u}_t | \Omega_n] &= \mathbb{E}[(\mathbf{F}_t - \mathbf{A}\mathbf{F}_{t-1})(\mathbf{F}_t - \mathbf{A}\mathbf{F}_{t-1})^\top | \Omega_n] \\ &= \mathbb{E}[\mathbf{F}_t \mathbf{F}_t^\top - 2\mathbf{F}_t \mathbf{F}_{t-1}^\top \mathbf{A}^\top + \mathbf{A} \mathbf{F}_{t-1} \mathbf{F}_{t-1}^\top \mathbf{A}^\top | \Omega_n] \\ &= \mathbb{E}[\mathbf{F}_t \mathbf{F}_t^\top | \Omega_n] - 2\mathbb{E}[\mathbf{F}_t \mathbf{F}_{t-1}^\top | \Omega_n] \mathbf{A}^\top + \mathbf{A} \mathbb{E}[\mathbf{F}_{t-1} \mathbf{F}_{t-1}^\top | \Omega_n] \mathbf{A}^\top \\ &= \mathbb{E}[\mathbf{F}_t | \Omega_n] \mathbb{E}[\mathbf{F}_t^\top | \Omega_n] + \text{Var}[\mathbf{F}_t | \Omega_n] \\ &\quad - 2 \left(\mathbb{E}[\mathbf{F}_t | \Omega_n] \mathbb{E}[\mathbf{F}_{t-1}^\top | \Omega_n] \text{Cov}[\mathbf{F}_t, \mathbf{F}_{t-1} | \Omega_n] \right) \mathbf{A}^\top \\ &\quad + \mathbf{A} \left(\mathbb{E}[\mathbf{F}_{t-1} | \Omega_n] \mathbb{E}[\mathbf{F}_{t-1}^\top | \Omega_n] + \text{Var}[\mathbf{F}_{t-1} | \Omega_n] \right) \mathbf{A}^\top \\ &= \mathbf{S}_{t|n} - 2\mathbf{S}_{t,t-1|n} \mathbf{A}^\top + \mathbf{A} \mathbf{S}_{t-1|n} \mathbf{A}^\top. \end{aligned}$$

Differentiating \mathbf{E} with respect to \mathbf{A} and Σ_u^{-1} we obtain:

$$\begin{aligned} \frac{\partial \mathbf{E}}{\partial \mathbf{A}} &= -\frac{1}{2} \Sigma_u^{-1} \frac{\partial}{\partial \mathbf{A}} \sum_{t=1}^n \mathbb{E} [\mathbf{u}_t^\top \mathbf{u}_t | \Omega_n] = -\frac{1}{2} \Sigma_u^{-1} \sum_{t=1}^n (-2\mathbf{S}_{t,t-1|n} + 2\mathbf{A} \mathbf{S}_{t-1|n}), \\ \frac{\partial \mathbf{E}}{\partial \Sigma_u^{-1}} &= \frac{n}{2} \Sigma_u - \frac{1}{2} \sum_{t=1}^n \mathbb{E} [\mathbf{u}_t^\top \mathbf{u}_t | \Omega_n] = \frac{n}{2} \Sigma_u - \frac{1}{2} (\mathbf{S}_{t|n} - 2\mathbf{S}_{t,t-1|n} \mathbf{A}^\top + \mathbf{A} \mathbf{S}_{t-1|n} \mathbf{A}^\top). \end{aligned}$$

Setting both derivatives equal to 0, we first get the result

$$\hat{\mathbf{A}} = \left(\sum_{t=1}^n \mathbf{S}_{t-1|n} \right)^{-1} \left(\sum_{t=1}^n \mathbf{S}_{t,t-1|n} \right),$$

and use this to obtain

$$\hat{\Sigma}_u = \frac{1}{n} \sum_{t=1}^n \left[\mathbf{S}_{t|n} - \hat{\mathbf{A}} (\mathbf{S}_{t-1,t|n}) \right].$$

Using the selection matrix \mathbf{W}_t defined in Section 4.3.1, with the property $\mathbf{X}_t =$

$\mathbf{W}_t \mathbf{X}_t + (\mathbf{I} - \mathbf{W}_t) \mathbf{X}_t$, we can expand out the expectation in the measurement equation part (C.1.3) and obtain:

$$\begin{aligned}
\mathbb{E} [\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_t^\top | \Omega_n] &= \mathbb{E} [(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top | \Omega_n] \\
&= \mathbb{E} \left[\left(\mathbf{W}_t(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t) + (\mathbf{I} - \mathbf{W}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t) \right) \right. \\
&\quad \left. \left(\mathbf{W}_t(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t) + (\mathbf{I} - \mathbf{W}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t) \right)^\top | \Omega_n \right] \\
&= \mathbb{E} \left[\mathbf{W}_t(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t) \mathbf{W}_t + (\mathbf{I} - \mathbf{W}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top \mathbf{W}_t \right. \\
&\quad \left. + \mathbf{W}_t(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top (\mathbf{I} - \mathbf{W}_t) \right. \\
&\quad \left. + (\mathbf{I} - \mathbf{W}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top (\mathbf{I} - \mathbf{W}_t) | \Omega_n \right].
\end{aligned}$$

Using the law of iterated expectations:

$$\mathbb{E}[(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top | \Omega_n] = \mathbb{E} [\mathbb{E}[(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top | \mathbf{F}, \Omega_n] | \Omega_n],$$

Bañbura and Modugno (2014) show

$$\mathbb{E}[\mathbf{W}_t(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top (\mathbf{I} - \mathbf{W}_t) | \mathbf{F}, \Omega_n] = 0,$$

$$\mathbb{E}[(\mathbf{I} - \mathbf{W}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top (\mathbf{I} - \mathbf{W}_t) | \mathbf{F}, \Omega_n] = (\mathbf{I} - \mathbf{W}_t) \boldsymbol{\Sigma}_\epsilon (\mathbf{I} - \mathbf{W}_t),$$

and

$$\begin{aligned}
\mathbb{E}[\mathbf{W}_t(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top \mathbf{W}_t | \Omega_n] &= \mathbf{W}_t \mathbf{X}_t \mathbf{X}_t^\top \mathbf{W}_t - \mathbf{W}_t \mathbf{X}_t \mathbb{E}[\mathbf{F}_t^\top | \Omega_n] \boldsymbol{\Lambda}^\top \mathbf{W}_t \\
&\quad - \mathbf{W}_t \boldsymbol{\Lambda} \mathbb{E}[\mathbf{F}_t | \Omega_n] \mathbf{X}_t^\top \mathbf{W}_t \\
&\quad + \mathbf{W}_t \boldsymbol{\Lambda} \mathbb{E}[\mathbf{F}_t \mathbf{F}_t^\top | \Omega_n] \boldsymbol{\Lambda}^\top \mathbf{W}_t.
\end{aligned}$$

