Parsimonious Time Series Modelling of High-dimensional Data with Linear and Non-Linear Models

Yiyong Luo

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

Department of Statistical Science

University College London

December 16, 2025

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

Abstract

Time series models perform statistical analysis and predictions based on sequential data, with two trends emerging in their development: high dimensionality and non-linearity. The first trend results from advances in data collection and storage capabilities, and the second arises from the dynamic evolution of sequential data that challenges traditional static linear models. In this thesis, we study these two trends in Bayesian Vector Autoregression (VAR), chosen for its flexibility and widespread applications across fields. The Bayesian framework enables the incorporation of well-designed priors for unknown parameters while providing insight into parameter uncertainty. The first part of the thesis focuses on addressing the proliferation of parameters, a challenge inherent to high dimensionality, through tensor VAR, which treats the coefficient matrix as a third-order tensor and conducts tensor decomposition to achieve parsimony. We apply different Bayesian techniques to induce shrinkage and achieve convergent Markov chains. In the second part, we introduce time-varying parameterization to the tensor VAR to capture the evolving interrelation among time series, enabling the model to accommodate both trends. The final part explores factor augmented VAR (FAVAR), an alternative VAR framework that reduces dimensionality by extracting factors from high-dimensional data. We develop a novel approach which combines time-varying parameterization in the VAR component with a proposed Grouped Sparse autoencoder, alleviating identifiability and interpretability issues of the standard autoencoder. To demonstrate the merits of the models proposed in these three parts, we apply them to macroeconomic data and functional magnetic resonance imaging data.

Impact Statement

Time series is a prominent research area in statistics, driven by its broad applications across disciplines such as econometrics, finance, neuroscience, and meteorology, among others. Recent advances in data collection and storage, along with the demand for capturing complex patterns, have led to two trends in time series modeling: high-dimensionality and non-linearity. This thesis aims to accommodate these two trends in time series modeling and retain the interpretability of the inferred models.

Chapters 3 and 4 concentrate on vector autoregression (VAR), a multivariate time series model which connects the time series with their lagged values linearly. The trend of high-dimensionality poses the challenge of over-parameterization to this model, as the number of parameters grows quadratically with the number of time series. To address this issue, we adopt the tensor VAR, a recent VAR variant that introduces tensor decomposition for dimension reduction and model parsimony. Chapter 3 introduces a novel Bayesian inference framework using a Markov Chain Monte Carlo (MCMC) scheme that efficiently estimates unknown parameters, resolves the mixing issue in tensor components, and allows for effective rank inference, an important parameter in the tensor decomposition. This contribution advances the development of Bayesian methods and the interpretation of statistical models with tensor structures. Chapter 4 extends this framework to the non-linear setting of time-varying parameter VAR (TVP-VAR), where over-parameterization is even more severe due to its dependence on both the number of variables and time points. By considering multiple tensor decompositions, we mitigate this issue and offer a new approach to statistical model selection by incorporating an information criterion and knee point detection. This work makes a contribution to the modeling of time-varying tensor components, a relatively underexplored area in time series research. We apply the proposed model to functional magnetic resonance imaging (fMRI) data, uncovering evolving connectivity patterns across brain regions.

Chapter 5 shifts focus to factor-augmented VAR (FAVAR), a model widely used in econometrics to handle high-dimensional time series via latent factor structures. Conventional FAVAR approaches rely on linear factor extraction, but we propose a novel framework using autoencoders to extract non-linear factors. The autoencoder is specifically designed to overcome the identifiability issues of standard implementations, thereby enhancing both interpretability and utility. This work not only addresses the dual challenges of high-dimensionality and non-linearity but also contributes to the emerging literature on integrating deep learning with time series analysis. Applied to U.S. economic data, our method improves forecasting performance and reveals a time-varying effect of monetary policy, highlighting its evolving impact over recent decades.

Acknowledgements

First of all, I would like to express my deepest and sincerest gratitude to my supervisors, Prof. Jim E. Griffin and Dr. Brooks Paige. Their insightful guidance, intellectual generosity, and constant encouragement have shaped every stage of my Ph.D. journey. I feel incredibly fortunate to have learned from them, and this work would not have been possible without their guidance. I sincerely hope to stay connected and continue learning from their wisdom, and I would be honored to have the opportunity to give back in the future.

I would like to thank the Department of Statistical Science at University College London for providing an enriching environment to study, conduct research, and learn from the brightest minds in the field. I am also grateful to Ms. Marina Lewis and Prof. Terry Soo for their kind support with administrative matters.

I am thankful to all the reviewers and audience members who engaged with my work during journal reviews, conferences, and seminars. Your constructive feedback and insightful questions have greatly contributed to shaping and strengthening this research.

To my parents, Bin Luo and Wenlei Jiang – thank you for supporting me throughout my study, celebrating every step of progress in my research, and for always believing in my potential. You have been my harbour through every challenge. I am also deeply grateful to my grandparents, Ruyu Ding, Guiyun Liu, Jiajun Jiang, and Shousheng Luo – your hopes and care fueled my dedication to science.

I would also like to thank Yijie Li, for the constant intellectual exchange and your passion for technology, which brightened my world and inspired me to keep pursuing knowledge. To Yijie and my friends – spending time together brought me joy and balance, making this journey more enjoyable and fulfilling.

Contents

List of Figures

List of Tables

Acronyms

Adam Adaptive Moment Estimation

ASIS Ancillarity-Sufficiency Interweaving Strategy

CP CANDECOMP/PARAFAC

DFM Dynamic Factor Model

ELBO Evidence Lower Bound

FAVAR Factor-augmented Factor Autoregression

FFBS Forward-filtering Backward-sampling

fMRI Functional Magnetic Resonance Imaging

GFC Global Financial Crisis

GS Grouped Sparse

IRF Impulse Response Function

M-DGDP Multiway Dirichlet Generalized Double Pareto

MCMC Markov Chain Monte Carlo

MGP Multiplicative Gamma Prior

MLP Multilayer Perceptron

NG Normal-gamma

PCA Principal Component Analysis

ROI Region of Interest

SSVS Stochastic Search Variable Selection

TIV Time-invariant

TVAR Tensor Vector Autoregression

TVP-VAR Time-varying Parameter Vector Autoregression

VAR Vector Autoregression

UCL Research Paper Declaration Form: referencing the doctoral candidate's own published work(s)

1. **For a research manuscript that has already been published** (if not yet published, please skip to section 2):

   (a) **What is the title of the manuscript?** Bayesian Inference of Vector Autoregressions with Tensor Decompositions.

   (b) **Please include a link to or doi for the work:**
   https://doi.org/10.1080/07350015.2024.2447302

   (c) **Where was the work published?** Journal of Business & Economic Statistics.

   (d) **Who published the work?** Taylor & Francis.

   (e) **When was the work published?** February, 2025.

   (f) **List the manuscript's authors in the order they appear on the publication:** Yiyong Luo, Jim E. Griffin.

   (g) **Was the work peer reviewed?** Yes.

   (h) **Have you retained the copyright?** Yes.

   (i) **Was an earlier form of the manuscript uploaded to a preprint server (e.g. medRxiv)? If 'Yes', please give a link or doi**:
   Yes. https://doi.org/10.48550/arXiv.2211.01727
   If 'No', please seek permission from the relevant publisher and check the box next to the below statement:

   □ *I acknowledge permission of the publisher named under 1d to include in this thesis portions of the publication named as included in 1c.*

2. **For a research manuscript prepared for publication but that has not yet been published** (if already published, please skip to section 3):

   (a) **What is the current title of the manuscript?**

   (b) **Has the manuscript been uploaded to a preprint server 'e.g. medRxiv' ?**
   If 'Yes', please please give a link or doi:

(c) **Where is the work intended to be published?**

(d) **List the manuscript's authors in the intended authorship order:**

(e) **Stage of publication:**

3. **For multi-authored work, please give a statement of contribution covering all authors** (if single-author, please skip to section 4): Yiyong Luo proposed the methods, implemented the model, conducted data analysis, and wrote the paper. Jim E. Griffin supervised the work, provided guidance and feedback during the project, and contributed to the writing of the paper.

4. **In which chapter(s) of your thesis can this material be found?** Chapter 3.

e-Signatures confirming that the information above is accurate (this form should be co-signed by the supervisor/ senior author unless this is not appropriate, e.g. if the paper was a single-author work):

Candidate: Yiyong Luo
Date: 01/06/2025

Supervisor/Senior Author signature (where appropriate): Jim E. Griffin
Date: 21/06/2025

UCL Research Paper Declaration Form: referencing the doctoral candidate's own published work(s)

1. **For a research manuscript that has already been published** (if not yet published, please skip to section 2):

   (a) **What is the title of the manuscript?**

   (b) **Please include a link to or doi for the work:**

   (c) **Where was the work published?**

   (d) **Who published the work?**

   (e) **When was the work published?**

   (f) **List the manuscript's authors in the order they appear on the publication:**

   (g) **Was the work peer reviewd?**

   (h) **Have you retained the copyright?**

   (i) **Was an earlier form of the manuscript uploaded to a preprint server (e.g. medRxiv)? If 'Yes', please give a link or doi**
   If 'No', please seek permission from the relevant publisher and check the box next to the below statement:

   □ *I acknowledge permission of the publisher named under 1d to include in this thesis portions of the publication named as included in 1c.*

2. **For a research manuscript prepared for publication but that has not yet been published** (if already published, please skip to section 3):

   (a) **What is the current title of the manuscript?** Time-varying Parameter Tensor Vector Autoregression.

   (b) **Has the manuscript been uploaded to a preprint server 'e.g. medRxiv' ?**
   If 'Yes', please please give a link or doi:
   Yes. https://doi.org/10.48550/arXiv.2505.07975

   (c) **Where is the work intended to be published?** Journal of the Royal Statistical Society, Series B (Statistical Methodology).

(d) **List the manuscript's authors in the intended authorship order:** Yiyong Luo, Jim E. Griffin.

(e) **Stage of publication:** Prepare for submission.

3. **In which chapter(s) of your thesis can this material be found?** Chapter 4

4. **For multi-authored work, please give a statement of contribution covering all authors** (if single-author, please skip to section 4): Yiyong Luo completed the methodological details of the model, wrote the code for simulation and real data application, and wrote the manuscript. Jim E. Griffin proposed time-varying parameterization of the model specified in Chapter 4, provided guidance and feedback during the project, and contributed to the writing of the manuscript.

e-Signatures confirming that the information above is accurate (this form should be co-signed by the supervisor/ senior author unless this is not appropriate, e.g. if the paper was a single-author work):

Candidate: Yiyong Luo
Date: 01/06/2025

Supervisor/Senior Author signature (where appropriate): Jim E. Griffin
Date: 21/06/2025

UCL Research Paper Declaration Form: referencing the doctoral candidate's own published work(s)

1. **For a research manuscript that has already been published** (if not yet published, please skip to section 2):

   (a) **What is the title of the manuscript?**

   (b) **Please include a link to or doi for the work:**

   (c) **Where was the work published?**

   (d) **Who published the work?**

   (e) **When was the work published?**

   (f) **List the manuscript's authors in the order they appear on the publication:**

   (g) **Was the work peer reviewd?**

   (h) **Have you retained the copyright?**

   (i) **Was an earlier form of the manuscript uploaded to a preprint server (e.g. medRxiv)? If 'Yes', please give a link or doi**

   If 'No', please seek permission from the relevant publisher and check the box next to the below statement:

   □ *I acknowledge permission of the publisher named under 1d to include in this thesis portions of the publication named as included in 1c.*

2. **For a research manuscript prepared for publication but that has not yet been published** (if already published, please skip to section 3):

   (a) **What is the current title of the manuscript?** Time-varying Factor Augmented Vector Autoregression with Grouped Sparse Autoencoder

   (b) **Has the manuscript been uploaded to a preprint server 'e.g. medRxiv' ?**
   If 'Yes', please please give a link or doi:
   Yes. https://doi.org/10.48550/arXiv.2503.04386

   (c) **Where is the work intended to be published?** Journal of Business & Economic Statistics.

(d) **List the manuscript's authors in the intended authorship order:** Yiyong Luo, Brooks Paige, Jim E. Griffin.

(e) **Stage of publication:** Prepare for submission.

3. **For multi-authored work, please give a statement of contribution covering all authors** (if single-author, please skip to section 4): Yiyong Luo proposed the model, wrote the code for the real data application, and wrote the manuscript. Brooks Paige and Jim E. Griffin provided guidance and feedback during the project, and contributed to the writing of the manuscript.

4. **In which chapter(s) of your thesis can this material be found?** Chapter 5

e-Signatures confirming that the information above is accurate (this form should be co-signed by the supervisor/ senior author unless this is not appropriate, e.g. if the paper was a single-author work):

Candidate: Yiyong Luo
Date: 01/06/2025

Supervisor/Senior Author signature (where appropriate): Jim E. Griffin
Date: 21/06/2025

Chapter 1

Introduction

Time series refer to sequences of data points indexed in temporal order. The analysis of time series has developed into a broad area of statistical research, encompassing a variety of research areas driven by a range of objectives, including, but not limited to, forecasting, nowcasting, causal inference, and change point detection. These time series analyses have been applied to many disciplines. For example, to monitor economic conditions (Bok et al., 2018), and to detect dynamic connectivity between brain regions (Seth et al., 2015). Accurately modeling temporal structure is a fundamental goal in time series analysis. Among the many methodological developments in this field, two trends have emerged: high dimensionality and non-linearity.

The first trend emerges from advancements in both hardware and software, which have enabled large-scale data collection and storage. In time series analysis, dimensionality can be understood along two primary axes: the temporal length and the number of variables. High dimensionality may therefore refer to (1) a large number of observations, (2) a large number of time series, or (3) a combination of both. This thesis focuses on the second case[1], where there are many variables at one time point, but the overall lengths of the time series are relatively short. If the research requires joint modeling of these time series, this trend potentially introduces over-parameterization, where an excessive number of parameters undermines model robustness and degrades out-of-sample performance. To address this issue, several approaches have been proposed to achieve parsimonious models, including regularization methods in the frequentist framework (Basu and Matteson, 2021), shrinkage priors in the Bayesian framework (Korobilis et al., 2022), and dimension reduction techniques (Ashraf et al., 2023), which are generally applicable in both frameworks.

The second trend, non-linearity, stems from the demand to model time series to better re-

[1] For a review of the other two cases, see Peña and Tsay (2021).

flect real-world complexity, thereby enabling more accurate analysis (Cheng et al., 2015). For example, Koop et al. (1996) advocated non-linear modeling because economic shocks may behave differently during recessions than during expansions. Additional sources of non-linearity include structural breaks, extreme events, and gradual changes driven by human behavior. In this thesis, non-linear models refer to those that capture non-linear and/or time-varying relationships among time series. In particular, models with time-varying parameters are classified as non-linear even when the underlying relationships are linear, because identical inputs to the linear function can produce different outputs at different time points.

The time series model considered in this thesis is the vector autoregression (VAR) (Sims, 1980), whose standard form models the linear interrelationship between multivariate time series and their lagged values. This thesis contributes to the methodological development of VAR models for three reasons. First, VARs provide flexible yet interpretable frameworks for capturing dynamic interactions among time series, supported by well-established estimation methods. Second, VARs support a wide range of empirical analyses—such as forecasting (Clark, 2011) and Granger causality (Granger, 1969) (introduced in Chapter 2.1.3)—which are useful to the econometric and neuroscience applications studied in this thesis. While alternative multivariate time series models (e.g., vector autoregressive moving average models; see a review in Düker et al., 2025) can also accommodate these analyses, their model specifications and implementations are typically less straightforward than those of VARs. Third, VARs are connected to many widely used multivariate time series models. For example, a VAR with time-varying parameters (which is a non-linear time series model in this thesis) (Primiceri, 2005) can be formulated as a state-space model; the vector error correction model is essentially a VAR modified to incorporate cointegration relationships[2]. These connections allow methodological insights developed for VARs to be extended to broader model classes. For these reasons, this thesis focuses on VAR. Readers interested in other multivariate time series models are referred to Lütkepohl (2013) and Tsay (2013).

This thesis estimates the parameters of VARs using Bayesian inference, implemented via Markov Chain Monte Carlo (MCMC). We adopt the Bayesian approach, rather than the frequen-

[2] The thesis will not cover cointegrated time series because the data sets analyzed either mitigate cointegration through appropriate data transformations or are inherently not cointegrated. For a comprehensive treatment of cointegration, see Juselius (2006).

tist one, for three reasons. First, embedding prior knowledge to VARs is particularly useful. For instance, the Minnesota prior (Litterman, 1979) encodes beliefs about the relative importance of lagged variables across different series, while hierarchical priors can facilitate model selection (Giannone et al., 2015). Second, the Bayesian framework treats parameters as random variables, allowing for direct quantification of uncertainty through posterior distributions and credible intervals. Third, Bayesian inference is the standard estimation framework of some specific models considered in this thesis. For example, time-varying parameter VAR (TVP-VAR) (Primiceri, 2005) uses the Gibbs sampler to simulate parameters, offering more stability relative to the frequentist method when the parameters evolve slowly over time.

We incorporate the trend of high dimensionality in VARs by considering two VAR-based frameworks: the tensor VAR (Wang et al., 2022a) and the factor-augmented VAR (FAVAR) (Bernanke et al., 2005). Both frameworks can be viewed as dimension reduction methods, but they operate on different spaces: the tensor VAR reduces the dimension of the parameter space, whereas the FAVAR extracts factors from the high-dimensional data to reduce the dimension of the data space. Studying both therefore allows the thesis to accommodate the high-dimensionality trend from two perspectives. Although these frameworks introduce parsimony on their own, we further enhance it by imposing shrinkage priors on the parameters. By combining dimension reduction with shrinkage, our study contributes to these two streams of high-dimensional time series modeling.

Regarding the trend of non-linearity, this thesis considers non-linear models in both the time series literature and deep learning literature. Within the tensor VAR framework, we study non-linearity by incorporating the idea of the TVP-VAR, which models parameters as random walks. We focus on the TVP-VAR because it is one of the most prominent non-linear extensions of the VAR, making it a natural direction in which to extend the time-invariant tensor VAR. Compared with other non-linear VAR models—such as the threshold VAR (Tsay, 1998) and the Markov-switching VAR (Krolzig and Krolzig, 1997)—the TVP-VAR is more straightforward to implement: it does not need to specify separate regimes, and its Bayesian inference can be carried out using the Kalman filter for state-space models (West and Harrison, 1997), which is relatively simple to compute. Within the FAVAR framework, while principal component analysis is typically used as a linear method to extract factors, we instead employ a deep learning approach, namely the autoencoder (Hinton and Salakhutdinov, 2006), to obtain more representative factors. Although other non-linear dimension-reduction methods—such as locally linear embedding (Roweis and Saul, 2000) from the manifold-learning literature and diffusion models (Ho et al., 2020) from the deep-learning literature—could be applied, We choose the autoencoder because prior research in econometrics has demonstrated its superior performance (Hauzenberger et al., 2023a), and its simple implementation is well-suited to macroeconomic applications where the number of observations is limited. In addition to the autoencoder, the VAR part of the FAVAR is modeled as a TVP-VAR to assume the evolution of factors to be dynamic.

This thesis is organized as follows. Chapter 2 reviews the literature on VAR and FAVAR models, covering both their standard formulations and extensions regarding the two trends. This chapter also introduces tensor decomposition, the dimension reduction technique which will be applied to tensor VARs, and outlines the deep learning methods used in the FAVAR. Chapter 3 adopts the tensor VAR (TVAR) to address the over-parameterization issue. An MCMC scheme is proposed to adaptively determine the tensor rank and to yield interpretable inferential results. Chapter 4 extends the TVAR framework by allowing its first moment to vary over time, contributing to the literature on TVP-VARs with high-dimensional settings. Chapter 5 focuses on the FAVAR model, incorporating a grouped sparse autoencoder for semi-identifiable and interpretable factor extraction. To capture the evolving dynamics of the factors, a TVP-VAR structure is employed. Finally, Chapter 6 concludes the thesis and outlines potential directions for future research.

We close this chapter by introducing the notations. For any positive integer $N \in \mathbb{N}$, we denote $[N] = 1, \ldots, N$. The scalar, vector and matrix are denoted as x, $\boldsymbol{x}$, and $\boldsymbol{X}$. The i-th and (i, j) element in $\boldsymbol{x}$ and $\boldsymbol{X}$ are denoted as $\boldsymbol{x}_i$ and $\boldsymbol{X}_{(i,j)}$, respectively. The N-by-N identity matrix is written as $\boldsymbol{I}_N$. The kronecker product of two matrices, $\boldsymbol{A} \in \mathbb{R}^{M \times N}$ and $\boldsymbol{B} \in \mathbb{R}^{I \times J}$, is represented as $\boldsymbol{A} \otimes \boldsymbol{B} = \begin{pmatrix} \boldsymbol{A}_{(1,1)}\boldsymbol{B} & \boldsymbol{A}_{(1,2)}\boldsymbol{B} & \cdots & \boldsymbol{A}_{(1,N)}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{A}_{(M,1)}\boldsymbol{B} & \boldsymbol{A}_{(M,2)}\boldsymbol{B} & \cdots & \boldsymbol{A}_{(M,N)}\boldsymbol{B} \end{pmatrix} \in \mathbb{R}^{MI \times NJ}$. We denote the set $S_{>a} = \{s \in S : s > a\}$, where $a \in \mathbb{R}$; similar notation is used for $S_{<a}$, $S_{\geq a}$, and $S_{\leq a}$.

Chapter 2

Literature Review

This chapter provides the literature review of vector autoregression (VAR) and its extension factor-augmented VAR (FAVAR). It also introduces the methodological background of tensor decomposition and deep learning, whose techniques will be applied in the subsequent chapters. The paradigms are outlined in the following paragraphs before reviewing these research areas.

Both VAR and FAVAR offer flexible approaches for capturing the dynamic interrelationships among time series. For each framework, the standard linear specification will be presented, followed by the Bayesian inference and applications. The extensions that accommodate high-dimensionality and non-linear dynamics will then be introduced. In this thesis, high dimensionality refers to settings in which the number of parameters is large relative to the number of observations. Linearity is defined as modeling a time series as a linear function of historical observations or latent factors with time-invariant parameters; otherwise, the model is considered non-linear. In particular, linear functions with time-varying parameters are classified as non-linear, as identical inputs to the linear function can produce different outputs at different time points.

Within the VAR and FAVAR literature, this review focuses on parametric models designed for time series observed at discrete time intervals. Parametric VAR and FAVAR models form the foundational structures of these frameworks and are widely employed across various research domains due to their simplicity. Nonparametric techniques, such as principal component analysis, can be incorporated, but the overarching model frameworks remain parametric. Recent developments have introduced nonparametric stochastic processes to model non-linear VARs; these will be briefly discussed in this chapter. Time series with discrete time intervals are defined as $\boldsymbol{y}_{1:T} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T) \in \mathbb{R}^T$, where each data point $t \in \mathbb{N}_{\leq T}$. Although certain continuous-time stochastic processes, such as the Ornstein-Uhlenbeck process (Uhlenbeck and

Ornstein, 1930), exhibit autoregressive behavior, such stochastic processes are beyond the scope of this review. This is because the time series examined in this thesis are low-frequency data from econometrics and neuroscience, which are appropriately treated within a discrete-time framework.

Tensor decompositions and autoencoders in the deep learning literature are two distinct approaches to dimensionality reduction. This thesis applies the former in Chapter 3 and 4, and the latter in Chapter 5. To facilitate comprehension of these approaches in the subsequent chapters, this chapter reviews key concepts, optimization methods, and applications in both research areas. The review of tensor decompositions focuses on two widely used methods: CANDECOMP/PARAFAC (CP) and Tucker decompositions. The CP is included due to its direct application in this thesis; though the Tucker is not implemented, it is closely related to the CP and is often referenced when specifying the CP. Other decompositions, which differ more substantially from the CP than the Tucker does, will not be discussed in this chapter. For an in-depth review of other tensor decompositions, readers are referred to Ji et al. (2019). The deep learning literature discussed in this chapter will focus on some foundational architectures, including the multilayer perceptron (MLP) and the standard autoencoder. While more expressive neural networks and architectures exist, they fall outside the scope of this thesis. This is because the use of autoencoders in macroeconomic research, the focus of Chapter 5, is still in its early stages, making it essential to first establish a solid methodological foundation. Up-to-date models beyond the MLP and autoencoders can be found in Goodfellow et al. (2016), Zhang et al. (2023), and Zhao et al. (2023).

The next four sections are organized as follows. Section 2.1 and 2.2 review VARs and FAVARs, respectively. Section 2.3 provides the methodological background of tensor decompositions, followed by Section 2.4, which focuses on deep learning.

2.1 Vector Autoregression

2.1.1 Modeling of Standard VAR

Vector autoregression (VAR) is a multivariate time series model that describes the linear interrelationship of the data. Since the advocacy of Sims (1980), it soon became the workhorse in econometrics due to its flexibility, which avoided the restrictive assumptions of traditional economic models, such as the dynamic simultaneous equation models developed by the Cowles

Commission (Achinstein, 1961). VARs have also gained its popularity in neuroscience as the key challenge in this field has evolved from identifying regional activations to analyzing functional connectivity underlying cognition, behavior, and consciousness (Seth et al., 2015). Beyond econometrics and neuroscience, VARs are applied in finance for modeling market dynamics (Carriero et al., 2012).

Let $\boldsymbol{y}_t \in \mathbb{R}^N$ $(t = P+1, \ldots, T)$ be the multivariate time series of interest, a standard reduced-form VAR with P lags, VAR(P), is written as

$$\boldsymbol{y}_t = \boldsymbol{\nu} + \boldsymbol{A}_1 \boldsymbol{y}_{t-1} + \cdots + \boldsymbol{A}_P \boldsymbol{y}_{t-P} + \boldsymbol{\epsilon}_t, \tag{2.1}$$

where $\boldsymbol{\nu} \in \mathbb{R}^N$ denotes the intercept terms, $\boldsymbol{A} = (\boldsymbol{A}_1, \ldots, \boldsymbol{A}_P) \in \mathbb{R}^{N \times NP}$ stores the coefficients which linearly connect the data and its lags. The error term $\boldsymbol{\epsilon}_t$ (or *innovation*) typically follows a Gaussian distribution, $\mathcal{N}(\mathbf{0}, \boldsymbol{\Omega})$. While this error term can follow non-Gaussian distributions (see Chiu et al. (2017) for Student t-distribution and Eltoft et al. (2006) for the Laplace distribution), this thesis adopts Gaussian errors because this distribution choice is standard in VARs. Moreover, other distributions aforementioned can be considered as Gaussian scale mixtures[3], so the inference based on Gaussian errors can be extended to these cases.

While the VAR can be applied to a variety of empirical contexts without requiring stringent assumptions about the underlying data, this model possesses useful properties and demonstrates generality under the *stability* condition. A VAR(P) is *stable* when all the roots of its characteristic polynomial are outside a unit circle, i.e., for all $z \in \mathbb{C}$ such that $\det\left(\boldsymbol{I}_N - \boldsymbol{A}_1 z - \cdots - \boldsymbol{A}_P z^P\right) = 0$, the modulus of z is greater than 1. Stability of a VAR implies that elements in $\boldsymbol{A}_p$ decay to 0 if p is sufficiently large[4]. The first property of a stable VAR(P) is that this model has a moving average (MA) representation,

$$\boldsymbol{y}_t = \sum_{q=0}^{\infty} \boldsymbol{\Phi}_q \boldsymbol{\nu} + \sum_{q=0}^{\infty} \boldsymbol{\Phi}_q \boldsymbol{\epsilon}_{t-q}, \tag{2.2}$$

where $\boldsymbol{\Phi}_0 = \boldsymbol{I}_N$, $\boldsymbol{\Phi}_q = \sum_{q'=1}^{q} \boldsymbol{\Phi}_{q-q'} \boldsymbol{A}_{q'}$, for $q > 0$. This representation is useful because it provides closed-form expressions of the unconditional mean and autocovariance of $\boldsymbol{y}_t$: $\mathbb{E}[\boldsymbol{y}_t] =$

[3] The error $\boldsymbol{\epsilon}_t$ $(t \in [T])$ is a Gaussian scale mixture when $\boldsymbol{\epsilon}_t \mid \tau \sim \mathcal{N}(\mathbf{0}, \tau\boldsymbol{\Sigma})$, where τ is a positive hyperparameter following some distributions, and $\boldsymbol{\Sigma}$ is a positive-definite matrix. If we marginalize out τ, the distribution of $\boldsymbol{\epsilon}_t$ is non-Gaussian. For example, when $\tau \sim \mathcal{G}^{-1}(\nu/2, \nu/2)$, the distribution of $\boldsymbol{\epsilon}_t$ is $t_\nu(\mathbf{0}, \boldsymbol{\Sigma})$, which is the multivariate-t distribution, see Appendix A.1 for its probability density function.

[4] Note that stability is a sufficient, but not a necessary condition for the decay of autoregressive coefficient matrices. A VAR can exhibit coefficient matrices that decay to zero without being stable. For example, condider a VAR(P) with $\boldsymbol{A}_1 = \boldsymbol{I}_N$ and $\boldsymbol{A}_p = \mathbf{0}$, for $1 < p \leq P$.

$\boldsymbol{\mu} = \sum_{q=0}^{\infty} \boldsymbol{\Phi}_q \boldsymbol{\nu}$ and $\mathbb{E}[(\boldsymbol{y}_t - \boldsymbol{\mu})(\boldsymbol{y}_{t-h} - \boldsymbol{\mu})'] = \sum_{q=0}^{\infty} \boldsymbol{\Phi}_{h+q} \boldsymbol{\Omega} \boldsymbol{\Phi}'_q$, for $h \in \mathbb{Z}$. Since these two terms do not depend on t, which is the definition of *stationarity*[5], stability implies this fundamental property in the time series literature. Another practical benefit of the MA representation is that it helps define the impulse response functions, which will be introduced later.

To illustrate the generality of a VAR under the stability condition, consider the *Wold's Decomposition Theorem* (Wold, 1938), which states that any stationary process can be written as a sum of a deterministic process (which can be forecasted without uncertainty) and an infinite-order MA process in the form of (2.2). Under the assumption of invertibility[6] (which is a standard assumption in empirical time series analysis (Komunjer and Ng, 2011)), the infinite-order MA process can be represented as an infinite-order VAR, which can be approximated as a finite-order VAR due to stability. Thus, the stochastic part of any stationary process can be approximated as a finite-order VAR.

Modeling a VAR requires selecting a finite lag order, P. Researchers typically determine the lag order in advance based on established knowledge in their specific domain. For example, Huber and Feldkircher (2019) studied the quarterly economic data with 5 lags, meaning that the current economy is influenced by the economic condition in the preceding 5 quarters. For the research that adopts an agnostic stance toward lag selection, several methods are available for determining the optimal lag order. Kilian and Ivanov (2001) provided multiple information criteria, including Akaike's Information Criterion (Akaike, 1998), Hannan-Quinn Criterion (Hannan, 1979) and Bayesian Information Criterion (Schwarz, 1978). Alternative approaches by Ahelegbey et al. (2016b) and Binks et al. (2024) impose priors on coefficients or lag order itself, enabling the selection of P based on the posterior.

2.1.2 Bayesian Inference of Standard VAR

Before proceeding to the Bayesian inference of standard VARs, it is beneficial to explore alternative formulations, as certain ones yield more mathematically elegant (conditional) posteriors relative to others in specific analytical contexts. The intercept terms are omitted hereafter as

[5] Another more strict definition of stationarity is that the joint density of any n consecutive data points (for $n \in \mathbb{Z}^+$) in the time series is the same. However, this strict definition is often impractical, so the weaker one defined above is typically adopted in practice.

[6] Denote L as the backshift operator such that $L^p \boldsymbol{y}_t = \boldsymbol{y}_{t-p}$, for $p \in \mathbb{Z}^+$, the infinite-order MA process in (2.2) is invertible when $\Phi(L) = \boldsymbol{I}_N + \boldsymbol{\Phi}_1 L + \boldsymbol{\Phi}_2 L^2 + \dots$ has an inverse $\Pi(L) = \boldsymbol{I}_N + \boldsymbol{\Pi}_1 L + \boldsymbol{\Pi}_2 L^2 + \cdots$ such that $\Pi(L)\Phi(L) = \boldsymbol{I}_N$.

$\boldsymbol{y}_{1:T}$ can be normalized to have zero means. The first formulation is simplified from (2.1) as

$$\boldsymbol{y}_t = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{\epsilon}_t, \tag{2.3}$$

where $\boldsymbol{x}_t = \left(\boldsymbol{y}_{t-1}', \ldots, \boldsymbol{y}_{t-P}'\right)'$. By concatenating the data across time, the second formulation is

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{A}' + \boldsymbol{E}, \tag{2.4}$$

where the t-th rows of $\boldsymbol{Y}$, $\boldsymbol{E} \in \mathbb{R}^{T\times N}$, $\boldsymbol{X} \in \mathbb{R}^{T\times NP}$ are $\boldsymbol{y}_t'$, $\boldsymbol{\epsilon}_t'$ and $\boldsymbol{x}_t'$, respectively. The third formulation vectorizes $\boldsymbol{A}'$ as $\boldsymbol{\alpha}$ and has the following expression,

$$\boldsymbol{y} = \boldsymbol{Z}\boldsymbol{\alpha} + \boldsymbol{\epsilon}, \tag{2.5}$$

where $\boldsymbol{Z} = (\boldsymbol{Z}_1', \ldots, \boldsymbol{Z}_T')' \in \mathbb{R}^{TN\times N^2P}$, $\boldsymbol{Z}_t = \boldsymbol{I}_N \otimes \boldsymbol{x}_t'$, $\boldsymbol{y} = (\boldsymbol{y}_1', \ldots, \boldsymbol{y}_T')'$ and $\boldsymbol{\epsilon} = (\boldsymbol{\epsilon}_1', \ldots, \boldsymbol{\epsilon}_T')' \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_T \otimes \boldsymbol{\Omega})$.

Conducting Bayesian inference for a standard VAR requires specifying prior distributions of $\boldsymbol{A}$ and $\boldsymbol{\Omega}$. These prior specifications can be categorized into two types: conjugate and non-conjugate. A typical conjugate prior in a standard VAR is the normal-inverse-Wishart prior[7],

$$\boldsymbol{A}', \boldsymbol{\Omega} \sim \mathcal{NIW}\left(\text{vec}\left(\underline{\boldsymbol{A}}'\right), \underline{\boldsymbol{V}}, \underline{\nu}, \underline{\boldsymbol{S}}\right), \tag{2.6}$$

where $\underline{\boldsymbol{A}} \in \mathbb{R}^{N\times NP}$, $\underline{\nu} \in \mathbb{R}$, and $\underline{\boldsymbol{V}} \in \mathbb{R}^{NP\times NP}$, $\underline{\boldsymbol{S}} \in \mathbb{R}^{N\times N}$ are positive definite matrices. The hierarchical representation of this prior is $\text{vec}(\boldsymbol{A}') = \boldsymbol{\alpha} \mid \boldsymbol{\Omega} \sim \mathcal{N}\left(\underline{\boldsymbol{\alpha}}, \boldsymbol{\Omega} \otimes \underline{\boldsymbol{V}}\right)$, $\boldsymbol{\Omega} \sim \mathcal{IW}\left(\underline{\nu}, \underline{\boldsymbol{S}}\right)$, where $\underline{\boldsymbol{\alpha}} = \text{vec}\left(\underline{\boldsymbol{A}}'\right)$, $\mathcal{IW}$ denotes the inverse-Wishart distribution with the description available in Appendix A.1. The specification of hyperparameters depends on the data characteristics. For example, $\underline{\boldsymbol{\alpha}}$ is a zero vector in general, but when time series in $\boldsymbol{y}_t$ exhibit random walk behavior, the elements corresponding to the first own-lag coefficients ($\boldsymbol{A}_{1,(i,i)}$ for $i \in [N]$) can be set to 1. $\underline{\boldsymbol{V}}$ is typically a diagonal matrix, with all non-zero elements set to a large value (such as 4) when prior knowledge is absent, or incorporating specific information such as the one in the Minnesota prior (Litterman, 1979), which will be introduced later. $\underline{\nu}$ and $\boldsymbol{S}$ are usually treated as known with default setting $\underline{\nu} = N + 3$ and $\underline{\boldsymbol{S}} = \boldsymbol{I}_N$[8]. Given the likelihood formulated in (2.4), the joint posterior of $\boldsymbol{A}'$ and $\boldsymbol{\Omega}$

[7] If a matrix $\boldsymbol{M} \in \mathbb{R}^{M\times N}$ and a symmetric positive definite matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{N\times N}$ follow the normal-inverse-Wishart prior, $\mathcal{NIW}\left(\text{vec}\left(\underline{\boldsymbol{M}}\right), \underline{\boldsymbol{V}}, \underline{\nu}, \underline{\boldsymbol{S}}\right)$, the corresponding probability density function is $|\boldsymbol{\Sigma}|^{-\frac{\underline{\nu}+N+M+1}{2}} \exp\left(-\frac{1}{2}\text{tr}\left(\boldsymbol{\Sigma}^{-1}(\boldsymbol{M}-\underline{\boldsymbol{M}})'\underline{\boldsymbol{V}}^{-1}(\boldsymbol{M}-\underline{\boldsymbol{M}}) + \boldsymbol{\Sigma}^{-1}\underline{\boldsymbol{S}}\right)\right)$.

[8] The rationale behind this default setting is that the prior means and covariances of the elements in $\boldsymbol{\Omega}$ exist when $\nu > N + 2$; hence, setting $\nu = N + 3$ ensures the existence of both the first and second moments while keeping the prior weakly informative. In addition, choosing $\boldsymbol{S}$ as an identity matrix implies that all elements share the same prior variance and are assumed to be *a priori* uncorrelated, providing a non-informative prior structure.

is $\mathcal{NIW}\left(\text{vec}\left(\overline{\boldsymbol{A}}\right), \overline{\boldsymbol{V}}, \overline{\nu}, \overline{\boldsymbol{S}}\right)$, where $\overline{\boldsymbol{A}} = \overline{\boldsymbol{V}}\left(\underline{\boldsymbol{V}}^{-1}\underline{\boldsymbol{A}}' + \boldsymbol{X}'\boldsymbol{Y}\right)$, $\overline{\boldsymbol{V}} = \left(\underline{\boldsymbol{V}}^{-1} + \boldsymbol{X}'\boldsymbol{X}\right)^{-1}$, $\overline{\nu} = \underline{\nu} + T$, $\overline{\boldsymbol{S}} = \underline{\boldsymbol{S}} + \underline{\boldsymbol{A}}\,\underline{\boldsymbol{V}}\,\underline{\boldsymbol{A}}' + \boldsymbol{Y}'\boldsymbol{Y} - \overline{\boldsymbol{A}}'\overline{\boldsymbol{V}}^{-1}\overline{\boldsymbol{A}}$.