Hence, we get the result

$$\begin{aligned}
\mathbb{E}[\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_t^\top | \boldsymbol{\Omega}_n] &= \mathbb{E}[(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)(\mathbf{X}_t - \boldsymbol{\Lambda} \mathbf{F}_t)^\top | \boldsymbol{\Omega}_n] \\
&= \mathbf{W}_t \mathbf{X}_t \mathbf{X}_t^\top \mathbf{W}_t - \mathbf{W}_t \mathbf{X}_t \mathbb{E}[\mathbf{F}_t^\top | \boldsymbol{\Omega}_n] \boldsymbol{\Lambda}^\top \mathbf{W}_t - \mathbf{W}_t \boldsymbol{\Lambda} \mathbb{E}[\mathbf{F}_t | \boldsymbol{\Omega}_n] \mathbf{X}_t^\top \mathbf{W}_t \\
&\quad + \mathbf{W}_t \boldsymbol{\Lambda} (\mathbb{E}[\mathbf{F}_t | \boldsymbol{\Omega}_n] \mathbb{E}[\mathbf{F}_t^\top | \boldsymbol{\Omega}_n] + \text{Var}[\mathbf{F}_t | \boldsymbol{\Omega}_n]) \boldsymbol{\Lambda}^\top \mathbf{W}_t + (\mathbf{I} - \mathbf{W}_t) \boldsymbol{\Sigma}_\epsilon^* (\mathbf{I} - \mathbf{W}_t) \\
&= \mathbf{W}_t \mathbf{X}_t \mathbf{X}_t^\top \mathbf{W}_t - 2 \mathbf{W}_t \mathbf{X}_t \mathbf{a}_{t|n} \boldsymbol{\Lambda}^\top \mathbf{W}_t + \mathbf{W}_t \boldsymbol{\Lambda} \mathbf{S}_{t|n} \boldsymbol{\Lambda}^\top \mathbf{W}_t + (\mathbf{I} - \mathbf{W}_t) \boldsymbol{\Sigma}_\epsilon (\mathbf{I} - \mathbf{W}_t) \\
&= \mathbf{W}_t (\mathbf{X}_t \mathbf{X}_t^\top - 2 \mathbf{X}_t \mathbf{a}_{t|n} \boldsymbol{\Lambda}^\top + \boldsymbol{\Lambda} \mathbf{S}_{t|n} \boldsymbol{\Lambda}^\top) \mathbf{W}_t + (\mathbf{I} - \mathbf{W}_t) \boldsymbol{\Sigma}_\epsilon^* (\mathbf{I} - \mathbf{W}_t),
\end{aligned}$$

where $\boldsymbol{\Sigma}_\epsilon^*$ is obtained from the previous EM iteration.

Differentiating \mathbf{E} with respect to $\boldsymbol{\Lambda}$ we obtain:

$$\frac{\partial \mathbf{E}}{\partial \boldsymbol{\Lambda}} = -\frac{1}{2} \boldsymbol{\Sigma}_\epsilon^{-1} \frac{\partial}{\partial \boldsymbol{\Lambda}} \sum_{t=1}^n \mathbb{E}[\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_t^\top | \boldsymbol{\Omega}_n] = -\frac{1}{2} \boldsymbol{\Sigma}_\epsilon^{-1} \sum_{t=1}^n \mathbf{W}_t (-2 \mathbf{X}_t \mathbf{a}_{t|n} + 2 \boldsymbol{\Lambda} \mathbf{S}_{t|n}) \mathbf{W}_t. \tag{C.1.4}$$

Setting this equal to 0, dividing through by $\mathbf{W}_t \boldsymbol{\Sigma}_\epsilon^{-1}$, and using the identity $\text{vec}(ABC) = (C^\top \otimes A) \text{vec}(B)$, for matrices A, B and C , we obtain the result

$$\text{vec}(\hat{\boldsymbol{\Lambda}}) = \left(\sum_{t=1}^n \mathbf{S}_{t|n} \otimes \mathbf{W}_t \right)^{-1} \text{vec} \left(\sum_{t=1}^n \mathbf{W}_t \mathbf{X}_t \mathbf{a}_{t|n}^\top \right).$$

This is the ordinary least-squares estimator for the loadings matrix as in a classic DFM. The work proposed in Chapter 4 estimates a regularised version of this using an ADMM algorithm. This ADMM algorithm along with the $\boldsymbol{\Lambda}$ estimator is fully derived in Appendix C.2.

Differentiating \mathbf{E} with respect to $\boldsymbol{\Sigma}_\epsilon^{-1}$ we obtain:

$$\frac{\partial \mathbf{E}}{\partial \boldsymbol{\Sigma}_\epsilon^{-1}} = \frac{n}{2} \boldsymbol{\Sigma}_\epsilon - \frac{1}{2} \sum_{t=1}^n \mathbf{W}_t (\mathbf{X}_t \mathbf{X}_t^\top - 2 \mathbf{X}_t \mathbf{a}_{t|n} \boldsymbol{\Lambda}^\top + \boldsymbol{\Lambda} \mathbf{S}_{t|n} \boldsymbol{\Lambda}^\top) \mathbf{W}_t + (\mathbf{I} - \mathbf{W}_t) \boldsymbol{\Sigma}_\epsilon^* (\mathbf{I} - \mathbf{W}_t).$$

Setting this equal to 0 and subbing in $\hat{\Lambda}$ we obtain the result

$$\hat{\Sigma}_{\epsilon} = \frac{1}{n} \sum_{t=1}^n \text{diag} \left[\mathbf{W}_t \left(\mathbf{X}_t \mathbf{X}_t^{\top} - 2 \mathbf{X}_t \mathbf{a}_{t|n}^{\top} \hat{\Lambda}^{\top} + \hat{\Lambda} \mathbf{S}_{t|n} \hat{\Lambda}^{\top} \right) \mathbf{W}_t + (\mathbf{I} - \mathbf{W}_t) \Sigma_{\epsilon}^* (\mathbf{I} - \mathbf{W}_t) \right].$$

C.2 ADMM Algorithm Derivations for Section 4.3.1

For the ADMM algorithm of Section 4.3.1, used to incorporate ℓ_1 regularisation into the estimation of the loadings parameter Λ , we consider the penalised and augmented Lagrangian

$$\mathcal{C}(\Lambda, \mathbf{Z}, \mathbf{U}) := -\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \Omega_n] + \alpha \|\mathbf{Z}\|_1 + \frac{\nu}{2} \|\Lambda - \mathbf{Z} + \mathbf{U}\|_F^2, \quad (\text{C.2.1})$$

where $\mathbf{Z} \in \mathbb{R}^{p \times r}$ is an auxiliary variable, $\mathbf{U} \in \mathbb{R}^{p \times r}$ are the (scaled) Lagrange multipliers and ν is the scaling term. The derivation for minimising with respect to \mathbf{Z} and \mathbf{U} are standard and can be found in Boyd et al. (2011). We now go into detail on the minimisation with respect to Λ for iteration $(k+1)$ of the ADMM algorithm:

$$\Lambda^{(k+1)} = \arg \min_{\Lambda} \mathcal{C}(\Lambda, \mathbf{Z}^{(k)}, \mathbf{U}^{(k)}).$$

We can re-write the Lagrangian (C.2.1) as

$$\mathcal{C}(\Lambda, \mathbf{Z}, \mathbf{U}) \propto -\mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \Omega_n] + \frac{\nu}{2} \text{tr} (\Lambda \Lambda^{\top} - 2(\mathbf{Z}^{(k)} - \mathbf{u}^{(k)}) \Lambda^{\top}),$$

by dropping terms not involving Λ and using the property $\text{tr}(AB) = \text{tr}(BA)$ for matrices A and B . Taking the derivative and using (C.1.4) we obtain

$$\begin{aligned} \frac{\partial \mathcal{C}(\Lambda, \mathbf{Z}^{(k)}, \mathbf{U}^{(k)})}{\partial \Lambda} &= -\frac{\partial}{\partial \Lambda} \mathbb{E} [\log \mathcal{L}(\mathbf{X}, \mathbf{F}; \boldsymbol{\theta}) | \Omega_n] + \frac{\nu}{2} \text{tr} (2\Lambda - 2(\mathbf{Z}^{(k)} - \mathbf{u}^{(k)})) \\ &= \frac{1}{2} \Sigma_{\epsilon}^{-1} \sum_{t=1}^n \mathbf{W}_t (-2 \mathbf{X}_t \mathbf{a}_{t|n} + 2 \Lambda \mathbf{S}_{t|n}) \mathbf{W}_t + \frac{\nu}{2} \text{tr} (2\Lambda - 2(\mathbf{Z}^{(k)} - \mathbf{u}^{(k)})). \end{aligned}$$

Setting this equal to 0 and re-arranging for $\mathbf{\Lambda}$ we have

$$\sum_{t=1}^n \mathbf{W}_t \Sigma_{\epsilon}^{-1} \mathbf{W}_t \mathbf{\Lambda} \mathbf{S}_{t|n} + \nu \mathbf{\Lambda} = \sum_{t=1}^n \mathbf{W}_t \Sigma_{\epsilon}^{-1} \mathbf{W}_t \mathbf{X}_t \mathbf{a}_{t|n} + \nu (\mathbf{Z}^{(k)} - \mathbf{U}^{(k)}).$$

Using the identity $\text{vec}(ABC) = (C^{\top} \otimes A) \text{vec}(B)$, for matrices A, B and C , we obtain the result

$$\text{vec}(\mathbf{\Lambda}^{(k+1)}) = \left(\sum_{t=1}^n \mathbf{S}_{t|n} \otimes \mathbf{W}_t \Sigma_{\epsilon}^{-1} \mathbf{W}_t + \nu \mathbf{I}_{pr} \right)^{-1} \text{vec} \left[\sum_{t=1}^n \mathbf{W}_t \Sigma_{\epsilon}^{-1} \mathbf{W}_t \mathbf{X}_t \mathbf{a}_{t|n}^{\top} + \nu (\mathbf{Z}^{(k)} - \mathbf{U}^{(k)}) \right].$$

Appendix D

Appendix to Chapter 6

D.1 An Extension for AR(1) Idiosyncratic Errors

The package `sparseDFM` allows the extension for the state-space framework (2.2.2)-(2.2.3) of a DFM introduced in Chapter 2.2.4 to have AR(1) idiosyncratic errors, as opposed to IID white noise errors. Following the methodology laid out in Bańbura and Modugno (2014), this is carried out by augmenting the state vector so it contains both the latent factors and the latent AR(1) errors. Formally, the AR(1) process is set up like so:

$$\begin{aligned}\boldsymbol{\epsilon}_{i,t} &= \tilde{\boldsymbol{\epsilon}}_{i,t} + \boldsymbol{\eta}_{i,t}, & \boldsymbol{\eta}_{i,t} &\sim N(0, \kappa), \\ \tilde{\boldsymbol{\epsilon}}_{i,t} &= \phi_i \tilde{\boldsymbol{\epsilon}}_{i,t-1} + \boldsymbol{e}_{i,t}, & \boldsymbol{e}_{i,t} &\sim N(0, \sigma_i^2),\end{aligned}$$

for $i = 1, \dots, p$ and $t = 1, \dots, n$, where κ is a very small number allowing us to easily set up the new state-space framework.

This new state-space framework is given by

$$\begin{aligned}\boldsymbol{X}_t &= \tilde{\boldsymbol{\Lambda}} \tilde{\boldsymbol{F}}_t + \boldsymbol{\eta}_t, & \boldsymbol{\eta}_t &\sim N(0, \boldsymbol{\Sigma}_\eta), \\ \tilde{\boldsymbol{F}}_t &= \tilde{\boldsymbol{A}} \tilde{\boldsymbol{F}}_{t-1} + \tilde{\boldsymbol{u}}_t, & \tilde{\boldsymbol{u}}_t &\sim N(0, \boldsymbol{\Sigma}_{\tilde{\boldsymbol{u}}}),\end{aligned}$$

where

$$\tilde{\mathbf{F}}_t = \begin{bmatrix} \mathbf{F}_t \\ \tilde{\boldsymbol{\epsilon}}_t \end{bmatrix}, \quad \tilde{\mathbf{u}}_t = \begin{bmatrix} \mathbf{u}_t \\ \mathbf{e}_t \end{bmatrix}, \quad \tilde{\boldsymbol{\Lambda}} = \begin{bmatrix} \boldsymbol{\Lambda} & \mathbf{I} \end{bmatrix}, \quad \tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\phi} \end{bmatrix},$$

$$\boldsymbol{\Sigma}_{\tilde{\mathbf{u}}} = \begin{bmatrix} \boldsymbol{\Sigma}_u & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_e \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Sigma}_\eta = \kappa \mathbf{I}_{r+p},$$

for $\boldsymbol{\phi} = \text{diag}(\phi_1, \dots, \phi_p)$ and $\boldsymbol{\Sigma}_e = \mathbb{E}[\mathbf{e}_t \mathbf{e}_t^\top] = \text{diag}(\boldsymbol{\sigma}_e^2)$ and $\boldsymbol{\sigma}_e^2 \in \mathbb{R}_+^p$ is a vector of error variances. The parameter estimates of this new augmented state-space model can be obtained very similarly to the derivations in Appendix C.1. See Bańbura and Modugno (2014) for details of these estimates: $\tilde{\boldsymbol{\theta}} = (\mathbf{A}, \boldsymbol{\phi}, \boldsymbol{\Lambda}, \boldsymbol{\Sigma}_u, \boldsymbol{\Sigma}_e)$.

D.1.1 Effect on Kalman Filter and Smoother Computation

This way of incorporating AR(1) idiosyncratic errors by augmenting them into the state vector leads to an increase in the dimension of this state vector. This of course slows down the computation of Kalman filter and smoother (KFS) equations. Bańbura and Modugno (2014) suggest the alternative approach by Jungbacker et al. (2009) is perhaps a computationally more efficient way to incorporate AR(1) errors into the DFM framework, however, is a lot more complex with time-varying parameters.

In the `sparseDFM` package it is recommended to use the univariate KFS for IID errors and multivariate KFS for AR(1) errors for the fastest possible computation. In Remark 3 of Chapter 4, we saw that the calculation of the cross-covariance matrix in the expectation step involving the univariate KFS equations is given by

$$\mathbf{P}_{t,t-1|n} = \mathbf{P}_{t|n} (\mathbf{P}_{t|t-1})^{-1} \mathbf{A} \mathbf{P}_{t-1|t-1}.$$

The matrix needing to be inverted, $\mathbf{P}_{t|t-1}$, is of dimension equal to the number of states. For this reason, the univariate KFS becomes slow when the number of states is high.