While the normal-inverse-Wishart prior enables an analytical posterior of unknown parameters, it restricts the variance-covariance matrix of $\boldsymbol{\alpha}$ to be a Kronecker product, which indicates that the prior covariances of coefficients for the i-th variable, $\boldsymbol{\Omega}_{(i,i)}\underline{\boldsymbol{V}}$, is proportional to one another, for $i \in [N]$. This limitation motivates the development of non-conjugate priors as more flexible alternatives. The non-conjugate priors typically take the form of $\boldsymbol{\alpha} \sim \mathcal{N}\left(\underline{\boldsymbol{\alpha}}, \underline{\boldsymbol{V}}\right)$, and $\boldsymbol{\Omega}$ still follows the above inverse-Wishart prior. Although the joint posterior no longer has an analytical form, full conditionals are available based on the formulation in (2.5),

$$\boldsymbol{\alpha} \mid \boldsymbol{\Omega}, \boldsymbol{y} \sim \mathcal{N}\left(\overline{\boldsymbol{\alpha}}, \overline{\boldsymbol{V}}\right), \quad \boldsymbol{\Omega} \mid \boldsymbol{\alpha}, \boldsymbol{y} \sim \mathcal{IW}\left(\overline{\nu}, \overline{\boldsymbol{S}}\right), \tag{2.7}$$

where $\overline{\boldsymbol{\alpha}} = \overline{\boldsymbol{V}}\left(\underline{\boldsymbol{V}}^{-1}\,\underline{\boldsymbol{\alpha}} + \boldsymbol{Z}'\left(\boldsymbol{I}_T \otimes \boldsymbol{\Omega}\right)^{-1}\boldsymbol{y}\right)$, $\overline{\boldsymbol{V}} = \left(\underline{\boldsymbol{V}}^{-1} + \boldsymbol{Z}'\left(\boldsymbol{I}_T \otimes \boldsymbol{\Omega}\right)^{-1}\boldsymbol{Z}\right)^{-1}$, $\overline{\nu} = T + \underline{\nu}$ and $\overline{\boldsymbol{S}} = \underline{\boldsymbol{S}} + \sum_{t=1}^{T}\left(\boldsymbol{y}_t - \boldsymbol{Z}_t\boldsymbol{\alpha}\right)\left(\boldsymbol{y}_t - \boldsymbol{Z}_t\boldsymbol{\alpha}\right)'$. Given these full conditionals, it is straightforward to sample $\boldsymbol{\alpha}$ and $\boldsymbol{\Omega}$ iteratively using the Gibbs sampler.

2.1.3 Applications of Standard VAR

After estimating $\boldsymbol{A}$ and $\boldsymbol{\Omega}$ given the data, the VAR enables multiple analytical applications. First, the model can make point and density forecasts. Assume that $\hat{\boldsymbol{A}}$ is the (sampled) posterior mean of $\boldsymbol{A}$ from the derivation or the MCMC, then the joint h-step-ahead point forecast ($h \in \mathbb{N}$ is the horizon) is $\hat{\boldsymbol{y}}_{T+h} = \mathbb{E}\left[\boldsymbol{y}_{T+h} \mid \boldsymbol{y}_{1:T}\right] = \hat{\boldsymbol{A}}\hat{\boldsymbol{x}}_{T+h}$, where $\hat{\boldsymbol{x}}_{T+h} = \left(\hat{\boldsymbol{y}}'_{T+h-1}, \ldots, \hat{\boldsymbol{y}}'_{T+h-P}\right)'$. The marginal point forecast is the corresponding elements in $\hat{\boldsymbol{y}}_{T+h}$. Density forecasts have become increasingly popular in VARs due to the extra uncertainty insights provided by the model (Clark, 2011). When the conjugate normal-inverse-Wishart prior is imposed, the joint 1-step-ahead density forecast of $\boldsymbol{y}_{T+1}$, $p\left(\boldsymbol{y}_{T+1} \mid \boldsymbol{y}_{1:T}\right)$, has a closed form as a multivariate-t distribution with $\bar{\nu} - N + 1$ degrees of freedom, mean $\hat{\boldsymbol{y}}_{T+1}$, and a scale matrix $\left(\bar{\nu} - N + 1\right)^{-1}\left(1 + \boldsymbol{x}'_{T+1}\bar{\boldsymbol{V}}\boldsymbol{x}_{T+1}\right)\bar{\boldsymbol{S}}$, see the definition of multivariate-t distribution in Appendix A.1. The marginal density forecast of $\boldsymbol{y}_{T+1,i}$ is a location-scale t distribution (see Appendix A.1) with $\bar{\nu} - N + 1$ degrees of freedom, location $\hat{\boldsymbol{y}}_{T+1,i}$, and scale as the square root of the (i,i) entry of the above scale matrix, for $i \in [N]$. For the cases when the prior is non-conjugate or $h > 1$, the density needs to be approximated as $p\left(\boldsymbol{y}_{T+h} \mid \boldsymbol{y}_{1:T}\right) = \frac{1}{L}\sum_{l=1}^{L} p\left(\boldsymbol{y}_{T+h} \mid \boldsymbol{y}_{1:T}, \boldsymbol{A}^{(l)}, \boldsymbol{\Omega}^{(l)}\right)$, where $\boldsymbol{A}^{(l)}$ and $\boldsymbol{\Omega}^{(l)}$ are samples generated from the posterior or the MCMC.

The VAR framework is also instrumental in assessing Granger causality (Granger, 1969), i.e., the directional connectivity among variables in $\boldsymbol{y}_t$. For example, central banks may use Granger causality analysis to inform policy decisions: if the growth rates of money supply are found to Granger-cause income growth, this suggests that maintaining a stable rate of money growth could help promote economic stability. Let $\mathcal{F}_t$ and $\mathcal{F}_t^{(-j)}$ denote the information set containing all variables up to time t and that excluding the j-th variable, respectively. The j-th variable in $\boldsymbol{y}_t$ is said to *Granger-cause* the i-th variable ($i, j \in [N]$ and $i \neq j$) if $\mathbb{E}\left[\left(\boldsymbol{y}_{t+h,i} - \mathbb{E}\left[\boldsymbol{y}_{t+h,i} \mid \mathcal{F}_t\right]\right)^2 \mid \mathcal{F}_t\right] < \mathbb{E}\left[\left(\boldsymbol{y}_{t+h,i} - \mathbb{E}\left[\boldsymbol{y}_{t+h,i} \mid \mathcal{F}_t^{(-j)}\right]\right)^2 \mid \mathcal{F}_t\right]$, for at least one horizon h. Given a time series model (not necessarily a VAR), one can apply hypothesis tests such as F-test and the likelihood ratio test to determine Granger causality. In the VAR framework, the definition of Granger causality simplifies to $\boldsymbol{A}_{p,(i,j)} \neq 0$ for any $p \in [P]$. Although exact zeros are rarely found in the coefficient matrix, a practical approach is to inspect the (sampled) posterior distribution of the coefficient. For example, if zero lies outside a confidence or credible interval of a coefficient, this coefficient is different from zero with statistical significance. Thus, the VAR facilitates an efficient implementation of Granger causality analysis.

The third main application of a VAR is impulse response analysis, which quantifies the responses of a variable in $\boldsymbol{y}_t$ to an unexpected shock of interest. For example, if $\boldsymbol{y}_t \in \mathbb{R}^3$ contains the CPI, the unemployment rate, and the Federal fund rates in the US economy, and the shock of interest is a monetary policy shock, then one can model the response of the CPI or the unemployment rate h periods after receiving an unexpected increase or decrease in the Federal fund rate. To properly interpret impulse responses, shocks must be uncorrelated across variables; otherwise, a shock arising from one variable would be entangled with simultaneous shocks from others, making the origin of the shock ambiguous. When $\boldsymbol{\Omega}$ is a diagonal matrix, the shocks are directly given by the error terms in $\boldsymbol{\epsilon}_t$ since they are uncorrelated. The MA representation in (2.2) facilitates the mathematical definition of the impulse response function (IRF) in this case. The IRF of the i-th variable with respect to a one unit shock of the j-th variable is defined as $\text{IRF}_{i,j}(h) = \frac{\partial \boldsymbol{y}_{t+h,i}}{\partial \boldsymbol{\epsilon}_{t,j}} = \boldsymbol{\Phi}_{h,(i,j)}$, for $h \in \mathbb{Z}$ and $i, j \in [N]$. When $\boldsymbol{\Omega}$ is not diagonal, the shocks are not $\boldsymbol{\epsilon}_t$ from the reduced-form VAR defined in (2.1). Instead, one needs the structural VAR (see Kilian and Lütkepohl (2017) for a review) with the following form,

$$\boldsymbol{P}\boldsymbol{y}_t = \boldsymbol{B}_1\boldsymbol{y}_{t-1} + \cdots + \boldsymbol{B}_P\boldsymbol{y}_{t-P} + \boldsymbol{\eta}_t, \tag{2.8}$$

where $\boldsymbol{\Omega} = \boldsymbol{P}^{-1}(\boldsymbol{P}^{-1})'$, $\boldsymbol{B}_p = \boldsymbol{P}\boldsymbol{A}_p$, for $p \in [P]$, and $\boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_N)$. The shocks are then defined as the uncorrelated elements in $\boldsymbol{\eta}_t$. By writing the MA representation (2.2) in terms of $\boldsymbol{\eta}_t$,

$$\boldsymbol{y}_t = \sum_{q=0}^{\infty} \boldsymbol{\Phi}_q \boldsymbol{\nu} + \sum_{q=0}^{\infty} \boldsymbol{\Phi}_q \boldsymbol{P}^{-1} \boldsymbol{\eta}_{t-q}, \tag{2.9}$$

the IRF is then $\text{IRF}_{i,j}(h) = \frac{\partial \boldsymbol{y}_{t+h,i}}{\partial \boldsymbol{\eta}_{t,j}}$, which is the (i,j) entry of the matrix $\boldsymbol{\Psi}_h = \boldsymbol{\Phi}_h \boldsymbol{P}^{-1}$. It is noteworthy that the structural VAR in (2.8) is identifiable up to an orthogonal rotation; that is, replacing $\boldsymbol{P}$ by $\tilde{\boldsymbol{P}} = \boldsymbol{P}\boldsymbol{Q}$, where $\boldsymbol{Q} \in \mathbb{R}^{N \times N}$ is an orthogonal matrix, does not change the likelihood of this structural VAR. Thus, identifying the shocks necessitates imposing restrictions on $\boldsymbol{P}^{-1}$, which provide additional interpretation of the shocks. The most common restriction is the recursive one proposed by Sims (1980), which assumes $\boldsymbol{P}^{-1}$ is a lower triangular matrix as the Cholesky component of $\boldsymbol{\Omega}$. This recursive restriction implies that a variable only responds contemporaneously to the shocks originating from the other variable ordered before it in $\boldsymbol{y}_t$. Other commonly applied restrictions include zero and sign restrictions (Uhlig, 2005), which assume some elements in $\boldsymbol{P}^{-1}$ to be zeros and to have specific signs, respectively; see Stock and Watson (2016) for a comprehensive review.

Forecast error variance decomposition (FEVD) is also a useful application of the VAR. This determines the proportion of the forecast error variance of $\boldsymbol{y}_{t+h}$ explained by the uncertainty of $\boldsymbol{y}_t$. For example, the FEVD helps analyze the volatility spillover, i.e., how the fluctuation in one sector, market, or country influences the uncertainty of others (Diebold and Yilmaz, 2012). To define the FVED, one can base on the MA representation in (2.2) and write $\boldsymbol{\epsilon}_{t+h}(t) = \boldsymbol{y}_{t+h} - \mathbb{E}\left[\boldsymbol{y}_{t+h} \mid \boldsymbol{y}_{1:t}\right] = \sum_{q=0}^{h-1} \boldsymbol{\Phi}_q \boldsymbol{\epsilon}_{t+h-q}$. Then, take the notation from the structural VAR in (2.8) with the recursive restriction, $\boldsymbol{\epsilon}_{t+h}(t)$ can be rewritten as $\sum_{q=0}^{h-1} \boldsymbol{\Psi}_q \boldsymbol{\eta}_{t+h-q}$. The FVED of the j-th variable to the i-th variable (i.e., the proportion of variance in $\boldsymbol{y}_{t+h,i}$ explained by the fluctuation in $\boldsymbol{y}_{t,j}$ for $i, j \in [N]$), is $\left(\sum_{q=0}^{h-1} (\boldsymbol{\Psi}_{q,(i,j)})^2\right) / \left(\sum_{q=0}^{h-1} \sum_{j'=1}^{N} (\boldsymbol{\Psi}_{q,(i,j')})^2\right)$.

The VAR model discussed so far represents the standard specification. Although it already serves as a powerful tool for time series analysis, it faces limitations when $\boldsymbol{y}_t$ is high-dimensional or exhibits time-varying dynamics. To accommodate these two important aspects of time series modeling, the remainder of this subsection will discuss their motivations, the challenges encountered, and the approaches developed to address them.

2.1.4 High-dimensional VAR

The use of high-dimensional data, typically involving 40 to hundreds of variables, in VAR models has been found to improve forecasting performance (Bańbura et al., 2010; Carriero et al., 2015; Giannone et al., 2015; Koop, 2013) and enhance structural analysis to better align with economic theory (Carriero et al., 2019; Huber and Feldkircher, 2019), compared to low-dimensional VARs with only 3 – 10 variables. The growing demand for analyzing disaggregated economic time series, rather than relying solely on aggregated key indicators, has also motivated the development of high-dimensional VARs. For instance, Korobilis and Pettenuzzo (2019) considered time series corresponding to different sectors of the industrial production index. These motivations, alongside the availability of large data sets in the big data era, encouraged the development of high-dimensional VARs. However, to succeed in modeling with VARs with high-dimensional data, one must address over-parameterization, which is especially an issue in the VAR for two reasons. First, the number of parameters in the coefficient matrix $\boldsymbol{A}$ and the variance-covariance matrix grows quadratically with the number of variables. Second, the application of the VAR usually involves low-frequency data, so the number of observations is relatively small. Over-parameterization directly leads to overfitting, where the model fits the in-sample data well but performs poorly out of sample. Moreover, this issue increases uncertainty in inference; for instance, impulse responses derived from an over-parameterized model may exhibit wide credible intervals. Therefore, addressing over-parameterization is crucial for VARs.

Research on high-dimensional VARs can be categorized into two strands. The first focuses on inducing sparsity or shrinkage in the parameter space, while the second reduces the number of parameters through dimension reduction techniques. This literature review will first discuss these two strands and then introduce approaches for alleviating the computational burden associated with the inference.

The first strand has gained popularity in both the frequentist and Bayesian frameworks. Given the scope of this thesis, the literature review will primarily focus on the latter, with a brief introduction to the former. Frequentist approaches in the VAR literature (Davis et al., 2016; Hsu et al., 2008; Lozano et al., 2009) typically incorporate the lasso penalty (Tibshirani, 1996) or its variants with respect to $\boldsymbol{A}$ into the loss function, which is usually either the mean squared error or the negative log-likelihood, while the ridge penalty (Hoerl and Kennard, 1970) has received less attention due to its limited interpretability in variable selection and relatively weaker forecasting performance[9]. Song and Bickel (2011), Shojaie and Michailidis (2010), and Nicholson et al. (2020) incorporated the autoregressive structure of the VAR by allowing the importance of lagged values to vary by lag order and distinguished between own-lag and cross-lag effects. While frequentist methods mainly induce sparsity in the coefficient matrix, shrinking some coefficients exactly to zero, Bayesian methods applied in VARs impose shrinkage priors to the coefficient matrix, typically based on the belief that all predictors could be important, though some parameters may have small magnitudes[10]. Most shrinkage priors applied in VARs take the non-conjugate prior form $\boldsymbol{\alpha} \sim \mathcal{N}\left(\underline{\boldsymbol{\alpha}}, \underline{\boldsymbol{V}}\right)$ with some hyperparameters, so the estimation of coefficients follows the full conditional posteriors stated in (2.7). Three types of priors will be introduced: the Minnesota, the spike-and-slab, and the global-local shrinkage priors.

The most prominent prior in the VAR framework is the Minnesota prior (Doan et al., 1984; Litterman, 1979, 1986), of which the name originates from the University of Minnesota and the Federal Reserve Bank of Minneapolis, where the authors were affiliated. This prior imposes three pieces of prior knowledge: one through $\underline{\boldsymbol{\alpha}}$ and two through $\underline{\boldsymbol{V}}$. First, $\underline{\boldsymbol{\alpha}}$ encodes prior beliefs about whether the data behave like random walks or not. With most elements in $\underline{\boldsymbol{\alpha}}$ being zero, the entries corresponding to the first own lags ($\boldsymbol{A}_{1,(i,i)}$ for $i \in [N]$) are set close to one in the former case and to zero in the latter. Second, the prior further emphasizes the importance of own lags by specifying $\underline{\boldsymbol{V}}$ as a diagonal matrix, with its diagonal elements divided into two groups,

$$\underline{\boldsymbol{V}}_{p,i,j} = \begin{cases} \lambda_1/p^2, & \text{if } i = j \\ \lambda_2 \hat{\sigma}_i^2/(p^2 \hat{\sigma}_j^2), & \text{if } i \neq j \end{cases}, \tag{2.10}$$

where $\underline{\boldsymbol{V}}_{p,i,j}$ is the prior variance of $\boldsymbol{A}_{p,(i,j)}$, for $i, j \in [N]$ and $p \in [P]$, λ_1 are λ_2 are positive real values to control the shrinkage levels with $\lambda_1 > \lambda_2$, $\hat{\sigma}_i^2$ is the ordinary least square (OLS) estimate of the variance of $\boldsymbol{y}_{t,i}$ via the VAR or the corresponding AR(P). Lastly, the Minnesota prior assumes that shorter lags have a greater impact on the data than longer lags, as shown in (2.10), where the shrinkage level on $\boldsymbol{A}_p$ increases with p through the p^2 term in the denominator.

[9] An exception is the study by Ballarin (2025), which investigated the VAR with the ridge penalty.

[10] Sparsity is also achievable in Bayesian VARs; for example, through the spike-and-slab prior as applied in Korobilis (2013b).

The Minnesota prior has been extended to multiple variants to form Minnesota-type priors. For example, Kadiyala and Karlsson (1997) changed p^2 in (2.10) to p. Bańbura et al. (2010) added dummy observations to the data to impose a conjugate prior variant of the Minnesota prior, which imposed the first and the third assumptions mentioned, but relaxed the second. Other extensions can be found in Section 3.1.1 of Karlsson (2013).

The hyperparameters λ_1 and λ_2 are crucial to the Minnesota because different values affect the performance of the VAR (Chan et al., 2019). Doan et al. (1984), Giannone et al. (2014), and Carriero et al. (2015), to name a few, selected these hyperparameters by comparing in-sample or out-of-sample forecasting performance. More recently, Giannone et al. (2015) imposed hyperpriors and sample λ_1 and λ_2 using Metropolis-Hastings steps. This procedure has become a standard in VAR applications and the corresponding Minnesota prior is called the hierarchical Minnesota prior. While the assumptions of the Minnesota priors are intuitively appealing for time series data, the next two types of priors adopt more flexible structures with fewer restrictions.

The spike-and-slab priors (Mitchell and Beauchamp, 1988) (see a review in O'Hara and Sillanpää (2009)) assume a coefficient to be negligible with a probability, and the corresponding mathematical expression with Gaussian specification is defined as follows:

$$\begin{aligned} \boldsymbol{\alpha}_i \mid \gamma_i &\sim (1-\gamma_i)\mathcal{N}\left(0, \kappa_{0,i}^2\right) + \gamma_i \mathcal{N}\left(0, \kappa_{1,i}^2\right), \\ \gamma_i &\sim \text{Bernoulli}(q_i), \end{aligned} \tag{2.11}$$

where $\boldsymbol{\alpha}_i$ is the i-th element in $\boldsymbol{\alpha}$, and q_i can either be set as 0.5 or assigned a Beta distribution, for $i \in [N^2P]$. By assuming $\kappa_{0,i}^2 \ll \kappa_{1,i}^2$, $\boldsymbol{\alpha}_i$ is inferred as negligible if $\gamma_i = 0$. When $\kappa_{0,i}$ and $\kappa_{1,i}$ are non-zero, this spike-and-slab prior is continuous and represents the stochastic search variable selection (SSVS) (George and McCulloch, 1993), which was firstly applied in VAR in George et al. (2008). The default specification of $\kappa_{k,i}^2 = c_k \widehat{\text{var}(\boldsymbol{\alpha}_i)}$, where $\widehat{\text{var}(\boldsymbol{\alpha}_i)}$ is the standard error of $\boldsymbol{\alpha}_i$ based on the OLS estimation, $c_0 = 0.01$ and $c_1 = 100$. Koop (2013) replaced $\widehat{\text{var}(\boldsymbol{\alpha}_i)}$ to the prior variance defined in the Minnesota prior and set $c_1 = 1$, so the prior is a combination of the SSVS and the Minnesota. Korobilis (2013b) applied the Dirac spike-and-slab prior (Mitchell and Beauchamp, 1988), corresponding to the case when $\kappa_{0,i} = 0$, and $\kappa_{1,i}$ can be specified in various ways, such as the Minnesota and the global-local shrinkage prior (which will be discussed later). The spike-and-slab priors attain interpretability from γ_i, which indicates whether a predictor is important to a response variable. The SSVS relaxes the computational burden of the Dirac spike-and-slab prior, which has 2^{N^2P} possible models to explore, and is considered as a benchmark in many VAR applications (Huber and Feldkircher, 2019; Korobilis, 2008).

The other type of flexible priors is the global-local shrinkage prior (Polson and Scott, 2010) with the following expression

$$\boldsymbol{\alpha}_i \sim \mathcal{F}_{\psi_i,\tau},\ \psi_i \sim \mathcal{F}_1,\ \tau \sim \mathcal{F}_2,$$

where $\mathcal{F}_{\psi_i,\tau}$ is the prior of $\boldsymbol{\alpha}_i$, with hyperparameters ψ_i and τ as well as their respective hyperpriors $\mathcal{F}_1$ and $\mathcal{F}_2$, for $i \in [N^2P]$. τ is the global parameter which shrinks the coefficients globally towards 0. ψ_i is the local parameter to retain a heavy tail in $\mathcal{F}_{\psi_i,\tau}$, which avoids overshrinkage of the coefficients. Table 2.1 provides a summary of global-local shrinkage priors applied in VARs. An overview of these priors and their application to VARs is presented below.

Prior	$\mathcal{F}_{\psi_i,\tau}$	$\mathcal{F}_1$	$\mathcal{F}_2$
Bayesian lasso	$\mathcal{N}\left(0, \psi_i\tau\right)$	$\mathcal{E}\left(1/2\right)$	$\mathcal{IG}\left(a_{\text{BL}}, b_{\text{BL}}\right)$
Normal-gamma	$\mathcal{N}\left(0, \psi_i\tau\right)$	$\mathcal{G}\left(\lambda, 1/2\right)$	$\mathcal{IG}\left(a_{\text{NG}}, b_{\text{NG}}/2\lambda\right)$
Dirichlet-Laplace	$\mathcal{DE}\left(\psi_i\tau\right)$	$\mathcal{D}(\boldsymbol{a}_{\text{DL}})$	$\mathcal{G}\left(N^2Pa_{\text{DL}}, 1/2\right)$
Horseshoe	$\mathcal{N}\left(0, \psi_i^2\tau^2\right)$	$\mathcal{C}^+(0,1)$	$\mathcal{C}^+(0,1)$
R2D2	$\mathcal{DE}\left(\sqrt{\psi_i\tau}\right)$	$\mathcal{G}(a_{R2}, a_{R2})$	$\mathcal{IG}\left(b_{R2}, a_{R2}/2\right)$

Table 2.1: Global-local shrinkage priors applied in VARs. The R2D2 means R^2-induced Dirichlet decomposition. $\mathcal{DE}$, $\mathcal{E}$, $\mathcal{G}$, $\mathcal{D}$, $\mathcal{C}^+$, $\mathcal{IG}$ denote double exponential, exponential, gamma, Dirichlet, half-Cauchy, and inverse-gamma distributions, respectively. See Appendix A.1 for the descriptions of these distributions. $\boldsymbol{a}_{DL} \in \mathcal{R}^{N^2P}$ has all elements being $a_{DL} \in \mathbb{R}_{>0}$.

The Bayesian lasso prior (Park and Casella, 2008) provides a Bayesian analogue to lasso regularization, as its posterior mode coincides with the maximum likelihood estimate obtained under lasso penalization, where the tuning parameter corresponds to $1/\sqrt{\tau}$. Gefang (2014) extended the Bayesian lasso to a Bayesian adaptive elastic-net prior, in which the prior $\mathcal{F}_{\psi_i,\tau}$ for each coefficient modifies the corresponding density for the Bayesian lasso by multiplying by a zero-mean normal density. Though the Bayesian lasso prior stems from a popular penalty in the VAR literature, it is prone to undershrink parameters with negligible magnitudes and overshrink those with large magnitudes because it has a light tail which leads to a high misclassification rate (Datta and Ghosh, 2013). Motivated by the search for a more flexible prior than the Bayesian lasso, Griffin and Brown (2010) proposed the normal-gamma distribution, with the hyperparameter λ in Table 2.1 governing its properties. When $\lambda = 1$, the normal-gamma prior reduces to the Bayesian lasso; as λ decreases, the prior becomes more peaked at zero and exhibits heavier tails. Huber and Feldkircher (2019) applied the normal-gamma to VARs and proposed three extensions by introducing semi-global shrinkage parameters that vary with the response variables, predictor variables, and lag orders. Instead of imposing priors to local parameters marginally, the Dirichlet-Laplace prior (Bhattacharya et al., 2015) elicits shrinkage by specifying a Dirichlet distribution to local parameters jointly. Small values in $\boldsymbol{a}_{\text{DL}} = (a_{\text{DL}}, \ldots, a_{\text{DL}})'$ cause most mass in $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_{N^2P})'$ to concentrate at a small subset, leading to the majority of coefficients being shrunk towards zero. Kastner and Huber (2020) used the Dirichlet-Laplace in VARs and showed that this prior outperforms other priors, such as the Minnesota and the normal-gamma, in forecasting. The horseshoe prior places a Beta$(0.5, 0.5)$ distribution to the shrinkage level, $(1 + \psi_i \tau)^{-1}$. A higher shrinkage level indicates that the corresponding coefficient is more likely to be shrunk towards 0. The Beta$(0.5, 0.5)$ distribution has high density near both 0 and 1, giving its probability density function a horseshoe-like shape, hence the name of the prior. This specification gives $\boldsymbol{\alpha}_i$ a prior that encourages strong shrinkage around zero while preserving heavy tails. Follett and Yu (2019) compared the horseshoe prior with the spike-and-slab prior used in Korobilis (2013b), and found that both priors produce similar results, while the horseshoe prior offers computational efficiency as a continuous prior without sacrificing interpretability. While the priors aforementioned are directly specified for coefficients, R^2-induced Dirichlet decomposition (R2D2) prior (Zhang et al., 2022b) simplifies the specification by imposing a Beta(a_{R2}, b_{R2}) prior to a single number—the coefficient of determination (R^2), which controls the global shrinkage levels of the coefficients. The authors showed that this Beta prior on R^2 can be induced by the prior specified in Table 2.1. Gruber and Kastner (2025) compared the R2D2 prior with other global-local shrinkage priors mentioned (except the Bayesian lasso) and showed that this prior is competitive to others.

While high-dimensional Bayesian VARs have seen substantial methodological development, their theoretical support is relatively limited, with two exceptions which pertain to the three types of priors considered. Ghosh et al. (2019) established posterior consistency for the coefficients under (non-)hierarchical Gaussian priors, which are conceptually related to the Minnesota and global-local shrinkage priors discussed. However, a key distinction is that the theoretical results rely on a conjugate prior, as defined in equation (2.6), whereas most of the high-dimensional Bayesian VARs mentioned so far adopted non-conjugate priors due to their flexibility and the ease of extending them to non-linear VARs (which will be discussed in this subsection). Incorporating the conjugate prior framework requires modifying the shrinkage priors to accommodate the $\underline{\boldsymbol{V}}$ in (2.6). The second theoretical contribution comes from Ghosh et al. (2021), who studied the asymptotic properties of spike-and-slab priors in the VAR framework, demonstrating consistency in estimating the coefficient matrix and identifying the correct set of non-negligible coefficients. As with Ghosh et al. (2019), the spike-and-slab priors considered in this literature review, such as SSVS and Dirac ones, differ from their formulation, as their setup assumes that the q_i in (2.11) depends on the number of non-negligible coefficients.

From a practical standpoint, Cross et al. (2020) and Gruber and Kastner (2025) conducted empirical comparisons of the three types of priors using US macroeconomic data sets. Their conclusions are broadly aligned: the Minnesota priors remain a strong benchmark that is difficult to outperform, while the performance of spike-and-slab priors (SSVS in particular) is the weakest among the three. The finding highlights the importance of incorporating prior information in VAR applications within econometrics. Notably, global-local shrinkage priors still show potential to outperform the Minnesota, particularly when a semi-group structure which distinguishes between own-lag and cross-lag effects is incorporated into the prior specification (Gruber and Kastner, 2025).

The review of high-dimensional VARs so far mainly focuses on the application of shrinkage priors to the coefficient matrix, but they can also be applied to the variance-covariance matrix, as shown in George et al. (2008) and Huber and Feldkircher (2019). Rather than imposing an inverse-Wishart prior to this matrix, one can decompose it analogously to the structural VAR, $\boldsymbol{\Omega} = \boldsymbol{H}^{-1}\boldsymbol{S}\left(\boldsymbol{H}^{-1}\right)'$, where $\boldsymbol{H}^{-1}$ is usually defined as a lower-triangular matrix with ones on the diagonal (i.e., the unit lower-triangular matrix) and $\boldsymbol{S}$ is a diagonal matrix with positive diagonal terms. A shrinkage prior can then be imposed on the non-zero off-diagonal elements of $\boldsymbol{H}^{-1}$, allowing more parsimonious modeling of contemporaneous relationships. The diagonal terms in $\boldsymbol{S}$ can be assigned non-informative inverse-gamma priors, which result in closed-form conditional posteriors. Detailed derivations of the conditional posteriors for $\boldsymbol{H}^{-1}$ and $\boldsymbol{S}$ are available in George et al. (2008) and Carriero et al. (2019).

An alternative strategy to induce parsimony in VAR coefficients is through graphical vector autoregression (graphical VAR) models, whose theoretical foundation can be found in Eichler (2012). In brief, a graphical VAR represents the (directed or undirected) connectivity among variables via binary connectivity matrices $\boldsymbol{G}_1, \ldots, \boldsymbol{G}_P$[11]. The (i, j) entry of $\boldsymbol{G}_p$ equals to 1 if the i-th variable and j-th variable p periods earlier are connected, and zero otherwise, for $i, j \in [N]$. Following the idea of graphical VAR, Ahelegbey et al. (2016a) and Ahelegbey et al. (2016b) defined the coefficient matrix $\boldsymbol{A}_p$ as the element-wise product of an unrestricted matrix and $\boldsymbol{G}_p$, of which the prior is

$$\boldsymbol{G}_{p,(i,j)} \sim \text{Bernoulli}(\omega_{p,(i,j)}),\ \omega_{p,(i,j)} \sim \text{Beta}(a, b),$$

where $a, b > 0$. By setting a and b such that $\omega_{p,(i,j)}$ has a small prior mean (e.g., $a = 1$ and $b = 5$), one induces sparsity in $\boldsymbol{G}_{p,(i,j)}$, and hence in $\boldsymbol{A}_{p,(i,j)}$.

The second strand of high-dimensional VARs addresses over-parameterization through dimension reduction techniques. One approach within this strand reduces the number of parameters by imposing a low-rank structure on the coefficient matrix. For example, reduced-rank VAR (Velu et al., 1986) and multivariate index autoregression (Reinsel, 1983) assume that each coefficient matrix $\boldsymbol{A}_p$, for $p \in [P]$, to have a low rank, $R \ll N$. These methods decompose $\boldsymbol{A}_p$ to a product of two matrices: $\boldsymbol{C}\boldsymbol{D}_p$ and $\boldsymbol{C}_p\boldsymbol{D}$, respectively, where $\boldsymbol{C}, \boldsymbol{C}_p \in \mathbb{R}^{N\times R}$ and $\boldsymbol{D}, \boldsymbol{D}_p \in \mathbb{R}^{R\times N}$. With an upper bound $R < \frac{N}{2}$, the number of parameters reduces from N^2P to $2NPR$. The Bayesian inference of these two models has been explored by Geweke (1996) and Carriero et al. (2016b), where the decomposed matrices following Gaussian priors. Note that these decompositions are identifiable up to a rotation, i.e., $\boldsymbol{A}_p$ is the same if the decomposition is $\boldsymbol{C}\boldsymbol{Q}\boldsymbol{Q}^{-1}\boldsymbol{D}_p$ or $\boldsymbol{C}_p\boldsymbol{Q}\boldsymbol{Q}^{-1}\boldsymbol{D}$, for some invertible matrix $\boldsymbol{Q} \in \mathbb{R}^{R\times R}$. While this indeterminacy issue does not affect forecasting, additional restrictions are required if the model interpretation is desired. For instance, the upper R-by-R block of $\boldsymbol{C}$ can be an identity matrix. In both models, the rank R is selected by the grid search over a range of possible ranks with some evaluation metrics, such as the posterior odds ratio applied in Geweke (1996), mean squared forecast error in Carriero et al. (2011), and marginal likelihood in Carriero et al. (2016b). Other approaches for selecting the rank have been discussed in Camba-Mendez et al. (2003) from a frequentist

[11] The standard formulation of a graphical VAR only specifies a single binary matrix for all lags, such as that in Corander and Villani (2006). This thesis uses multiple binary matrices following Ahelegbey et al. (2016a) and Ahelegbey et al. (2016b).

perspective, such as the information criterion.

Several studies have extended the low-rank approaches. Koop et al. (2019) proposed the compressed VAR by applying the same formulation of the reduced-rank VAR. Instead of directly estimating the decomposed matrices, the authors randomly generated $\boldsymbol{D}_p$ from a distribution, and only inferred the corresponding $\boldsymbol{C}$ for each random draw. Bayesian model averaging (Raftery et al., 1997) was then used to assign weights across the set of random decompositions. This model was applied for forecasting financial time series in Taveeapiradeecharoen et al. (2019) and housing sentiment in Gupta et al. (2019). More recently, Wang et al. (2022a) introduced an alternative low-rank structure to the parameter space, namely tensor VAR (TVAR). Rather than decomposing the coefficient by the matrix multiplication as described above, the TVAR uses tensor decomposition, which will be discussed in Chapter 3. A VAR coefficient matrix can also have a low tree-rank representation, Duan et al. (2023) expressed all non-zero coefficients by the union of R spanning trees[12], each with N nodes. The benefit of using this tree structure is that connectivity of variables can be modeled by only $R(N-1)$ parameters, instead of $N(N-1)$ in a standard VAR setting,

The dynamic factor model (DFM) (Geweke, 1977) adopts an alternative dimension reduction technique with the assumption that the information in the high-dimensional data can be explained by low-dimensional factors, and these factors follow a VAR model

$$\boldsymbol{y}_t = \boldsymbol{\Lambda}\boldsymbol{f}_t + \boldsymbol{\epsilon}_t, \tag{2.12}$$

$$\boldsymbol{f}_t = \boldsymbol{A}_1\boldsymbol{f}_{t-1} + \cdots + \boldsymbol{A}_p\boldsymbol{f}_{t-P} + \boldsymbol{u}_t, \tag{2.13}$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{N\times K}$ is the factor loading, and $\boldsymbol{f}_t, \boldsymbol{u}_t \in \mathbb{R}^K$ denote the factors and the error term, respectively. The DFM has been extensively reviewed in the literature (Barigozzi, 2018; Doz and Fuleky, 2020; Stock and Watson, 2010, 2016) due to its effectiveness in handling high-dimensional data. One notable extension of the DFM is the factor-augmented VAR (FAVAR), which will be introduced in Section 2.2. Since DFMs and FAVARs share many similarities in their Bayesian inference, implementation details, and applications, this literature review leaves these aspects to Section 2.2.

Unlike for the coefficient matrix, dimension reduction techniques are not widely applied

[12] Let $G = (V, E)$ be an undirected graph, where V is the set of N nodes and $E \subseteq V \times V$ is the set of edges. A *spanning tree* of G is a subgraph $T = (V, E_T)$ satisfying: (1) T contains all nodes of G; (2) T is connected, i.e., for any $u, v \in V$ there exists a path in T between u and v; (3) T is acyclic.

in the variance-covariance matrix $\boldsymbol{\Omega}$ in VARs. A notable exception is the factor modeling of the error term (Pitt and Shephard, 1999) applied in Kastner and Huber (2020) and Hauzenberger et al. (2023b). Specifically, the error term is written as $\boldsymbol{\epsilon}_t = \boldsymbol{\Lambda} \boldsymbol{f}_t + \boldsymbol{\eta}_t$, with $\boldsymbol{f}_t$ and $\boldsymbol{\eta}_t$ following multivariate normal distributions with zero means and diagonal variance-covariance matrices, $\boldsymbol{\Sigma}_f$ and $\boldsymbol{\Sigma}_\eta$, respectively. This factor model decomposes the $\boldsymbol{\Omega}$ to $\boldsymbol{\Lambda}\boldsymbol{\Sigma}_f\boldsymbol{\Lambda}' + \boldsymbol{\Sigma}_\eta$, reducing the number of parameters if the number of factors is lower than $((2N+1-(8N+1)^{1/2})/2$ [13].

In addition to developing shrinkage priors and dimension reduction techniques, many studies have focused on alleviating the computational burden of high-dimensional VAR estimation. A major source of this burden arises from the need to compute the Cholesky decomposition of $\bar{\boldsymbol{V}}$, the posterior variance matrix defined in (2.7). In the standard formulation, this step has a computational complexity of $\mathcal{O}(N^6P^3)$. Carriero et al. (2019) adopted the decomposition, $\boldsymbol{\Omega} = \boldsymbol{H}^{-1}\boldsymbol{S}\left(\boldsymbol{H}^{-1}\right)'$ (see the definition of $\boldsymbol{H}$ and $\boldsymbol{S}$ on page 44) and proposed an equation-by-equation algorithm to sample parameters in each row of the coefficient matrix, rather than sampling all coefficients at once. This procedure reduced the computation complexity of the Cholesky decomposition to $\mathcal{O}(N^4P^3)$. Kastner and Huber (2020) also adopted an equation-by-equation algorithm, but mitigated the Cholesky decomposition by applying the sampling scheme in Bhattacharya et al. (2016), which relies on matrix multiplication and linear system solutions. The algorithm achieves a computation complexity of $\mathcal{O}(N^3PT^2)$, and the computational gains are most pronounced when $NP > T$. Beyond the MCMC schemes discussed, Hajargasht and Woźniak (2018), Gefang et al. (2023), and Chan and Yu (2022) implemented variational Bayes (see Jordan et al. (1999) for a review) to accelerate the computation. This methodology converts a sampling problem to an optimization problem, $\min_{q(\boldsymbol{A},\boldsymbol{\Omega}))\in\mathcal{Q}} D_{KL}(q(\boldsymbol{A},\boldsymbol{\Omega}) \,\|\, p(\boldsymbol{A},\boldsymbol{\Omega} \mid \boldsymbol{y}))$, where $\mathcal{Q}$ denotes a family of distributions, $q(\boldsymbol{A},\boldsymbol{\Omega})$ is the variational posterior, and $D_{KL}(q(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta})) = \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta})} \mathrm{d}\boldsymbol{\theta}$ is the Kullback-Leibler divergence between distribution $q(\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$. Instead of exploring the parameter space of $\boldsymbol{A}$ and $\boldsymbol{\Omega}$ stochastically as the MCMC does, variational Bayes optimizes the parameters in the variational posterior deterministically, which can substantially speed up the computation when the parameter space is high-dimensional. As shown in Chan and Yu (2022), fitting a VAR(4) with 100 variables and 300 observations takes about 70 minutes using the MCMC, but the variational

[13] This upper bound is called Ledermann bound (Ledermann, 1937), which assumes that the variance-covariance matrix $\boldsymbol{\Sigma}_f = \boldsymbol{I}_K$.

method only takes about 7 minutes.