To test the timing comparisons of the KFS methods in IID and AR(1) scenarios,

we can simulate data from a DFM with $\mathbf{\Lambda} = N(\mathbf{0}_p, \mathbf{I}_p)$, $\mathbf{\Sigma}_\epsilon = \mathbf{I}_p$, $\mathbf{A} = 0.8 \times \mathbf{I}_r$ and $\mathbf{\Sigma}_u = (1 - 0.8^2) \times \mathbf{I}_r$ using 2 factors and fit a DFM in `sparseDFM` using `alg = "EM"` (Bańbura and Modugno, 2014) with `err = "IID"` and `err = "AR1"`. Figure D.1.1 shows the average time from 10 experiments with a fixed number of observations, $n = 100$, and varying dimension p , with the IID case on the left and the AR(1) case on the right. The IID case shows the clear computational gain of the univariate KFS approach, and this becomes even more prominent for larger p . The AR(1) setting clearly takes a longer amount of time to run and the univariate filtering is slower than the classic multivariate approach.

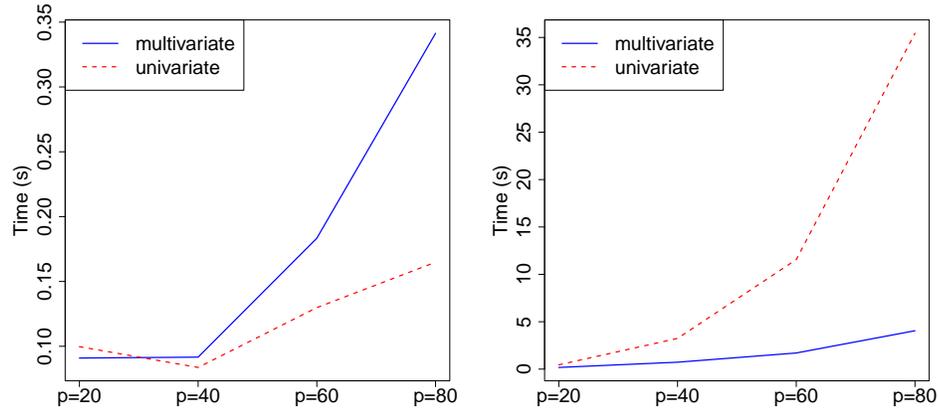


Figure D.1.1: Average time from 10 experiments the EM algorithm (setting `alg = "EM"`) takes with the multivariate and univariate KFS methods with `err = "IID"` on the left and `err = "AR1"` on the right.

D.2 Table of Functions of the `sparseDFM` Package

The inputs and outputs for the main function `sparseDFM()` were provided in Chapter 6.2.2. The table below presents a full list of functions the user of the package can employ.

Function	Brief Description
<code>sparseDFM()</code>	The main function to implement DFM estimation. See Chapter 6.2.2.
<code>kalmanMultivariate()</code>	Function to run the classic multivariate Kalman filter and smoother equations of Shumway and Stoffer (1982).
<code>kalmanUnivariate()</code>	Function to run the fast univariate Kalman filter and smoother equations of Koopman and Durbin (2000).
<code>fillNA()</code>	Function to interpolate missing data in \mathbf{X} by using the median of the series that are then smoothed with a simple moving window. This is used to obtain a balanced dataset for the PCA initialisation fit.
<code>transformData()</code>	Function to transform a non-stationary data matrix to become stationary by taking first/second (log) differences/growth rates. See Chapter 6.2.3.
<code>tuneFactors()</code>	Function to determine the number of factors to use based on the Bai and Ng (2002) information criteria. A plot of IC values and the variance explained from PCA are produced.
<code>missing_data_plot()</code>	Visualise the amount of missing data in a data matrix or data frame.
<code>raggedEdge()</code>	Function to generate a ragged edge structure for the end of sample of a data matrix/frame.
<code>logspace()</code>	Function to produce a vector of log10 space values useful for the LASSO tuning parameter α .

Bibliography

- Andreou, E., Gagliardini, P., Ghysels, E., and Rubin, M. (2019). Inference in group factor models with an application to mixed-frequency data. *Econometrica*, 87(4):1267–1305.
- Andreou, E., Ghysels, E., and Kourtellis, A. (2011). Forecasting with mixed-frequency data. In *The Oxford Handbook of Economic Forecasting*, pages 225–245. Oxford University Press.
- Angelini, E., Henry, J., and Marcellino, M. (2006). Interpolation and backdating with a large information set. *Journal of Economic Dynamics and Control*, 30(12):2693–2724.
- Angelini, E., , M., Runstler, G., et al. (2010). Estimating and forecasting the euro area monthly national accounts from a dynamic factor model. *OECD Journal: Journal of Business Cycle Measurement and Analysis*, 2010(1):1–22.
- Aruoba, S. B., Diebold, F. X., and Scotti, C. (2009). Real-time measurement of business conditions. *Journal of Business & Economic Statistics*, 27(4):417–427.
- Bach, F., Jenatton, R., Mairal, J., Obozinski, G., et al. (2011). Convex optimization with sparsity-inducing norms. *Optimization for Machine Learning*, 5:19–53.
- Baffigi, A., Golinelli, R., and Parigi, G. (2004). Bridge models to forecast the Euro area GDP. *International Journal of forecasting*, 20(3):447–460.
- Bai, J. (2003). Inferential theory for factor models of large dimensions. *Econometrica*, 71(1):135–171.

- Bai, J. and Li, K. (2016). Maximum likelihood estimation and inference for approximate factor models of high dimension. *The Review of Economics and Statistics*, 98(2):298–309.
- Bai, J. and Ng, S. (2002). Determining the number of factors in approximate factor models. *Econometrica*, 70(1):191–221.
- Bańbura, M., Giannone, D., Modugno, M., and Reichlin, L. (2013). Now-casting and the real-time data flow. 2:195–237.
- Bańbura, M., Giannone, D., and Reichlin, L. (2011). Nowcasting. In *The Oxford Handbook of Economic Forecasting*, chapter 7. Oxford University Press.
- Bańbura, M. and Modugno, M. (2014). Maximum likelihood estimation of factor models on datasets with arbitrary pattern of missing data. *Journal of Applied Econometrics*, 29(1):133–160.
- Bańbura, M. and Rünstler, G. (2011). A look into the factor model black box: publication lags and the role of hard and soft data in forecasting GDP. *International Journal of Forecasting*, 27(2):333–346.
- Barigozzi, M. and Luciani, M. (2022). Quasi maximum likelihood estimation and inference of large approximate dynamic factor models via the EM algorithm. *arXiv Preprint arXiv:1910.03821*.
- Bean, C. R. (2016). *Independent review of UK economic statistics*. HM Treasury.
- Belloni, A., Chernozhukov, V., et al. (2013). Least squares after model selection in high-dimensional sparse models. *Bernoulli*, 19(2):521–547.
- Belloni, A., Chernozhukov, V., and Wang, L. (2011). Square-root lasso: pivotal recovery of sparse signals via conic programming. *Biometrika*, 98(4):791–806.