2.1.5 Non-linear VAR

While linear VAR models are already highly useful, non-linear VARs offer greater flexibility in capturing the time-varying nature of underlying data structures, leading to more accurate forecasts (Clark, 2011; D'Agostino et al., 2013) and more reliable structural analyses (Cogley and Sargent, 2005; Primiceri, 2005). The remainder of the subsection will first introduce the time-varying parameter VAR (TVP-VAR) model (Primiceri, 2005), given its widespread adoption and the representative role in modeling both time-varying coefficient and variance-covariance matrices. Other nonlinear VARs will then be discussed.

The TVP-VAR was motivated by the debate over the causes of the Great Moderation – a period roughly spanning from the mid-1980s to the early 2000s, characterized by a significant decline in the volatility of the U.S. macroeconomic variables such as output growth and inflation. A central question was whether this stabilization resulted from a structural shift in monetary policy, particularly actions taken by the Federal Reserve, or from favorable shocks ("good luck"). To allow for both possibilities, the TVP-VAR incorporates time variation in both the coefficient and the variance-covariance matrices, as reflected in the following specification:

$$\boldsymbol{y}_t = \boldsymbol{A}_{t,1}\boldsymbol{y}_{t-1} + \cdots + \boldsymbol{A}_{t,P}\boldsymbol{y}_{t-P} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{H}_t^{-1}\boldsymbol{S}_t\left(\boldsymbol{H}_t^{-1}\right)'\right), \tag{2.14}$$

where the coefficient $\boldsymbol{A}_t = (\boldsymbol{A}_{t,1}, \ldots, \boldsymbol{A}_{t,P})$ are time-varying, the covariance-variance matrix $\boldsymbol{\Omega}_t$ follows the decomposition with $\boldsymbol{H}_t$ being a unit lower triangular matrix and $\boldsymbol{S}_t = \text{diag}(\boldsymbol{s}_t)$, for $\boldsymbol{s}_t = (s_{t,1}, \ldots, s_{t,N})'$. The time-varying parameters are assumed to evolve gradually over time, making the (geometric[14]) random walk a natural choice for modeling these gradual changes,

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} + \boldsymbol{u}_t^{\alpha}, \quad \boldsymbol{u}_t^{\alpha} \sim \mathcal{N}\left(\boldsymbol{0}, \Sigma_{\alpha}\right), \tag{2.15}$$

$$\boldsymbol{h}_t = \boldsymbol{h}_{t-1} + \boldsymbol{u}_t^{h}, \quad \boldsymbol{u}_t^{h} \sim \mathcal{N}\left(\boldsymbol{0}, \Sigma_{h}\right), \tag{2.16}$$

$$\log \boldsymbol{s}_t = \log \boldsymbol{s}_{t-1} + \boldsymbol{u}_t^{s}, \quad \boldsymbol{u}_t^{s} \sim \mathcal{N}\left(\boldsymbol{0}, \Sigma_{s}\right), \tag{2.17}$$

where $\boldsymbol{\alpha}_t = \text{vec}(\boldsymbol{A}_t')$, $\boldsymbol{h}_t$ denotes all non-zero and non-unit elements in $\boldsymbol{H}_t$. Σ_α and Σ_s are unconstrained symmetric positive definite matrices, and Σ_h is block diagonal such that the elements within the same row in $\boldsymbol{H}_t$ are correlated, while elements across different rows are

[14] Geometric random walk is defined in (5.8), where the logarithm of a parameter follows a random walk.

uncorrelated[15]. The geometric random walk modeling of the marginal variance, $\boldsymbol{s}_t$, is referred to as *stochastic volatility*, a prominent framework for expressing the marginal variance in VARs, see Clark and Mertens (2023) and Chan (2024) for the reviews. Although the TVP-VAR model may appear linear at first glance, since the lagged values remain linearly related to the observed time series, it is non-linear for two reasons. First, the geometric random walk specification introduces inherent non-linearity into the model. Second, even when $\boldsymbol{s}_t$ is assumed to be time-invariant and the model is a linear Gaussian state-space model (LGSSM)[16], with a transition equation for $(\boldsymbol{\alpha}_t', \boldsymbol{h}_t')'$ and a measurement equation for $\boldsymbol{y}_t$, practitioners (particularly econometricians) still treat the model as non-linear. This is because identical lagged values can produce different conditional means across time. Several model structures related to the TVP-VAR model have been proposed before and after its introduction. Canova (1993) and Cogley and Sargent (2001) assumed $\boldsymbol{\Omega}_t$ to be static. Cogley and Sargent (2005) considered both time-varying coefficients and stochastic volatility, but treated $\boldsymbol{H}_t$ as time-invariant. Instead of random walks, Kastner and Huber (2020) and Huber and Feldkircher (2019) considered AR(1) for elements in $\log \boldsymbol{s}_{t,i}$.

Del Negro and Primiceri (2015) provided the Bayesian inferential scheme of the TVP-VAR[17], which implements a blocked Gibbs sampler over four blocks: $\boldsymbol{\alpha}_{1:T}$, $\boldsymbol{h}_{1:T}$, $\boldsymbol{s}_{1:T}$ and $\{\boldsymbol{\Sigma}_\alpha, \boldsymbol{\Sigma}_h, \boldsymbol{\Sigma}_s\}$. The sampling of $\boldsymbol{\alpha}_{1:T}$ depends on an LGSSM, consisting of the measurement equation transformed from (2.14),

$$\boldsymbol{y}_t = \boldsymbol{Z}_t\boldsymbol{\alpha}_t + \boldsymbol{\epsilon}_t,$$

see the definition of $\boldsymbol{Z}_t$ in equation (2.5), and the transition equation (5.6), with $\boldsymbol{\alpha}_{1:T}$ being the latent states. Given this structure, the forward-filtering backward-sampling (FFBS) algorithm

[15] Most TVP-VAR studies assumes $\boldsymbol{\Sigma}_h$ as block diagonal to simplify the inference, which will be discussed in the next paragraph in the main content. Instead of sampling all elements in $\boldsymbol{h}_t$ at once, block diagonal restriction allows an equation-by-equation sampling of $\boldsymbol{h}_t$ according to the row indices of $\boldsymbol{H}_t$.

[16] The generalized linear Gaussian state-space model is defined in Harvey (1990), West and Harrison (1997) and Kitagawa and Gersch (2012) as

$$\boldsymbol{x}_t = \boldsymbol{G}_t\boldsymbol{x}_{t-1} + \boldsymbol{u}_t, \quad \boldsymbol{u}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_t), \tag{2.18}$$
$$\boldsymbol{y}_t = \boldsymbol{M}_t\boldsymbol{x}_t + \boldsymbol{\varepsilon}_t, \quad \boldsymbol{\varepsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_t), \tag{2.19}$$

where $\boldsymbol{y}_t$ denotes the observed variable and $\boldsymbol{x}_t$ is the latent state vector. The latent states evolve according to a VAR(1) process with transition matrix $\boldsymbol{G}_t$ and innovation covariance matrix $\boldsymbol{Q}_t$. The observations are linearly related to the states through the measurement matrix $\boldsymbol{M}_t$, with observation noise having covariance matrix $\boldsymbol{R}_t$.

[17] Although Primiceri (2005) introduced an earlier version of the algorithm, Del Negro and Primiceri (2015) identified an error in the sampling order and proposed a corrected version.

developed by Frühwirth-Schnatter (1994) and Carter and Kohn (1994) is a natural choice for sampling the latent states. The FFBS combines a forward Kalman filter pass to compute the filtering distributions and uses a backward simulation step to draw from the joint posterior of the state sequence given other parameters and the data, see Appendix A.2 for details. Similarly, the second block $\boldsymbol{h}_{1:T}$ can also be considered as the states in an LGSSM because the unit lower triangular $\boldsymbol{H}_t$ allows $\boldsymbol{H}_t(\boldsymbol{y}_t - \boldsymbol{A}_t\boldsymbol{x}_t) = \boldsymbol{S}_t^{1/2}\boldsymbol{\eta}_t$ to be rewritten as a measurement equation,

$$\boldsymbol{y}_t^* = \boldsymbol{M}_t\boldsymbol{h}_t + \boldsymbol{S}_t^{1/2}\boldsymbol{\eta}_t,$$

where $\boldsymbol{y}_t^* = \boldsymbol{y}_t - \boldsymbol{A}_t\boldsymbol{x}_t$, and $\boldsymbol{M}_t = \begin{pmatrix} 0 & \cdots & \cdots & 0 \\ -\boldsymbol{y}_{t,1}^* & 0 & \cdots & 0 \\ 0 & -\boldsymbol{y}_{t,[2]}^* & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -\boldsymbol{y}_{t,[N-1]}^* \end{pmatrix} \in \mathbb{R}^{N \times \frac{N(N-1)}{2}}$, with $\boldsymbol{y}_{t,[i]}^* = (\boldsymbol{y}_{t,1}^*, \ldots, \boldsymbol{y}_{t,i}^*)'$, for $i \in [N-1]$, and (5.7) is the transition equation. Since $\boldsymbol{\Sigma}_h$ is block diagonal, the non-zero and non-unit elements in each row of $\boldsymbol{H}_t$ can be sampled by an FFBS scheme. For the third block, the corresponding state-space model regarding $\boldsymbol{s}_t$ has the measurement equation,

$$\boldsymbol{y}_t^{**} = \log \boldsymbol{s}_t + \boldsymbol{\epsilon}_t^*,$$

where $\boldsymbol{y}_{t,i}^{**} = \log\left((\tilde{\boldsymbol{y}}_{t,i}^*)^2 + c\right)$, $\tilde{\boldsymbol{y}}_t^* = \boldsymbol{H}_t\boldsymbol{y}_t^*$, c is a small positive value (e.g. 0.001) to ensure the logarithm is well-defined, $\boldsymbol{\epsilon}_{t,i}^* \sim \log\chi_1^2$ is the log of a Chi-squared random variable with 1 degree of freedom, for $i \in [N]$, and (5.8) is the transition equation. Although this system is a non-Gaussian state-space model due to the distribution of $\boldsymbol{\epsilon}_{t,i}^*$, Kim et al. (1998) showed that a mixture of normal distributions can approximate this distribution. Specifically, $p(\boldsymbol{\epsilon}_{t,i}^*) \approx \sum_{j=1}^{7} p_j\phi\left(\boldsymbol{\epsilon}_{t,i}^*; m_j, v_j^2\right)$, where $\phi(\cdot)$ denotes the probability density function (PDF) of a normal distribution and the values of $\{p_j, m_j, v_j^2\}_{j=1}^{7}$ are in Appendix A.1. Given the mixture indicator $\boldsymbol{z}_{1:T}$ (which has the conditional posterior $p(\boldsymbol{z}_{t,i} = j \mid \boldsymbol{y}_{t,i}^{**}, \boldsymbol{s}_{t,i}) \propto p_j\phi\left(\boldsymbol{y}_{t,i}^{**}; \log s_{t,i} + m_j, v_j^2\right)$, for $t \in [T]$ and $i \in [N]$), one can approximate the non-Gaussian state-space model to a Gaussian one, and sample $\boldsymbol{s}_{1:T}$ using the FFBS. Finally, the fourth block about the variance-covariance matrices can be readily sampled when the corresponding priors are inverse-Wishart distributions.

Many studies about TVP-VARs, including those mentioned as well as Havlicek et al. (2010), Nakajima (2011), and Antonakakis et al. (2020), to name a few, used fewer than 10 variables to mitigate over-parameterization and the computational burden of the TVP-VAR.

These challenges stem from the time-invariant VARs and are exacerbated in TVP-VARs as the number of parameters grows with time. Similar to the solutions introduced in the framework of high-dimensional VARs, both shrinkage priors and dimension reduction techniques alleviate the issues. Belmonte et al. (2014) imposed the Bayesian lasso on parameters in a non-centered parameterization of the TVP-VAR (Frühwirth-Schnatter and Wagner, 2010),

$$\boldsymbol{y}_t = \boldsymbol{Z}_t \left(\boldsymbol{\alpha}_0 + (\boldsymbol{\Sigma}_{\boldsymbol{\alpha}})^{\frac{1}{2}} \tilde{\boldsymbol{\alpha}}_t \right) + \boldsymbol{\epsilon}_t,$$

$$\tilde{\boldsymbol{\alpha}}_t = \tilde{\boldsymbol{\alpha}}_{t-1} + \boldsymbol{u}_t^{\tilde{\alpha}}, \quad \boldsymbol{u}_t^{\tilde{\alpha}} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{I}_{N^2 P}\right),$$

where $\boldsymbol{\alpha_0}$ and $\tilde{\boldsymbol{\alpha}}_t$ denote the time-invariant and time-varying parts of the coefficients. The shrinkage prior is imposed on both $\boldsymbol{\alpha}_0$ and the diagonal terms of $(\boldsymbol{\Sigma}_{\boldsymbol{\alpha}})^{\frac{1}{2}}$, leading to a parsimonious model with generally small coefficient magnitudes and limited time variation. Prüser (2021) and Huber et al. (2021) also applied this technique with different prior choices. Koop et al. (2009), Eisenstat et al. (2016), and Chan (2023) proposed the spike-and-slab priors to Σ_α, Σ_h and Σ_s to determine whether the corresponding block of parameters is time-invariant or time-varying. In the dimension reduction strand, Brune et al. (2022) and Cubadda et al. (2025) proposed time-varying variants of the reduced-rank VAR and the multivariate index autoregression, respectively, by modeling the decomposed parameters as random walks. Chan et al. (2020) applied a DFM to the time-varying coefficients, i.e., $\boldsymbol{\alpha}_t$ was regarded as the $\boldsymbol{y}_t$ in (2.12), and (2.13) was replaced by random walks. While Chan et al. (2020) followed the assumption in the TVP-VAR that the parameters evolve gradually, Fischer et al. (2023) extended the work by adding discrete factors to account for abrupt changes. Time-varying DFM (Del Negro and Otrok, 2008) reduced the number of parameters by compressing information in high-dimensional observed data to low-dimensional factors.

Moving to the modeling of $\boldsymbol{\Omega}_t$ in high-dimensional TVP-VARs, Carriero et al. (2016a) proposed common stochastic volatility, motivated by the observation that the volatilities of many U.S. macroeconomic time series are highly correlated. They decomposed $\boldsymbol{\Omega}_t$ to $s_t\boldsymbol{\Sigma}$, where a scalar s_t explains all the time variation in $\boldsymbol{\Omega}_t$, and $\boldsymbol{\Sigma}$ is a constant positive definite matrix. The evolution of $\log \boldsymbol{s}_{t,i}$ (for $i \in [N]$) follows an AR(1) without the intercept to avoid identifiability issues. Instead of specifying only one factor, the factor stochastic volatility mentioned in the review of high-dimensional VARs assumes multiple factors to govern the stochastic volatility.

Apart from imposing shrinkage priors and dimension reductions, some research contributes to improving computational efficiency during the inference of the TVP-VAR. Inspired by Raftery et al. (2010), Koop and Korobilis (2013) simplified the sampling by approximating $\boldsymbol{\Omega}_t$, $\boldsymbol{\Sigma}_\alpha$, $\boldsymbol{\Sigma}_h$, and $\boldsymbol{\Sigma}_s$ using terms derived from the Kalman filter with forgetting factors, so that the sampling of these matrices can be omitted. Loaiza-Maya and Nibbering (2022) proposed the variational inference for state-space models and took the TVP-VAR as an example.

While the TVP-VAR assumes the change of parameters over time is gradual, a class of non-linear VARs allows parameters to shift according to different regimes. For instance, the model may include separate parameter sets corresponding to periods of economic expansion and downturn. This class of VARs with M regimes can be expressed as

$$\boldsymbol{y}_t = \sum_{m=1}^{M} \left(\boldsymbol{A}_{m,1}\boldsymbol{y}_{t-1} + \cdots + \boldsymbol{A}_{m,P}\boldsymbol{y}_{t-P} + \boldsymbol{\epsilon}_t^{(m)} \right) G(z_t; \boldsymbol{\theta}_m),$$

where $\boldsymbol{A}_{m,1}, \ldots, \boldsymbol{A}_{m,P}$ correspond to the coefficients in the m-th regime ($m \in [M]$) and $\boldsymbol{\epsilon}_t^{(m)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Omega}_m)$ denotes the error term in this regime, $G(z_t; \boldsymbol{\theta}_m)$ is a function with parameter $\boldsymbol{\theta}_m$, z_t is an observed or latent variable to determine the regime, which could be an endogenous lagged variable (Fry-Mckibbin and Zheng, 2016), a linear combination of lagged values (Galvao and Marcellino, 2014) or a latent variable to specify the regime explicitly (Krolzig and Krolzig, 1997), and $\sum_{m=1}^{M} G(z_t; \boldsymbol{\theta}_m) = 1$. The number of regimes is typically chosen by the specific research questions, but information criteria can also be applied, see (Kwon et al., 2008) for an example. The choice of $G(z_t; \boldsymbol{\theta}_m)$ defines the specific non-linear VAR models, which will be introduced in the following paragraphs.

If $G(z_t; \boldsymbol{\theta}_m) = \mathbb{I}_{z_t \in R_m}$, where $\mathbb{I}$ is the indicator function, z_t is an observed lagged value, and R_m is a range within which z_t can fall, then the model is a threshold VAR (Tsay, 1998), which assumes the VAR coefficients change if z_t passes some thresholds. Huber (2014) provided the MCMC scheme of threshold VAR, which imposed a normal prior on the boundaries of R_m and sampled these boundaries using Metropolis-Hastings. Teräsvirta et al. (2010) and Hubrich and Teräsvirta (2013) reviewed threshold VARs, and the recent developments of this model with high-dimensional data have been studied in the frequentist literature (Liu and Chen, 2020; Zhang et al., 2022a), while the Bayesian approaches are rarely found.

Markov-switching (MS) VAR (Krolzig and Krolzig, 1997) uses $G(z_t; \boldsymbol{\theta}_m) = \mathbb{I}_{z_t = m}$ by defining $z_t \in [M]$ as a latent variable to indicate the regime. As revealed by the model name, $z_{1:T}$ is a discrete Markov chain with the transition probability, $p(z_t = m \mid z_{t-1} = m') = p_{mm'}$, such that $\sum_{m=1}^{M} p_{mm'} = 1$, for $m, m' \in [M]$. The MS-VAR has been widely applied in econometrics (Rubio-Ramirez et al., 2005; Sims et al., 2008; Sims and Zha, 2006). Similar to other VAR models discussed above, research on the MS-VAR has increasingly focused on incorporating high-dimensional data (Li et al., 2022; Maung, 2021) and time-varying parameters, such as the time-varying transition probabilities investigated in Bazzi et al. (2017) and the gradual time variation within each regime (Inayati et al., 2024).

While the interpretation of threshold VARs and MS-VARs focuses on parameters within individual regimes, the smooth-transition (ST) VAR (Anderson and Vahid, 1998), as discussed in the surveys by Dijk et al. (2002) and Hubrich and Teräsvirta (2013), introduces a smoothed approach by considering the distance of z_t (an observed and lagged value) from a particular value. The ST-VAR typically involves two regimes, with $G(z_t; \boldsymbol{\theta}_1) = 1/(1 + \exp(-\gamma(z_t - c)))$ for the logistic ST-VAR, or $G(z_t; \boldsymbol{\theta}_1) = \exp(-\gamma(z_t - c)^2)$ for the exponential ST-VAR, and $G(z_t; \boldsymbol{\theta}_2) = 1 - G(z_t; \boldsymbol{\theta}_1)$, where γ is a positive smoothing parameter, and c is the value of z_t at which the data is within the first regime. As z_t moves away from c, the data gradually approaches the second regime. Gefang and Strachan (2009) and Bruns and Piffer (2024) provided the MCMC schemes of ST-VAR with non-conjugate and conjugate priors on coefficients and variance-covariance matrices.

The non-linear VAR models defined so far rely on parametric approaches to describe the dynamics of the parameters. In contrast, nonparametric VAR models offer a more flexible method to capture the relationship between $\boldsymbol{y}_t$ and its lagged values. Härdle et al. (1998) contributed to the early work of nonparametric VARs by employing the local polynomial technique (Fan, 1996) to optimize the parameters conditional on the lagged values. More recently, Kalli and Griffin (2018) adopted a Dirichlet process mixture model (Ferguson, 1973) to model the joint distribution of $\boldsymbol{y}_t$ and $\boldsymbol{x}_t$, then used Bayes Theorem to construct an infinite mixture model of $p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)$. Hauzenberger et al. (2025) employed Gaussian processes, for a textbook review see Williams and Rasmussen (2006), to construct two types of functions taking the own-lag and cross-lag values as inputs, respectively. Huber and Rossini (2022) structured the VAR as a tree model using the Bayesian additive regression tree (Chipman et al., 2010), which was then applied to model the coefficients of VARs based on lagged values in Hauzenberger et al. (2022).

Many deep learning techniques can be considered as non-linear VARs when they take lagged values as the input and $\boldsymbol{y}_t$ as the output. Some examples include: multilayer perceptron (Rosenblatt, 1958), recurrent neural network (Rumelhart et al., 1986), long short-term memory (Hochreiter and Schmidhuber, 1997), and transformer (Vaswani et al., 2017), see Wang et al. (2024b) for a comprehensive review and recent development of these models. These techniques have demonstrated strong forecasting performance in Zhang (2003), Yu et al. (2019), and Hewamalage et al. (2021), among others, and the corresponding impulse response analysis and Granger causality have been explored in Cabanilla and Go (2019) and Tank et al. (2021), respectively. The rise of deep learning has established a distinct domain of time series analysis, which diverges from the models considered in this thesis. Therefore, this section will not explore this research field in depth, but the methodological background about deep learning which is related to this thesis will be introduced in Section 2.4.

2.2 Factor Augmented Vector Autoregression

2.2.1 Modeling of Standard FAVAR

The previous subsection reviewed VARs, which can handle high-dimensional data by implementing techniques to construct parsimonious models. One of the techniques is the dynamic factor model (DFM), which compresses high-dimensional data to a few factors and models the dynamics of these factors with a VAR. While the DFM has been extensively applied in macroeconomic research, it cannot capture interactions among key observed variables, since they are excluded from the VAR component. To address this challenge and retain the analysis of high-dimensional data, Bernanke et al. (2005) proposed factor-augmented vector autoregression (FAVAR), which was built upon the DFM with the following expression:

$$\begin{pmatrix} \boldsymbol{d}_t \\ \boldsymbol{y}_t \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Lambda}_{df} & \boldsymbol{\Lambda}_{dy} \\ \boldsymbol{0} & \boldsymbol{I}_N \end{pmatrix} \begin{pmatrix} \boldsymbol{f}_t \\ \boldsymbol{y}_t \end{pmatrix} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}), \tag{2.20}$$

$$\begin{pmatrix} \boldsymbol{f}_t \\ \boldsymbol{y}_t \end{pmatrix} = \boldsymbol{A}_1 \begin{pmatrix} \boldsymbol{f}_{t-1} \\ \boldsymbol{y}_{t-1} \end{pmatrix} + \cdots + \boldsymbol{A}_P \begin{pmatrix} \boldsymbol{f}_{t-P} \\ \boldsymbol{y}_{t-P} \end{pmatrix} + \boldsymbol{u}_t, \quad \boldsymbol{u}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Omega}), \tag{2.21}$$

where $\boldsymbol{d}_t \in \mathbb{R}^M$, $\boldsymbol{y}_t \in \mathbb{R}^N$, $\boldsymbol{f}_t \in \mathbb{R}^K$ denotes the high-dimensional data, key variables (or observed factors) and latent factors, respectively, with $N + K \ll M$, $\boldsymbol{d}_{1:T}$ and $\boldsymbol{y}_{1:T}$ are standardized so that the intercepts are omitted. $\boldsymbol{\Lambda}_{df} \in \mathbb{R}^{M\times K}$ and $\boldsymbol{\Lambda}_{dy} \in \mathbb{R}^{M\times N}$ are the factor loadings which loads the low-dimensional (un)observed factors to $\boldsymbol{d}_t$, for $t \in [T]$, $\boldsymbol{A} = (\boldsymbol{A}_1, \ldots, \boldsymbol{A}_P)$

represents the coefficient matrix of the VAR, $\boldsymbol{\Sigma}$ and $\boldsymbol{\Omega}$ correspond to the respective variance-covariance matrices in the factor model and the VAR, where $\boldsymbol{\Sigma}$ is a diagonal matrix with only the first M diagonal terms being non-zero.

The FAVAR model is both related to and distinct from DFMs and VARs. Compared to the DFM defined in (2.12) and (2.13), the FAVAR incorporates latent and observed factors in (2.20) to explain the comovement of the high-dimensional data, instead of only the latent factors in the DFM. Similarly, (2.21) models the dynamics of both types of factors. These extensions enable the FAVAR to express how variables in $\boldsymbol{y}_t$ interact with each other. In contrast to the VAR in (2.1), which models all the observed variables, the FAVAR separates them into $\boldsymbol{d}_t$ and $\boldsymbol{y}_t$, and models only the low-dimensional $\boldsymbol{y}_t$ in its VAR part. This structure allows the FAVAR to handle high-dimensional data without incurring over-parameterization.

Before delving into the inferential schemes of the FAVAR, two questions need to be answered for the model setup. The first concerns how to partition the data to $\boldsymbol{y}_t$ and $\boldsymbol{d}_t$. The most straightforward approach is to partition it based on the research question. For example, in analyzing the domestic and international transmission of monetary policy shocks to the UK economy, one can include the policy rate that the Bank of England controls and key UK macroeconomic variables in $\boldsymbol{y}_t$, while incorporating short-term interest rates from other countries and other UK macroeconomic indicators in $\boldsymbol{d}_t$. Alternatively, a more agnostic and data-driven approach can be taken. For instance, Koop and Korobilis (2014) fixed $\boldsymbol{y}_t$ as three key macroeconomic variables and used the dynamic model selection (Raftery et al., 2010) to determine $\boldsymbol{d}_t$ from 17 financial variables for each time point to construct a financial condition index. The second question is to choose the number of latent factors. The information criteria proposed by Bai and Ng (2002) for the DFM is a prominent method, which can be easily extended to the FAVAR as shown in Abbate et al. (2016), Daniele and Schnaitmann (2019), and Boivin et al. (2020), among others. Specifically, this type of information criteria is defined as $\text{IC}(K) = \text{MSE}\left(\boldsymbol{d}_{1:T}, \hat{\boldsymbol{d}}_{1:T}^{(K)}\right) + Kg(M,T)$, where MSE denotes mean squared error, $\hat{\boldsymbol{d}}_{1:T}^{(K)}$ is the conditional mean of $\boldsymbol{d}_{1:T}$ given there are K latent factors, $g(M,T)$ is a penalty term satisfying $g(M,T) \to 0$ and $c_{M,T}^2 g(M,T) \to \infty$ as $M,T \to \infty$, where $c_{M,T} = \min\{\sqrt{M}, \sqrt{T}\}$. Since factors can be estimated using the principal component analysis, another straightforward method is to inspect the eigenvalues of $\boldsymbol{D}'\boldsymbol{D}$, where $\boldsymbol{D} = (\boldsymbol{d}_1, \ldots, \boldsymbol{d}_T)'$. If the first K eigenvalues, ordered in descending magnitude, explain a sufficiently large proportion of the variance in $\boldsymbol{D}$, then K can be taken as the number of factors.

2.2.2 Bayesian Inference of Standard FAVAR

The FAVAR typically has two inferential schemes proposed by Bernanke et al. (2005) to estimate latent factors and unknown parameters: one- and two-step procedures. The one-step procedure considers the FAVAR as a state-space model with $\boldsymbol{f}_t$ being the latent states, so a Gibbs sampler can iteratively sample $\boldsymbol{f}_{1:T}$ using algorithms such as the FFBS introduced in the TVP-VAR and then sample other parameters from their conditional posteriors given $\boldsymbol{f}_{1:T}$. The benefit of this fully Bayesian method is that it can model the uncertainty of all unknown terms. Since the dynamics of $\boldsymbol{f}_{1:T}$ are explicitly defined as in (2.21), the inferred latent factors naturally adhere to this VAR model. However, there are two disadvantages of the one-step procedure. First, the corresponding MCMC is complicated because the Markov chains may exhibit autocorrelation due to the interdependence between factors and loadings in their full conditionals. While longer chains could address this issue, this solution increases computational costs given the high dimensionality of $\boldsymbol{d}_t$. Second, factor loadings $\boldsymbol{\Lambda}_{df}$, $\boldsymbol{\Lambda}_{dy}$ and $\boldsymbol{f}_{1:T}$ are only identified up to some linear transformations. Specifically, $\tilde{\boldsymbol{\Lambda}}_{df}$, $\tilde{\boldsymbol{\Lambda}}_{dy}$ and $\tilde{\boldsymbol{f}}_{1:T}$ lead to the same likelihood if $\tilde{\boldsymbol{\Lambda}}_{df} = \boldsymbol{\Lambda}_{df}\boldsymbol{Q}_1$, $\tilde{\boldsymbol{\Lambda}}_{dy} = \boldsymbol{\Lambda}_{dy} + \boldsymbol{\Lambda}_{df}\boldsymbol{Q}_2$, $\tilde{\boldsymbol{f}}_t = \boldsymbol{Q}_1^{-1}\boldsymbol{f}_t - \boldsymbol{Q}_1^{-1}\boldsymbol{Q}_2\boldsymbol{y}_t$, for an invertible matrix $\boldsymbol{Q}_1 \in \mathbb{R}^{K\times K}$ and a matrix $\boldsymbol{Q}_2 \in \mathbb{R}^{K\times N}$. Note that $\left(\tilde{\boldsymbol{f}}_t', \boldsymbol{y}_t'\right)'$ still follows a VAR model[18]. Although the indeterminacy does not affect forecasting or impulse response analysis, identifying the factors is essential for interpretation – which is typically a key objective in empirical applications. The FAVAR requires $K^2 + KN$ (corresponding to the number of elements in $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$) restrictions to identify the factors (Bai et al., 2016). Bernanke et al. (2005) restricted the upper K-by-K block of $\boldsymbol{\Lambda}_{df}$ and the upper K-by-N block of $\boldsymbol{\Lambda}_{dy}$ as identity and zero matrices, respectively. Bai et al. (2016) provided three sets of restrictions to loadings and $\boldsymbol{\Omega}$, and proved the consistency of the estimation based on these restrictions. Alternatively, Beyeler and Kaufmann (2021), motivated by the idea that sparsity helps identifiability (Kaufmann and Schumacher, 2019), proposed a hierarchical spike-and-slab prior to induce sparsity in the two

[18] Let $\boldsymbol{Q} = \begin{pmatrix} \boldsymbol{Q}_1^{-1} & -\boldsymbol{Q}_1^{-1}\boldsymbol{Q}_2 \\ \boldsymbol{0} & \boldsymbol{I}_N \end{pmatrix}$, $\tilde{\boldsymbol{f}}_t$ and $\boldsymbol{y}_t$ are modeled as $\begin{pmatrix} \tilde{\boldsymbol{f}}_t \\ \boldsymbol{y}_t \end{pmatrix} = \boldsymbol{Q}\boldsymbol{A}_1\boldsymbol{Q}^{-1}\begin{pmatrix} \tilde{\boldsymbol{f}}_{t-1} \\ \boldsymbol{y}_{t-1} \end{pmatrix} + \cdots + \boldsymbol{Q}\boldsymbol{A}_P\boldsymbol{Q}^{-1}\begin{pmatrix} \tilde{\boldsymbol{f}}_{t-P} \\ \boldsymbol{y}_{t-P} \end{pmatrix} + \tilde{\boldsymbol{u}}_t$, $\tilde{\boldsymbol{u}}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}\boldsymbol{\Omega}\boldsymbol{Q}')$.

loading matrices. They further employed an ex-post identification scheme to address the issues of permutation and sign switching.

The two-step procedure firstly extracts unobserved factors based on principal components of $\boldsymbol{D}$, which was justified by Stock and Watson (2002) as consistent estimates of factors, and then considers these extracted factors as observed to infer the remaining parameters. The standard principal component analysis (PCA) extracted these components by optimizing

$$\min_{\boldsymbol{C},\boldsymbol{\Lambda}_{dc}} \|\boldsymbol{D}-\boldsymbol{C}\boldsymbol{\Lambda}_{dc}'\|_F^2, \text{ subject to } \boldsymbol{\Lambda}_{dc}'\boldsymbol{\Lambda}_{dc}=\boldsymbol{I}_K, \tag{2.22}$$

where $\boldsymbol{C}\in\mathbb{R}^{T\times K}$ and $\boldsymbol{\Lambda}_{dc}\in\mathcal{R}^{N\times K}$, $\|\boldsymbol{M}\|_F=\sqrt{\text{tr}(\boldsymbol{M}\boldsymbol{M}^T)}$ denotes the Frobenius norm of a matrix $\boldsymbol{M}$, tr$(\cdot)$ is the trace of a matrix. The columns of the optimized $\boldsymbol{\Lambda}_{dc}$ are the eigenvectors of $\boldsymbol{D}'\boldsymbol{D}$ corresponding to the K largest eigenvalues and $\boldsymbol{C}=\boldsymbol{D}\boldsymbol{\Lambda}_{dc}$. By ordering these columns according to the descending eigenvalues, the identifiability issue can be readily solved. Note that $\boldsymbol{c}_t$, for $\boldsymbol{C}=(\boldsymbol{c}_1,\dots,\boldsymbol{c}_T)'$, is typically not directly treated as $\boldsymbol{f}_t$ (see Belviso and Milani (2006) for an exception), because some of the information stored in $\boldsymbol{c}_t$ can be explained by $\boldsymbol{y}_t$. To avoid $\boldsymbol{f}_t$ spanning on $\boldsymbol{y}_t$, i.e., there is no matrix $\boldsymbol{M}\in\mathbb{R}^{K\times N}$ such that $\boldsymbol{f}_t=\boldsymbol{M}\boldsymbol{y}_t$, for $t\in[T]$, Bernanke et al. (2005) modified $\boldsymbol{c}_t$ to $\boldsymbol{f}_t$ by implementing a linear regression $\boldsymbol{c}_t=\boldsymbol{B}_s\boldsymbol{c}_t^s+\boldsymbol{B}_y\boldsymbol{y}_t+\boldsymbol{e}_t$, where $\boldsymbol{c}_t^s$ denotes the principal components of those variables in $\boldsymbol{d}_t$ which do not respond contemporaneously to the monetary policy shock. After estimating this linear regression using the OLS, one can obtain $\boldsymbol{f}_t$ as $\boldsymbol{c}_t-\boldsymbol{B}_y\boldsymbol{y}_t$. Boivin et al. (2010) proposed another method to modify principal components. They firstly used $\boldsymbol{c}_{1:T}$ as $\boldsymbol{f}_{1:T}$ to estimate $\boldsymbol{\Lambda}_{df}$ and $\boldsymbol{\Lambda}_{dy}$, then treated the principal components of $\boldsymbol{D}-\boldsymbol{Y}\boldsymbol{\Lambda}_{dy}'$, as $\boldsymbol{f}_{1:T}$. One concern about the two-step procedure is that the VAR may not be the optimal model to explain the evolution of latent factors. This concern motivated Dufour and Stevanović (2013) to replace the VAR with a VARMA, which added lagged errors to the VAR[19]. Nevertheless, the VAR is still the most common model to express factors because (1) empirical studies found that the performance of a FAVAR estimated using the two-step procedure is similar to the one using the one-step procedure (Bernanke et al., 2005); (2) the estimation of the VAR parameters is straightforward and the further analysis can be easily derived (Lütkepohl, 2013).

[19] see a recent review about the VARMA in Düker et al. (2025).

2.2.3 Applications of Standard FAVAR

The applications of the FAVAR are similar to those of the VAR, with forecasting, impulse response analysis, and variance decomposition being the most common ones, whereas Granger causality is rarely applied. If a FAVAR is used for forecasting, $\boldsymbol{y}_{T+h}$ is usually of interest, for some horizon $h \in \mathbb{N}$, instead of the high-dimensional data $\boldsymbol{d}_{T+h}$ (though this forecasting task is feasible). The forecasting procedure is the same as a standard VAR.

The impulse response analysis from an estimated FAVAR is more complex than that using a VAR. First, the source of the shocks can be any unexpected value in the (un)observed factors, as long as one can interpret them. For example, Belviso and Milani (2006) applied the PCA to different categories of macroeconomic data, allowing the factors to be interpreted as the index of each data category. Second, the response to a shock can either be about the variables in the observed factors or those in the high-dimensional data. In contrast to the forecasting tasks, the impulse response analysis of $\boldsymbol{d}_t$ is more important in some studies. For example, Stenvall (2024) investigated the responses of employment growth in different sectors (these are the variables in $\boldsymbol{d}_t$) to financial shocks, and found that the responses were heterogeneous across manufacturing, construction, financial services, and the information and communication sectors. The impulse responses of $\boldsymbol{y}_{t+h}$ are the same as those using a standard VAR. Recall that if the variance-covariance matrix is non-diagonal, then one can decompose $\boldsymbol{\Omega}$ in (2.21) to $\boldsymbol{P}^{-1}(\boldsymbol{P}^{-1})'$ (see (2.8) for details), and the response of the i-th time series in $\boldsymbol{y}_{t+h}$ to a shock in the j-th variable in $(\boldsymbol{f}'_t, \boldsymbol{y}'_t)$ is $\boldsymbol{\Psi}_{h,(i+K,j)}/\boldsymbol{P}^{-1}_{(j,j)}$, for $i \in [N]$ and $j \in [N+K]$. The impulse response of $\boldsymbol{d}_{t,m}$ to a shock in the same variable in $(\boldsymbol{f}'_t, \boldsymbol{y}'_t)$ is $(\boldsymbol{\Lambda\Psi}_h)_{(m,j)}$, where $\boldsymbol{\Lambda} = \begin{pmatrix} \boldsymbol{\Lambda}_{df} & \boldsymbol{\Lambda}_{dy} \\ \boldsymbol{0} & \boldsymbol{I}_N \end{pmatrix}$ and $m \in [M]$.

The FAVAR framework can also be used to compute two types of variance decompositions, which assess the proportion of the variance in $\boldsymbol{d}_{t+h}$ or $\boldsymbol{y}_{t+h}$ that is attributed to the variation in selected data of interest. The first type quantifies the proportion of variance explained by the comovement within $\boldsymbol{d}_t$. For the m-th variable in $\boldsymbol{d}_{t+h}$, this variance proportion is written as $\frac{\boldsymbol{\Lambda}_{(m,\cdot)}\boldsymbol{\Omega}\boldsymbol{\Lambda}'_{(m,\cdot)}}{\boldsymbol{\Lambda}_{(m,\cdot)}\boldsymbol{\Omega}\boldsymbol{\Lambda}'_{(m,\cdot)}+\boldsymbol{\Sigma}_{(m,m)}}$, for $m \in [M]$. The second type is the forecast error variance decomposition, which is closely related to the one applied in VAR, so one can refer to Section 2.1.3 for this decomposition applied to $\boldsymbol{y}_{t+h}$ in the FAVAR. When $\boldsymbol{d}_{t+h}$ is of interest, the proportion of forecast error variance of $\boldsymbol{d}_{t+h,m}$ (given the information available up to time point t) explained by the j-th variable in $(\boldsymbol{f}'_t, \boldsymbol{y}'_t)'$

is $\sum_{h'=0}^{h-1}\left(\sum_{i=1}^{N+K}\mathbf{\Lambda}_{(m,i)}\mathbf{\Psi}_{h',(i,j)}\right)^2/(\sum_{j'=1}^{N+K}\sum_{h'=0}^{h-1}\left(\sum_{i=1}^{N+K}\mathbf{\Lambda}_{(m,i)}\mathbf{\Psi}_{h',(i,j')}\right)^2+\mathbf{\Sigma}_{(m,m)})$, for $m \in [M]$ and $j \in [N+K]$.