- Bernal, J. L., Cummins, S., and Gasparrini, A. (2017). Interrupted time series regression for the evaluation of public health interventions: a tutorial. *International journal of epidemiology*, 46(1):348–355.
- Bhattacharya, A., Pati, D., Pillai, N. S., and Dunson, D. B. (2015). Dirichlet–laplace priors for optimal shrinkage. *Journal of the American Statistical Association*, 110(512):1479–1490.
- Bok, B., Caratelli, D., Giannone, D., Sbordone, A. M., and Tambalotti, A. (2018). Macroeconomic nowcasting and forecasting with big data. *Annual Review of Economics*, 10:615–643.
- Boot, J. C. G., Feibes, W., and Lisman, J. H. C. (1967). Further methods of derivation of quarterly figures from annual data. *Applied Statistics*, 16(1):65–75.
- Bournay, J. and Laroque, G. (1979). Réflexions sur la méthode d’élaboration des comptes trimestriels. *Annales de l’INSEE*, (36):3–30.
- Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.
- Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122.
- Brave, S. A., Butters, R. A., and Kelley, D. (2022). A practitioner’s guide and matlab toolbox for mixed frequency state space models. *Journal of Statistical Software*, 104:1–40.
- Breitung, J. and Eickmeier, S. (2011). Testing for structural breaks in dynamic factor models. *Journal of Econometrics*, 163(1):71–84.

- Brockwell, P. J. and Davis, R. A. (2009). *Time series: theory and methods*. Springer science & business media.
- Bühlmann, P. and Mandozzi, J. (2014). High-dimensional variable screening and bias in subsequent inference, with an empirical comparison. *Computational Statistics*, 29:407–430.
- Bühlmann, P. and Van De Geer, S. (2011). *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media.
- Buigut, S. and Kapar, B. (2023). How did Brexit impact EU trade? Evidence from real data. *The World Economy*, 46:1566–1581.
- Carroll, J. B. (1953). An analytical solution for approximating simple structure in factor analysis. *Psychometrika*, 18(1):23–38.
- Castle, J., Hendry, D., and Kitov, O. (2017). Forecasting and nowcasting macroeconomic variables: A methodological overview. *Handbook on rapid estimates, Eurostat*, pages 53–107.
- Causey, B. and Trager, M. L. (1981). Derivation of solution to the benchmarking problem: Trend revision. *Unpublished research notes, US Census Bureau, Washington DC*.
- Chen, B. (2007). *An empirical comparison of methods for temporal disaggregation at the national accounts*. Office of Directors Bureau of Economic Analysis, Washington, DC.
- Chen, J. and Chen, Z. (2008). Extended bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3):759–771.
- Cheung, Y.-W. and Lai, K. S. (1995). Lag order and critical values of the augmented dickey–fuller test. *Journal of Business & Economic Statistics*, 13(3):277–280.

- Choi, H. and Varian, H. (2012). Predicting the present with google trends. *Economic record*, 88:2–9.
- Cholette, P.-A. (1984). Adjusting sub-annual series to yearly benchmarks. *Surv. Methodol.*, 10(1):35–49.
- Cholette, P. A. and Dagum, E. B. (1994). Benchmarking time series with autocorrelated survey errors. *International Statistical Review/Revue Internationale de Statistique*, pages 365–377.
- Chow, G. C. and Lin, A.-l. (1971). Best linear unbiased interpolation, distribution, and extrapolation of time series by related series. *The review of Economics and Statistics*, pages 372–375.
- Chronopoulos, I., Chrysikou, K., and Kapetanios, G. (2022). High dimensional generalised penalised least squares. *arXiv preprint arXiv:2207.07055*.
- Ciammola, A., Di Palma, F., and Marini, M. (2005). Temporal disaggregation techniques of time series by related series: A comparison by a monte carlo experiment. Technical report, Eurostat, Working Paper.
- Clements, M. P. and Galvão, A. B. (2009). Forecasting US output growth using leading indicators: An appraisal using MIDAS models. *Journal of Applied Econometrics*, 24(7):1187–1206.
- Corcoran, P., Griffin, E., Arensman, E., Fitzgerald, A. P., and Perry, I. J. (2015). Impact of the economic recession and subsequent austerity on suicide and self-harm in Ireland: An interrupted time series analysis. *International journal of epidemiology*, 44(3):969–977.
- Croux, C. and Exterkate, P. (2011). Sparse and robust factor modelling. Tinbergen Institute Discussion Paper TI 122/4.

- Dagum, E. B. and Cholette, P. A. (2006). *Benchmarking, temporal distribution, and reconciliation methods for time series*. Springer.
- De Jong, P. (1991). The diffuse kalman filter. *The Annals of Statistics*, 19(2):1073–1083.
- De Jong, P. and Mackinnon, M. J. (1988). Covariances for smoothed estimates in state space models. *Biometrika*, 75(3):601–602.
- De Valk, S., de Mattos, D., and Ferreira, P. (2019). Nowcasting: An R package for predicting economic variables using Dynamic Factor Models. *The R Journal*, 11(1):230–244.
- Del Negro, M. and Otrok, C. (2008). Dynamic factor models with time-varying parameters: measuring changes in international business cycles. *FEB of New York Staff Report*, 1(326).
- Denton, F. T. (1971). Adjustment of monthly or quarterly series to annual totals: an approach based on quadratic minimization. *Journal of the american statistical association*, 66(333):99–102.
- Despois, T. and Doz, C. (2023). Identifying and interpreting the factors in factor models via sparsity: Different approaches. *Journal of Applied Econometrics*, 38:451–667.
- Dhingra, S., Huang, H., Ottaviano, G., Paulo Pessoa, J., Sampson, T., and Van Reenen, J. (2017). The costs and benefits of leaving the eu: trade effects. *Economic Policy*, 32(92):651–705.
- Di Fonzo, T. (1990). The estimation of m disaggregate time series when contemporaneous and temporal aggregates are known. *The Review of Economics and Statistics*, 72(1):178–182.
- Di Fonzo, T. and Marini, M. (2011). Simultaneous and two-step reconciliation of systems

- of time series: methodological and practical issues. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 60(2):143–164.
- Di Fonzo, T. and Marini, M. (2015). Reconciliation of systems of time series according to a growth rates preservation principle. *Statistical Methods & Applications*, 24:651–669.
- Diron, M. (2008). Short-term forecasts of euro area real GDP growth: an assessment of real-time performance based on vintage data. *Journal of forecasting*, 27(5):371–390.
- Doz, C. and Fuleky, P. (2020). Dynamic factor models. In *Macroeconomic Forecasting in the Era of Big Data*, pages 27–64. Springer.
- Doz, C., Giannone, D., and Reichlin, L. (2011). A two-step estimator for large approximate dynamic factor models based on Kalman filtering. *Journal of Econometrics*, 164(1):188–205.
- Doz, C., Giannone, D., and Reichlin, L. (2012). A quasi-maximum likelihood approach for large, approximate dynamic factor models. *Review of Economics and Statistics*, 94(4):1014–1024.
- Du, J., Satoglu, E. B., and Shepotylo, O. (2023). How did Brexit affect UK trade? *Contemporary Social Science*, 18(2):1–18.
- Durbin, J. and Koopman, S. J. (2012). *Time Series Analysis by State Space Methods*. Oxford University Press.
- Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *The Annals of Statistics*, 32(2):407–451.
- Eurostat (2017). *Handbook on rapid estimates: 2017 edition*. Publications Office.
- Eurostat (2018). European Statistical System (ESS) guidelines on temporal disaggregation, benchmarking and reconciliation. Technical report, Eurostat.

- Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360.
- Fernandez, R. B. (1981). A methodological note on the estimation of time series. *The Review of Economics and Statistics*, 63(3):471–476.
- Ferrara, L. and Simoni, A. (2022). When are Google Data Useful to Nowcast GDP? An Approach via Preselection and Shrinkage. *Journal of Business & Economic Statistics*, 0(0):1–15.
- Fisher, A. J. (2015). Toward a dynamic model of psychological assessment: Implications for personalized care. *Journal of Consulting and Clinical Psychology*, 83(4):825.
- Forni, M. and Gambetti, L. (2021). Policy and business cycle shocks: A structural factor model representation of the us economy. *Journal of Risk and Financial Management*, 14(8):371.
- Froni, C. and Marcellino, M. (2014a). A comparison of mixed frequency approaches for nowcasting Euro area macroeconomic aggregates. *International Journal of Forecasting*, 30(3):554–568.
- Froni, C. and Marcellino, M. (2014b). Mixed-frequency structural models: Identification, estimation, and policy analysis. *Journal of Applied Econometrics*, 29(7):1118–1144.
- Froni, C., Marcellino, M., and Schumacher, C. (2015). Unrestricted mixed data sampling (MIDAS): MIDAS regressions with unrestricted lag polynomials. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 178(1):57–82.
- Frale, C., Marcellino, M., Mazzi, G. L., and Proietti, T. (2011). Euromind: a monthly indicator of the euro area economic conditions. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 174(2):439–470.

- Freeman, R., Manova, K., Prayer, T., Sampson, T., et al. (2022). Unravelling deep integration: UK trade in the wake of Brexit. *Working Paper, Centre for Economic Performance, London School of Economics and Political*.
- Friel, N., McKeone, J. P., Oates, C. J., and Pettitt, A. N. (2017). Investigation of the widely applicable bayesian information criterion. *Statistics and Computing*, 27(3):833–844.
- Fuleky, P. (2019). *Macroeconomic forecasting in the era of big data: Theory and practice*, volume 52. Springer.
- Fusacchia, I., Salvatici, L., and Winters, L. A. (2022). The consequences of the trade and cooperation agreement for the uk’s international trade. *Oxford Review of Economic Policy*, 38(1):27–49.
- Galvão, A. B. (2013). Changes in predictive ability with mixed frequency data. *International Journal of Forecasting*, 29(3):395–410.
- Geweke, J. (1977). The dynamic factor analysis of economic time series. In *Latent Variables in Socio-Economic Models*. North-Holland.
- Geweke, J. F. and Singleton, K. J. (1981). Maximum likelihood ”confirmatory” factor analysis of economic time series. *International Economic Review*, 22(1):37–54.
- Ghysels, E., Santa-Clara, P., and Valkanov, R. (2004). The MIDAS Touch: Mixed Data Sampling Regression Models. University of california at los angeles, anderson graduate school of management, Anderson Graduate School of Management, UCLA.
- Ghysels, E., Sinko, A., and Valkanov, R. (2007). Midas regressions: Further results and new directions. *Econometric reviews*, 26(1):53–90.
- Giannone, D., Reichlin, L., and Small, D. (2008). Nowcasting: The real-time informational content of macroeconomic data. *Journal of monetary economics*, 55(4):665–676.

- Grassi, S., Proietti, T., Frale, C., Marcellino, M., and Mazzi, G. (2015). EuroMInd-C: A disaggregate monthly indicator of economic activity for the Euro area and member countries. *International Journal of Forecasting*, 31(3):712–738.
- Guérin, P. and Marcellino, M. (2013). Markov-switching midas models. *Journal of Business & Economic Statistics*, 31(1):45–56.
- Guerrero, V. M. and Nieto, F. H. (1999). Temporal and contemporaneous disaggregation of multiple economic time series. *Test*, 8(2):459–489.
- Hamilton, J. D. (2020). *Time series analysis*. Princeton university press.
- Hand, D. J., Babb, P., Zhang, L.-C., Allin, P., Wallgren, A., Wallgren, B., Blunt, G., Garrett, A., Murtagh, F., Smith, P. W., et al. (2018). Statistical challenges of administrative and transaction data. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 181(3):555–605.
- Harvey, A. (1996). Intervention analysis with control groups. *International Statistical Review/Revue Internationale de Statistique*, 64(3):313–328.
- Harvey, A. C. (1990). *Forecasting, structural time series models and the Kalman filter*. Cambridge university press.
- Hastie, T., Taylor, J., Tibshirani, R., and Walther, G. (2007). Forward stagewise regression and the monotone lasso. *Electronic Journal of Statistics*, 1:1–29.
- Hastie, T., Tibshirani, R., and Wainwright, M. (2015). *Statistical learning with sparsity: the lasso and generalizations*. CRC press.
- Hecq, A., Ternes, M., and Wilms, I. (2022). Hierarchical regularizers for mixed-frequency vector autoregressions. *Journal of Computational and Graphical Statistics*, 31(4):1076–1090.

- Hesterberg, T., Choi, N. H., Meier, L., and Fraley, C. (2008). Least angle and l1 penalized regression: A review. *Statistics Surveys*, 2:61–93.
- Holmes, E. E., Ward, E. J., and Kellie, W. (2012). Marss: Multivariate autoregressive state-space models for analyzing time-series data. *The R Journal*, 4(1):11.
- Hopp, D. (2021). nowcastDFM. <https://github.com/dhopp1/nowcastDFM> [Accessed: (01/06/2023)].
- Hu, Y., Chi, E. C., and Allen, G. I. (2016). ADMM algorithmic regularization paths for sparse statistical machine learning. In *Splitting Methods in Communication, Imaging, Science, and Engineering*, pages 433–459. Springer International Publishing.
- Jarmin, R. S. (2019). Evolving measurement for an evolving economy: thoughts on 21st century US economic statistics. *Journal of Economic Perspectives*, 33(1):165–84.
- Jennrich, R. I. and Sampson, P. (1966). Rotation for simple loadings. *Psychometrika*, 31(3):313–323.
- Johansen, S. (1988). Statistical analysis of cointegration vectors. *Journal of economic dynamics and control*, 12(2-3):231–254.
- Jungbacker, B., Koopman, S.-J., and van der Wel, M. (2009). Dynamic factor analysis in the presence of missing data. Technical report, Tinbergen Institute Discussion Paper.
- Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23(3):187–200.
- Kariya, T. and Kurata, H. (2004). *Generalized least squares*. John Wiley & Sons.
- Kock, A. B. (2016). Consistent and conservative model selection with the adaptive lasso in stationary and nonstationary autoregressions. *Econometric Theory*, 32(1):243–259.