As in the previous subsection about VARs, the remainder of this subsection introduces the development of the standard FAVAR model in high-dimensionality and non-linearity, with an emphasis on the latter, since the standard FAVAR is already designed to handle high-dimensional data.

2.2.4 High-dimensional FAVAR

The research about constructing parsimonious models in the FAVAR framework is limited. Most of the exceptions focus on imposing a Minnesota prior to the VAR part (Belke and Osowski, 2019; Korobilis, 2013a; Lu and Zhu, 2023). Lin and Michailidis (2020) added a lasso penalty with respect to $\mathbf{\Lambda}_{dy}$ and the VAR coefficients to its least squares-based loss function to achieve sparsity in both equations. Beyeler and Kaufmann (2021) induced sparsity by imposing a spike-and-slab prior to the factor loadings to facilitate both interpretation and identifiability of factors.

2.2.5 Non-linear FAVAR

Compared to high-dimensional extensions, the exploration of non-linearity in FAVAR models has received more attention in the literature. The discussion begins by examining models that incorporate a non-linear VAR component, as they build directly on the nonlinear VAR framework introduced earlier. It then reviews studies that integrate non-linear factor structures into the FAVAR framework, both with and without a non-linear VAR. Finally, alternative non-linear dimension reduction techniques will be introduced.

FAVARs with non-linear VARs share the same motivation as the models described in Section 2.1.5, since there are structural changes in the dynamics of (un)observed factors. Korobilis (2013a) followed Primiceri (2005) to propose the FAVAR with a TVP-VAR and estimated the model using the two-step procedure, so the estimation of the factor model and the TVP-VAR can be separated. This model framework has been extensively applied in Bianchi et al. (2009), Baumeister et al. (2010), Ellis et al. (2014), Ren et al. (2020), Prüser and Schlösser (2020), and Ren et al. (2020). Huber and Fischer (2018) adopted a Markov-switching VAR with two regimes. This method was then applied in Corrado et al. (2021) to analyze the effect of quantitative easing.

The factor model part can also be non-linear because the (un)observed factors may become more or less important to some variables in different periods (Breitung and Eickmeier, 2011; Stock and Watson, 2009). It has been shown in DFMs that time-varying factor loadings helped examine how global business cycle (e.g., factors extracted from GDP variables from different countries) affects individual countries (Del Negro and Otrok, 2008) and improved forecasting (Banerjee et al., 2008). Since the FAVAR is a variant of the DFM, it is natural to consider time-varying loading in the FAVAR framework. Liu et al. (2011) modeled the elements in the time-varying loading, $(\boldsymbol{\Lambda}_{t,df}, \boldsymbol{\Lambda}_{t,dy})$, as random walks,

$$\boldsymbol{\lambda}_t = \boldsymbol{\lambda}_{t-1} + \boldsymbol{u}_t^{\lambda}, \boldsymbol{u}_t^{\lambda} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_{\lambda}\right), \tag{2.23}$$

where $\boldsymbol{\lambda}_t = \text{vec}\left((\boldsymbol{\Lambda}_{t,df}, \boldsymbol{\Lambda}_{t,dy})\right)$, and followed Primiceri (2005) to construct a VAR with constant coefficients and time-varying variance-covariance matrix, $\boldsymbol{\Omega}_t$. The authors adopted the one-step procedure to estimate unknown parameters by iterating three FFBSs for sampling $\boldsymbol{f}_{1:T}$, $\boldsymbol{\lambda}_{1:T}$ and $\boldsymbol{\Omega}_{1:T}$, and sampled time-invariant VAR coefficients given the factors and other parameters. Eickmeier et al. (2015) constructed a more flexible model by adding the time-varying VAR coefficients, $\boldsymbol{\alpha}_t = \text{vec}(\boldsymbol{A}_t)$, as in (5.6), and modeled the loadings as in (2.23). This model has been applied in Abbate et al. (2016) for studying the transmission of financial shocks. Koop and Korobilis (2014) constructed a similar model but assumed all the variance-covariance matrices of the FAVAR system ($\boldsymbol{\Sigma}_t$ and $\boldsymbol{\Omega}_t$) and the random walks ($\boldsymbol{\Sigma}_{\lambda}$ and $\boldsymbol{\Sigma}_{\alpha}$) to be time-varying. After extracting the factors from the PCA, the authors used two Kalman filters with forgetting factors, proposed by Koop and Korobilis (2013), to estimate the loadings and VAR coefficients. An alternative model was proposed by Hacioglu and Tuzcuoglu (2016), which employed a similar idea as the threshold VAR, by constructing each element in $\boldsymbol{\lambda}_t$ as $\boldsymbol{\lambda}_{t,i} = \boldsymbol{\lambda}^*_{t,i}\mathbb{I}(\boldsymbol{\lambda}^*_{t,i} > \delta_i)$, where $\boldsymbol{\lambda}^*_{t,i}$ follows an AR(1) process, for $i \in [M(K+N)]$, and all the other parameters are set to be constant.

All the above linear and non-linear FAVARs estimated factors via a linear state-space model or used the PCA to extract factors. An alternative model to achieve non-linear FAVARs is to extract factors from non-linear dimension reduction methods. Fu et al. (2024) applied the local PCA (Su and Wang, 2017) to optimize the principal components as

$$\min_{\boldsymbol{C}^{(t)}, \boldsymbol{\Lambda}_{t,dc}} \|\boldsymbol{D}^{(t)} - \boldsymbol{C}^{(t)}\boldsymbol{\Lambda}'_{t,dc}\|_F^2, \text{ subject to } \boldsymbol{\Lambda}'_{t,dc}\boldsymbol{\Lambda}_{t,dc} = \boldsymbol{I}_K, \tag{2.24}$$

where $\boldsymbol{D}^{(t)} = \left(\boldsymbol{D}_1^{(t)}, \ldots, \boldsymbol{D}_M^{(t)}\right) \in \mathbb{R}^{T\times M}$, $\boldsymbol{D}_m^{(t)} = \left(k_{\tilde{h},(1,t)}^{1/2}\boldsymbol{D}_{(1,m)}, \ldots, k_{\tilde{h},(T,t)}\boldsymbol{D}_{(T,m)}\right)'$, $k_{\tilde{h},(t',t)}^{1/2} = \tilde{h}^{-1}K\left(\frac{t-t'}{T\tilde{h}}\right)$, $\tilde{h}$ is bandwidth and $K(\cdot)$ is a kernel, for $t, t' \in [T]$. Then the latent factors were estimated as $\boldsymbol{f}_t = (\boldsymbol{\Lambda}_{t,dc}'\boldsymbol{\Lambda}_{t,dc})^{-1}\boldsymbol{\Lambda}_{t,dc}'\boldsymbol{d}_{t,dc}$. For each time point, the local PCA assigned more weights to the observations of which the time indices are close to this time point, and applied the PCA to obtain the principal components.

Klieber (2024) considered two methods to extract factors non-linearly, namely the locally linear embedding (Roweis and Saul, 2000) from the manifold-based learning literature, see Izenman (2012) for an introduction, and autoencoder (which will be introduced in Section 2.4) from the deep learning literature. Locally linear embedding firstly finds the k nearest neighbors of $\boldsymbol{d}_t$ based on Euclidean distance, with the time indices of these neighbors being stored in $N(t)$, and obtained the weights $\hat{\boldsymbol{W}} = \text{argmin}_W \sum_{t=1}^T \left\|\boldsymbol{d}_t - \sum_{t'\in\mathcal{N}(t)} W_{tt'}\boldsymbol{d}_{t'}\right\|_2$, where $\hat{\boldsymbol{W}} \in \mathbb{R}^{T\times T}$, and the (t, t') entry is non-zero when $t' \in N(t)$, for $t, t' \in [T]$, $\|\cdot\|_2$ denotes the L2 norm. Then the factors were extracted by optimizing

$$\min_{\boldsymbol{F}} \|\boldsymbol{F} - \boldsymbol{W}\boldsymbol{F}\|_F^2, \text{ subject to } \boldsymbol{F}'\boldsymbol{F} = \boldsymbol{I}_K, \tag{2.25}$$

where $\boldsymbol{F} \in \mathbb{R}^{T\times K}$. The columns of the optimized $\boldsymbol{F}$ are the eigenvectors corresponding to the K largest eigenvalues of $(\boldsymbol{I}_T - \boldsymbol{W})'(\boldsymbol{I}_T - \boldsymbol{W})$. This dimension reduction method assumes that the linear relationship between one data point $\boldsymbol{d}_t$ and its neighbors is preserved in $\boldsymbol{f}_t$, for $t \in [T]$, which cannot be expressed as a linear transformation from $\boldsymbol{d}_t$ to $\boldsymbol{f}_t$. For the autoencoder, Klieber (2024) modeled neural networks to extract factors and reconstruct $\boldsymbol{d}_{1:T}$. As the model architecture and optimization procedures in this paper follow standard practices in the deep learning literature, a detailed discussion will be provided in Section 2.4. The author showed that the non-linear dimension reduction techniques yield tighter credible intervals of the IRFs before and during the pandemic period, compared to the result of the PCA.

2.3 Methodological Background about Tensor Decomposition

Section 2.1.4 and 2.2 discussed dimension reduction methods for analyzing multivariate time series, focusing on techniques applied either directly to the data or the model parameters, both of which can be represented as matrices. However, dimension reduction is not limited to matrices; it can be extended to tensors, leading to tensor decompositions.

Since Chapters 3 and 4 apply tensor decompositions to the VAR coefficients, this subsection aims to provide an overview of this dimension reduction technique. Section 2.3.1 introduces the concept of tensors and the associated operations which will be used in this thesis. Section 2.3.2 shows two widely used tensor decompositions: CANDECOMP/PARAFAC (CP) and Tucker decompositions. As the ranks corresponding to these decompositions are important components to be determined, Section 2.3.3 presents methods for the rank determination. Finally, applications of tensor decompositions to time series models are reviewed in Section 2.3.4.

2.3.1 Concept of Tensor and Related Operations

An array is a *J-th-order tensor* if it has J *modes* (or dimensions), with each mode having length $I_j \in \mathbb{N}$, for $j \in [J]$. Vectors and matrices correspond to the first- and second-order tensors, respectively, and the arrays with more than 2 modes are referred to as higher-order tensors (or multidimensional data in some research areas). This thesis will use "tensor" to refer to the higher-order tensor for brevity.

Many research fields collect or represent the data in the form of tensors, including neuroscience (Zhou et al., 2013), macroeconomics (Billio et al., 2023), finance (Han et al., 2022), computer vision (Liu et al., 2012), and recommendation systems (Bi et al., 2021), among others. Taking the data set applied in Chen et al. (2022) as an example, the international multi-category export data set was represented as a fourth-order tensor with dimension 22×22×15×84, of which the (i_1, i_2, i_3, i_4) entry being the export of products in category i_3 from country i_1 to country i_2 at time point i_4, for $i_1, i_2 \in [22]$, $i_3 \in [15]$ and $i_4 \in [84]$. While it is feasible to reshape tensors into vectors or matrices, then apply univariate or multivariate statistical models to analyze the data, directly modeling tensor data preserves the inherent structural information, enhancing the interpretability of the model results.

The primary technique for modeling tensors is tensor decomposition. To facilitate the discussion about tensor decompositions in this section and the subsequent chapters, basic notations and operations from the tensor literature are introduced below. To avoid abuse of notation, only those essential for understanding tensor decompositions will be introduced. See Kolda and Bader (2009) and Ji et al. (2019) for a more comprehensive overview of notations and operations about tensors.

In this section and hereafter, a J-th-order tensor is denoted by a curly capital letter, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times \cdots \times I_J}$, with the $(i_1, \ldots, i_J)$ entry being $\boldsymbol{\mathcal{X}}_{(i_1,\ldots,i_J)}$. The j-th mode in $\boldsymbol{\mathcal{X}}$ is referred to as the

j-mode. In addition to extracting scalar elements, there are mainly two ways to extract subarrays from a tensor: *fiber* and *slice*. The mode-j fiber of $\boldsymbol{\mathcal{X}}$, $\boldsymbol{\mathcal{X}}_{(i_1,\dots,i_{j-1},\cdot,i_{j+1},\dots,i_J)} \in \mathbb{R}^{I_j}$, is a vector obtained by fixing the indices of all modes except the j-th one. The slice is analogous to the fiber, but leaves two indices unfixed, producing a matrix rather than a vector. The *tensor norm* of $\boldsymbol{\mathcal{X}}$ is defined as $\|\boldsymbol{\mathcal{X}}\| = \sqrt{\sum_{i_1=1}^{I_1} \cdots \sum_{i_J=1}^{I_J} \left(\boldsymbol{\mathcal{X}}_{(i_1,\dots,i_J)}\right)^2}$. Similar to the diagonal matrix, there is *superdiagonal* tensor, $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{I\times\cdots\times I}$, of which the non-zero entries lie along the diagonal, i.e., the $(i,\dots,i)$ entry of the tensor, for $i \in [I]$.

As mentioned earlier, the tensor can be transformed to a matrix. This operation is referred to as *matricization*, and there are J possible matricizations to a J-th-order tensor. Although the matricization here is not to enable multivariate models to analyze the tensor, this operation is useful in tensor decompositions. The mode-j matricization of $\boldsymbol{\mathcal{X}}$ is denoted as $\boldsymbol{\mathcal{X}}_{(j)} \in \mathbb{R}^{I_j \times I_1\cdots I_{j-1} I_{j+1}\cdots I_J}$, where the i_j-th row corresponds to the vectorization of $\boldsymbol{\mathcal{X}}_{(\dots,i_j,\dots)}$. Specifically, the (i_j, k) element in $\boldsymbol{\mathcal{X}}_{(j)}$ corresponds to $\boldsymbol{\mathcal{X}}_{(i_1,\dots,i_j,\dots,i_J)}$ with $k = 1 + \sum\limits_{j'=1,\, j'\neq j}^{J} (i_{j'} - 1) \prod\limits_{m=1,\, m\neq j}^{j'-1} I_m$. Apart from the matricization, three tensor operations about multiplication are used in this thesis. The first one is the *outer product* of two tensors, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1\times\cdots\times I_J}$ and $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{K_1\times\cdots\times K_L}$,

$$\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \circ \boldsymbol{\mathcal{Y}}, \tag{2.26}$$

where $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{I_1\times\cdots\times I_J\times K_1\times\cdots\times K_L}$ with the $(i_1,\dots,i_J,k_1,\dots,k_L)$ entry being $\boldsymbol{\mathcal{X}}_{(i_1,\dots,i_J)}\boldsymbol{\mathcal{Y}}_{(k_1,\dots,k_L)}$, for $i_j \in [I_j]$, $k_l \in [K_l]$, $j \in [J]$ and $l \in [L]$. This outer product can also be applied to vectors and matrices. The second product operation is the *mode-j product* of a tensor $\boldsymbol{\mathcal{X}}$ (with the dimensionality defined above) and a matrix $\boldsymbol{Y} \in \mathcal{R}^{K\times I_j}$,

$$\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times_j \boldsymbol{Y},$$

where $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{I_1\times\cdots\times I_{j-1}\times K\times I_{j+1}\times\cdots\times I_J}$ has the $(i_1,\dots,i_{j-1},k,i_{j+1},\dots,i_J)$ entry as $\sum_{i_j=1}^{I_j} \boldsymbol{\mathcal{X}}_{(i_1,\dots,i_J)}\boldsymbol{Y}_{k,i_j}$, for $i_j \in [I_j]$, $k \in [K]$ and $j \in [J]$. The last operation is the *mode-J contracted product* of an $(L+J)$-th-order tensor, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{K_1\times\cdots\times K_L\times I_1\times\cdots\times I_J}$, and a $(J+M)$-th-order tensor, $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1\times\cdots\times I_J\times N_1\times\cdots\times N_M}$,

$$\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \bar{\times}_J \boldsymbol{\mathcal{Y}}$$

where $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{K_1\times\cdots\times K_L\times N_1\times\cdots\times N_M}$ has the $(k_1\dots,k_L,n_1,\dots,n_M)$ entry as $\sum_{i_1=1}^{I_1}\cdots\sum_{i_J=1}^{I_J}$ $\boldsymbol{\mathcal{X}}_{(k_1,\dots,k_L,i_1,\dots,i_J)}\boldsymbol{\mathcal{Y}}_{(i_1,\dots,i_J,n_1,\dots,n_M)}$.

2.3.2 Tensor Decompositions

CANDECOMP/PARAFAC (CP) Decomposition

The CP decomposition was first proposed by Hitchcock (1927) and popularized by Carroll and Chang (1970) and Harshman et al. (1970). While this decomposition technique has a long history, the term "CP decomposition" was later formalized by Kiers (2000). A rank-R CP decomposition of $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times \cdots \times I_J}$ is written as

$$\boldsymbol{\mathcal{X}} \approx \hat{\boldsymbol{\mathcal{X}}} = \sum_{r=1}^{R} \boldsymbol{\mathcal{X}}^{(r)} = \sum_{r=1}^{R} \boldsymbol{\beta}_1^{(r)} \circ \cdots \circ \boldsymbol{\beta}_J^{(r)}, \tag{2.27}$$

where $\hat{\boldsymbol{\mathcal{X}}}$ is an approximation of $\boldsymbol{\mathcal{X}}$, which is a sum of R tensors with the same dimensionality, $\boldsymbol{\mathcal{X}}^{(r)}$ for $r \in [R]$. $\boldsymbol{\mathcal{X}}^{(r)}$ can be further decomposed to J vectors, $\boldsymbol{\beta}_j^{(r)} \in \mathbb{R}^{I_j}$ for $j \in [J]$, of which the elements are referred to as *margins*. By stacking these vectors along each tensor mode, that is, setting $\boldsymbol{B}_j = (\boldsymbol{\beta}_j^{(1)}, \dots, \boldsymbol{\beta}_j^{(R)}) \in \mathbb{R}^{I_j \times R}$ for $j \in [J]$, one obtains matrices commonly referred to as *factor matrices* or *loadings*, with the latter terminology adopted in this thesis. The CP decomposition in (2.27) can be represented as $\boldsymbol{\mathcal{X}} \approx [\![\boldsymbol{B}_1, \dots, \boldsymbol{B}_J]\!]_{\text{CP}}$.

It is important to connect and distinguish between the rank R used in the CP decomposition and the concept of CP rank defined by Hitchcock (1927). The CP rank is the smallest value of R such that the CP decomposition is *exact*, meaning that the approximation in (2.27) becomes an equality. Unlike determining the matrix rank, finding the CP rank is NP-hard[20] (Håstad, 1989), so the CP decomposition practically only ever approximates a tensor with the rank R, which need not equal the CP rank. However, in the statistical literature on tensors, these two concepts are often used interchangeably due to the introduction of the error term (see (3.1) in conjunction with (3.2) for an example).

Sidiropoulos and Bro (2000) showed that the CP decomposition is identifiable up to scaling and permutation. In particular, $\hat{\boldsymbol{\mathcal{X}}} = [\![\boldsymbol{B}_1, \dots, \boldsymbol{B}_J]\!]_{\text{CP}} = [\![\tilde{\boldsymbol{B}}_1, \dots, \tilde{\boldsymbol{B}}_J]\!]_{\text{CP}}$, if $\tilde{\boldsymbol{B}}_j$ (for $j \in [J]$) follows the transformations below.

1. Scaling: $\tilde{\boldsymbol{B}}_j = \boldsymbol{B}_j \boldsymbol{Q}_j$, and $\boldsymbol{Q}_j$ is an R-by-R diagonal matrix satisfying $\prod_{j=1}^{J} \boldsymbol{Q}_{j,(r,r)} = 1$, for $r \in [R]$.
2. Permutation: $\tilde{\boldsymbol{B}}_j = \boldsymbol{B}_j \boldsymbol{\Pi}$ for an arbitrary R-by-R column-wise permutation matrix $\boldsymbol{\Pi}$.

[20] Informally, a problem is NP-hard if it is at least as hard as the hardest problem in the nondeterministic-polynomial (NP) class, in which the solutions of these NP problems can be verified using a deterministic machine (e.g. computer) in polynomial time. The hardness is in the sense that the solution of an NP-hard problem can be used to solve any NP problem.

When interpreting the CP decomposition, it is necessary to determine the margin scale and the order of the loading columns. Faber et al. (2003) addressed the indeterminacy *ex post* by scaling the margins so that the first $J-1$ loadings have unit vectors, and ordered the columns according to the L2 norm of the columns in the last loading. Zhou et al. (2013) imposed constraints on the loadings before decomposing the tensor. Specifically, the margins in the first rows of the first $J-1$ loadings are equal to one, and those margins in the first row of the last loading are in descending order.

Determining the CP decomposition of a tensor is equivalent to optimizing $\operatorname{argmin}_{\hat{\boldsymbol{\mathcal{X}}}} \|\boldsymbol{\mathcal{X}} - \hat{\boldsymbol{\mathcal{X}}}\|$, subject to $\hat{\boldsymbol{\mathcal{X}}}$ defined in (2.27). The most popular algorithm of this optimization problem is the alternating least square (ALS) (Carroll and Chang, 1970; Harshman et al., 1970), which updates the loadings iteratively. Algorithm 1 presents the ALS, where $\odot$ in line 4 means Khatri–Rao product such that $\boldsymbol{B}_j \odot \boldsymbol{B}_{j'} = \left[\boldsymbol{\beta}_j^{(1)} \otimes \boldsymbol{\beta}_{j'}^{(1)}, \cdots, \boldsymbol{\beta}_j^{(R)} \otimes \boldsymbol{\beta}_{j'}^{(R)}\right]$, and the superscipt "+" in line 5 means the Moore–Penrose inverse. The update in line 5 is because the CP decomposition can be rewritten as $\boldsymbol{\mathcal{X}}_{(j)} \approx \boldsymbol{B}_j \boldsymbol{Z}^{(j)}$.

Algorithm 1 Alternating least square

```
1: **Initialize** $B_j \in \mathbb{R}^{I_j \times R}$ for $j \in [J]$
2: **repeat**
3:     **for** $j = 1, \ldots, J$ **do**
4:         $\boldsymbol{Z}^{(j)} \leftarrow (\boldsymbol{B}_J \odot \cdots \odot \boldsymbol{B}_{j+1} \odot \boldsymbol{B}_{j-1} \odot \cdots \odot \boldsymbol{B}_1)'$
5:         $\boldsymbol{B}_j \leftarrow \boldsymbol{\mathcal{X}}_{(j)}(\boldsymbol{Z}^{(j)})^+$
6:     **end for**
7: **until** fit ceases to improve or maximum iteration exhausted.
8: **Return** $\{\boldsymbol{B}_j : j \in [J]\}$
```

Tucker Decomposition

The Tucker decomposition was first introduced by Tucker (1963). Instead of having only one rank in the CP decomposition, the Tucker decomposition has the following expression with J ranks

$$\begin{aligned} \boldsymbol{\mathcal{X}} \approx \hat{\boldsymbol{\mathcal{X}}} &= \sum_{r_1=1}^{R_1} \cdots \sum_{r_J=1}^{R_J} \boldsymbol{\mathcal{G}}_{(r_1,\ldots,r_J)} \boldsymbol{\beta}_1^{(r_1)} \circ \cdots \circ \boldsymbol{\beta}_J^{(r_J)}, \\ &= \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{B}_1 \times_2 \cdots \times_J \boldsymbol{B}_J, \end{aligned} \tag{2.28}$$

where $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{R_1 \times \cdots \times R_J}$ is the *core tensor*. Let $\boldsymbol{B}_j = (\boldsymbol{\beta}_j^{(1)}, ..., \boldsymbol{\beta}_j^{(R_j)}) \in \mathbb{R}^{I_j \times R_j}$, for $j \in [J]$, the Tucker decomposition is written as $\boldsymbol{\mathcal{X}} \approx [\![\boldsymbol{\mathcal{G}}, \boldsymbol{B}_1, \ldots, \boldsymbol{B}_J]\!]_{\text{Tucker}}$ for brevity. Note that the CP decomposition can be represented as a Tucker decomposition with the core tensor being superdiagonal and $R_1 = \cdots = R_J = R$. The exact Tucker decomposition exists when $R_j = \text{rank}(\boldsymbol{\mathcal{X}}_{(j)})$, for all $j \in [J]$. Although $\text{rank}(\boldsymbol{\mathcal{X}}_{(j)})$ is much easier to obtain compared to the CP rank, these ranks are often large when $\boldsymbol{\mathcal{X}}$ is noisy, hindering effective dimension reduction. Thus, R_j ($j \in [J]$) are typically set to be smaller, which leads to the approximation in (2.28).

The Tucker decomposition is invariant to rotations. Specifically, $\boldsymbol{\mathcal{X}} \approx [\![\boldsymbol{\mathcal{G}}, \boldsymbol{B}_1, \ldots, \boldsymbol{B}_J]\!]_{\text{Tucker}} = [\![\tilde{\boldsymbol{\mathcal{G}}}, \boldsymbol{B}_1\boldsymbol{Q}_1, \ldots, \boldsymbol{B}_J\boldsymbol{Q}_J]\!]_{\text{Tucker}}$, where $\boldsymbol{Q}_j \in \mathbb{R}^{R_j \times R_j}$ is an arbitrary invertible matrix, for $j \in [J]$, and $\tilde{\boldsymbol{\mathcal{G}}} = \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{Q}_1^{-1} \times_2 \cdots \times_J \boldsymbol{Q}_J^{-1}$. This indeterminacy poses a challenge to the interpretation of the Tucker decomposition, which can be overcome by orthogonally rotating $\boldsymbol{B}_j$ according to some criteria. Let $\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_J$ defined above denote the rotation matrices and assume they are orthogonal, Kiers (1997) optimized these orthogonal matrices by maximizing the orthomax criterion, $\sum_{j=1}^J w_j \text{IC}_{\text{OR}}(\tilde{\boldsymbol{\mathcal{G}}}_{(j)}, \gamma_j)$, where $w_j = (R_1 \cdots R_{j-1}R_{j+1} \cdots R_J)^{-1}$, and $\text{IC}_{\text{OR}}(\tilde{\boldsymbol{\mathcal{G}}}_{(j)}, \gamma_j)$ quantifies the efficiency of information stored in the mode-j matricization of $\tilde{\boldsymbol{\mathcal{G}}}$,

$$\text{IC}_{\text{OR}}(\tilde{\boldsymbol{\mathcal{G}}}_{(j)}, \gamma_j) = \sum_{l=1}^{w_j^{-1}} \left[\sum_{i=1}^{R_j} \left(\tilde{\boldsymbol{\mathcal{G}}}_{(j),(i,l)} \right)^4 - \frac{\gamma_j}{R_j} \left(\sum_{i=1}^{R_j} \tilde{\boldsymbol{\mathcal{G}}}_{(j),(i,l)} \right)^2 \right],$$

where $\gamma_j \in [0, 1]$ controls the specific orthogonal rotation applied to each loading. If $\gamma_j = 0$ or 1, the rotation corresponds to quartimax (Neuhaus and Wrigley, 1954) or varimax (Kaiser, 1958). Kiers (1998) extended the criterion to a more flexible one by considering both the core tensor and loadings, $\sum_{j=1}^J w_j \text{IC}_{\text{OR}}(\tilde{\boldsymbol{\mathcal{G}}}_{(j)}, \gamma_j) + w_j^* \text{IC}_{\text{OR}}(\tilde{\boldsymbol{B}}_{(j)}, \gamma_j^*)$, where w_j^* are the weights corresponding to the j-th loading, $\tilde{\boldsymbol{B}}_j = \boldsymbol{B}_j\boldsymbol{Q}_j$, and $\gamma_j^* \in [0, 1]$. More recently, Chen et al. (2022) directly applied varimax rotations to the loadings. An alternative method to address the indeterminacy issue is through the higher-order singular value decomposition (HOSVD) (De Lathauwer et al., 2000a), which is an algorithm to compute the Tucker decomposition deterministically and will be introduced later. Since the decomposition is fully deterministic, the core tensor and loadings are identifiable.

Two algorithms are widely applied to compute the Tucker decomposition given fixed $R_1, \ldots, R_J$. The first algorithm is the HOSVD as shown in Algorithm 2. When $R_j = \text{rank}(\boldsymbol{\mathcal{X}}_{(j)})$, for $j \in [J]$, the HOSVD yields an exact decomposition. However, when R_j is set to a lower value (as is typically the case), the HOSVD can only optimize $\boldsymbol{B}_j$ in the sense that it maximizes the variance explained in a matricization of $\boldsymbol{\mathcal{X}}$, without accounting for the interaction with other loadings and the core tensor. This motivated the higher-order orthogonal iteration (HOOI) (De Lathauwer et al., 2000b), which is analogous to the ALS as it optimizes $\operatorname{argmin}_{\hat{\boldsymbol{\mathcal{X}}}} \|\boldsymbol{\mathcal{X}} - \hat{\boldsymbol{\mathcal{X}}}\|$ iteratively, as presented in Algorithm 3.

Algorithm 2 Higher-order singular value decomposition

```
for $j = 1, \ldots, J$ do
    $\boldsymbol{B}_j \leftarrow R_j$ leading left singular vector of $\boldsymbol{\mathcal{X}}_{(j)}$
end for
$\boldsymbol{\mathcal{G}} \leftarrow \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{B}_1' \times \cdots \times_J \boldsymbol{B}_J'$
Return $\{\boldsymbol{\mathcal{G}}, \boldsymbol{B}_j : j \in [J]\}$
```

Algorithm 3 High-order orthogonal iteration

```
Initialize $B_j \in \mathbb{R}^{I_j \times R}$ for $j \in [J]$
repeat
    for $j = 1, \ldots, J$ do
        $\boldsymbol{\mathcal{Y}} \leftarrow \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{B}_1' \times_2 \cdots \times_{j-1} \boldsymbol{B}_{j-1}' \times_{j+1} \boldsymbol{B}_{j+1}' \times_{j+2} \cdots \times_J \boldsymbol{B}_J'$
        $\boldsymbol{B}_j \leftarrow R_j$ leading singular vectors of $\boldsymbol{\mathcal{Y}}_{(j)}$
    end for
until fit ceases to improve or maximum iteration exhausted.
$\boldsymbol{\mathcal{G}} \leftarrow \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{B}_1' \times_2 \cdots \times_J \boldsymbol{B}_J'$
Return $\{\boldsymbol{\mathcal{G}}, \boldsymbol{B}_j : j \in [J]\}$
```

2.3.3 Rank Determination of Tensor Decompositions

Rank determination is a crucial step in tensor decompositions, with various methods proposed to address this challenge, primarily falling into two categories: performance-based and model-based approaches. The performance-based approaches determine the rank(s) by evaluating the tensor decompositions with different metrics. Bro and Kiers (2003) introduced an evaluation method for the CP decomposition by exploiting its relationship with the Tucker decomposition. Given a rank, the CP decomposition obtained through the ALS corresponds to a Tucker decomposition with the CP loadings and a superdiagonal core tensor, of which the non-zero entries are ones. The authors compared this core tensor with $\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{B}_1' \times_2 \cdots \times_J \boldsymbol{B}_J'$, the Tucker core tensor obtained via HOSVD or HOOI using the CP loadings, and selected the rank that minimized the difference between the two core tensors. Timmerman and Kiers (2000) defined the variance explained by the Tucker decomposition as $\text{ExpVar}_s = 1 - \frac{\|\boldsymbol{\mathcal{X}} - \hat{\boldsymbol{\mathcal{X}}}_s\|}{\|\boldsymbol{\mathcal{X}}\|}$, where $s = R_1 + \cdots + R_j$, and $\hat{\boldsymbol{\mathcal{X}}}_s$ denotes the fitted tensor that maximizes the variance explained among all fitted tensors whose sum of ranks equals s. The authors then selected the value of s and the corresponding ranks, which exhibited the highest fitting improvement ratio, defined as $\frac{\text{ExpVar}_s - \text{ExpVar}_{s-1}}{\text{ExpVar}_{s+1} - \text{ExpVar}_s}$. While this method requires repeatedly fitting the Tucker decomposition using the HOOI, Kiers and Der Kinderen (2003) proposed a faster alternative that determines the rank with a single fit via the HOSVD. Ceulemans and Kiers (2006) adopted a similar procedure and incorporated the number of margins into this improvement ratio.

Information criteria are also popular metrics to determine the ranks in the performance-based approaches. The likelihood part of the information criteria is typically

$$\prod_{i_1=1}^{I_1} \cdots \prod_{i_J=1}^{I_J} \phi(\boldsymbol{\mathcal{X}}_{(i_1,\ldots,i_J)}; \hat{\boldsymbol{\mathcal{X}}}_{(i_1,\ldots,i_J)}, \sigma^2), \tag{2.29}$$

where $\phi(\cdot)$ is the PDF of a normal distribution and σ^2 is the variance of each tensor entry, and the penalty part depends on the specific criterion applied. Zhou et al. (2013), Li and Zhang (2017), Sun and Li (2017) and Li et al. (2018) selected the rank(s) with the lowest BIC. Guhaniyogi and Spencer (2021) and Spencer et al. (2022) used Deviance Information Criterion (DIC) (Spiegelhalter et al., 2002). Han et al. (2022) followed Bai and Ng (2002) to propose a class of information criteria for decomposing tensor time series with the Tucker decomposition.

The model-based approaches consider penalty terms or shrinkage priors when decomposing a tensor. For the former, a penalty term modifies the loss function as

$$\|\boldsymbol{\mathcal{X}} - \hat{\boldsymbol{\mathcal{X}}}\| + \lambda R(\boldsymbol{\theta}),$$

where λ is the tuning parameter, $R(\cdot)$ is the penalty term, which regularizes components $\boldsymbol{\theta} = \{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_J\}$ or $\boldsymbol{\theta} = \{\boldsymbol{\mathcal{G}}, \boldsymbol{B}_1, \ldots, \boldsymbol{B}_J\}$ for the CP or Tucker decomposition, respectively. The dimensions of the core tensor and loadings are initially set according to the maximum candidate rank(s), and the rank(s) will then be determined by eliminating those negligible margins. Bazerque et al. (2013) constructed the penalty term as the sum of the Frobenius norm of loadings in the CP decomposition. Shi et al. (2017) and Karim et al. (2020) applied the lasso penalty to regularize the core tensors.

Imposing shrinkage priors on loadings or the core tensor is also prominent in rank determination. This approach aims to compute the posterior of $\boldsymbol{\theta}$ (as defined above) given the observed tensor, with the likelihood defined in (2.29). Unlike the three algorithms discussed, this Bayesian framework employs MCMC or variational inference to sample $\boldsymbol{\theta}$. Mørup and Hansen (2009) imposed the scaled mixture of normal distributions, including Student-t and Laplace distributions, to margins in both the CP and Tucker decompositions, and determined the rank(s) by eliminating the margins with small magnitudes. This procedure has been extended to tensor completion, i.e., imputing missing values in a tensor, in Zhao et al. (2015a) and Zhao et al. (2015b). Guhaniyogi et al. (2017) proposed the multiway Dirichlet generalized double Pareto (M-DGDP) prior, which will be discussed in Chapter 3, on the loadings of the CP decomposition. The Dirichlet distribution in the M-DGDP governs the shrinkage levels of margins in a loading. Similar to the Dirichlet-Laplace prior discussed in Section 2.1.4, when the parameters of the Dirichlet distribution are small, most margins are shrunk toward zero, with only a small subset retained as important. The M-DGDP has been applied in Billio et al. (2023) and Zhang et al. (2021). Another class of priors applied to the margins is the increasing shrinkage prior, where the shrinkage levels of the margins or core tensor elements grow with the corresponding rank(s). The multiplicative gamma prior (MGP) (Bhattacharya and Dunson, 2011) is particularly popular within this class due to its increasing shrinkage property and adaptive shrinkage mechanism (see Chapter 3 for a detailed illustration about the MGP), which allows rank determination during the inference. The earliest application of the MGP to tensor decomposition can be found in Rai et al. (2014), where the core tensor corresponding to the CP decomposition followed the MGP. In contrast, Chapter 3 and Fan et al. (2022) imposed the MGP on the loadings of the CP decomposition. Takayama et al. (2022) compared the MGP with the scale mixture of normal distributions and found that the MGP determined the rank more accurately. An alternative increasing shrinkage prior is the multiway stick-breaking shrinkage prior proposed by Guhaniyogi and Spencer (2021), which replaced the Dirichlet prior in the M-DGDP with a stick-breaking process (Sethuraman, 1994).

2.3.4 Applications of Tensor Decompositions

Tensor decompositions were initially applied in fields such as psychometrics, chemometrics, and neuroimaging following their introduction in the 20th century (Kolda and Bader, 2009). Entering the 21st century, they have attracted increasing interest in statistics and machine learning, driven by the growing demand for analyzing data collected across multiple dimensions. Many fundamental statistical and machine learning models, such as linear regression and PCA, have been extended to incorporate tensor decompositions, see Ji et al. (2019) and Bi et al. (2021) for recent reviews. In deep learning, tensors are not only used as input structures, but tensor decompositions are also employed to compress and optimize the storage of trained model parameters (Bacciu and Mandic, 2020). The remainder of this subsection provides a review of tensor decomposition methods applied to time series models, in which applications can be broadly classified into tensor decompositions in the data space and the parameter space.

Applications of tensor decompositions to the data space can be viewed as extensions of factor models, such as the one defined in (2.12). The key difference is that tensor decompositions are applied to matrix- or tensor-valued time series, rather than to vector-valued multivariate time series traditionally studied in the factor models. For the matrix one, Chang et al. (2023) treated $\boldsymbol{Y}_{1:T} \in \mathbb{R}^{I_1 \times I_2 \times T}$ as a third-order tensor with time as one of the modes, and modeled this data using a CP decomposition. This factor model can be written as

$$\boldsymbol{Y}_t = \boldsymbol{B}_1 \left(\boldsymbol{\beta}_{3,t}' \odot \boldsymbol{B}_2 \right)' + \boldsymbol{E}_t,$$

where $\boldsymbol{\beta}_{3,t} \in \mathbb{R}^R$ represents the factors of $\boldsymbol{Y}_t$, $\boldsymbol{E}_t \in \mathbb{R}^{I_1 \times I_2}$. Since the third loading $\boldsymbol{B}_3 = \left(\boldsymbol{\beta}_3^{(1)}, \dots, \boldsymbol{\beta}_3^{(R)} \right) \in \mathbb{R}^{T \times R}$ is the only loading related to time, the authors constructed R AR(1) processes to forecast $\boldsymbol{Y}_t$. When the time series are in the tensor form, i.e., $\boldsymbol{\mathcal{Y}}_t \in \mathbb{R}^{I_1 \times \cdots \times I_J}$, for $t \in [T]$, Chen et al. (2022) applied the Tucker decomposition with a time-varying core tensor,

$$\boldsymbol{\mathcal{Y}}_t = \boldsymbol{\mathcal{G}}_t \times_1 \boldsymbol{B}_1 \times_2 \cdots \times_J \boldsymbol{B}_J + \boldsymbol{\mathcal{E}}_t,$$

where $\boldsymbol{\mathcal{E}}_t$ is the error tensor. This model has been extended to the CP decomposition by Han et al. (2024) and applied to missing value imputation by Cen and Lam (2025). Chen (2024) provided a review of this tensor factor model, with the recent development focusing on the theoretical work under different conditions of $\boldsymbol{\mathcal{E}}_t$. Babii et al. (2022) followed Chang et al. (2023) to construct time as one mode and decomposed $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times \cdots \times I_J \times T}$ using a Tucker decomposition, so the last loading was regarded as factors, instead of the core tensor.