- Koop, G., McIntyre, S., and Mitchell, J. (2020a). UK regional nowcasting using a mixed frequency vector autoregressive model with entropic tilting. *Journal of the Royal Statistical Society: Series A*, 183(1):91–119.
- Koop, G., McIntyre, S., Mitchell, J., and Poon, A. (2020b). Regional output growth in the United Kingdom: More timely and higher frequency estimates from 1970. *Journal of Applied Econometrics*, 35(2):176–197.
- Koopman, S. J. and Durbin, J. (2000). Fast filtering and smoothing for multivariate state space models. *Journal of Time Series Analysis*, 21(3):281–296.
- Kose, M. A., Otrok, C., and Whiteman, C. H. (2008). Understanding the evolution of world business cycles. *Journal of International Economics*, 75(1):110–130.
- Krantz, S. and Bagdziunas, R. (2022). *dfms: Dynamic Factor Models*. R Package Version 0.1.3.
- Kremer, J., Lombardo, G., von Thadden, L., and Werner, T. (2006). Dynamic stochastic general equilibrium models as a tool for policy analysis. *CESifo Economic Studies*, 52(4):640–665.
- Kristensen, J. T. (2017). Diffusion indexes with sparse loadings. *Journal of Business & Economic Statistics*, 35(3):434–451.
- Kuzin, V., Marcellino, M., and Schumacher, C. (2011). MIDAS vs. mixed-frequency VAR: Nowcasting GDP in the euro area. *International Journal of Forecasting*, 27(2):529–542.
- Kuzin, V., Marcellino, M., and Schumacher, C. (2013). Pooling versus model selection for nowcasting GDP with many predictors: Empirical evidence for six industrialized countries. *Journal of Applied Econometrics*, 28(3):392–411.
- Labonne, P. and Weale, M. (2020). Temporal disaggregation of overlapping noisy

- quarterly data: estimation of monthly output from UK value-added tax data. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 183(3):1211–1230.
- Lê, S., Josse, J., and Husson, F. (2008). FactoMineR: A package for multivariate analysis. *Journal of Statistical Software*, 25(1):1–18.
- Lee, D. and Baldick, R. (2016). Load and wind power scenario generation through the generalized dynamic factor model. *IEEE Transactions on Power Systems*, 32(1):400–410.
- Lin, T., Ma, S., and Zhang, S. (2015). On the global linear convergence of the ADMM with MultiBlock variables. *SIAM Journal on Optimization*, 25(3):1478–1497.
- Linden, A. (2015). Conducting interrupted time-series analysis for single-and multiple-group comparisons. *The Stata Journal*, 15(2):480–500.
- Litterman, R. B. (1983). A random walk, markov model for the distribution of time series. *Journal of Business & Economic Statistics*, 1(2):169–173.
- Luciani, M. (2015). Monetary policy and the housing market: A structural factor analysis. *Journal of Applied Econometrics*, 30(2):199–218.
- Magnus, J. R. and Neudecker, H. (2019). *Matrix Differential Calculus with Applications in Statistics and Econometrics*. Wiley.
- Marcellino, M. (1999). Some consequences of temporal aggregation in empirical analysis. *Journal of Business & Economic Statistics*, 17(1):129–136.
- Marcellino, M. and Schumacher, C. (2010). Factor MIDAS for nowcasting and forecasting with ragged-edge data: A model comparison for German GDP. *Oxford Bulletin of Economics and Statistics*, 72(4):518–550.
- Mariano, R. S. and Murasawa, Y. (2003). A new coincident index of business cycles based on monthly and quarterly series. *Journal of applied Econometrics*, 18(4):427–443.

- Mariano, R. S. and Murasawa, Y. (2010). A coincident index, common factors, and monthly real GDP. *Oxford Bulletin of Economics and Statistics*, 72(1):27–46.
- Matheson, M. T. (2011). *New indicators for tracking growth in real time*. International Monetary Fund.
- Meinshausen, N. and Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473.
- Miralles, J. M. P., Lladosa, L.-E. V., and Vallés, R. E. (2003). On the performance of the Chow-Lin procedure for quarterly interpolation of annual data: Some Monte-Carlo analysis. *Spanish Economic Review*, 5(4):291–305.
- Mitchell, J., Smith, R. J., Weale, M. R., Wright, S., and Salazar, E. L. (2005). An indicator of monthly GDP and an early estimate of quarterly GDP growth. *The Economic Journal*, 115(501):F108–F129.
- Moauro, F. and Savio, G. (2005). Temporal disaggregation using multivariate structural time series models. *The Econometrics Journal*, 8(2):214–234.
- Molenaar, P. (1985). A dynamic factor model for the analysis of multivariate time series. *Psychometrika*, 50(2):181–202.
- Ng, S. (2013). Variable selection in predictive regressions. *Handbook of economic forecasting*, 2:752–789.
- ONS (2017a). Index of Services QMI. <https://www.ons.gov.uk/economy/nationalaccounts/uksectoraccounts/methodologies/indexofservicesqmi>[Accessed: (01/06/2023)].
- ONS (2017b). ONS methodology working paper series no. 11 – Identifying caravan homes in Zoopla data: June 2017. <https://www.ons.gov.uk/methodology/methodologicalpublications/generalmethodology/onsworkingpaperseries/>

- onsmethodologyworkingpaperseriesno11identifyingcaravanhomesinzoopladatajune2017[Accessed: (01/06/2023)].
- ONS (2019a). Faster indicators of UK economic activity: road traffic data for England. <https://datasciencecampus.ons.gov.uk/projects/faster-indicators-of-uk-economic-activity-road-traffic-data-for-england/>[Accessed: (01/06/2023)].
- ONS (2019b). Introducing GDP for the countries of the UK and the regions of England. <https://www.ons.gov.uk/economy/grossdomesticproductgdp/methodologies/introducinggdpforthecountriesoftheukandtheregionsofengland>[Accessed: (01/06/2023)].
- ONS (2021). Research into the use of scanner data for constructing UK consumer price statistics. <https://www.ons.gov.uk/economy/inflationandpriceindices/articles/researchintotheuseofscannerdataforconstructingukconsumerpricestatistics/2021-04-06>[Accessed: (01/06/2023)].
- ONS (2022a). Economic activity and social change in the UK, real-time indicators: 1 December 2022. <https://www.ons.gov.uk/economy/economicoutputandproductivity/output/bulletins/economicactivityandsocialchangeintheukrealtimeindicators/1december2022>[Accessed: (01/06/2023)].
- ONS (2022b). GDP monthly estimate, UK: September 2022. <https://www.ons.gov.uk/economy/grossdomesticproductgdp/bulletins/gdpmonthlyestimateuk/september2022>[Accessed: (01/06/2023)].
- ONS (2022c). UK National Accounts, The Blue Book: 2022. <https://www.ons.gov.uk/economy/grossdomesticproductgdp/bulletins/bluebook>