Tensor decompositions have been applied to decompose the parameters in multivariate, matrix, and tensor time series models. Within the VAR framework considered in Section 2.1, Wang et al. (2022a) introduced tensor VAR (TVAR) by transforming the coefficient matrix $\boldsymbol{A} \in \mathbb{R}^{N \times NP}$ to a third-order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{N \times N \times P}$, and employed the Tucker decomposition to solve the over-parameterization issue. This application of tensor decompositions motivated the research presented in Chapter 3 and 4, so more details will be provided in these two chapters. Zhang et al. (2021), Fan et al. (2022) and Chan and Qi (2024) also proposed variants of TVARs, specifically tailored for time-varying coefficients, multi-subject data, and stochastic volatilities, respectively. Huang et al. (2024) and Harris et al. (2021) extended this technique to model the parameters in the VARMA and the VAR(1) with a sliding window structure. For the matrix time series, Hecq et al. (2024) extended the matrix autoregressive model (Chen et al., 2021) by constructing this model as

$$\boldsymbol{Y}_t = \boldsymbol{\mathcal{A}}\bar{\times}_2\boldsymbol{Y}_{t-1} + \boldsymbol{E}_t,$$

where $\boldsymbol{Y}_t, \boldsymbol{E}_t \in \mathbb{R}^{N\times M}$, $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{N\times M\times N\times M}$, then reduced the number of parameters in $\boldsymbol{\mathcal{A}}$ through the Tucker decomposition. To model tensor time series, Billio et al. (2023) and Wang et al. (2024a) proposed tensor autoregressive models with Bayesian and frequentist estimation methods, respectively. The mathematical expression of the tensor autoregression is

$$\boldsymbol{\mathcal{Y}}_t = \sum_{p=1}^{P} \boldsymbol{\mathcal{A}}_p \bar{\times}_J \boldsymbol{\mathcal{Y}}_{t-p} + \boldsymbol{\mathcal{E}}_t,$$

where $\boldsymbol{\mathcal{Y}}_t, \boldsymbol{\mathcal{E}}_t \in \mathbb{R}^{I_1\times\cdots\times I_J}$, $\boldsymbol{\mathcal{A}}_p \in \mathbb{R}^{I_1\times\cdots\times I_J\times I_1\times\cdots\times I_J}$, for $p \in [P]$. The number of parameters can be reduced by applying tensor decompositions to $\boldsymbol{\mathcal{A}}_p$, leading to more efficient estimation. An alternative tensor autoregression was proposed by Li and Xiao (2021), who adopted the Tucker decomposition with the lagged tensor time series as the core tensor,

$$\boldsymbol{\mathcal{Y}}_t = \boldsymbol{\mathcal{Y}}_{t-1} \times_1 \boldsymbol{A}_1 \times \cdots \times_J \boldsymbol{A}_J,$$

where $\boldsymbol{A}_j \in \mathbb{R}^{I_j\times I_j}$ represents the coefficients, for $j \in [J]$.

2.4 Methodological Background of Deep Learning

Over the past few decades, deep learning has evolved into a thriving field within machine learning, achieving remarkable success across a wide range of applications. Originating from the early work on artificial neural networks in the 1940s (McCulloch and Pitts, 1943), deep learning made relatively limited progress in tackling complex tasks until the mid-2000s, when a combination of increased computational power, the availability of high-quality data, and algorithmic innovations enabled a new wave of developments. Since then, deep learning has demonstrated exceptional performance across diverse fields, from computer vision (Voulodimos et al., 2018) and natural language processing (Otter et al., 2020) to robotics (Soori et al., 2023) and biology (Jumper et al., 2021), to name a few.

Motivated by the popularity of deep learning, this thesis applies one of the deep learning models, autoencoder, to the FAVAR framework in Chapter 5. To support this application, this subsection provides the methodological background of deep learning. Section 2.4.1 introduces the multilayer perceptron, which forms the foundation of most deep learning models; the training procedure is also described based on this neural network. Section 2.4.2 then presents an overview of the standard autoencoder. Throughout this subsection, the index $t \in [T]$ is used to denote a data point to maintain notational coherence with the rest of the thesis, though t does not necessarily represent a time point in this context.

2.4.1 Multilayer Perceptron

The multilayer perceptron (MLP) (Rosenblatt, 1958) is a fundamental architecture in deep learning. Mathematically, the MLP models a target $\boldsymbol{y}_t$ based on some features $\boldsymbol{x}_t$ as

$$\hat{\boldsymbol{y}}_t = (g_L \circ \cdots \circ g_1)_{\boldsymbol{\theta}}(\boldsymbol{x}_t), \tag{2.30}$$

where $\hat{\boldsymbol{y}}_t \in \mathbb{R}^N$, $\boldsymbol{x}_t \in \mathbb{R}^M$, L denotes the number of layers in the MLP, $g_l(\cdot)$ (for $l \in [L]$) represents a function, $\circ$ gives function composition (not the outer product defined in Section 2.3.1), $\boldsymbol{\theta}$ stores all the parameters of the MLP. This architecture illustrates the essence of deep learning: the neural network and non-linearity. Suppose each element in $\boldsymbol{x}_t$, $\hat{\boldsymbol{y}}_t$, and the outputs of $g_l(\cdot)$, $\boldsymbol{h}_{t,l}$ (for $l \in [L]$), is a *node* (or *neuron*), Figure 2.1 presents the graphical model of (2.30) as a network, namely *feedforward neural network*, in which the nodes in each layer are connected to those in the subsequent layer. While $g_l(\cdot)$, which connects $\boldsymbol{h}_{t,l-1}$ and $\boldsymbol{h}_{t,l}$, could be linear functions (for $l \in [L]$), the Universal Approximation Theorem (Cybenko, 1989) and its extensions (see Pinkus (1999) for a survey) showed that the MLP can approximate any continuous function with arbitrary precision. To accommodate non-linearity, $g_l(\cdot)$ is defined as

$$\boldsymbol{h}_{t,l} = g_l(\boldsymbol{h}_{t,l-1}) = \sigma_l(\boldsymbol{W}_l \boldsymbol{h}_{t,l-1} + \boldsymbol{b}_l), \tag{2.31}$$

where $\boldsymbol{h}_{t,l} \in \mathbb{R}^{D_l}$ denotes the l-th hidden layer, for $l \in [L-1]$, $\boldsymbol{h}_{t,0}$ and $\boldsymbol{h}_{t,L}$ are the input ($\boldsymbol{x}_t$) and output ($\hat{\boldsymbol{y}}_t$) layers, respectively, the parameters $\boldsymbol{W}_l \in \mathbb{R}^{D_l \times D_{l-1}}$ and $\boldsymbol{b}_l \in \mathbb{R}^{D_l}$ are referred to as *weights* and *bias*. $\sigma_l(\cdot)$ is an *activation function*, which could be an identity or a non-linear function depending on the specific application. When $l < L$, $\sigma_l(\cdot)$ is typically a non-linear function, such as the most common ones in Table 2.2. For $l = L$, $\sigma_L(\cdot)$ is often chosen as an identity function as $\boldsymbol{y}_t$ is real-valued. For those cases when $\boldsymbol{y}_t$ has some restrictions, e.g., lying on a simplex, $\sigma_L(\cdot)$ can also be non-linear.

The loss function of the MLP, $\mathcal{L}(\boldsymbol{y}_{1:T}, \hat{\boldsymbol{y}}_{1:T})$, evaluates the model performance and is used during training. The specification of this loss function is related to the likelihood of $\boldsymbol{y}_{1:T}$ as-

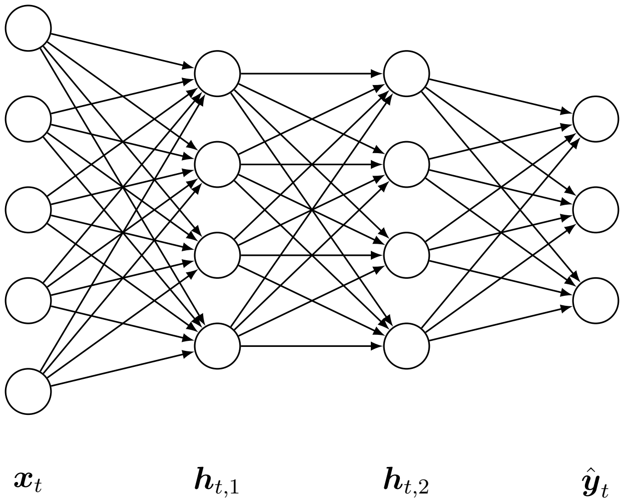


Figure 2.1: Graphical model of an MLP with 3 layers (2 hidden layers).

Activation Function	$\sigma_l(x)$
ReLU	$\max(0, x)$
Tanh	$\frac{e^x - e^{-x}}{e^x + e^{-x}}$
Sigmoid	$\frac{1}{1+e^{-x}}$
Softplus	$\log(1 + e^x)$

Table 2.2: Commonly applied activation functions in deep learning.

sumed. If $\boldsymbol{y}_t$ follows a Gaussian distribution with $\hat{\boldsymbol{y}}_t$ being the mean, the loss function is specified as the mean squared error. When a Laplace distribution is considered instead, the loss function becomes the mean absolute error. If the purpose of the MLP is classification, the cross-entropy loss is applied[21].

The training procedure of the MLP computes the gradient of the loss function with respect to $\boldsymbol{\theta}$ and updates the parameters using an optimization algorithm to minimize the loss function. The method to compute this gradient is the *backpropagation* (Rumelhart et al., 1986), which recursively applies the chain rule from the output layer back to the input layer. The most classical optimization algorithm is the gradient descent (Cauchy et al., 1847), which updates the parameters with

$$\boldsymbol{\theta}^{(j)} \leftarrow \boldsymbol{\theta}^{(j-1)} - \alpha \nabla_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(j-1)}} \mathcal{L}(\boldsymbol{y}_{1:T}, \hat{\boldsymbol{y}}_{1:T}),$$

where j is the index of the update, with the maximum value referred to as *epoch*, α is the learn-

[21] In this case, $\hat{\boldsymbol{y}}_t$ lies on a simplex to represent probabilities, and the cross-entropy is defined as $-\sum_{t=1}^{T} \boldsymbol{y}_t \log \hat{\boldsymbol{y}}_t$.

ing rate with the default value of 0.001. While gradient descent updates the parameters using the entire training dataset, referred to as a *batch*, mini-batch gradient descent accelerates training by randomly selecting a *minibatch*, i.e., a subset of the data, at each iteration and updating the parameters based on this subset. When the minibatch has only one data point, this optimization algorithm is referred to as stochastic gradient descent (SGD) (Kiefer and Wolfowitz, 1952). Although SGD enables efficient training, its updates are prone to fluctuations because each update is based on a single data point. For the same reason, the algorithm may also get stuck in local minima. This motivated momentum methods (Polyak, 1964), which smooth the optimization path and help the updates escape local minima by adjusting the gradients dynamically,

$$\begin{aligned} \boldsymbol{v}^{(j)} &\leftarrow \beta_1 \boldsymbol{v}^{(j-1)} + (1-\beta_1)\nabla_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(j-1)}}\mathcal{L}(\boldsymbol{y}_{\mathcal{B}^{(j)}}, \hat{\boldsymbol{y}}_{\mathcal{B}^{(j)}}), \\ \boldsymbol{\theta}^{(j)} &\leftarrow \boldsymbol{\theta}^{(j-1)} - \alpha \boldsymbol{v}^{(j)}, \end{aligned} \tag{2.32}$$

where $\boldsymbol{v}^{(j)}$ denotes the *velocity*, which is an exponentially weighted moving average of past gradients, β_1 is the decay rate, which specifies how fast the elements in $\boldsymbol{v}^{(j)}$ decay to 0, $\mathcal{B}^{(j)}$ is the minibatch at the j-th iteration. An alternative adaptive method is the root mean square propagation (RMSprop) (Tieleman and Hinton, 2012), which introduces an adaptive learning rate using the following expression,

$$\boldsymbol{s}^{(j)} \leftarrow \beta_2 \boldsymbol{s}^{(j-1)} + (1-\beta_2)\left(\nabla_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(j-1)}}\mathcal{L}(\boldsymbol{y}_{\mathcal{B}^{(j)}}, \hat{\boldsymbol{y}}_{\mathcal{B}^{(j)}})\right)^{\odot 2}, \tag{2.33}$$

$$\boldsymbol{\theta}^{(j)} \leftarrow \boldsymbol{\theta}^{(j-1)} - \alpha \frac{\nabla_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(j-1)}}\mathcal{L}(\boldsymbol{y}_{\mathcal{B}^{(j)}}, \hat{\boldsymbol{y}}_{\mathcal{B}^{(j)}})}{\sqrt{\boldsymbol{s}^{(j)}+c}}, \tag{2.34}$$

where $\boldsymbol{s}^{(j)}$ accumulates the exponentially weighted moving average of squared gradients, β_2 is the corresponding decay rate, $(\cdot)^{\odot 2}$ denotes element-wise square, c is a small constant to ensure numerical stability. Adaptive Moment Estimation (Adam) (Kingma, 2014) combines the advantages of the momentum method and RMSprop by allowing both adaptive gradients and gradient variances. Specifically, Adam uses both (2.32) and (2.33), followed by a bias correction step, $\hat{\boldsymbol{v}}^{(j)} = (1-\beta_1)^{-1}\boldsymbol{v}^{(j)}$ and $\hat{\boldsymbol{s}}^{(j)} = (1-\beta_2)^{-1}\boldsymbol{s}^{(j)}$, then updates $\boldsymbol{\theta}$ with

$$\boldsymbol{\theta}^{(j)} \leftarrow \boldsymbol{\theta}^{(j-1)} - \alpha \frac{\hat{\boldsymbol{v}}^{(j)}}{\sqrt{\hat{\boldsymbol{s}}^{(j)}+c}}.$$

2.4.2 Autoencoder

The autoencoder (Hinton and Salakhutdinov, 2006; LeCun, 1987) is a deep learning model that compresses the high-dimensional data to a lower dimension. A standard autoencoder comprises three parts: encoder, factors (often referred to as *bottleneck* or *code* in the deep learning literature), and decoder, with the goal of getting factors from the encoder and reconstructing the high-dimensional data using the decoder. The mathematical expression of a standard symmetric autoencoder is

$$\boldsymbol{f}_t = g^e_{\boldsymbol{\phi}}\left(\boldsymbol{x}_t\right) = \left(g^e_L \circ \cdots \circ g^e_1\right)_{\boldsymbol{\phi}}\left(\boldsymbol{x}_t\right), \tag{2.35}$$

$$\hat{\boldsymbol{x}}_t = g^d_{\boldsymbol{\theta}}\left(\boldsymbol{f}_t\right) = \left(g^d_L \circ \cdots \circ g^d_1\right)_{\boldsymbol{\theta}}\left(\boldsymbol{f}_t\right). \tag{2.36}$$

where $g^e_{\boldsymbol{\phi}}(\cdot)$ and $g^d_{\boldsymbol{\theta}}(\cdot)$ are the encoder and decoder with parameters $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$, respectively, each representing an MLP with L layers, following the architecture of a symmetric autoencoder[22]. By constructing $g^e_{\boldsymbol{\phi}}(\cdot)$ and $g^d_{\boldsymbol{\theta}}(\cdot)$ as linear functions and specifying the loss function as the mean squared error between $\boldsymbol{x}_t$ and $\hat{\boldsymbol{x}}_t$ (for t in a minibatch), the autoencoder is equivalent to the PCA as proven by Baldi and Hornik (1989). When the MLP incorporates non-linear activation functions, the factors are the non-linear low-dimensional representations, which are potentially more expressive than the principal components. For example, Wang et al. (2016) showed that autoencoders are more effective than the PCA and other non-linear dimension reduction methods, such as the locally linear embedding discussed in Section 2.2.5, at capturing repetitive structures. While linear factor models, such as the one defined in (2.12) and (2.22), only require a single equation to express the connection between the factors and the data, the autoencoder generally needs both encoder and decoder because it is nontrivial to optimize factors and the parameters in a non-linear autoencoder simultaneously. Combining the encoder and decoder enables an effective training procedure, as introduced in Section 2.4.1.

Multiple variants of the standard autoencoder have been proposed primarily for two purposes: enhancing the representational power of the factors and modeling them as latent variables. The first purpose typically involves modifying the loss function to prevent the autoencoder from simply copying the input through the encoder and pasting it as the output of the decoder. Sparse autoencoder (Ng, 2011) adds a penalty term on the nodes to the loss function to induce sparsity or shrinkage. Contractive autoencoder (Rifai et al., 2011a,b) considers another type of penalty term which shrinks the Jacobian of $\boldsymbol{f}_t$ with respect to $\boldsymbol{x}_t$, with the assumption that factors are not sensitive to the change in $\boldsymbol{x}_t$. Alternatively, denoising autoencoder (Vincent et al., 2008) modifies the data set such that the input of the autoencoder is

[22] If the encoder and decoder have different numbers of layers, this autoencoder is asymmetric and has been studied by Majumdar and Tripathi (2017).

corrupted as a noisy version of $\boldsymbol{x}_t$ and aims to reconstruct the original $\boldsymbol{x}_t$, for $t \in [T]$. For the second purpose, variational autoencoder (VAE) (Kingma and Welling, 2014) assumes the factors to be stochastic, rather than deterministic in the standard form. Instead of inferring $\boldsymbol{f}_t$ with its posterior $p(\boldsymbol{f}_t \mid \boldsymbol{x}_t)$, the encoder of a VAE outputs the parameters of a variational posterior $q_{\boldsymbol{\phi}}(\boldsymbol{f}_t \mid \boldsymbol{x}_t)$. The decoder is treated as a generative model, which determines the likelihood, $p_{\boldsymbol{\theta}}(\boldsymbol{x_t} \mid \boldsymbol{f_t})$. Given a prior of $\boldsymbol{f}_t$ as $p(\boldsymbol{f}_t)$, the parameters in encoder and decoder are optimized by maximizing the evidence lower bond of the marginal likelihood $p(\boldsymbol{x}_t)$, $\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{f}_t \mid \boldsymbol{x}_t)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_t \mid \boldsymbol{f}_t)] - D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{f}_t \mid \boldsymbol{x}_t) \,\|\, p(\boldsymbol{f}_t))$. After optimizing parameters, a VAE can generate new data by sampling from the prior of $\boldsymbol{f}_t$ and passing the low-dimensional data to trained decoder.

The application of autoencoders mainly focuses on dimension reduction, which uses the bottleneck to implement downstream tasks, such as classification (Khozeimeh et al., 2021), clustering (Song et al., 2013), visualization (Hinton and Salakhutdinov, 2006), and fine-tuning (Erhan et al., 2010). Apart from dimension reduction, the VAE is regarded as a generative model, capable of generating new data points by sampling from the latent space, thereby enabling the model to create new data points to facilitate scientific discovery. For example, Gómez-Bombarelli et al. (2018), Kusner et al. (2017), and Dai et al. (2018) applied variational autoencoders to generate molecules which have the potential for drug design.

Chapter 3

Bayesian Tensor Vector Autoregression

As reviewed in Section 2.1.4, VARs are prone to over-parameterization because the number of parameters grows quadratically with the number of variables. This issue is particularly pronounced in macroeconomic applications, where data are typically collected at low frequencies (e.g., monthly or quarterly) and the number of variables is relatively large (often exceeding 20). This chapter contributes to the high-dimensional VAR literature by building on the work of Wang et al. (2022a), which applied tensor decomposition to a third-order tensor representing the VAR coefficients, leading to a tensor VAR (TVAR) model. Specifically, we employ the CP decomposition and estimate the parameters using a Bayesian approach, in contrast to the Tucker decomposition and frequentist estimation method proposed by Wang et al. (2022a). The motivation for adopting this methodology to address over-parameterization is fourfold. First, tensor decomposition aligns with dense modeling of the parameters, which is in line with studies that questioned the suitability of sparse modeling in macroeconomic research, see Giannone et al. (2021) for the "illusion of sparsity". Second, a TVAR with an appropriate choice of rank is parsimonious without imposing any penalty term or shrinkage prior (although incorporating these techniques results in further parsimony). Third, the TVAR is a useful model for explaining macroeconomic data since its reconstruction provides insights into the economy, and margins are interpretable as shown in Wang et al. (2022a) and Chen et al. (2022). Fourth, tensor structures with Bayesian inference have been successfully applied in time series models apart from VARs, which have been discussed in Section 2.3.4.

Our first contribution is to infer the rank using the multiplicative gamma prior (MGP) (Bhattacharya and Dunson, 2011), which is imposed on the loadings of the CP decomposition. We then propose an adaptive inferential scheme that allows the rank to be reduced during the inference process. Our contribution is closely related to the method proposed by Fan et al.

(2022), who also employed the MGP prior and introduced an adaptive algorithm. However, our approach differs from theirs in two aspects. First, Fan et al. (2022) incorporated both the horseshoe prior and the MGP as the prior on the loadings. Second, they followed a different criterion to reduce the rank.

The second contribution is to retain the interpretability of a TVAR based on the MCMC result. From a Bayesian perspective, a fundamental prerequisite for a TVAR to be interpretable is the convergence of margin (i.e., elements of the loadings) Markov chains, but this prerequisite cannot be achieved using the traditional MCMC scheme because the indeterminacy of the CP decomposition can lead to poorly mixing MCMC result, which in turn produces posterior distributions that are difficult to interpret. We improve the mixing of the MCMC algorithm by introducing a Gibbs sampler including a variant of the Ancillarity-Sufficiency Interweaving Strategy (ASIS) (Yu and Meng, 2011) with three interweaving steps, inspired by the ASIS algorithm for factor models (Kastner et al., 2017). Even if the mixing of margins is not essential in some instances, e.g., one does not interpret margins and only regards the mixing of entries in the tensor itself as important, this contribution is still beneficial because achieving good mixing of margins provides a solid foundation for entries in the VAR coefficient matrix to mix well. Additionally, we introduce a post-processing procedure aimed at identifying the margins.

We examine the utility of TVARs through two U.S. macroeconomic data sets with medium and large sizes. We consider two specifications of TVARs that treat the coefficient matrix in two ways: (1) the matricization of a third-order tensor, and (2) a sum of the matricization of a third-order tensor and a matrix with only non-zero entries for own lags. The first one corresponds to the original TVAR idea of Wang et al. (2022a), and the second one accommodates the main feature of the Minnesota-type priors (see Section 2.1.4 for details), i.e., the own lags of a variable are more informative than lags of other dependent variables. In point and density forecasts, these two TVARs obtain the best results for joint forecasting tasks and are competitive to standard VARs with a range of standard prior choices. We demonstrate how to interpret margins by applying our model to the large-scale data and constructing factors as linear combinations of lagged data. The TVARs can effectively reduce the number of parameters, and the factors constructed can summarize the dynamics of the data set. The additional own-lag matrix in the second TVAR structure introduces more parameters but allows the tensor to focus on exploring the cross-lag effects.

The chapter is organized as follows. Section 3.1 explains the TVAR and its interpretation. Section 3.2 provides the MCMC schemes. Section 3.3 introduces the post-processing procedure. Section 3.4 shows results from simulation experiments. Section 3.5 presents the forecasting performance and interpretation of TVARs using the real data set. Section 3.6 concludes our work.

3.1 Tensor VAR

3.1.1 Model Specification

Consider the VAR(P) presented in (2.3) (P is the lag order), TVAR(P) models the multivariate time series $\boldsymbol{y}_t \in \mathbb{R}^N$ as

$$\boldsymbol{y}_t = \boldsymbol{\mathcal{A}}_{(1)}\boldsymbol{x}_t + \boldsymbol{\epsilon}_t, \tag{3.1}$$

where $\boldsymbol{\mathcal{A}}_{(1)} = \boldsymbol{A} \in \mathbb{R}^{N\times NP}$ is the mode-1 matricization of a third-order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{N\times N\times P}$, with $\boldsymbol{\mathcal{A}}_{(i_1,i_2,p)}$ corresponding to the $(i_1,\, i_2)$ entry in $\boldsymbol{A}_p$, for $i_1,\, i_2 \in [N]$ and $p \in [P]$. The error term $\boldsymbol{\epsilon}_t$ follows a multivariate normal distribution with zero mean and a time-varying covariance matrix $\boldsymbol{\Omega}_t$. We factorize $\boldsymbol{\Omega}_t$ according to Huber and Feldkircher (2019) such that $\boldsymbol{\Omega}_t = \boldsymbol{H}^{-1}\boldsymbol{S}_t(\boldsymbol{H}^{-1})'$, where $\boldsymbol{H}^{-1}$ is a unit lower triangular matrix, and $\boldsymbol{S}_t$ is a time-varying diagonal matrix with diagonal terms $(s_{t,1}, ..., s_{t,N})$. Although the number of entries in $\boldsymbol{\mathcal{A}}$ is the same as that in $\boldsymbol{A}$, the tensor $\boldsymbol{\mathcal{A}}$ can be decomposed via a rank-R CP decomposition,

$$\boldsymbol{\mathcal{A}} = \sum_{r=1}^{R} \boldsymbol{\mathcal{A}}^{(r)} = \sum_{r=1}^{R} \boldsymbol{\beta}_1^{(r)} \circ \boldsymbol{\beta}_2^{(r)} \circ \boldsymbol{\beta}_3^{(r)}, \tag{3.2}$$

where $\boldsymbol{\mathcal{A}}^{(r)}$ is a third-order tensor with the same dimension as $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\beta}_1^{(r)}$, $\boldsymbol{\beta}_2^{(r)} \in \mathbb{R}^N$ and $\boldsymbol{\beta}_3^{(r)} \in \mathbb{R}^P$ store the margins of $\boldsymbol{\mathcal{A}}$, for $r \in [R]$, $\circ$ denotes the outer product defined in (2.26). Following the notation given in Section 2.3.2, this CP decomposition is written as $\boldsymbol{\mathcal{A}} = [\![\boldsymbol{B}_1, \boldsymbol{B}_2, \boldsymbol{B}_3]\!]_{\text{CP}}$, where $\boldsymbol{B}_j = (\boldsymbol{\beta}_j^{(1)}, \cdots, \boldsymbol{\beta}_j^{(R)}) \in \mathbb{R}^{I_j\times R}$, for $j \in [3]$, $I_1 = I_2 = N$ and $I_3 = P$. Another useful representation of the margins is $\boldsymbol{B} = (\boldsymbol{B}_1', \boldsymbol{B}_2', \boldsymbol{B}_3')' \in \mathbb{R}^{(2N+P)\times R}$ to which we refer as a *tensor matrix*, then $\boldsymbol{\mathcal{A}}^{(r)}$ is constructed by margins in the r-th column of $\boldsymbol{B}$. With an upper bound $N^2P/(2N+P)$ of R, the number of parameters in the TVAR(P) reduces from N^2P in the coefficient matrix to $(2N+P)R$ in $\boldsymbol{B}$, so a low-rank structure in the CP decomposition alleviates over-parameterization.

The model in (3.1) represents the original Tensor VAR (Wang et al., 2022a), which does not distinguish between the own-lag and cross-lag effects. In Section 3.5, we empirically find that introducing this distinction allows us to achieve better forecasting performance and interpretability, so we build an extension of (3.1), called Own-lag TVAR, following the assumption of the Minnesota-type priors, the own-lag effects are more powerful than the cross-lag effects. In particular, we add a matrix $\boldsymbol{D}$, the concatenation of P N-by-N diagonal matrices, to give

$$\boldsymbol{y}_t = \boldsymbol{\mathcal{A}}_{(1)}\boldsymbol{x}_t + \boldsymbol{D}\boldsymbol{x}_t + \boldsymbol{\epsilon}_t, \tag{3.3}$$

so $\boldsymbol{D}$ can only affect entries corresponding to own lags.

3.1.2 Model Interpretation

The TVAR connects $\boldsymbol{y}_t^* = \boldsymbol{y}_t - \boldsymbol{D}\boldsymbol{x}_t$[23] with the past information through $\boldsymbol{B}_1, \boldsymbol{B}_2, \boldsymbol{B}_3$ in the following reconstruction

$$\boldsymbol{y}_t^* = \boldsymbol{B}_1\boldsymbol{\mathcal{I}}_{(1)}\text{vec}(\boldsymbol{B'}_2\boldsymbol{X}_t\boldsymbol{B}_3) + \boldsymbol{\epsilon}_t = \sum_{r=1}^{R}\boldsymbol{B}_{1,(\cdot,r)}\sum_{i_2=1}^{N}\sum_{i_3=1}^{P}\boldsymbol{\beta}_{2,i_2}^{(r)}\boldsymbol{\beta}_{3,i_3}^{(r)}\boldsymbol{y}_{t-i_3,i_2} + \boldsymbol{\epsilon}_t, \tag{3.4}$$

where $\boldsymbol{\mathcal{I}}_{(1)} \in \mathbb{R}^{R\times R^2}$ is the mode-1 matricization of a third-order superdiagonal tensor $\boldsymbol{\mathcal{I}}$ with ones on non-zero entries, $\boldsymbol{X}_t = (\boldsymbol{y}_{t-1}, \ldots, \boldsymbol{y}_{t-P})$, vec$(\cdot)$ is the vectorization operation which transforms $\boldsymbol{B'}_2\boldsymbol{X}_t\boldsymbol{B}_3 \in \mathbb{R}^{R\times R}$ to an R^2-dimensional vector, $\boldsymbol{B}_{1,(\cdot,r)}$ is the r-th column of $\boldsymbol{B}_1$, $\boldsymbol{\beta}_{2,i_2}^{(r)}, \boldsymbol{\beta}_{3,i_3}^{(r)}$ are the (i_2, r) and (i_3, r) entries of $\boldsymbol{B}_2$ and $\boldsymbol{B}_3$, respectively, $\boldsymbol{y}_{t-i_3,i_2}$ is the i_2-th entry in $\boldsymbol{y}_{t-i_3}$.

Following Wang et al. (2022a), we can relate (3.4) to a factor model (such as the one in (2.12)), where $\boldsymbol{B}_1$ is the factor loading and $\boldsymbol{\mathcal{I}}_{(1)}\text{vec}(\boldsymbol{B'}_2\boldsymbol{X}_t\boldsymbol{B}_3)$ contains R observable factors. Since the i_1-th row in $\boldsymbol{B}_1$ describes the linear relationship between $\boldsymbol{y}_{t,i_1}$ and factors, for $i_1 \in [N]$, we refer to $\boldsymbol{B}_1$ as "response loading". The formation of factors describes how past information is combined. We use $\sum_{i_2=1}^{N}\sum_{i_3=1}^{P}\boldsymbol{\beta}_{2,i_2}^{(r)}\boldsymbol{\beta}_{3,i_3}^{(r)}\boldsymbol{y}_{t-i_3,i_2}$ in (3.4) to understand this formation. If $\boldsymbol{\beta}_{2,i_2}^{(r)} = 0$, the r-th factor will not contain information from any lagged values of $\boldsymbol{y}_{t,i_2}$. Similarly, $\boldsymbol{\beta}_{3,i_3}^{(r)} = 0$ results in no information about the i_3-th lag of $\boldsymbol{y}_t$ in the r-th factor. Therefore, the i_2-th row of $\boldsymbol{B}_2$ contains the effect from the i_2-th variable to $\boldsymbol{y}_t$, and the i_3-th row of $\boldsymbol{B}_3$ is related to the effect from the i_3-th lag to $\boldsymbol{y}_t$. This interpretation was also discussed in Wang et al. (2022a), who called $\boldsymbol{B}_2$ and $\boldsymbol{B}_3$ "predictor loading" and "temporal loading", respectively.

Another way to explain the CP decomposition in the TVAR is that it separates the lag effect

[23] We include $\boldsymbol{D}$ for completion. $\boldsymbol{D}$ is a zero matrix if we apply (3.1).

from the variable-wise effect because it decomposes $\boldsymbol{A}_p$ as $\sum_{r=1}^{R}(\boldsymbol{\beta}_1^{(r)} \circ \boldsymbol{\beta}_2^{(r)})\boldsymbol{\beta}_{3,p}^{(r)}$. The first two vectors $\boldsymbol{\beta}_1^{(r)}$ and $\boldsymbol{\beta}_2^{(r)}$ (for $r \in [R]$) do not depend on the index of $\boldsymbol{A}_p$, suggesting that all lagged coefficients matrices share these vectors. The only difference among these matrices reflects on the different entries in $\boldsymbol{\beta}_3^{(r)}$.

3.2 Bayesian Inference

3.2.1 Prior Specification

The rank R in the CP decomposition is an important component which controls the parsimony of the TVAR. To select this parameter, we impose the MGP (Bhattacharya and Dunson, 2011) on the tensor matrix $\boldsymbol{B}$ because this prior has the increasing shrinkage property, enabling margins with higher column indices to have higher degrees of shrinkage. As a result, the rank can be lowered if some columns in $\boldsymbol{B}$ have magnitudes negligibly small. To be specific, a margin $\boldsymbol{\beta}_{j,i_j}^{(r)}$ (the (i_j, r) entry of $\boldsymbol{B}_j$) follows the prior below for $j \in [3]$, $r \in [R]$, $i_1,\ i_2 \in [N]$ and $i_3 \in [P]$,

$$\boldsymbol{\beta}_{j,i_j}^{(r)} \sim \mathcal{N}\left(0,\ \left(\sigma_{j,i_j}^{(r)}\right)^2\right),\ \left(\sigma_{j,i_j}^{(r)}\right)^2 = \phi_{(r,j,i_j)}^{-1}\tau_r^{-1},$$

$$\phi_{(r,j,i_j)} \sim \text{Gamma}\left(\nu/2, \nu/2\right), \tau_r = \prod_{l=1}^{r}\delta_l,$$

$$\delta_1 \sim \text{Gamma}\left(a_1, 1\right),\ \delta_l \sim \text{Gamma}\left(a_2, 1\right),\ 1 < l < R,$$

where $\phi_{(r,j,i_j)}$ is a local parameter for the margin with the same index. We store all these local parameters in a matrix $\boldsymbol{\Phi}$ in which each entry corresponds to an entry in the tensor matrix $\boldsymbol{B}$ with the same indices. The increasing shrinkage property is induced by τ_r since $\mathbb{E}(\tau_r) = \prod_{l=1}^{r}\mathbb{E}(\delta_l) = a_1 a_2^{r-1}$ increases with r, when $a_2 > 1$. Hyperparameter ν is set to be known, and a_1 and a_2 will be inferred with Gamma priors. Durante (2017) showed that both $\mathbb{E}(\tau_r)$ and $\mathbb{E}(\tau_r^{-1})$ increase with r when $1 < a_2 < 2$. This result means that the MGP has the increasing shrinkage property only when $a_2 > 2$. Thus, we set priors for a_1 and a_2 as Gamma(5,1) to have the increasing shrinkage property with a high probability.

In the case of own-lag TVAR, we impose a variant of normal-gamma prior defined in Huber and Feldkircher (2019) to each non-zero entry in $\boldsymbol{D}$. Let $d_{i,p}$ denote the own-lag coefficient for the p-th lag of the i-th response, then its prior is written as

$$d_{i,p} \sim \mathcal{N}\left(0,\ \left(2/\lambda_d^2\right)\psi_d^{(i,p)}\right),\ \psi_d^{(i,p)} \sim \text{Gamma}(a_d, a_d),\ \text{for } i \in [N] \text{ and } p \in [P], \tag{3.5}$$

where $\lambda_d^2 \sim \mathcal{G}(0.01, 0.01)$ and $a_d \sim \mathcal{E}(1)$. For $\boldsymbol{\Omega}_t$, we adopt a similar prior as in (3.5) on the non-zero entries of $\boldsymbol{H}$ and model diagonal terms of $\boldsymbol{S}_t$ using stochastic volatility applied in Huber and Feldkircher (2019),

$$\ln(s_{t,i}) \mid \ln(s_{t-1,i}), \mu_i, \psi_i, \sigma_i \sim \mathcal{N}\left(\mu_i + \psi_i \left(\ln(s_{t-1,i}) - \mu_i\right), \sigma_i^2\right), \tag{3.6}$$

$$\ln(s_{0,i}) \mid \mu_i, \psi_i, \sigma_i \sim \mathcal{N}\left(\mu_i, \sigma_i^2 / \left(1 - \psi_i^2\right)\right). \tag{3.7}$$

All the hyperpriors are available in Appendix B.1.1, and the full conditionals can be found in Appendix B.1.

3.2.2 An Overview of Inferential Scheme

To illustrate the strengths of our inferential scheme, we contrast it with a widely used one in the tensor literature. In the traditional scheme (Billio et al., 2023; Fan et al., 2022; Guhaniyogi et al., 2017; Zhang et al., 2021), $\boldsymbol{\beta}_j^{(r)}$ is sampled from $p\left(\boldsymbol{\beta}_j^{(r)} \mid \boldsymbol{\beta}_{-j}^{(r)}, \boldsymbol{B}_{(\cdot,-r)}, \boldsymbol{y}_{1:T}, \left(\boldsymbol{\sigma}_j^{(r)}\right)^2\right)$, for $r \in [R]$ and $j \in [J]$ (J is 3 in our case), where $\boldsymbol{\beta}_{-j}^{(r)}$ contains all $\boldsymbol{\beta}_{j'}^{(r)}$ with $j' \neq j$, $\boldsymbol{B}_{(\cdot,-r)}$ is $\boldsymbol{B}$ discarding its r-th column, $\left(\boldsymbol{\sigma}_j^{(r)}\right)^2$ contains all prior variance corresponding to $\boldsymbol{\beta}_j^{(r)}$. These full conditionals are then incorporated into a usual Gibbs sampler, so each $\boldsymbol{\beta}_j^{(r)}$ sampled depends on other margins, and in turn, other margins are sampled given $\boldsymbol{\beta}_j^{(r)}$ and other parameters. The rank R is fixed to a large value during the inference and can be determined to a smaller value *a posteriori*.

This inferential scheme neglects the convergence of margin Markov chains because authors are more interested in the tensor itself, so they pay more attention to the convergence of the tensor elements rather than the margins. The convergence issue arises from the indeterminacy of margins, mentioned in Section 2.3.2, which leads to poor mixing of the Markov chains, consequently hindering convergence. We consider the convergence of margins to be an important aspect for two reasons. First, margins in TVARs are potentially interpretable as shown in Wang et al. (2022a); second, as the literature on TVARs grows, one cannot guarantee that the Markov chains in a more complex model, e.g., regime-switching TVARs, still converge. Apart from the convergence issue, it is computationally expensive to infer the rank using the above MCMC scheme since it assumes R to be fixed during the inference. To address these issues, we propose three modifications to our inferential framework. Two of these modifications aim to alleviate the poor mixing contributing to the convergence issue. The third modification enhances computational efficiency.

Firstly, we reduce the dependence between columns within $\boldsymbol{B}_j$, for $j \in [3]$, by introducing a block sampler, which divides margins into three blocks according to the three loadings mentioned in Section 3.1.1. This block sampler is feasible because a TVAR can be written as

$$\begin{aligned} \boldsymbol{y}_t^* &= \left(\boldsymbol{x}_t' \left(\boldsymbol{B}_3 \otimes \boldsymbol{B}_2\right) \boldsymbol{\mathcal{I}}_{(1)}' \otimes \boldsymbol{I}_N\right) \operatorname{vec}(\boldsymbol{B}_1) + \boldsymbol{\epsilon}_t && (3.8)\\ &= \boldsymbol{B}_1 \boldsymbol{\mathcal{I}}_{(1)} \left((\boldsymbol{B}_3' \boldsymbol{X}_t') \otimes \boldsymbol{I}_R\right) \operatorname{vec}(\boldsymbol{B}_2') + \boldsymbol{\epsilon}_t && (3.9)\\ &= \boldsymbol{B}_1 \boldsymbol{\mathcal{I}}_{(1)} \left(\boldsymbol{I}_R \otimes (\boldsymbol{B}_2' \boldsymbol{X}_t)\right) \operatorname{vec}(\boldsymbol{B}_3) + \boldsymbol{\epsilon}_t. && (3.10) \end{aligned}$$

Therefore, margins in one loading can be sampled jointly to reduce their dependence on each other.

Secondly, we do *not* use a standard Gibbs sampler to sample loadings. Instead, we introduce a variant of the ASIS, containing four different parameterizations, to reduce the parameter autocorrelation during the sampling. Given a rank value in each sample iteration, the interweaving Gibbs sampler interweaves between full conditional distributions under a base parameterization and the other three (one for each loading).

Lastly, the rank R in our case is adaptively inferred similarly to Bhattacharya and Dunson (2011) to speed up computation. In the following three subsections, we introduce the interweaving Gibbs sampler for a fixed rank in Section 3.2.3 and the adaptive inferential scheme of the rank in Section 3.2.4.

3.2.3 Interweaving Gibbs Sampler

In principle, we could run a standard Gibbs sampler to infer margins and other parameters, but in practice, Markov chains of margins suffer from poor mixing since these chains are highly autocorrelated. We circumvent margins with poor mixing by introducing a variant of the ASIS, which unfolds its strategy from its name: sampling the same block of parameters by interweaving two sampling schemes corresponding to two data augmentations – ancillary statistic and sufficient statistic. The benefit of the ASIS is that the mixing will be at least as good as the one using only one data augmentation, and a low correlation between these two augmentations leads to faster convergence and better mixing compared to using either augmentation alone. Because of these benefits, the ASIS has been applied to many models, including stochastic volatility (Kastner and Frühwirth-Schnatter, 2014) and factor models (Kastner et al., 2017).

Our ASIS parameterizations are more related to those in Kastner et al. (2017) for sampling factor loadings and factors due to the tensor structure. The tensor structure in the TVAR leads to four parameterizations instead of two in Kastner et al. (2017). The first parameterization, which we call the base one, is $\boldsymbol{B}_1$, $\boldsymbol{B}_2$, and $\boldsymbol{B}_3$ described in Section 3.1.1. The remaining three parameterizations come from specifications of scaling indeterminacy of the CP decomposition (see Section 2.3.2 for details). Specifically, $\boldsymbol{\mathcal{A}} = [\![\boldsymbol{B}_1,\, \boldsymbol{B}_2,\, \boldsymbol{B}_3]\!]_{\text{CP}} = [\![\boldsymbol{B}_1^*,\, \boldsymbol{B}_2^*,\, \boldsymbol{B}_3]\!]_{\text{CP}}$ when $\boldsymbol{B}_1^*$, $\boldsymbol{B}_2^*$ are transformed from

$$\boldsymbol{B}_1^* = \boldsymbol{B}_1\boldsymbol{D}_1^{-1}, \boldsymbol{B}_2^* = \boldsymbol{B}_2\boldsymbol{D}_1, \tag{3.11}$$

where $\boldsymbol{D}_1$ is a diagonal matrix with non-zero, non-infinite diagonal entries.

There are infinite choices of $\boldsymbol{D}_1$ to get this equivalence, but since our objective is boosting the mixing of margins, we restrict $\boldsymbol{D}_1$ to be related to $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$. We choose $\boldsymbol{D}_1 = \text{diag}\left(\boldsymbol{\beta}_{1,1}^{(1)}, \ldots, \boldsymbol{\beta}_{1,1}^{(R)}\right)$ for further demonstration. This choice restricts the first row of $\boldsymbol{B}_1^*$ to be ones. Other choices of $\boldsymbol{D}_1$ can be investigated in future work. After the transformation, we are able to write the model in terms of $\boldsymbol{B}_1^*$, $\boldsymbol{B}_2^*$, and $\boldsymbol{D}_1$ for the second parameterization. For $i_1,\, i_2 \in [N]$, we have

$$\boldsymbol{\beta}_{1,1}^{*(r)} = 1,\, \boldsymbol{\beta}_{1,i_1}^{*(r)} \sim \mathcal{N}\left(0, \left(\frac{\sigma_{1,i_1}^{(r)}}{\boldsymbol{\beta}_{1,1}^{(r)}}\right)^2\right),\, \boldsymbol{\beta}_{2,i_2}^{*(r)} \sim \mathcal{N}\left(0, \left(\sigma_{2,i_2}^{(r)}\boldsymbol{\beta}_{1,1}^{(r)}\right)^2\right). \tag{3.12}$$

The above parameterization only improves the mixing of margins in $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, so we also need a parameterization to improve the mixing of margins in $\boldsymbol{B}_3$. An obvious choice is to pair $\boldsymbol{B}_2$ and $\boldsymbol{B}_3$. At this point, each $\boldsymbol{B}_j$ has been paired at least once, but we conjecture that an additional pair of $\boldsymbol{B}_1$ and $\boldsymbol{B}_3$ would provide better mixing than just considering three parameterizations because the mixing would be improved across margins in each pair of $\boldsymbol{B}_j$'s. Transformations of these two pairs are similar to the one for $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$,

$$\boldsymbol{B}_2^{**} = \boldsymbol{B}_2\boldsymbol{D}_2^{-1}, \boldsymbol{B}_3^{**} = \boldsymbol{B}_3\boldsymbol{D}_2;\; \boldsymbol{B}_3^{***} = \boldsymbol{B}_3\boldsymbol{D}_3^{-1}, \boldsymbol{B}_1^{***} = \boldsymbol{B}_1\boldsymbol{D}_3, \tag{3.13}$$

where $\boldsymbol{D}_2$ and $\boldsymbol{D}_3$ are diagonal matrices with non-zero, non-infinite diagonal entries.

Similarly, we choose the diagonal entries in $\boldsymbol{D}_2$ to be the first row of $\boldsymbol{B}_2$, and likewise for those in $\boldsymbol{D}_3$ (as the first row of $\boldsymbol{B}_3$). These lead to the last two parameterizations, which are presented in terms of $\boldsymbol{B}_2^{**}$, $\boldsymbol{B}_3^{**}$, $\boldsymbol{D}_2$, and $\boldsymbol{B}_3^{***}$, $\boldsymbol{B}_1^{***}$, $\boldsymbol{D}_3$, respectively. For $i_1,\, i_2 \in [N]$, $i_3 \in [P]$, the following two expressions represent the two parameterizations

$$\boldsymbol{\beta}_{2,1}^{**(r)} = 1,\, \boldsymbol{\beta}_{2,i_2}^{**(r)} \sim \mathcal{N}\left(0, \left(\frac{\sigma_{2,i_2}^{(r)}}{\boldsymbol{\beta}_{2,1}^{(r)}}\right)^2\right),\, \boldsymbol{\beta}_{3,i_3}^{**(r)} \sim \mathcal{N}\left(0, \left(\sigma_{3,i_3}^{(r)}\boldsymbol{\beta}_{2,1}^{(r)}\right)^2\right), \tag{3.14}$$

$$\boldsymbol{\beta}_{3,1}^{***(r)} = 1,\ \boldsymbol{\beta}_{3,i_3}^{***(r)} \sim \mathcal{N}\left(0, \left(\frac{\sigma_{3,i_3}^{(r)}}{\boldsymbol{\beta}_{3,1}^{(r)}}\right)^2\right),\ \boldsymbol{\beta}_{1,i_1}^{***(r)} \sim \mathcal{N}\left(0, \left(\sigma_{1,i_1}^{(r)}\boldsymbol{\beta}_{3,1}^{(r)}\right)^2\right). \tag{3.15}$$

In each MCMC iteration, the loadings are sampled under these four parameterizations. The sampling using the base parameterization is stated in Appendix B.1.2, and we focus on sampling margins under the other three parameterizations introduced in this subsection. For $\boldsymbol{\beta}_{1,1}^{(r)}$, its normal prior implies that $\left(\boldsymbol{\beta}_{1,1}^{(r)}\right)^2$ has a gamma prior, Gamma$\left(\frac{1}{2}, \frac{1}{2\left(\sigma_{1,1}^{(r)}\right)^2}\right)$. The full conditional of $\left(\boldsymbol{\beta}_{1,1}^{(r)}\right)^2$ under (3.12) is a Generalized Inverse Gaussian ($\mathcal{GIG}$), see Appendix A.1 for the definition,

$$\left(\boldsymbol{\beta}_{1,1}^{(r)}\right)^2 \mid \boldsymbol{B}_{1,(\cdot,r)}^{*}, \boldsymbol{B}_{2,(\cdot,r)}^{*} \sim \mathcal{GIG}\left(0, \sum_{i_2=1}^{M}\left(\frac{\boldsymbol{\beta}_{2,i_2}^{*(r)}}{\sigma_{2,i_2}^{(r)}}\right)^2, \sum_{i_1=2}^{M}\left(\frac{\boldsymbol{\beta}_{1,i_1}^{*(r)}}{\sigma_{1,i_1}^{(r)}}\right)^2 + \left(\frac{1}{\sigma_{1,1}^{(r)}}\right)^2\right), \tag{3.16}$$

Similarly, we can get full conditionals of $\left(\boldsymbol{\beta}_{2,1}^{(r)}\right)^2$ under (3.14) and $\left(\boldsymbol{\beta}_{3,1}^{(r)}\right)^2$ under (3.15)

$$\left(\boldsymbol{\beta}_{2,1}^{(r)}\right)^2 \mid \boldsymbol{B}_{2,(\cdot,r)}^{**}, \boldsymbol{B}_{3,(\cdot,r)}^{**} \sim \mathcal{GIG}\left(\frac{M-P}{2}, \sum_{i_3=1}^{P}\left(\frac{\boldsymbol{\beta}_{3,i_3}^{**(r)}}{\sigma_{3,i_3}^{(r)}}\right)^2, \sum_{i_2=2}^{M}\left(\frac{\boldsymbol{\beta}_{2,i_2}^{**(r)}}{\sigma_{2,i_2}^{(r)}}\right)^2 + \left(\frac{1}{\sigma_{2,1}^{(r)}}\right)^2\right), \tag{3.17}$$

$$\left(\boldsymbol{\beta}_{3,1}^{(r)}\right)^2 \mid \boldsymbol{B}_{3,(\cdot,r)}^{***}, \boldsymbol{B}_{1,(\cdot,r)}^{***} \sim \mathcal{GIG}\left(\frac{P-M}{2}, \sum_{i_1=1}^{M}\left(\frac{\boldsymbol{\beta}_{1,i_1}^{***(r)}}{\sigma_{1,i_1}^{(r)}}\right)^2, \sum_{i_3=2}^{P}\left(\frac{\boldsymbol{\beta}_{3,i_3}^{***(r)}}{\sigma_{3,i_3}^{(r)}}\right)^2 + \left(\frac{1}{\sigma_{3,1}^{(r)}}\right)^2\right). \tag{3.18}$$

Algorithm 4 outlines how to interweave sampling under the base parameterization with the second one described in (3.12). Similar algorithms can be applied to the third and fourth parameterizations, incorporating full conditionals in (3.17) and (3.18). Combining these three algorithms leads to a Gibbs sampler, of which the full algorithm can be found in Appendix B.1.5. If we only sample margins using Step (a), the algorithm is just a standard Gibbs sampler with the base parameterization. Every interweaving step starts at the base parameterization, then switches to an alternative parameterization and swaps back to the base one. Note that $\boldsymbol{B}_2$ in Step (d) has superscript $\widetilde{\text{new}}$. This is because $\boldsymbol{B}_2$ is included in two interweaving steps, but we only store one sample for $\boldsymbol{B}_2$ in each iteration. It will be easier to distinguish between the one stored (with superscript "new") and the one left (with superscript $\widetilde{\text{new}}$). One can find the same superscripts in the full algorithm.

Algorithm 4 Interweave between the base parameterization and the one in (3.12).

Step (a): Update $\boldsymbol{B}_1^{\text{old}}$ under the base parameterization.

Step (b): Store the first row of $\boldsymbol{B}_1^{\text{old}}$ into $\boldsymbol{D}_1$ and determine $\boldsymbol{B}_1^*$ and $\boldsymbol{B}_2^*$.

Step (c): Sample $\left(\beta_{1,1}^{\text{new}(r)}\right)^2$ for $r = 1, \ldots, R$ using the corresponding full conditional in (3.16) and store sampled values into $\boldsymbol{D}_1$.

Step (d): Update $\boldsymbol{B}_1^{\text{new}}$ and $\boldsymbol{B}_2^{\text{n\tilde{e}w}}$ with transformation $\boldsymbol{B}_1^{\text{new}} = \boldsymbol{B}_1^* \boldsymbol{D}_1$, $\boldsymbol{B}_2^{\text{n\tilde{e}w}} = \boldsymbol{B}_2^* \boldsymbol{D}_1^{-1}$.

It is worth stressing that the interweaving strategy improves the mixing of entries in $\boldsymbol{B}$ up to column permutations and sign-switching issues. Thus, we propose a post-processing procedure to identify the margins *a posteriori* in Section 3.3.

3.2.4 Adaptive Inference of Rank

We aim to infer the rank by finding inactive columns in $\boldsymbol{B}$, i.e., those columns which do not contribute much to the tensor $\boldsymbol{\mathcal{A}}$. An adaptive algorithm, inspired by Bhattacharya and Dunson (2011), is displayed in Algorithm 5.

Algorithm 5 Adaptive Inference of Rank

```
Initialize $R^*$, $\alpha_0$, $\alpha_1$ and set a criterion
while iteration $\tilde{l} < l \leq l_{\text{burn-in}}$ do
    Sample $u$ from Uniform(0,1) and let $R^{(l)}$ be the rank at iteration $l$
    if $p(l) \geq u$ then
        $k$=#{inactive columns}
        if $k > 0$ then
            Remove inactive columns in $\boldsymbol{B}$ and related parameters
            $R^{(l)} = R^{(l-1)} - k$
        else
            Add one column to $\boldsymbol{B}$ and expand related parameters
            $R^{(l)} = R^{(l-1)} + 1$
        end if
    end if
end while
```

In this algorithm, we initialize the rank as $R^* = \lceil 5 \log N \rceil$, which is the same as for the number of factors in Bhattacharya and Dunson (2011). Empirically, this initialization is large enough to estimate the coefficient matrix. We discard inactive columns in the l-th iteration with probability $p(l) = \exp(\alpha_0 + \alpha_1 l)$, where $\alpha_0 \leq 0$, $\alpha_1 < 0$. Since $p(l)$ gets smaller as l

increases, R is less likely to change during the inference. Lastly, we set a criterion to decide whether a column in $\boldsymbol{B}$ is active. Specifically, this criterion is related to the proportion of small magnitudes in $\boldsymbol{\mathcal{A}}^{(r)}$, for $r \in [R]$. For ease of explanation, we omit l here. We regard an entry in $\boldsymbol{\mathcal{A}}^{(r)}$ to have a small magnitude if its absolute value is smaller than a threshold γ_1, e.g., $\gamma_1 = 10^{-3}$. If the proportion of small magnitudes in $\boldsymbol{\mathcal{A}}^{(r)}$ is larger than another threshold γ_2 set *a-priori*, e.g., $\gamma_2 = 0.9$, then we regard the r-th column in $\boldsymbol{B}$ as inactive. We use the simulation study to determine γ_1 and γ_2 so as to minimize the rank inferred while simultaneously ensuring accurate inference of the coefficient matrix. More details about choosing γ_1 and γ_2 are available in Appendix B.2.1. Note that the choice of γ_1 and γ_2 depends on the scale of time series. To overcome the sensitivity due to the scale, we standardize all time series in the real data application. The number of time series (N) also affects the selection. For example, γ_1 can be selected as a lower value as N increases. The experimental results in this chapter rely on fixed choices of γ_1 and γ_2, and these results demonstrate that the tensor VAR performs well under these settings; however, how these thresholds depend on N remains an open question for future work.

Adaptive inference begins after the $\tilde{l}$-th iteration to stabilize Markov chains and stops at the last iteration during the burn-in period to allow easy interpretation of margins. If the number of inactive columns is greater than 0, we remove these columns in $\boldsymbol{B}$ and remove corresponding parameters in $\boldsymbol{\Phi}$, $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_R)$, $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_R)$. The rank will then be shrunk to a smaller number of active columns. If the algorithm does not detect any inactive column, we first sample a new column in $\boldsymbol{\Phi}$, a new entry in $\boldsymbol{\delta}$ and subsequently compute the new entry in $\boldsymbol{\tau}$. A new column in $\boldsymbol{B}$ will also be sampled using these newly-sampled hyperparameters.

3.3 Post-Processing Procedure

The interweaving algorithm allows Markov chains to improve mixing, but it does not completely solve the indeterminacy of the CP decomposition, which is the origin of the non-convergence of Markov chains. Therefore, we propose a post-processing procedure to identify margins *a posteriori*. Note that there exist methods to identify margins *a priori*, such as the methods described in Section 2.3.2. We opt to maintain an unrestricted tensor decomposition because it can incorporate the increasing shrinkage property of the MGP, enabling us to infer the rank.

The procedure proposed is inspired by the Match-Sign-Factor algorithm in the R package **infinitefactor** (Poworoznek et al., 2021). This algorithm performs a greedy search to rotate factor loadings and factors in factor models, and we apply a variant of this algorithm to TVARs. Our algorithm is presented in Algorithm 6, along with an explanation divided into two parts: (1) solve column permutations by the label-matching method (up to line 12); (2) solve sign-switching issues by the sign-matching method.

Column permutations in $\boldsymbol{B}$ are equivalent to those in $\boldsymbol{B}_3$, so if we solve the equivalent issue in $\boldsymbol{B}_3$, we will automatically solve column permutations in $\boldsymbol{B}$. There are analogous equivalences related to $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, but the empirical finding in Figure B.1 shows that the label matching related to $\boldsymbol{B}_3$ gives the best mixing results in the simulation study. The label matching needs a *pivot* matrix $\boldsymbol{B}_3^{(\text{pivot})}$ as a template to align $\boldsymbol{B}_3$ sampled in each iteration, i.e., columns in $\boldsymbol{B}_3$ after label being matched will have the same order as that of columns in $\boldsymbol{B}_3^{(\text{pivot})}$. Following Poworoznek et al. (2021), $\boldsymbol{B}_3^{(\text{pivot})}$ is the one with the median of the condition number $\kappa = \sigma_{\text{max}}(\boldsymbol{B}_3)$, where $\sigma_{\text{max}}(\boldsymbol{B}_3)$ is the maximal singular value of $\boldsymbol{B}_3$.

After choosing the pivot, we compute the Euclidean distance between columns in $\boldsymbol{B}_3$ in each iteration and $\left(\boldsymbol{B}_3^{(\text{pivot})}, -\boldsymbol{B}_3^{(\text{pivot})}\right)$, and store the distances into an R-by-$2R$ distance matrix $\boldsymbol{\Theta}$ with row and column indices corresponding to columns in $\boldsymbol{B}_3$ and $\left(\boldsymbol{B}_3^{(\text{pivot})}, -\boldsymbol{B}_3^{(\text{pivot})}\right)$, respectively. As shown in Algorithm 6, a greedy algorithm then starts from the lowest Euclidean distance to align the corresponding column in $\boldsymbol{B}_3$ to that in $\boldsymbol{B}_3^{(\text{pivot})}$ or -$\boldsymbol{B}_3^{(\text{pivot})}$, and these columns will not be matched again. The label matching is finished after repeating the procedure for R times.

Next, we explain the sign-matching method. For $j \in [2]$, $r \in [R]$, we determine whether to flip the sign of $\boldsymbol{B}_{j,(\cdot,r)}$ by comparing its distances to both $\boldsymbol{B}_{j,(\cdot,r)}^{(\text{pivot})}$ and -$\boldsymbol{B}_{j,(\cdot,r)}^{(\text{pivot})}$. The general guideline for flipping signs in $\boldsymbol{B}_{3,(\cdot,r)}$ is to do so only if this procedure identifies the tensor, i.e., the tensors before and after sign-matching are the same. If not, we leave the sign unchanged.

3.4 Simulation Results

3.4.1 Data and Implementation

We assess the merits of inferring ranks using the MGP and the adaptive inferential scheme in 3.4.2, compared to the M-DGDP (Guhaniyogi et al., 2017) prior commonly used in the tensor literature. Section 3.4.3 shows that the interweaving strategy can improve the mixing of mar-

Algorithm 6 Match labels and signs

```
Find a pivot matrix $\boldsymbol{B}_3^{(\text{pivot})}$ and its corresponding tensor matrix $\boldsymbol{B}^{(\text{pivot})}$
for each iteration do
    Compute the $R$-by-$2R$ distance matrix $\boldsymbol{\Theta}$
    for $r = 1, \dots, R$ do
        Find $(r_1^*, r_2^*) = \underset{r_1, r_2}{\text{argmin}}\, \boldsymbol{\Theta}_{r_1, r_2}$
        if $r_2^* \leq R$ then
            Match the $r_1^*$-th column in $\boldsymbol{B}$ to the $r_2^*$-th column in $\boldsymbol{B}^{(\text{pivot})}$.
            Change the $r_1$-th row, $r_2$-th and $(R + r_2)$-th columns in $\boldsymbol{\Theta}$ to infinity.
        else
            Match the $r_1^*$-th column in $\boldsymbol{B}$ to the $(r_2^* - R)$-th column in $\boldsymbol{B}^{(\text{pivot})}$.
            Change the $r_1$-th row, $(r_2 - R)$-th and $r_2$-th columns in $\boldsymbol{\Theta}$ to infinity.
        end if
        for $j = 1, 2$ do
            Compute distance $d_1 = d\left(\boldsymbol{B}_{j,(\cdot,r)}, \boldsymbol{B}_{j,(\cdot,r)}^{(\text{pivot})}\right)$ and $d_2 = d\left(\boldsymbol{B}_{j,(\cdot,r)}, -\boldsymbol{B}_{j,(\cdot,r)}^{(\text{pivot})}\right)$
            if $d_1 \leq d_2$ then
                Keep signs in $\boldsymbol{B}_{j,(\cdot,r)}$. Record $\text{ind}_{j,r} = 1$
            else
                Flip signs in $\boldsymbol{B}_{j,(\cdot,r)}$. Record $\text{ind}_{j,r} = -1$
            end if
        end for
        if $\text{ind}_{1,r}\text{ind}_{2,r} = 1$ then
            Keep the signs in $\boldsymbol{B}_{3,(\cdot,r)}$
        else
            Flip the signs in $\boldsymbol{B}_{3,(\cdot,r)}$
        end if
    end for
end for
```

gins, and the post-processing procedure identifies the margins. We will leave the comparison of predictive performance to the real data example. The data generated in this simulation study includes three scenarios with different combinations of the number of time series and rank (N, R): $(10, 3)$, $(20, 5)$, and $(50, 10)$. The lag order is $P = 3$. We assume that the true rank increases with the number of time series. Kolda and Bader (2009) and the reference therein summarized the CP ranks of some specific third-order tensors, but the rank of a tensor applied in a VAR with lag order exceeding 2 was not specified. Only an upper bound of the CP rank is available, which is $\min(N^2, NP)$.

In each scenario, we generate 25 data sets, each with 200 observations from a VAR(3) model with the model parameters independently generated. The coefficient matrix of each model is the mode-1 matricization of a tensor from a CP decomposition, and the covariance matrix is an identity matrix. Margins of the CP decomposition follow uniform distributions with different parameters, see Table 3.1 for more details. All time series are checked for stationarity via the Dickey-Fuller test and the Kwiatkowski–Phillips–Schmidt–Shin test with a significance level set at 5%[24]. All data sets are regarded as stationary with statistical significance.

	(10,3)	(20,5)	(50,10)
B_1	U(-1,1)	U(-1,1)	U(-1,1)
B_2	U(-1,1)	U(-1,1)	U(-0.6,0.6)
$B_{3,(1,\cdot)}$	U(-1,1)	U(-1,1)	U(-0.6,0.6)
$B_{3,(2,\cdot)}$	U(-0.5,0.5)	U(-0.2,0.2)	U(-0.2,0.2)
$B_{3,(3,\cdot)}$	U(-0.1,0.1)	U(-0.1,0.1)	U(-0.1,0.1)

Table 3.1: Uniform distributions of margins in different locations indicated by rows and different combinations of N and R indicated by columns.

We apply the MGP to both simulation experiments by setting $\nu = 3$ as shown in Bhattacharya and Dunson (2011), $\gamma_1 = 10^{-3}$, and $\gamma_2 = 0.9$. A table illustrating the sensitivity to the choice of γ_1 and γ_2 is available in Appendix B.2.1. Our chosen combination of γ_1 and γ_2 gives the most parsimonious model and the narrowest 90% credible interval of inferred rank. Apart from the MGP, we briefly introduce the M-DGDP prior, which is a global-local shrinkage prior proposed for tensor margins with the following expression,

$$\boldsymbol{\beta}_j^{(r)} \sim \mathcal{N}\left(\mathbf{0}, (\phi_r\tau)\boldsymbol{W}_{jr}\right), \; w_{jr,k} \sim \mathcal{E}\left(\lambda_{jr}/2\right), \; \lambda_{jr} \sim \mathcal{G}\left(a_\lambda, b_\lambda\right),$$

$$\boldsymbol{\Phi} = (\phi_1, \ldots, \phi_R)' \sim \mathcal{D}\left(\alpha, \ldots, \alpha\right), \; \tau \sim \mathcal{G}\left(a_\tau, b_\tau\right),$$

where $\boldsymbol{W}_{jr} = \text{diag}\left(w_{jr,1}, \ldots, w_{jr,I_j}\right)$, for $I_1 = I_2 = N$ and $I_j = P$ in our case. α is uniformly distributed on a grid with values equally placed on $[R^{-3}, R^{-0.01}]$, and R is the rank set in advance. We follow the same setting of hyperparameters as in Guhaniyogi et al. (2017), i.e., $a_\lambda = 3$ and $b_\lambda = \sqrt[6]{a_\lambda}$, $a_\tau = R\alpha$, $b_\tau = \alpha\sqrt[3]{R}$.

For both priors, the initialization of the rank is $\lceil 5\log(N) \rceil$, but the adaptive inferential scheme is only applied when using the MGP after the iteration reaches 200 in the burn-in period. For the M-DGDP, the rank is determined *a posteriori* by removing negligible margins as in

[24] Another method of checking stationarity is to examine whether the spectral radius of the companion matrix is smaller than 1, i.e., whether the maximum eigenvalue of $\begin{pmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 & \boldsymbol{A}_3 \\ \boldsymbol{I}_N & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_N & \mathbf{0} \end{pmatrix}$ is smaller than 1.

Algorithm 5. We implement all simulations with Intel(R) Xeon(R) Gold 6140 CPU 2.30GHzr and R 4.2.0.

3.4.2 Rank Selection

The first simulation assesses our approach to infer the rank R. Both samplers with MGP and M-DGDP were run for 10,000 iterations after 10,000 burn-in and incorporated the interweaving strategy. We record the performance of MGP and M-DGDP in Table 3.2 including four metrics: (1) mean squared error (MSE) of the coefficient matrix for coefficient accuracy; (2) averaged effective sample size (ESS) of coefficients for sampling efficiency; (3) averaged rank inferred (R) for rank accuracy; and (4) approximate running time for computational efficiency.

According to Table 3.2, both models estimate coefficient matrices with similar accuracy under the MSE. The MGP can infer most ranks lower than the true ones[25]. While the M-DGDP has a competitive performance in inferring the ranks, the adaptive shrinkage algorithm proposed for the MGP accelerates computation since the running time of the MGP grows more slowly with N and R compared to the growth rate of the M-DGDP. This leads to a large difference if $N = 50$ and $R = 10$, where the inference with the MGP runs more than 5 times faster than the M-DGDP. The MGP also explores coefficient posteriors more efficiently, as suggested by the ESS per unit running time.

(N, R)	method	MSE	R	ESS	running time (hr)
(10,3)	MGP	0.006	4	3977.539	0.45
	M-DGDP	0.006	3	3938.573	1.16
(20, 5)	MGP	0.008	4	2657.043	0.585
	M-DGDP	0.008	5	2644.262	2.60
(50, 10)	MGP	0.006	7	2125.425	2.52
	M-DGDP	0.006	10	2315.662	13.34

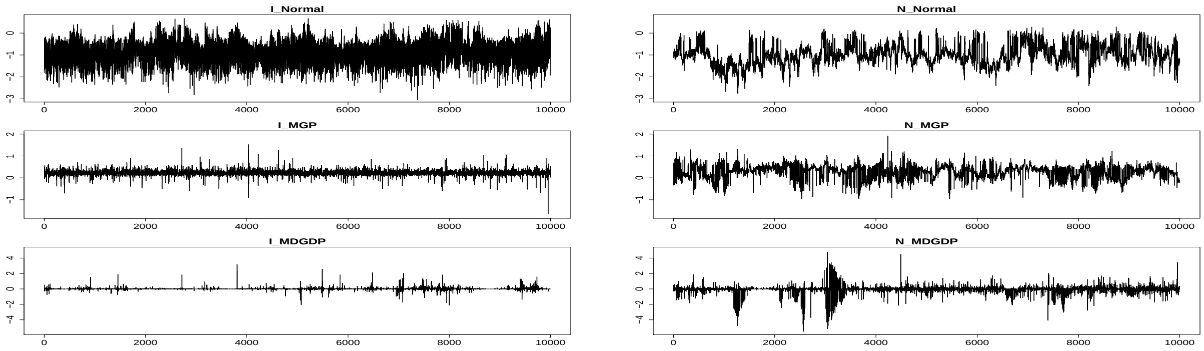
Table 3.2: Performance of MGP and M-DGDP in 25 simulations for different dimensionality combinations.

[25] A possible reason is that the data generating process (DGP) does not specify an increasing shrinkage property to margins. As the M-DGDP prior shrinks only a subset of columns toward zero while assigning the remaining columns the same level of shrinkage, this prior is better aligned with the DGP and therefore estimates the rank more accurately than the MGP does. Although the MGP prior tends to underestimate the rank, this unnecessarily implies that the MGP is inferior to the M-DGDP prior. Firstly, the MGP leads to more parsimonious models without compromising the accuracy of coefficient estimation. Secondly, if we choose the DGP to have an increasing shrinkage property, the MGP may estimate the ranks more accurately.

3.4.3 Quality of Markov Chains

The second simulation investigates the quality of Markov chains, i.e., whether the interweaving strategy and the post-processing procedure contribute to the mixing and convergence of Markov chains. We choose three prior settings (standard normal, MGP, M-DGDP) to infer margins with/without interweaving. The burn-in period still has 10,000 iterations, but we change the number of iterations after burn-in to 100,000 to demonstrate results with longer chains.

We first focus on the interweaving strategy by conducting the post-processing procedure to both samples with/without interweaving. To give an insight into the effect of interweaving to the mixing, Figure 3.1 shows trace plots of the margin $\boldsymbol{\beta}_{1,1}^{(1)}$ when $N = 10$ and $R = 3$ based on different prior settings with/without interweaving. Even though we used the label- and sign-matching methods, trace plots without interweaving still suffer from the mixing problem, while the interweaving strategy substantially improves mixing. The autocorrelations (ACFs) of all draws of $\boldsymbol{\beta}_{1,1}^{(1)}$ after the burn-in period, see Figure 3.2, also support the merit of the interweaving strategy. The ACFs from the interweaving strategy decay quickly, with only the one from the standard normal prior showing non-negligible values by 100 lags. All three of these ACFs without interweaving remain large for many lags[26].

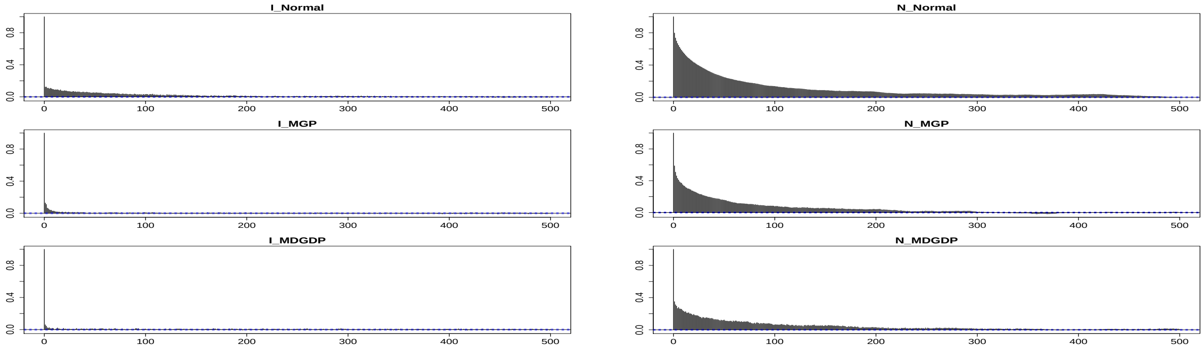


(a) With interweaving **(b)** Without interweaving

Figure 3.1: Trace plots of the first 10,000 draws of $\boldsymbol{\beta}_{1,1}^{(1)}$ in $N = 10$, $R = 3$ scenario after burn-in period. The inferential scheme adopts standard normal (top), MGP (middle), and M-DGDP (bottom) as priors and applies with (left panel) and without (right panel) interweaving strategy.

We follow the procedure in Kastner et al. (2017) to compute the inefficiency factor (IF) of each margin in different scenarios and prior settings. A smaller IF means that the sampling of a

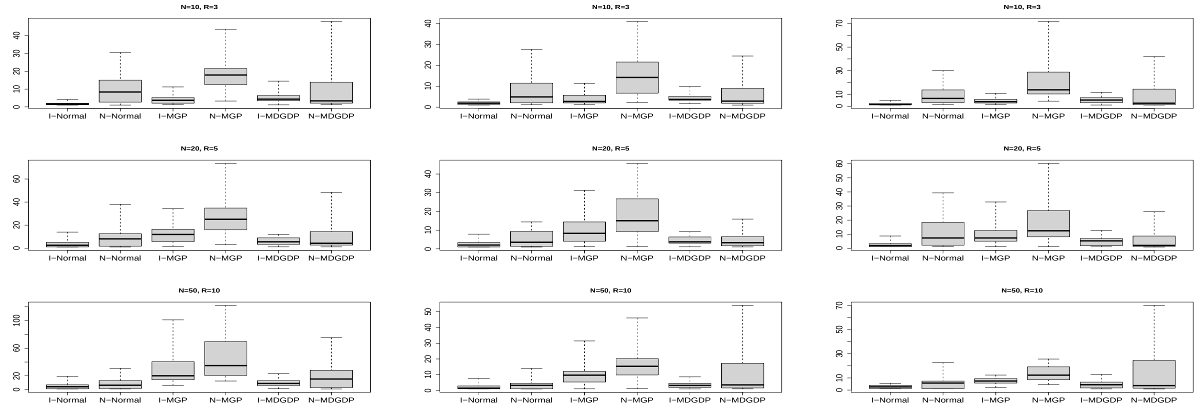
[26] Although we focus on the mixing of the Markov chains from a single run, another way to demonstrate the convergence of MCMC samples is to run the inference multiple times and overlay the trace plots.



(a) With interweaving **(b)** Without interweaving

Figure 3.2: Autocorrelations of $\boldsymbol{\beta}_{1,1}^{(1)}$ in $N = 10$, $R = 3$ scenario after the burn-in period. The inferential scheme adopts standard normal (top), MGP (middle), and M-DGDP (bottom) as priors and applies with (left panel) and without (right panel) interweaving strategy.

parameter is more efficient. Figure 3.3 displays boxplots of IFs where each panel corresponds to a scenario with a combination of (N, R) and $\boldsymbol{B}_j$. Each boxplot contains 25 data points from the 25 simulation data sets. Each data point in a boxplot is the IF of the (1,1) entry of $\boldsymbol{B}_j$, for $j \in [3]$, inferred from one data set. We exclude outliers because there are only a handful of them, and this exclusion allows us to focus on the medians and quantiles of IFs. Overall, most IFs with interweaving have lower median values and less variation than their counterparts without interweaving.



(a) $\boldsymbol{\beta}_{1,1}^{(1)}$ **(b)** $\boldsymbol{\beta}_{2,1}^{(1)}$ **(c)** $\boldsymbol{\beta}_{3,1}^{(1)}$

Figure 3.3: Boxplots of inefficiency factor of the 1-1 entry of $\boldsymbol{B}_1$ (left), $\boldsymbol{B}_2$ (middle) and $\boldsymbol{B}_3$ (right) from different scenarios: $(N, R) = (10, 3)$ (top), $(N, R) = (20, 5)$ (middle) and $(N, R) = (50, 10)$ (bottom). Inferential schemes with and without interweaving are represented as "I-" and "N-", respectively, followed by a prior setting.

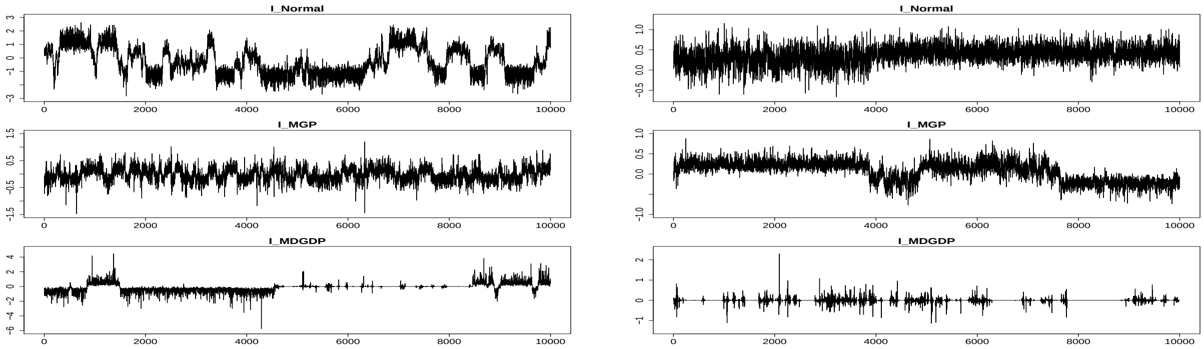
We then use the stable Gelman-Rubin method (Vats and Knudson, 2021) to diagnose the

N=10, R=3	Interweaving	Non-interwoven	**N=20, R=5**	Interweaving	Non-interwoven	**N=50, R=10**	Interweaving	Non-interwoven
Normal	1.000	0.847	Normal	0.996	0.916	Normal	0.996	0.978
MGP	0.998	0.866	MGP	0.986	0.740	MGP	0.940	0.770
MDGDP	0.996	0.871	MDGDP	0.998	0.819	MDGDP	0.989	0.858

Table 3.3: Averaged proportions of margins which are convergent according to stable Gelman-Rubin Statistics.

convergence of the margin Markov chains. The reason why we apply the stable Gelman-Rubin instead of the Gelman-Rubin (Gelman and Rubin, 1992) is twofold: (1) the Gelman-Rubin is suitable when the simulation has multiple Markov chains for each parameter, while our simulation only has one Markov chain for each parameter. The stable Gelman-Rubin can be applied to both multiple and single Markov chains; (2) The conventional Gelman-Rubin threshold of 1.1 implies an approximation of ESS of 5 according to Vats and Knudson (2021), and the authors propose a threshold depending on the parameter dimension and a significance level. The results are presented in Table 3.3, where each cell is the averaged proportion of margins of which the Markov chains are determined as convergent. Overall, the algorithm with interweaving achieves over 90% convergent Markov chains in all scenarios and with all prior choices. All proportions are higher based on the results from the interweaving algorithm compared to the non-interwoven ones. We also include the Geweke diagnostic (Geweke, 1991) in Appendix B.2.1, with most interweaving results having a better convergence performance.