- [//www.ons.gov.uk/economy/grossdomesticproductgdp/compendium/unitedkingdomnationalaccountsthebluebook/2022](http://www.ons.gov.uk/economy/grossdomesticproductgdp/compendium/unitedkingdomnationalaccountsthebluebook/2022)[Accessed: (01/06/2023)].
- ONS (2022d). Value Added Tax (VAT) flash estimates. <https://www.ons.gov.uk/economy/economicoutputandproductivity/output/datasets/valueaddedtaxvatflashestimates>[Accessed: (01/06/2023)].
- Pavía-Miralles, J. M. and Cabrer-Borrás, B. (2007). On estimating contemporaneous quarterly regional gdp. *Journal of Forecasting*, 26(3):155–170.
- Pavía-Miralles, J. M. et al. (2010). A survey of methods to interpolate, distribute and extra-polate time series. *Journal of Service Science and Management*, 3(04):449.
- Pfeffermann, D. (2015). Methodological issues and challenges in the production of official statistics: 24th annual morris hansen lecture. *Journal of Survey Statistics and Methodology*, 3(4):425–483.
- Poncela, P., Ruiz, E., and Miranda, K. (2021). Factor extraction using Kalman filter and smoothing: This is not just another survey. *International Journal of Forecasting*, 37(4):1399–1425.
- Proietti, T. (2006). Temporal disaggregation by state space methods: Dynamic regression methods revisited. *The Econometrics Journal*, 9(3):357–372.
- Proietti, T. (2011a). Estimation of common factors under cross-sectional and temporal aggregation constraints. *International Statistical Review*, 79(3):455–476.
- Proietti, T. (2011b). Multivariate temporal disaggregation with cross-sectional constraints. *Journal of Applied Statistics*, 38(7):1455–1466.
- Proietti, T. and Moauro, F. (2006). Dynamic factor analysis with non-linear temporal aggregation constraints. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 55(2):281–300.

- Quenneville, B., Picard, F., and Fortier, S. (2013). Calendarization with interpolating splines and state space models. *Journal of the Royal Statistical Society: SERIES C: Applied Statistics*, pages 371–399.
- Quilis, E. M. (2018). Temporal disaggregation of economic time series: The view from the trenches. *Statistica Neerlandica*, 72(4):447–470.
- Reid, S., Tibshirani, R., and Friedman, J. (2016). A study of error variance estimation in lasso regression. *Statistica Sinica*, 26(1):35–67.
- Reis, R. and Watson, M. W. (2010). Relative goods’ prices, pure inflation, and the phillips correlation. *American Economic Journal: Macroeconomics*, 2(3):128–157.
- Rohe, K. and Zeng, M. (2020). Vintage factor analysis with varimax performs statistical inference. *arXiv Preprint arXiv:2004.05387*.
- Rossi, N. et al. (1982). A note on the estimation of disaggregate time series when the aggregate is known. *The Review of Economics and Statistics*, 64(4):695–696.
- Rünstler, G. (2016). On the design of data sets for forecasting with dynamic factor models. 35:629–662.
- Sargent, T. J., Sims, C. A., et al. (1977). Business cycle modeling without pretending to have too much a priori economic theory. *New Methods in Business Cycle Research*, 1:145–168.
- Sax, C. and Eddelbuettel, D. (2018). Seasonal Adjustment by X-13ARIMA-SEATS in R. *Journal of Statistical Software*, 87(11):1–17.
- Sax, C. and Steiner, P. (2013). Temporal disaggregation of time series. *The R Journal*, 5(2).
- Schorfheide, F. and Song, D. (2015). Real-time forecasting with a mixed-frequency var. *Journal of Business & Economic Statistics*, 33(3):366–380.

- Schumacher, C. (2011). Forecasting with factor models estimated on large datasets: a review of the recent literature and evidence for german gdp. *Jahrbücher für Nationalökonomie und Statistik*, 231(1):28–49.
- Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.
- Shumway, R. H. and Stoffer, D. S. (1982). An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, 3(4):253–264.
- Silva, J. S. and Cardoso, F. (2001). The chow-lin method using dynamic models. *Economic modelling*, 18(2):269–280.
- Silva, L., Figueiredo Filho, D., and Fernandes, A. (2020). The effect of lockdown on the covid-19 epidemic in brazil: evidence from an interrupted time series design. *Cadernos de Saúde Pública*, 36:e00213920.
- Stock, J. H. and Watson, M. (2011). Dynamic factor models. *Oxford Handbooks Online*.
- Stock, J. H. and Watson, M. W. (2002). Forecasting using principal components from a large number of predictors. *Journal of the American Statistical Association*, 97(460):1167–1179.
- Stram, D. O. and Wei, W. W. (1986). A methodological note on the disaggregation of time series totals. *Journal of Time Series Analysis*, 7(4):293–302.
- Strang, G. (1993). *Introduction to linear algebra*, volume 3. Wellesley-Cambridge Press Wellesley, MA.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

- Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108.
- Tsay, R. S. (2013). *Multivariate time series analysis: with R and financial applications*. John Wiley & Sons.
- UK Parliament (2021). Brexit timeline: events leading to the UK’s exit from the European Union. <https://commonslibrary.parliament.uk/research-briefings/cbp-7960/>[Accessed: (01/06/2023)].
- UK Parliament (2023). Trade: Key economic indicators. <https://commonslibrary.parliament.uk/research-briefings/sn02815/>[Accessed: (01/06/2023)].
- U.S. Census Bureau (2022). X-13ARIMA-SEATS Seasonal Adjustment Program. <https://www.census.gov/data/software/x13as.References.html>[Accessed: (01/06/2023)].
- van de Geer, S., Bühlmann, P., Ritov, Y., and Dezeure, R. (2014). On asymptotically optimal confidence regions and tests for high-dimensional models. *The Annals of Statistics*, 42(3):1166 – 1202.
- van de Geer, S., Bühlmann, P., and Zhou, S. (2011). The adaptive and the thresholded Lasso for potentially misspecified models (and a lower bound for the Lasso). *Electronic Journal of Statistics*, 5:688 – 749.
- Watson, M. W. and Engle, R. F. (1983). Alternative algorithms for the estimation of dynamic factor, mimic and varying coefficient regression models. *Journal of Econometrics*, 23(3):385–400.
- Wickham, H., Hester, J., Chang, W., and Bryan, J. (2022). *devtools: Tools to Make Developing R Packages Easier*. R Package Version 2.4.5.

- Wu, H., Chan, S., Tsui, K., and Hou, Y. (2013). A new recursive dynamic factor analysis for point and interval forecast of electricity price. *IEEE Transactions on Power Systems*, 28(3):2352–2365.
- Yu, G. and Bien, J. (2019). Estimating the error variance in a high-dimensional linear model. *Biometrika*, 106(3):533–546.
- Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.
- Zheng, Z., Fan, Y., and Lv, J. (2014). High dimensional thresholded regression and shrinkage effect. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, 76(3):627–649.
- Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429.
- Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286.
- Zou, Hui and Hastie, Trevor (2012). Elastic-Net for sparse estimation and sparse PCA. <https://CRAN.R-project.org/package=devtools>[Accessed: (01/06/2023)].