Lastly, we demonstrate the necessity of the post-processing procedure. Figure 3.4 displays trace plots of the whole draws (with thinning of 10) of two selected margins inferred with the interweaving strategy, and we exclude the post-processing procedure at this time. All three panels in Figure 3.4a and the middle panel in Figure 3.4b have sign-switching issues. If we do not match signs, the interpretation of margins will be infeasible because the posterior mode or mean of some margins would be zero, but they should be non-zero. The top panel in Figure 3.4b provides evidence of column permutations, with the sample mean moving from 0 to 0.5. The bottom panel in Figure 3.4b has neither sign switching nor column permutations, but the M-DGDP does not guarantee convergence only with the interweaving strategy due to the evidence provided in Figure 3.4a.



(a) Scenario $(N, R) = (10, 3)$

(b) Scenario $(N, R) = (20, 5)$

Figure 3.4: Trace plots of $\beta_{1,1}^{(1)}$ in $N = 10$, $R = 3$ scenario (left) and $\beta_{1,2}^{(1)}$ in $N = 20$, $R = 5$ scenario (right) after burn-in period. The inferential scheme adopts standard normal (top), MGP (middle), and M-DGDP (bottom) as priors and applies with the interweaving strategy.

3.5 Real Data Application

3.5.1 Data and Implementation

We use the U.S. macroeconomic data extracted from Federal Reserve Economic Data[27] (McCracken and Ng, 2020) to assess the utility of TVARs. The data spans from 1959Q1 to 2019Q4, and are transformed to stationarity and standardized to have mean zero and variance one to avoid scaling issues. We construct medium-scale and large-scale data sets by selecting 20 and 40 variables, respectively, as referred to in Korobilis and Pettenuzzo (2019). A full description of the variables selected and their transformations can be found in Appendix B.3. The selected 40 variables can be divided into 8 categories: (i) output and income, (ii) consumption, orders and inventories, (iii) labor market, (iv) prices, (v) interest rate, (vi) money and credit, (vii) stock market and (viii) exchange rate. Since no variables in the categories of money and credit, and the stock market are selected into the medium-scale data set, we also construct an alternative 20-variable data set that contains variables from all 8 categories. We use this alternative data set to examine the robustness of forecasting performance with results available in Appendix B.2.2. Since the decomposition of the covariance matrix $\boldsymbol{\Omega}_t$ has a lower triangular matrix $\boldsymbol{H}$ in the model, the order of time series matters. We follow Bernanke et al. (2005) by splitting time series into slow, fast groups and the Federal Funds Rate (FEDFUNDS). The slow group contains variables that respond to a shock of FEDFUNDS with a lag, and variables in the fast group respond to it contemporaneously. The order is slow variables, FEDFUNDS, and fast variables.

[27] The data is available at https://research.stlouisfed.org/econ/mccracken/fred-databases/.

For each data set, we follow Korobilis and Pettenuzzo (2019) to estimate various VAR models with 5 lags[28]. TVARs with and without the additional own-lag matrix $\boldsymbol{D}$ are denoted as tensor MGP own-lag and tensor MGP, respectively. For these two TVARs, we use the same choice of γ_1 and γ_2 as in the simulation study. For competitors, we include standard VARs with the hierarchical Minnesota (Giannone et al., 2015), horseshoe (Carvalho et al., 2009), and a specification of normal-gamma (NG) prior introduced to VARs by Huber and Feldkircher (2019). The detail of the first two priors can be found in Section 2.1.4, and Section 3.2 described the last prior. For the hierarchical Minnesota prior defined in (2.10), λ_1 and λ_2 have a prior $\mathcal{G}$(0.01,0.01) and are inferred using a Metropolis-Hastings step. We follow Follett and Yu (2019) and Huber and Feldkircher (2019) to infer the hyperparameters defined in the horseshoe and the NG. Priors of $\boldsymbol{H}$ and stochastic volatility $\boldsymbol{S}_t$, for $t = 1, \ldots, T$, are the same for all models. The MCMC sampler runs 10,000 iterations after the 10,000 burn-in period.

Note that the decomposition of $\boldsymbol{\Omega}_t$ employs a triangular system due to the lower triangular matrix $\boldsymbol{H}$, which might lead to the ordering issue when estimating the parameters. This issue has been discussed in Carriero et al. (2019), Chan et al. (2024), and Arias et al. (2023), among others. Thus, we also provide the forecasting performance of which we apply a non-restrictive matrix $\boldsymbol{H}$, as defined in Chan et al. (2024). The results and further discussion about this order-invariant model are available in Appendix B.2.2.

3.5.2 Forecasting Results

Before delving into the evaluation of forecasting performance, we compare TVARs and standard VARs with the NG prior in computational time and number of parameters (margins or coefficients) inferred. As shown in Table 3.4, fewer parameters were inferred within the TVAR framework, leading to the reduced computing time of this framework compared to standard VARs. For the medium-scale data set, TVARs require at least six times fewer parameters than standard VARs. Similarly, for the large-scale data set, TVARs infer fewer than 10% of the parameters compared to those inferred from standard VARs. In term of running time, tensor and

[28] Many macroeconomic studies (see Bernanke et al. (2005), Huber and Feldkircher (2019), and Korobilis and Pettenuzzo (2019), among others) use lag lengths that span slightly more than one year of past data (5 lags for quarterly data, and 13 lags for monthly data). This is because impulse responses to macroeconomic shocks often peak beyond the one-year horizon; for instance, the response of GDP to a federal funds rate shock in Christiano et al. (1999) reaches its maximum after about five quarters. This lag order choice allows the model to effectively capture this kind of peaks.

standard VARs take a similar amount of time to infer the medium-scale data set, but the former requires approximately one-third of the time taken by the latter when we switch to the large-scale data set. The inference using tensor MGP is faster than tensor MGP own-lag because the latter requires additional time to infer the own-lag matrix. Note that the code for both TVARs and standard VARs has been accelerated by the Rcpp package.

	Number of Parameters		Runnning Time (hr)	
	Medium	Large	Medium	Large
Tensor MGP	187.18	257.36	0.95	3.14
Tensor MGP Own-lag	272.19	456.18	1.07	3.28
Standard VAR	2000	8000	1.30	10.39

Table 3.4: Averaged number of parameters and running time of tensor MGP, tensor MGP own-lag and standard VARs with the NG prior.

We follow the expanding window procedure to assess the forecasting performance of our models. Specifically, we first fit each VAR model with the historical data from 1959Q1 to 1984Q4, then get 1-, 2-, and 4-step-ahead forecasts for 1985Q1, 1985Q2, and 1985Q4, respectively. Next, we expand the historical data with the endpoint at 1985Q1 and conduct the multi-step-ahead forecasting again. This procedure is repeated iteratively and stops after conducting the 1-step-ahead forecast of 2019Q4.

We evaluate the forecasting performance of TVARs and standard VARs with both joint and marginal results. For the marginal ones, we select 7 variables which are salient to the U.S. economy, as shown in Table 3.5 and 3.6. The metrics for the forecasting evaluation are mean squared forecast error (MSFE), mean absolute error (MAE), and averaged log predictive likelihood (ALPL), see Appendix B.2.2 for mathematical expressions. All marginal metrics are relative to a standard VAR with a flat prior, taking the 7 time series selected as responses.

Results about point forecasts evaluated by MSFE and MAE can be found in Appendix B.2.2. Overall, TVARs achieve better joint and marginal performance than standard VARs. Table 3.5 and Table 3.6 present density forecasting performance from the medium and large data sets. TVARs have competitive performance when making joint density forecasts. They also outperform standard VARs in marginal forecasts since they are the best models in 12 and 13 out of 21 cases for medium and large data sets, respectively. Forecasts of FEDFUNDS, GDP, and UNRATE are more favorable when using TVARs, while standard VARs have better performance in forecasting PAYEMS and GDPDEFL. In comparing the performance of the two models within TVARs, Tensor MGP own-Lag demonstrates superior results to tensor MGP. If we focus on individual models in standard VARs, the hierarchical Minnesota prior is the best among these three priors. The superior performance of tensor MGP Own-lag and the hierarchical Minnesota highlights the importance of own-lag effect in economic data. When comparing each marginal evaluation in these two tables, most ALPLs in Table 3.6 are larger than those in Figure 3.5, indicating that the high-dimensionality is advantageous for the marginal forecasting.

Model	Horizon	ALPL							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	-16.378	0.170	0.151	0.637	0.177	0.150	0.124	0.160
	2	**-17.820**	0.416	0.227	0.634	0.240	0.284	0.141	0.128
	4	**-19.460**	0.671	**0.179**	0.498	0.196	0.306	0.110	0.077
Tensor MPG Own-lag	1	-16.184	**0.190**	0.147	**0.682**	**0.191**	**0.172**	0.133	0.163
	2	-17.852	0.424	**0.229**	**0.656**	**0.249**	0.289	0.144	0.127
	4	-19.567	0.702	0.171	**0.526**	**0.207**	**0.310**	**0.113**	0.081
Minnesota	1	**-15.921**	0.129	**0.183**	0.519	0.141	0.164	**0.181**	**0.187**
	2	-18.126	**0.443**	0.210	0.507	0.202	**0.301**	0.134	**0.141**
	4	-19.897	**0.754**	0.142	0.379	0.152	0.291	0.086	0.082
NG	1	-16.463	0.126	0.126	0.640	0.131	0.153	0.149	0.162
	2	-18.277	0.402	0.193	0.588	0.183	0.272	0.130	0.126
	4	-19.995	0.724	0.140	0.448	0.170	0.281	0.096	0.081
Horseshoe	1	-17.333	-0.164	0.090	0.633	0.112	0.048	0.168	0.152
	2	-18.394	0.214	0.199	0.626	0.162	0.223	**0.146**	0.130
	4	-19.464	0.632	0.141	0.495	0.156	0.257	0.104	**0.108**

Table 3.5: ALPL of joint and marginal variables using the medium-scale data set. The best forecasts are in bold.

3.5.3 Interpretation

Since tensor MGP own-lag performs better than tensor MGP, we demonstrate how to interpret a TVAR by fitting it with the whole large-scale data set (N= 40). The TVAR infers a rank of 3, reducing the number of parameters in the coefficient matrix from 8,000 (standard VAR(5)) to 455.

A TVAR can be interpreted as a factor model with observable factors according to (3.4), Figure 3.5 shows these factors are consistent with recession periods reported by the National Bureau of Economic Research (NBER) (available on https://fred.stlouisfed.org/series/USRECQ). The first factor has wider credible intervals during or after the NBER recession periods. The second factor peaks during these recession periods and has rela-

Model	Horizon	ALPL							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
	1	-24.520	0.078	0.126	0.670	0.151	0.135	0.103	**0.178**
Tensor MGP	2	-29.790	0.401	0.231	**0.686**	0.213	0.286	0.133	0.151
	4	-33.847	0.703	0.171	0.532	**0.172**	**0.353**	0.108	0.099
	1	**-23.809**	**0.101**	0.143	**0.688**	**0.159**	**0.172**	0.116	0.175
Tensor MPG Own-lag	2	-30.338	0.389	0.240	0.673	**0.217**	0.298	0.138	0.151
	4	-35.631	0.686	**0.176**	**0.533**	0.171	0.334	**0.113**	**0.101**
	1	-26.576	-0.073	**0.147**	0.534	0.103	0.035	**0.133**	0.174
Minnesota	2	**-29.600**	0.330	**0.252**	0.570	0.173	0.212	**0.148**	**0.162**
	4	**-32.545**	0.736	0.175	0.445	0.157	0.243	0.105	0.095
	1	-28.455	0.081	0.133	0.518	0.107	0.167	0.130	0.172
NG	2	-32.823	**0.421**	0.218	0.518	0.163	**0.316**	0.136	0.145
	4	-36.715	**0.793**	0.154	0.386	0.159	0.312	0.104	0.085
	1	-27.915	0.064	0.129	0.584	0.114	0.138	0.124	**0.178**
Horseshoe	2	-31.462	0.408	0.238	0.580	0.178	0.295	0.144	0.158
	4	-34.874	0.784	0.165	0.431	0.165	0.299	0.104	0.097

Table 3.6: ALPL of joint and marginal variables using the large-scale data set. The best forecasts are in bold.

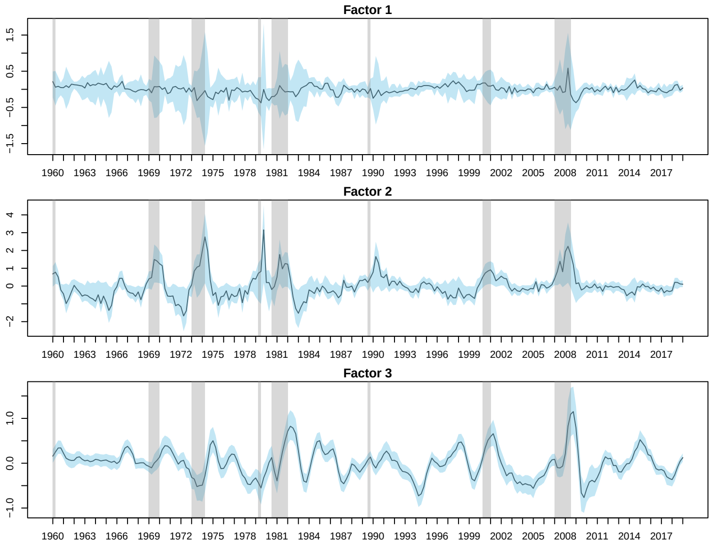
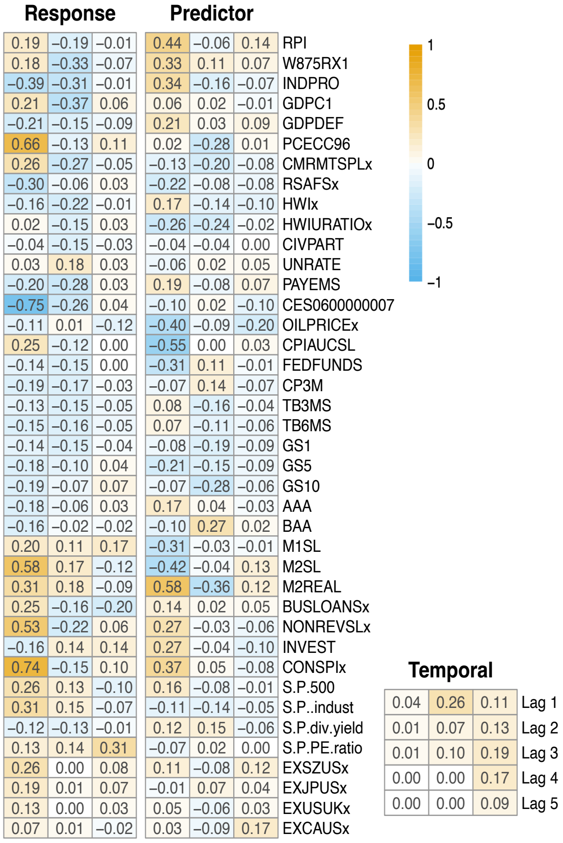


Figure 3.5: Time series plots of factors with median (solid line) and 80% credible interval (blue shade). The vertical grey shades correspond to the U.S. recession periods. The factors are derived from the inferential results of tensor MGP own-lag.

Factor 1	Factor 2	Factor 3
M2REAL (0.44)	PAYEMS (-0.84)	S&P PE ratio (0.62)
NONREVSLx (0.41)	UNRATE (0.76)	M2REAL (0.41)
CONSPIx (0.40)	INDPRO (-0.72)	BUSLOANSx (-0.31)
BUSLOANSx (0.32)	HWIURATIOx (-0.69)	INVEST (0.30)
PCECC96 (0.26)	HWIx (-0.62)	M2SL (0.28)

Table 3.7: Variables with the top 5 correlations with the factors.

tively high values during the recession of 1960-1961 and the dot-com bubble in the early 2000s. The third factor peaks after recession periods, and the reason will be explained later based on Figure 3.6. Furthermore, we present the variables that exhibit the five highest magnitudes of correlation with these three factors in Table 3.7. The first factor shows a high correlation with variables from the money and credit category, while the second factor is highly correlated to the variables from the labor market and industrial production. The correlations associated with PAYEMS and UNRATE are reversed, indicating that the second factor is positively linked to unemployment. Proceeding to the third factor, M2REAL and BUSLOANx are both found in the first and third columns in Table 3.7, but we consider the third factor to bear a connection with the financial market due to its high correlation with the S&P price earning ratio. It may seem surprising that none of the factors shows a strong connection with interest rates, but all three factors have a non-negligible correlation to interest rates according to the full correlation displayed in Appendix B.2.3.



Response			Predictor			
0.19	−0.19	−0.01	0.44	−0.06	0.14	RPI
0.18	−0.33	−0.07	0.33	0.11	0.07	W875RX1
−0.39	−0.31	−0.01	0.34	−0.16	−0.07	INDPRO
0.21	−0.37	0.06	0.06	0.02	−0.01	GDPC1
−0.21	−0.15	−0.09	0.21	0.03	0.09	GDPDEF
0.66	−0.13	0.11	0.02	−0.28	0.01	PCECC96
0.26	−0.27	−0.05	−0.13	−0.20	−0.08	CMRMTSPLx
−0.30	−0.06	0.03	−0.22	−0.08	−0.08	RSAFSx
−0.16	−0.22	−0.01	0.17	−0.14	−0.10	HWIx
0.02	−0.15	0.03	−0.26	−0.24	−0.02	HWIURATIOx
−0.04	−0.15	−0.03	−0.04	−0.04	0.00	CIVPART
0.03	0.18	0.03	−0.06	0.02	0.05	UNRATE
−0.20	−0.28	0.03	0.19	−0.08	0.07	PAYEMS
−0.75	−0.26	0.04	−0.10	0.02	−0.10	CES0600000007
−0.11	0.01	−0.12	−0.40	−0.09	−0.20	OILPRICEx
0.25	−0.12	0.00	−0.55	0.00	0.03	CPIAUCSL
−0.14	−0.15	0.00	−0.31	0.11	−0.01	FEDFUNDS
−0.19	−0.17	−0.03	−0.07	0.14	−0.07	CP3M
−0.13	−0.15	−0.05	0.08	−0.16	−0.04	TB3MS
−0.15	−0.16	−0.05	0.07	−0.11	−0.06	TB6MS
−0.14	−0.15	−0.04	−0.08	−0.19	−0.09	GS1
−0.18	−0.10	0.04	−0.21	−0.15	−0.09	GS5
−0.19	−0.07	0.07	−0.07	−0.28	−0.06	GS10
−0.18	−0.06	0.03	0.17	0.04	−0.03	AAA
−0.16	−0.02	−0.02	−0.10	0.27	0.02	BAA
0.20	0.11	0.17	−0.31	−0.03	−0.01	M1SL
0.58	0.17	−0.12	−0.42	−0.04	0.13	M2SL
0.31	0.18	−0.09	0.58	−0.36	0.12	M2REAL
0.25	−0.16	−0.20	0.14	0.02	0.05	BUSLOANSx
0.53	−0.22	0.06	0.27	−0.03	−0.06	NONREVSLx
−0.16	0.14	0.14	0.27	−0.04	−0.10	INVEST
0.74	−0.15	0.10	0.37	0.05	−0.08	CONSPIx
0.26	0.13	−0.10	0.16	−0.08	−0.01	S.P.500
0.31	0.15	−0.07	−0.11	−0.14	−0.05	S.P..indust
−0.12	−0.13	−0.01	0.12	0.15	−0.06	S.P.div.yield
0.13	0.14	0.31	−0.07	0.02	0.00	S.P.PE.ratio
0.26	0.00	0.08	0.11	−0.08	0.12	EXSZUSx
0.19	0.01	0.07	−0.01	0.07	0.04	EXJPUSx
0.13	0.00	0.03	0.05	−0.06	0.03	EXUSUKx
0.07	0.01	−0.02	0.03	−0.09	0.17	EXCAUSx

Temporal			
0.04	0.26	0.11	Lag 1
0.01	0.07	0.13	Lag 2
0.01	0.10	0.19	Lag 3
0.00	0.00	0.17	Lag 4
0.00	0.00	0.09	Lag 5

Figure 3.6: Posterior mean of response, predictor, and temporal loadings inferred from tensor MGP own-lag.

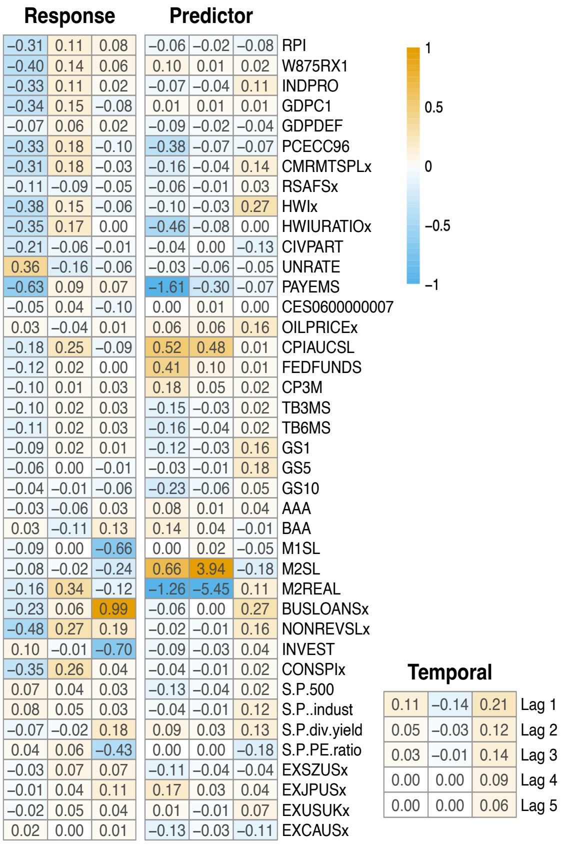
Next, we use Figure 3.6 to answer two questions: 1) which lagged time series contribute to the factors; 2) what is the effect from factors to responses. Figure 3.6 depicts the posterior mean of response, predictor, and temporal loadings. Larger margin magnitudes are associated with more deeply saturated hues.

The first question is answered by the predictor and temporal loading. The columns with the same index in these two loadings reveal how the corresponding factor is constructed. For the first factor representing money and credit, we inspect the top 5 margin magnitudes (M2REAL, CPIAUCSL, RPI, M2SL, and OILPRICEx) in the first column of the predictor loading, and show that price is the main category contributing to this factor. The negative margins of CPIAUCSL and OILPRICEx indicate that prices have a negative effect on the first factor. This conclusion is further strengthened by the opposite signs of M2REAL and M2SL margins since M2SL drops while M2REAL rises with decreasing prices. Additionally, the positive RPI margin, which is adjusted by inflation, supports this conclusion. In the first column of temporal loading, the first lag is suggested to be the most important one because its magnitude is the largest within the corresponding column. Combined with the findings from the predictor and temporal loading, the first factor is formed by the prices one quarter ago. We follow a similar method to investigate the formation of the second factor and get the following findings. First, a decline in real M2 money supply (M2REAL) and personal consumption expenditures (PCECC96) contributes to an increase in this factor about unemployment. Second, the factor grows with the increase of credit risk because of the opposite signs of BAA and GS10, representing the spread between the corresponding two yields. Akin to the formation of the first factor, the first lag exhibits the most significant contribution to the formation of the second factor. Lastly, we focus on the columns corresponding to the third factor and find two differences compared to other columns: 1) margins with relatively high magnitudes are related to the financial market, for example, oil price (OILPRICEx) in the commodity market, exchange rates (EXSZUsx and EXCAUSx) in the foreign exchange market; 2) the column in the temporal loading spans in all five lags, which explains why the third factor peaks after the recession periods.

The second question is answered by the response loading, which has the same definition as the factor loading in a factor model if one considers the factors from the tensor MGP own-lag as factor scores. Each column of the response loading shows how each factor impacts the responses. In the first column, margins corresponding to variables in the money and credit category have high magnitudes, which follows expectation because the first factor represents this category. Assume that the first factor is positively associated with money supply given the evidence in Table 3.7, we can explain the negative margins of interest rates: during economic downturns, both rate cuts and quantitative easing are applied as part of the monetary policy toolkit to boost economic activity. Similarly, the positive margins in the exchange rate category suggest the depreciation of the U.S. dollar when the money supply increases in the U.S. Moving to the second column, the negative margins in the income and output category have high magnitudes, suggesting an increase in this unemployment factor (the second factor) results in the slowdown of economic activities. Negative margins of interest rates show the expectation of interest rate reduction, given that the second factor rises. If we look at the loading corresponding to the third factor, it is unsurprising that the largest margin corresponds to S&P PE ratio because the third factor is highly correlated to this variable.

3.5.4 Effect of $\boldsymbol{D}$

This section discusses the effect of the own-lag matrix $\boldsymbol{D}$. First of all, the own-lag matrix is beneficial to model economic time series, as Figure 3.8 displays the posterior mean of non-zero elements in the own-lag matrix. Each row and column corresponds to one variable and a lag order, respectively. Own-lag effect is found in all categories except interest rate, with the first own-lag coefficient having the highest magnitudes. Moving to the comparison between TVARs with and without the own-lag matrix, we find that this matrix allows the tensor to explore more cross-lag effects. This is because Figure 3.6 does not present a strong own-lag effect, as the variables with high margin magnitudes in the response loading do not coincide with their counterparts in the predictor loading. We use tensor MGP (without $\boldsymbol{D}$) to conduct the same experiment as in Section 3.5.3. Figure 3.7 depicts the posterior mean of the loadings without $\boldsymbol{D}$, showing that the same variable (PAYEMS) is associated with the largest margin magnitudes in the first columns of response and predictor loadings. This pattern holds for the second and third columns as well, with the corresponding variables being M2REAL and BUSLOANS. In addition, Table 3.8 shows that these large margins in PAYEMS, M2REAL and BUSLOANS, presented in Figure 3.7 can distort the coefficients in such a manner that the rows and columns



Response			Predictor			
–0.31	0.11	0.08	–0.06	–0.02	–0.08	RPI
–0.40	0.14	0.06	0.10	0.01	0.02	W875RX1
–0.33	0.11	0.02	–0.07	–0.04	0.11	INDPRO
–0.34	0.15	–0.08	0.01	0.01	0.01	GDPC1
–0.07	0.06	0.02	–0.09	–0.02	–0.04	GDPDEF
–0.33	0.18	–0.10	–0.38	–0.07	–0.07	PCECC96
–0.31	0.18	–0.03	–0.16	–0.04	0.14	CMRMTSPLx
–0.11	–0.09	–0.05	–0.06	–0.01	0.03	RSAFSx
–0.38	0.15	–0.06	–0.10	–0.03	0.27	HWIx
–0.35	0.17	0.00	–0.46	–0.08	0.00	HWIURATIOx
–0.21	–0.06	–0.01	–0.04	0.00	–0.13	CIVPART
0.36	–0.16	–0.06	–0.03	–0.06	–0.05	UNRATE
–0.63	0.09	0.07	–1.61	–0.30	–0.07	PAYEMS
–0.05	0.04	–0.10	0.00	0.01	0.00	CES0600000007
0.03	–0.04	0.01	0.06	0.06	0.16	OILPRICEx
–0.18	0.25	–0.09	0.52	0.48	0.01	CPIAUCSL
–0.12	0.02	0.00	0.41	0.10	0.01	FEDFUNDS
–0.10	0.01	0.03	0.18	0.05	0.02	CP3M
–0.10	0.02	0.03	–0.15	–0.03	0.02	TB3MS
–0.11	0.02	0.03	–0.16	–0.04	0.02	TB6MS
–0.09	0.02	0.01	–0.12	–0.03	0.16	GS1
–0.06	0.00	–0.01	–0.03	–0.01	0.18	GS5
–0.04	–0.01	–0.06	–0.23	–0.06	0.05	GS10
–0.03	–0.06	0.03	0.08	0.01	0.04	AAA
0.03	–0.11	0.13	0.14	0.04	–0.01	BAA
–0.09	0.00	–0.66	0.00	0.02	–0.05	M1SL
–0.08	–0.02	–0.24	0.66	3.94	–0.18	M2SL
–0.16	0.34	–0.12	–1.26	–5.45	0.11	M2REAL
–0.23	0.06	0.99	–0.06	0.00	0.27	BUSLOANSx
–0.48	0.27	0.19	–0.02	–0.01	0.16	NONREVSLx
0.10	–0.01	–0.70	–0.09	–0.03	0.04	INVEST
–0.35	0.26	0.04	–0.04	–0.01	0.02	CONSPIx
0.07	0.04	0.03	–0.13	–0.04	0.02	S.P.500
0.08	0.05	0.03	–0.04	–0.01	0.12	S.P..indust
–0.07	–0.02	0.18	0.09	0.03	0.13	S.P.div.yield
0.04	0.06	–0.43	0.00	0.00	–0.18	S.P.PE.ratio
–0.03	0.07	0.07	–0.11	–0.04	–0.04	EXSZUSx
–0.01	0.04	0.11	0.17	0.03	0.04	EXJPUSx
–0.02	0.05	0.04	0.01	–0.01	0.07	EXUSUKx
0.02	0.00	0.01	–0.13	–0.03	–0.11	EXCAUSx

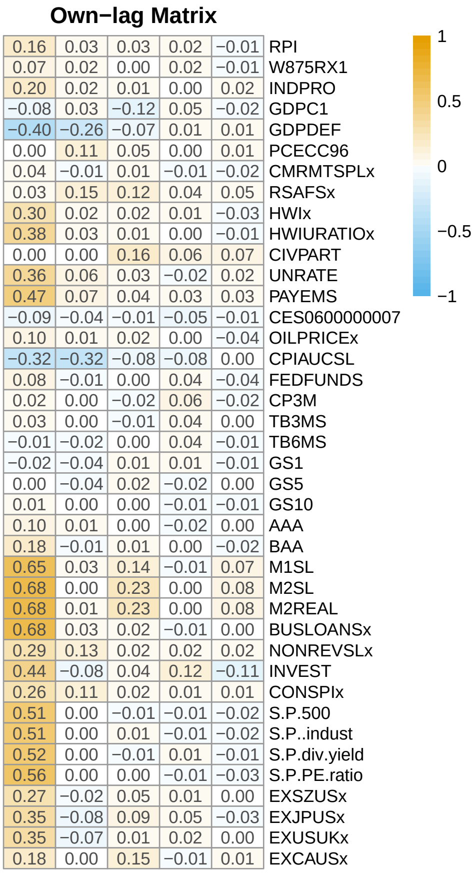
Temporal			
0.11	–0.14	0.21	Lag 1
0.05	–0.03	0.12	Lag 2
0.03	–0.01	0.14	Lag 3
0.00	0.00	0.09	Lag 4
0.00	0.00	0.06	Lag 5

Figure 3.7: Posterior mean of response, predictor and temporal loadings inferred from tensor MGP.

corresponding to these three variables in the coefficient matrix exhibit a higher proportion of large magnitudes compared to their counterparts associated with other variables. In this table, the proportions associated with the 3 variables are 2 to 6 times larger than those corresponding to other variables, when the results of the Tensor MGP are considered. However, when we apply the tensor MGP own-lag, the proportions across variables appear similar.

		Response		Predictor	
		3 Variables	Other Variables	3 Variables	Other Variables
Tensor MGP	>0.001	0.687	0.381	0.720	0.378
	>0.01	0.183	0.049	0.265	0.043
	>0.1	0.01	0.003	0.027	0.002
Tensor MGP Own-lag	>0.001	0.595	0.438	0.472	0.478
	>0.01	0.04	0.027	0.052	0.025
	>0.1	0	0	0	0

Table 3.8: Proportion of coefficient magnitudes greater than certain values (0.001, 0.01 and 0.1) corresponding to a group of variables. “Response” and “Predictor” mean whether this group of variables are considered as responses or predictors. If they are responses, we split coefficients by rows; otherwise, we split them by columns. “3 Variables” means the group of PAYEMS, M2REAL, and BUSLOANS, and “Other Variables” corresponds to the rest variables.



0.16	0.03	0.03	0.02	−0.01	RPI
0.07	0.02	0.00	0.02	−0.01	W875RX1
0.20	0.02	0.01	0.00	0.02	INDPRO
−0.08	0.03	−0.12	0.05	−0.02	GDPC1
−0.40	−0.26	−0.07	0.01	0.01	GDPDEF
0.00	0.11	0.05	0.00	0.01	PCECC96
0.04	−0.01	0.01	−0.01	−0.02	CMRMTSPLx
0.03	0.15	0.12	0.04	0.05	RSAFSx
0.30	0.02	0.02	0.01	−0.03	HWIx
0.38	0.03	0.01	0.00	−0.01	HWIURATIOx
0.00	0.00	0.16	0.06	0.07	CIVPART
0.36	0.06	0.03	−0.02	0.02	UNRATE
0.47	0.07	0.04	0.03	0.03	PAYEMS
−0.09	−0.04	−0.01	−0.05	−0.01	CES0600000007
0.10	0.01	0.02	0.00	−0.04	OILPRICEx
−0.32	−0.32	−0.08	−0.08	0.00	CPIAUCSL
0.08	−0.01	0.00	0.04	−0.04	FEDFUNDS
0.02	0.00	−0.02	0.06	−0.02	CP3M
0.03	0.00	−0.01	0.04	0.00	TB3MS
−0.01	−0.02	0.00	0.04	−0.01	TB6MS
−0.02	−0.04	0.01	0.01	−0.01	GS1
0.00	−0.04	0.02	−0.02	0.00	GS5
0.01	0.00	0.00	−0.01	−0.01	GS10
0.10	0.01	0.00	−0.02	0.00	AAA
0.18	−0.01	0.01	0.00	−0.02	BAA
0.65	0.03	0.14	−0.01	0.07	M1SL
0.68	0.00	0.23	0.00	0.08	M2SL
0.68	0.01	0.23	0.00	0.08	M2REAL
0.68	0.03	0.02	−0.01	0.00	BUSLOANSx
0.29	0.13	0.02	0.02	0.02	NONREVSLx
0.44	−0.08	0.04	0.12	−0.11	INVEST
0.26	0.11	0.02	0.01	0.01	CONSPIx
0.51	0.00	−0.01	−0.01	−0.02	S.P.500
0.51	0.00	0.01	−0.01	−0.02	S.P..indust
0.52	0.00	−0.01	0.01	−0.01	S.P.div.yield
0.56	0.00	0.00	−0.01	−0.03	S.P.PE.ratio
0.27	−0.02	0.05	0.01	0.00	EXSZUSx
0.35	−0.08	0.09	0.05	−0.03	EXJPUSx
0.35	−0.07	0.01	0.02	0.00	EXUSUKx
0.18	0.00	0.15	−0.01	0.01	EXCAUSx

Figure 3.8: Posterior mean of non-zero entries in the own-lag matrix inferred using the large-scale data set. Each row corresponds to a variable, and each column is for a lag order.

3.6 Conclusion and Discussion

In this chapter, we apply the multiplicative gamma prior (MGP) to margins and use an adaptive inferential scheme to infer the rank. To overcome the convergence issue, we introduce an interweaving Gibbs sampler to allow better mixing of Markov chains and match labels and signs after the inference.

Since tensor decomposition is a dimension reduction technique, TVARs are closely related to the reduced-rank VAR mentioned in Section 2.1.4. A detailed discussion of these two structures is available in the introduction section of Wang et al. (2022a). In short, reduced-rank VAR only applies the low-rank assumption to the matricization of the tensor, but TVAR makes the same assumption to all three matricizations (model-1, -2, and -3). Following this connection, we find that reduced-rank VAR is a special case of the TVAR with the following expression

$$\boldsymbol{\mathcal{A}} = \sum_{r=1}^{R} \boldsymbol{\mathcal{A}}^{(r)} = \sum_{r=1}^{R} \boldsymbol{\beta}_1^{(r)} \circ \boldsymbol{C}^{(r)},$$

where $\boldsymbol{C}^{(r)}$ is an N-by-P matrix, and $\circ$ is the outer product of a vector and a matrix such that $\boldsymbol{\beta}_{1,i_1}^{(r)}\boldsymbol{C}^{(r)}$ equals to $\boldsymbol{\mathcal{A}}_{(i_1,\cdot,\cdot)}^{(r)}$, for $i_1 \in [N]$. If we decompose $\boldsymbol{C}^{(r)}$ to $\boldsymbol{\beta}_2^{(r)} \circ \boldsymbol{\beta}_3^{(r)}$, then we retain Equation (3.2). The comparison between TVAR and reduced-rank VAR is worth further investigation.

One caveat about the post-processing procedure is that MCMC samples after the procedure become misaligned with the ordering implicitly assumed by the MGP prior after column switching. This is because the MGP prior imposes an increasing shrinkage structure across columns, so this prior is not invariant to column permutations. For instance, if the first and second columns of $\boldsymbol{B}$ are permuted, the prior originally associated with the second column is effectively reassigned to the first column, and vice versa. Although this misalignment does not affect the primary purpose of employing the MGP, i.e., rank selection, it does pose a problem if one wishes to inspect posteriors, such as studying how the MGP induces shrinkage on a column-by-column basis. To assess the severity of this issue, one could record the frequency of column switching in the post-processing procedure. If the frequency is high, a potential solution would be to adopt a permutation-invariant prior after the rank has been identified.

The TVAR can be extended in several aspects. First, margins in the TVAR can be time-varying, which motivates Chapter 4. Second, a similar MCMC scheme can be applied to the Tucker decomposition introduced in Section 2.3.2. Last, the MGP is not the only increasing shrinkage prior; other priors within this class are worth investigation, such as the multiway stick-breaking shrinkage prior mentioned in Section 2.3.3 and the cumulative shrinkage prior proposed by (Legramanti et al., 2020).

Chapter 4

Bayesian Time-varying Tensor Vector Autoregression

This chapter introduces a natural extension of the tensor VAR framework, which incorporates time variation into the model. Our motivation stems from the over-parameterization issue inherent in the TVP-VAR model discussed in Section 2.1.5, where coefficients evolve according to random walks. As a result, empirical applications of TVP-VARs are typically limited to fewer than ten variables to avoid this issue. To preserve the high-dimensional structure of the data, recall that two main strands have been proposed in the literature to mitigate over-parameterization in high-dimensional VARs, both time-invariant (TIV) and time-varying: (1) shrinkage priors and (2) dimension reduction techniques (see Section 2.1.4 for details). Our work falls within the second strand using dimension reduction, where we develop a time-varying parameter tensor VAR (TVP-TVAR).

In addition to advancing the TVP-VAR literature, this work also contributes to the tensor modeling literature. Although time-varying tensors are gaining popularity, see Section 2.3.4 for applied examples, research on models that treat tensors as time-varying parameters remains scarce. Among the few existing studies, Harris et al. (2021) and Chen et al. (2023) constructed the tensors with time corresponding to one dimension without explicitly specifying the time variation; Zhang et al. (2021) developed a model where a subset of elements within the tensor decomposition is active at each time point, controlled by a time-varying rank value. To the best of our knowledge, the explicit modeling of time-varying margins has not been explored in the existing literature.

We follow the previous chapter to treat the coefficient matrix as a third-order tensor and decompose it via the CP decomposition with a fixed rank. According to Section 3.1.2, margins in this decomposition can be divided into three loadings: response, predictor, and temporal. Our model assumes one loading to be time-varying, while maintaining the other two as time-invariant, resulting in three configurations. Although incorporating multiple time-varying loadings is possible, we restrict our model to these three configurations as they provide distinct interpretations of how temporal dynamics manifest in relationships between time series and their lags. Additionally, modeling only one time-varying loading is computationally efficient. By specifying the evolution of time-varying margins as random walks, we can treat the TVP-TVAR as a state-space model, facilitating straightforward Bayesian inference.

Rank determination is an important step in the CP decomposition. Rather than adopting the model-based approach of imposing shrinkage priors on the loading matrices, as introduced in the previous chapter, we employ an evaluation-based method to select the rank, motivated by the following considerations. First, the TVP-TVAR model is already complex; introducing additional model components such as an increasing shrinkage prior within a time-varying model would increase the implementation burden for Bayesian inference. Second, the TVP-TVAR also requires selecting a suitable configuration for modeling time variation. While shrinkage priors can be used to control the extent of time variation in TVP-VARs, evaluation-based approaches in the TVP-TVAR framework offer a more straightforward implementation for the simultaneous selection of both rank and temporal configuration. The specific evaluation metric we use is the Deviance Information Criterion (DIC) (Spiegelhalter et al., 2002), as it is designed for Bayesian estimates[29]. This criterion has also been applied by Spencer et al. (2022) and Guhaniyogi and Spencer (2021) in the tensor literature.

Since the TVP-TVAR represents a state-space model with time-varying parameters being latent variables, several DIC variants established in latent variable models are available for model selection (Celeux et al., 2006). Choosing an appropriate DIC variant has received increasing attention in time series research (Chan and Eisenstat, 2018; Chan and Grant, 2016b; Li et al., 2020) and other statistical domains (Ariyo et al., 2022, 2020; Merkle et al., 2019). We compare two types of DICs extensively investigated in the literature: conditional and marginal DICs. The conditional DIC evaluates the likelihood conditional on latent variables, while

[29] An alternative evaluation metric is marginal likelihood. While the estimation of this metric has been studied in related time series models (e.g., state space models in Chan and Eisenstat (2015)), the estimation is not as straightforward as the DIC.

marginal DIC integrates them out. We specify two conditional and one marginal DICs based on the TVP-TVAR framework. While most references mentioned favor the marginal DIC due to concerns about uncertainty and complexity bias in conditional DICs, our simulation study reveals that one particular conditional DIC demonstrates advantages for TVP-TVARs – it yields more reliable results measured by lower Monte Carlo errors compared to alternative DICs, and accurately identifies the true model configuration. Based on these findings, we opt for this conditional DIC for model selection. Although higher ranks lead to lower chosen DICs, the improvement diminishes gradually as the rank increases, aligning with the finding in Maity et al. (2021) that DIC distinguishes underfitted models but not overfitted ones. Thus, instead of determining the rank corresponding to the minimum DIC, we implement the "kneedle" algorithm (Satopaa et al., 2011) to a sequence of DICs across ranks for knee point detection, which improves the performance of identifying the optimal rank.

We demonstrate the utility of TVP-TVARs using functional magnetic resonance imaging (fMRI) data sets collected by Wehbe et al. (2014) from multiple subjects while reading a chapter in *Harry Potter and the Sorcerer's Stone* (Rowling, 2012). The result from model selection indicates that the dynamics of the fMRI data are time-varying, which reflects the conclusion of Gaschler-Markefski et al. (1997) about the non-stationarity of fMRI data. The CP decompositions with selected ranks manage to reduce the number of parameters by over 90%, relative to standard VARs. The Granger causality analysis from the selected model shows that the number of directional brain connectivity is consistent with the narrative progression. Different regions of interest function as primary signal emitters or receivers at various time points.

This chapter is structured as follows. Section 4.1 specifies the model framework. Section 4.2 describes the posterior sampler of unknown parameters. Section 4.3 discusses DIC variants and details our implementation of the "kneedle" algorithm. A Monte Carlo study in Section 4.4 supports one of the conditional DICs for model selection. Section 4.5 provides the empirical results of applying TVP-TVARs to the fMRI data. Section 4.6 concludes this chapter.

4.1 Methodology

We adopted the mathematical expression of TVAR in (3.1) and incorporate temporal dynamics into the tensor, $\boldsymbol{\mathcal{A}}$, to formulate a time-varying parameter TVAR (TVP-TVAR),

$$\boldsymbol{y}_t = \boldsymbol{\mathcal{A}}_{t,(1)}\boldsymbol{x}_t + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Omega}\right), \tag{4.1}$$

where $\boldsymbol{y}_t \in \mathbb{R}^N$, $\boldsymbol{x}_t = (\boldsymbol{y}'_{t-1}, \dots, \boldsymbol{y}'_{t-P})' \in \mathbb{R}^{NP}$, $\boldsymbol{\mathcal{A}}_{t,(1)} = \boldsymbol{A}_t = (\boldsymbol{A}_{t,1}, \dots, \boldsymbol{A}_{t,P})$ corresponds to the mode-1 matricization of the time-varying tensor $\boldsymbol{\mathcal{A}}_t$. We follow Zhang et al. (2021) to model the variance-covariance matrix $\boldsymbol{\Omega}$ as time-invariant. We define three CP decompositions of $\boldsymbol{\mathcal{A}}_t$ with the ranks being fixed over time: (1) $\boldsymbol{\mathcal{A}}_t = [\![\boldsymbol{B}_{t,1}, \boldsymbol{B}_2, \boldsymbol{B}_3]\!]_{\text{CP}}$, (2) $\boldsymbol{\mathcal{A}}_t = [\![\boldsymbol{B}_1, \boldsymbol{B}_{t,2}, \boldsymbol{B}_3]\!]_{\text{CP}}$, (3) $\boldsymbol{\mathcal{A}}_t = [\![\boldsymbol{B}_1, \boldsymbol{B}_2, \boldsymbol{B}_{t,3}]\!]_{\text{CP}}$. Each decomposition sets one loading as time-varying while keeping the other two time-invariant. We refer to a TVP-TVAR with P lags and the j-th loading being time-varying as TVP-TVAR(P, j), for $j \in [3]$.

While the CP decomposition can include more than one time-varying loading, we restrict it to the above three configurations mainly because each of them offers a distinct interpretation of how temporal dynamics manifest in the relationships between $\boldsymbol{y}_t$ and its lags. As discussed in Section 3.1.2, each row of the response loading explains how an individual variable in $\boldsymbol{y}_t$ connects to a representation of past information, $\boldsymbol{B'}_2\boldsymbol{X}_t\boldsymbol{B}_3$, where $\boldsymbol{X}_t = (\boldsymbol{y}_{t-1}, \dots, \boldsymbol{y}_{t-P})$; while each row of the predictor (temporal) loading determines how a specific variable (lag) contributes to this representation. Extended from this interpretation of loadings in a TVAR, TVP-TVAR$(P, 1)$ models a static construction of the past information, but the response of $\boldsymbol{y}_t$ to the past information evolves over time. Conversely, TVP-TVAR $(P, 2)$ and $(P, 3)$ operate under the opposite framework: the transformation of lagged values into the representation of past information varies temporally, while the relationship between $\boldsymbol{y}_t$ and this representation is fixed. In particular, these two decompositions allow time-varying effects from individual variables or lags to the representation. Two additional considerations support our focus on these three configurations: identifiability constraints and computational efficiency. Regarding identifiability, the tensor decomposition corresponding to each of the three configurations is identified up to time-invariant scaling and permutation. In contrast, incorporating multiple time-varying loadings in the CP decomposition would cause both these scaling and permutation transformations to be time-dependent, posing challenges in inferring the parameters.

Throughout the rest of this chapter, we denote the time-varying loading in TVP-TVAR (P, j) as $\boldsymbol{B}_{t,j}$ and the time-invariant ones as $\boldsymbol{B}_{j'}$ for $j', j \in [3]$ and $j' \neq j$. Following the standard structure in TVP-VAR (Primiceri, 2005), we model the evolution of $\boldsymbol{b}_{t,j} = \text{vec}\left(\boldsymbol{B}_{t,j}\right)$, for $t = 2, \dots, T$, as a random walk,

$$\boldsymbol{b}_{t,j} = \boldsymbol{b}_{t-1,j} + \boldsymbol{\eta}_{t,j}, \quad \boldsymbol{\eta}_{t,j} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{Q}_j\right), \tag{4.2}$$

where $\boldsymbol{Q}_j = \text{diag}\left(q_{j,1}, \ldots, q_{j,I_jR}\right)$ is a diagonal matrix, for $I_1, I_2 = N$, $I_3 = P$, $q_{j,k}$ follows an inverse-gamma prior, $\mathcal{IG}\left(a_k, b_k\right)$, for $k \in [I_jR]$. We impose normal priors to $\boldsymbol{b}_{1,j}$ and $\boldsymbol{b}_{j'} = \text{vec}(\boldsymbol{B}_{j'})$, $\mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_j\right)$ and $\mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_{j'}\right)$, respectively. Following the spirit of the Minnesota-type prior, which posits that the shorter lags contain more information than longer lags, we specify

$$\boldsymbol{\Sigma}_j = \begin{cases} \sigma^2 \mathbf{I}_{NR}, & j = 1 \text{ or } 2 \\ \sigma^2 \mathbf{I}_R \otimes \text{diag}\left(1, \frac{1}{2^2}, \ldots, \frac{1}{P^2}\right), & \text{otherwise} \end{cases},$$

where $\mathbf{I}_{NR}$ and $\mathbf{I}_R$ are identity matrices with dimensions specified in their subscripts. $\boldsymbol{\Sigma}_{j'}$, the variance-covariance matrix of $\boldsymbol{b}_{j'}$ share the same expression. $\boldsymbol{\Sigma}_3$ imposes an increasing shrinkage property to $\boldsymbol{B}_3$ (or the initialization of $\boldsymbol{B}_{t,3}$), wherein the shrinkage level increases with the loading row index, thereby prioritizing information from shorter lags over longer lags in the past information representation. These margin priors imply that the (i_1, i_2, p) entry of the tensor initialization, decomposed by $\boldsymbol{B}_{1,j}$ and $\boldsymbol{B}_{j'}$'s, has zero mean and variance $R\sigma^6/p^2$. Finally, $\boldsymbol{\Omega}$ follows an inverse-Wishart prior, $\boldsymbol{\Omega} \sim \mathcal{IW}\left(\nu, \boldsymbol{S}\right)$.

Two points are noteworthy if the time-varying margins follow random walks as defined in (4.2). First, if σ^2 is a known parameter, margins are identifiable up to sign switching and permutation, instead of the scaling and permutation described. This is because, take TVP-TVAR(P,1) as an example, if the two CP decompositions $[\![\boldsymbol{B}_{t,1}, \boldsymbol{B}_2, \boldsymbol{B}_3]\!]_{\text{CP}}$ and $[\![\tilde{\boldsymbol{B}}_{t,1}, \tilde{\boldsymbol{B}}_2, \tilde{\boldsymbol{B}}_3]\!]_{\text{CP}}$ provide the same tensor, then $\tilde{\boldsymbol{b}}_{1,1} = \text{vec}\left(\tilde{\boldsymbol{B}}_{1,1}\right)$ follows a multivariate normal distribution with zero mean and variance-covariance matrix $\left(\boldsymbol{Q}_1 \otimes \boldsymbol{I}_{I_j}\right) \boldsymbol{\Sigma}_1 \left(\boldsymbol{Q}_1 \otimes \boldsymbol{I}_{I_j}\right)'$, which equals to $\boldsymbol{\Sigma}_1$ only when diagonal entries in the diagonal scaling matrix $\boldsymbol{Q}_1$ are 1 or -1. The same explanation also applies to margins in the time-invariant loadings. According to Chapter 3, the scaling indeterminacy of the CP decomposition is one of the sources of high autocorrelation in the MCMC samples of margins. To mitigate high autocorrelation from this source, we assume σ^2 to be known. Second, TVP-TVAR(P, j) implies that the VAR coefficients evolve as random walks, which leads to a state-space representation,

$$\boldsymbol{y}_t = \left(\boldsymbol{I}_N \otimes \boldsymbol{x}_t'\right) \boldsymbol{\alpha}_t + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} + \boldsymbol{u}_t^{\alpha}, \tag{4.3}$$

where $\boldsymbol{\alpha}_t = \text{vec}(\boldsymbol{A}_t')$. Suppose the model is TVP-TVAR($P, 1$), then $\boldsymbol{u}_{t,i}^{\alpha}$ (for $i \in [N^2P]$) is independent to other elements in $\boldsymbol{u}_t^{\alpha}$ and follow

$$\boldsymbol{u}_{t,i}^{\alpha} \sim \mathcal{N}\left(0, \sum_{r=1}^{R} \left(\boldsymbol{B}_{2,(i_2,r)}\right)^2 \left(\boldsymbol{B}_{3,(i_3,r)}\right)^2 q_{1,N(r-1)+i_1}\right),$$

for $i = NP(i_1 - 1) + N(i_3 - 1) + i_2$, where $\boldsymbol{B}_{j',(i_{j'},r)}$ is the $(i_{j'}, r)$ entry of $\boldsymbol{B}_{j'}$, for $i_1, i_2 \in [N]$ and $i_3 \in [P]$. If the model is TVP-TVAR$(P, 2)$ or TVP-TVAR$(P, 3)$, one can change the three terms in the normal distribution accordingly.

4.2 Posterior Computation

We now turn to the MCMC algorithm to estimate unknown parameters in (4.1) and (4.2) given the priors specified in the previous section and a fixed rank R. We adopt a Gibbs sampler to sample $\{\boldsymbol{b}_{1:T,j}, \boldsymbol{b}_{j'}, \boldsymbol{Q}_j, \boldsymbol{\Omega}\}$[30]. The sampler cycles through the following steps.

Step 1. We sample $\boldsymbol{b}_{1:T,j}$ using the forward filtering backward sampling algorithm (Carter and Kohn, 1994; Frühwirth-Schnatter, 1994). This approach is feasible because, analogous to the rearrangements in TVARs, see Wang et al. (2022a), TVP-TVAR(P, j) can be rearranged as

$$\boldsymbol{y}_t = \boldsymbol{Z}_{t,j}\boldsymbol{b}^*_{t,j} + \boldsymbol{\epsilon}_t \tag{4.4}$$

where $\boldsymbol{Z}_{t,1} = \left(\boldsymbol{x}'_t\left(\boldsymbol{B}_3 \otimes \boldsymbol{B}_2\right)\boldsymbol{\mathcal{I}}'_{(1)}\right) \otimes \boldsymbol{I}_N$, $\boldsymbol{Z}_{t,2} = \boldsymbol{B}_1\boldsymbol{\mathcal{I}}_{(1)}\left(\left(\boldsymbol{B}'_3\boldsymbol{X}'_t\right) \otimes \boldsymbol{I}_R\right)$, $\boldsymbol{Z}_{t,3} = \boldsymbol{B}_1\boldsymbol{\mathcal{I}}_{(1)}$ $\left(\boldsymbol{I}_R \otimes \left(\boldsymbol{B}'_2\boldsymbol{X}_t\right)\right)$, $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{R\times R\times R}$ is a superdiagonal tensor with ones on non-zero entries, $\boldsymbol{b}^*_{t,j} = \boldsymbol{b}_{t,j}$ for $j = 1$ and 3, $\boldsymbol{b}^*_{t,2} = \text{vec}\left(\boldsymbol{B}'_{t,2}\right)$. Incorporating this equation with the random walk of $\boldsymbol{b}_{t,j}$ yields a linear state-space model.

Step 2. $\boldsymbol{b}_{j'}$ comprises two blocks corresponding to loadings specified in the previous section, with each block having similar full conditionals. Based on (4.4), the full conditionals of $\boldsymbol{b}_{j'}$, for $j' = 1$ or 3, or that of $\boldsymbol{b}^*_{j'}$, for $j = 2$, is $\mathcal{N}\left(\bar{\boldsymbol{\mu}}_{j'}, \bar{\Sigma}_{j'}\right)$ with

$$\overline{\Sigma}^{-1}_{j'} = \Sigma^{-1}_{j'} + \sum_{t=1}^{T} \boldsymbol{Z}'_{t,j'}\boldsymbol{\Omega}^{-1}\boldsymbol{Z}_{t,j'}, \quad \bar{\boldsymbol{\mu}}_{j'} = \overline{\Sigma}_{j'} \sum_{t=1}^{T} \boldsymbol{Z}'_{t,j'}\boldsymbol{\Omega}^{-1}\boldsymbol{y}_t,$$

where $\boldsymbol{B}_j$ in $\boldsymbol{Z}_{t,j'}$ changes to $\boldsymbol{B}_{t,j}$.

Step 3. The by-product of samples from Step 1 and 2 is the time-varying tensor $\boldsymbol{\mathcal{A}}_t$. The variance-covariance matrix $\boldsymbol{\Omega}$ is sampled given $\boldsymbol{\mathcal{A}}_{t,(1)}$ and $\boldsymbol{y}_{1:T}$ from

$$\mathcal{IW}\left(T + \nu, \sum_{t=1}^{T}\left(\boldsymbol{y}_t - \boldsymbol{\mathcal{A}}_{t,(1)}\boldsymbol{x}_t\right)\left(\boldsymbol{y}_t - \boldsymbol{\mathcal{A}}_{t,(1)}\boldsymbol{x}_t\right)' + \boldsymbol{S}\right).$$

Step 4. Sample $q_{j,k}$, for $k \in [I_jR]$, from

$$\mathcal{IG}\left(a_k + \frac{T}{2}, b_k + \frac{1}{2}\sum_{t=2}^{T}\left(\boldsymbol{b}_{t,j,k} - \boldsymbol{b}_{t-1,j,k}\right)^2\right),$$

where $\boldsymbol{b}_{t,j,k}$ is the k-th element in $\boldsymbol{b}_{t,j}$.

[30] We do not use the interweaving strategy applied in Chapter 3 because the application in the current chapter does not require margin Markov chains to mix well.

4.3 Model Selection

Given the TVP-TVAR framework and a data set, we need to select a model configuration to describe the time variation of loadings and a rank value. There are 4 configurations corresponding to TVAR(P) and TVP-TVAR(P, j), for $j \in [3]$, and we select the rank from 1 to R^*, a predefined upper bound. While existing tensor-structured models generally do not select model configuration, rank selection has been extensively investigated and can be divided into model-based and evaluation-based methods, see Section 2.3.3 for a review. Model-based methods impose shrinkage priors to margins with a rank of R^* and shrink the rank by removing redundant margins from the model. This approach requires only one MCMC simulation, rather than multiple simulations with different rank values, thus enhancing computational efficiency; however, given the complexity of the TVP-TVAR, we opt not to impose any additional shrinkage prior. Alternatively, one can employ the evaluation-based method by selecting an evaluation metric and incrementally increasing the rank from 1 until the metric no longer indicates improvement in model performance. Compared to the model-based methods, one advantage of this approach is that we can select both model configuration and rank simultaneously. Several evaluation metrics are available for model selection, including the Bayes factor (BF) (Jeffreys, 1935) and the Deviance Information Criterion (DIC) (Spiegelhalter et al., 2002). The DIC offers practical advantages, as it can be more readily evaluated from MCMC outputs compared to the BF[31]. Given the widespread adoption of DIC in both Bayesian time series (Bai and Wang, 2015; Chan and Eisenstat, 2018; Chan and Grant, 2016b; Li et al., 2020) and tensor-structured models (Guhaniyogi and Spencer, 2021; Spencer et al., 2022), we use this metric for model selection.

To facilitate further discussion about DIC, we briefly introduce it in the TVP-TVAR framework. The deviance of parameter $\boldsymbol{\theta}$, $D(\boldsymbol{\theta})$, is defined as $D(\boldsymbol{\theta}) = -2 \log p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{\theta}\right) + 2 \log h\left(\boldsymbol{y}_{1:T}\right)$, where $h\left(\boldsymbol{y}_{1:T}\right)$ is a fully-specified standardizing term which only depends on $\boldsymbol{y}_{1:T}$. The DIC balances goodness-of-fit and model complexity by a summation of two terms,

$$\mathrm{DIC}\left(\boldsymbol{\theta}\right) = \overline{D(\boldsymbol{\theta})} + p_D,$$

[31] Beyond the scope of this paper, the DIC has additional advantages over BF. Notably, the latter suffers Jeffreys–Lindley paradox (see detailed description in Robert (2014)) and is not well-defined with improper priors, i.e., the corresponding marginal likelihood is arbitrary up to a multiplicative constant, whereas the former does not have these issues.

where $\overline{D(\boldsymbol{\theta})} = -2\mathbb{E}_{\boldsymbol{\theta}}\left(\log p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{\theta}\right)\right) + 2\log h\left(\boldsymbol{y}_{1:T}\right)$ is the posterior mean deviance and p_D denotes the effective number of parameters. In particular, p_D is the difference between posterior mean deviance and deviance of posterior mean, $\overline{D(\boldsymbol{\theta})} - D(\hat{\boldsymbol{\theta}})$. Since $h\left(\boldsymbol{y}_{1:T}\right)$ is independent of $\boldsymbol{\theta}$, one can set $h\left(\boldsymbol{y}_{1:T}\right)$ as 1 and write the DIC as

$$\mathrm{DIC}\left(\boldsymbol{\theta}\right) = -4\mathbb{E}_{\boldsymbol{\theta}}\left(\log p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{\theta}\right)\right) + 2\log p\left(\boldsymbol{y}_{1:T} \mid \hat{\boldsymbol{\theta}}\right), \tag{4.5}$$

where we approximate the first term by the MCMC samples of $\boldsymbol{\theta}$ and plug the sample mean $\hat{\boldsymbol{\theta}}$ into the second term.

Since the TVP-TVAR can be written as state-space models, as described in (4.3) and (4.4), $\boldsymbol{\mathcal{A}}_{1:T}$ and $\boldsymbol{b}_{t,j}$ serve as latent variables in these two respective equations. Following Celeux et al. (2006), if $\boldsymbol{\theta}$ includes latent variables, we regard the DIC in (4.5) as conditional DIC because the likelihood is conditional on these latent variables. This conditional DIC has two variants based on the specification of $\boldsymbol{\theta}$. The first variant, denoted as $\mathrm{DIC}^{c,1}$, is formulated with $\boldsymbol{\theta} = \{\boldsymbol{\mathcal{A}}_{1:T}, \boldsymbol{\Omega}\}$. The second variant, $\mathrm{DIC}^{c,2}$, corresponds to the specification where $\boldsymbol{\theta} = \{\boldsymbol{b}_{1:T,j}, \boldsymbol{b}_{j'}, \boldsymbol{\Omega}\}$. These two variants are distinct due to the difference in their deviances of posterior mean, $p\left(\boldsymbol{y}_{1:T} \mid \hat{\boldsymbol{\theta}}\right)$. Specifically, both conditional DIC variants need to compute the time-varying tensor based on $\hat{\boldsymbol{\theta}}$. Take the TVP-TVAR$(P,1)$ as an example, these tensors correspond to $\mathbb{E}\left[[\![\boldsymbol{B}_{t,1}, \boldsymbol{B}_2, \boldsymbol{B}_3]\!]_{\mathrm{CP}}\right]$ and $[\![\mathbb{E}\left[\boldsymbol{B}_{t,1}\right], \mathbb{E}\left[\boldsymbol{B}_2\right], \mathbb{E}\left[\boldsymbol{B}_3\right]]\!]_{\mathrm{CP}}$ in $\mathrm{DIC}^{c,1}$ and $\mathrm{DIC}^{c,2}$, respectively, and these expressions can differ because the loadings may have posterior correlation. An alternative DIC widely studied in latent variable models is marginal DIC, denoted as DIC^{m}, which takes the form of (4.5) with $\boldsymbol{\theta} = \{\boldsymbol{b}_{j'}, \boldsymbol{Q}_j, \boldsymbol{\Omega}\}$. This DIC marginalizes $\boldsymbol{\mathcal{A}}_{1:T}$ and $\boldsymbol{b}_{1:T,j}$ in $\mathrm{DIC}^{c,1}$ and $\mathrm{DIC}^{c,2}$, respectively, and uses an integrated likelihood $p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{\theta}\right) = \int p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{\mathcal{A}}_{1:T}, \boldsymbol{\Omega}\right) p\left(\boldsymbol{\mathcal{A}}_{1:T} \mid \boldsymbol{b}_{j'}, \boldsymbol{Q}_j\right) d\boldsymbol{\mathcal{A}}_{1:T} = \int p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{b}_{1:T,j}, \boldsymbol{b}_{j'}, \boldsymbol{\Omega}\right) p\left(\boldsymbol{b}_{1:T,j} \mid \boldsymbol{b}_{j'}, \boldsymbol{Q}_j\right) d\boldsymbol{b}_{1:T,j}$ to evaluate model performance. A closed form expression for $\log p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{\theta}\right)$ is available in Appendix C.1.

Various studies favor marginal DIC over conditional ones for two reasons. Firstly, conditional DICs potentially exhibit higher Monte Carlo error than marginal DIC since it depends on the latent variables (Chan and Grant, 2016a; Merkle et al., 2019). Secondly, the conditional DICs tend to choose the most complex model. For instance Chan and Grant (2016a) empirically showed that the more factors a factor model assumes, the lower the corresponding conditional DIC is. However, we show that the concern about the first limitation can be invalid without a specific condition, and the second limitation of conditional DICs can be addressed by knee point detection.

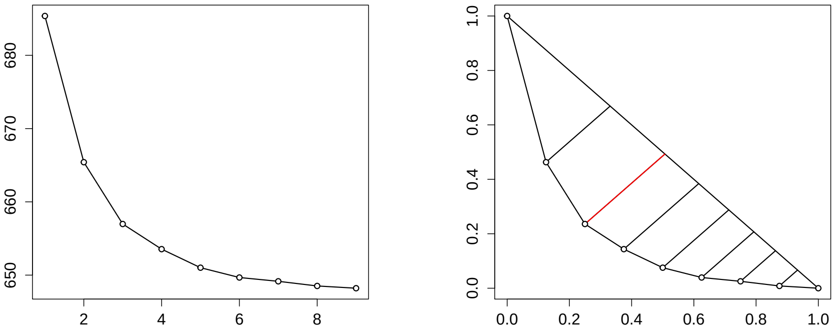
For the first limitation about the Monte Carlo error of DICs, we emphasize that this estimation uncertainty stems from the posterior uncertainty of $\boldsymbol{\theta}$. Due to the indeterminacy of the CP decomposition, margins defined in Section 4.1 are identifiable up to sign switching and permutation, resulting in highly autocorrelated margin Markov chains[32]. The consequence of this autocorrelation is that margins present high sample variance within a single MCMC simulation, and their sample means exhibit high variation given multiple simulations (see Figure 4.2, 4.3 and 4.4 in the next section for an illustration). DIC^{c,1} and DIC^{m}, which incorporate margins, inherently demonstrate higher Monte Carlo error compared to DIC^{c,1}, whose parameters of interest do not suffer from this indeterminacy issue. Therefore, we prefer DIC^{c,1} for model selection.

We regard the model complexity about the second limitation in two aspects: model configuration and rank choice. For the former, TVP-TVAR$(P, 1)$ and TVP-TVAR$(P, 2)$ are more complex due to more parameters included relative to those in TVP-TVAR$(P, 3)$ and time-invariant TVAR. According to the Monte Carlo study about DIC^{c,1} in Section 4.4, this conditional DIC does not favor a more complex model configuration over the one that generates the data. Next, we move to the rank choice. While higher ranks do yield lower DIC^{c,1}, the model improvement gradually diminishes when the rank exceeds a certain value, based on the simulation results in the next section (see Figure 4.1a). This reflects the theoretical results in Maity et al. (2021) that the DIC tends to distinguish underfitted models but not overfitted models[33]. Therefore, a useful tool to prevent DIC^{c,1} from choosing the highest available rank is knee point detection, which detects the point of maximum curvature in a function.

Multiple definitions of maximum curvature result in different algorithms for knee point detection. For example, Zhao et al. (2008) defined an angle-based curvature to select the number of clusters using a BIC curve; Satopaa et al. (2011) proposed “kneedle” to find the maximum distance between a normalized curve and a straight line (the detailed definition will be provided). We choose “kneedle” for knee point detection because it has a clearer visu-

[32] Although the autocorrelation arising from the scaling indeterminacy has been mitigated, autocorrelation due to sign switching and permutation still occurs.

[33] The DIC described in Maity et al. (2021) corresponds to (4.5), but it cannot be considered as the conditional DIC because their model does not have any latent structure.



(a) Unnormalized.

(b) Normalized.

Figure 4.1: Example of unnormalized (left) and normalized (right) conditional DICs (presented in circles) computed from the data set generated from TVP-TVAR$(3, 1)$ with a rank of 3. The straight lines connecting the circles and $y = -x + 1$ represent the distance between them, with the red line representing the maximum distance.

alization of maximum curvature and gives higher accuracy in detecting knees compared to the angle-based one. Specifically, locating the maximum curvature in a discrete sequence $\text{DIC}^{c,1}_{\mathcal{M}} = \left(\text{DIC}^{c,1}_{\mathcal{M},1}, \dots, \text{DIC}^{c,1}_{\mathcal{M},R^*}\right)$, which stores the $\text{DIC}^{c,1}$'s with model configuration $\mathcal{M}$ and different rank values, involves two steps. The first step is to normalize each conditional DIC to $\widetilde{\text{DIC}}^{c,1}_{\mathcal{M},R} = \left(\text{DIC}^{c,1}_{\mathcal{M},R} - \text{DIC}^{c,1}_{\mathcal{M},R_{\text{min}}}\right) / \left(\text{DIC}^{c,1}_{\mathcal{M},R_{\text{max}}} - \text{DIC}^{c,1}_{\mathcal{M},R_{\text{min}}}\right)$, where R_{min} and R_{max} represent the ranks associated with the minimum and maximum DICs, and convert the corresponding rank R to $(R - 1)/(R^* - 1)$. The curve in Figure 4.1b represents these normalized DICs computed using the simulation data in Section 4.4. Then, we identify the point with the maximum curvature as that with the maximum distance to the line which passes points $(0, \widetilde{\text{DIC}}^{c,1}_{\mathcal{M},1})$ and $(1, \widetilde{\text{DIC}}^{c,1}_{\mathcal{M},R^*})$. As illustrated in Figure 4.1b, this maximum curvature occurs at the third point, so we select the rank as 3, which is the true rank value defined in the next section.

4.4 A Monte Carlo Study

4.4.1 Data and Implementation

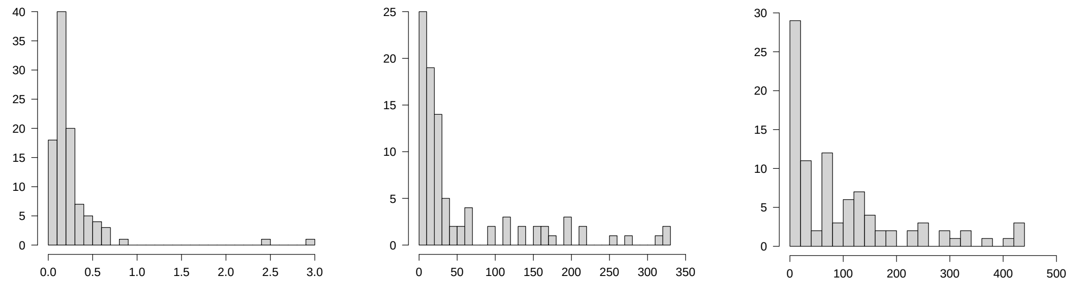
This section demonstrates the utility of $\text{DIC}^{c,1}$ (henceforth referred to as DIC for brevity) in model selection. We generate 100 3-variable datasets from each of four model configurations: TVAR(3), TVP-TVAR$(3, j)$, for $j \in [3]$. Each data set contains 200 observations, and the corresponding tensors are generated from CP decompositions with a rank of 3. We set σ^2, the multiplier in the margin variance, as 0.5, resulting in prior variances of 0.375, 0.094, and 0.042 for the coefficients in $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{A}_3$ (or their corresponding initializations in time-varying models), respectively. $\boldsymbol{\Omega}$ is sampled from the inverse-Wishart distribution stated in Section 4.1, with $\nu = 6$ and $\boldsymbol{S} = \mathbf{I}_3$, so that the prior mean of $\boldsymbol{\Omega}$ is $0.5\mathbf{I}_3$. $\boldsymbol{Q}_j$, the variance-covariance matrix in the random walk process of time-varying margins, is a diagonal matrix with all non-zero elements being 0.01.

With $N = P = 3$, the maximum rank that can be selected is 9, as $R \leq \{N^2, NP\}$ (Kolda and Bader, 2009). Therefore, there are 36 (9×4) possible models to select from for each data set. We estimate parameters in TVP-TVAR models using the algorithm described in Section 4.2, while for TVAR models, we modify Step 2 in the algorithm to incorporate three blocks of loadings rather than two, followed by implementation of the remaining two steps in the algorithm. To compute the DIC of each model and data set, we run 10 parallel Markov chains, each having 10,000 iterations with 1,000 burn-in. Following parameter estimation, we calculate the DIC for knee point detection by averaging the DICs from each of the 10 runs. The “kneedle” algorithm then helps determine the rank choice for each of the four model configurations. After recording the four corresponding DICs, we select the model configuration with the lowest value.

4.4.2 Simulation Result

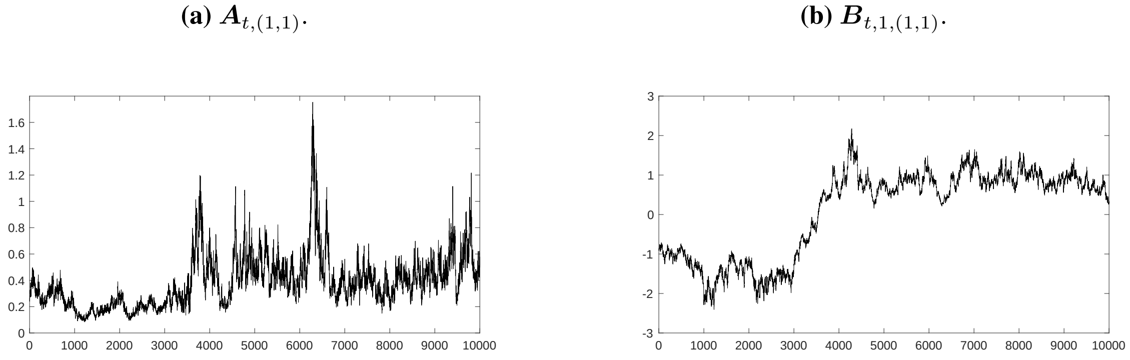
First, we compare the Monte Carlo errors corresponding to all the DIC variants mentioned in Section 4.3, and show the reason why we prefer DIC^{c,1}. The Monte Carlo error of a data set is calculated as the sample standard deviation of the 10 DICs obtained from the 10 parallel MCMC runs. A lower Monte Carlo error indicates that the corresponding estimation of this DIC variant is more reliable. As illustrated in Figure 4.2a, most Monte Carlo errors of DIC^{c,1} are below 1, whereas those in Figure 4.2b and 4.2c range from 0 to 500, with at most 10 being smaller than 1. These findings support our decision to use DIC^{c,1} for model selection in the TVP-TVAR framework.

Figure 4.3 and 4.4 explain why DIC^{c,1} is more reliable than other DICs. Figure 4.3 presents the trace plots of the (1,1)-entry in the coefficient matrix $\boldsymbol{A}_t$ alongside its decomposed margins, the $(1, 1)$ entry in $\boldsymbol{B}_{t,1}$, $\boldsymbol{B}_2$ and $\boldsymbol{B}_3$, at time point $t = 100$. One can also use $(1, 2)$ or $(1, 3)$ entries in the loadings for demonstration. These trace plots show that the Markov chain of the coefficient mixes well, whereas the margin ones have higher autocorrelation which cannot



(a) DIC^{c,1}. **(b)** DIC^{c,2}. **(c)** DIC^{m}

Figure 4.2: Histograms of Monte Carlo errors of conditional and marginal DICs, computed using data sets generated from TVP-TVAR$(3, 1)$. The model configuration and rank associated with these DICs align with the true data generating process. The middle and right panels restrict the display of Monte Carlo errors to a maximum of 350 and 500, respectively, which accounts for 93 data sets. Each Monte Carlo error is scaled by $1/\sqrt{10}$ for visualization clarity.



(a) $\boldsymbol{A}_{t,(1,1)}$. **(b)** $\boldsymbol{B}_{t,1,(1,1)}$.

(c) $\boldsymbol{B}_{2,(1,1)}$. **(d)** $\boldsymbol{B}_{3,(1,1)}$.

Figure 4.3: Trace plots of coefficients and margins sampled using TVP-TVAR(3,1) with a rank of 3.

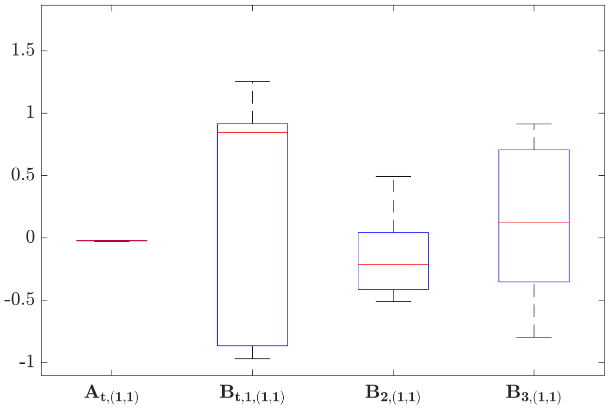


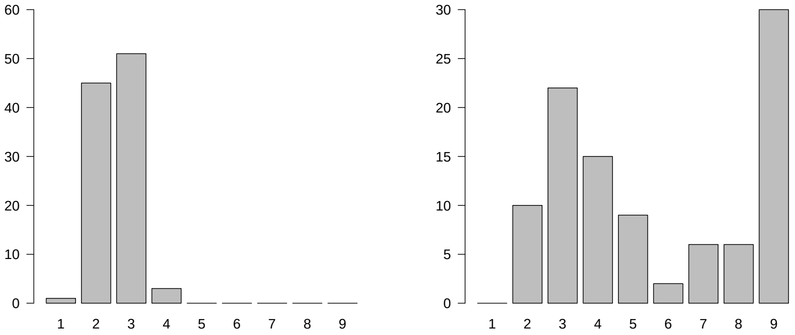
Figure 4.4: Boxplots of sample mean of coefficients and margins across 10 MCMC runs for the inference of one data set, generated from TVP-TVAR(3,1) with a rank of 3.

be solved by thinning. Notably, the sample variance of the coefficient is much lower than those margin counterparts. This autocorrelation, stemming from the indeterminacy of the CP decomposition, also elevates uncertainty of margin sample mean, as presented in Figure 4.4. This figure depicts boxplots of sample means of these parameters examined in Figure 4.3. Each boxplot displays 10 data points, with each point representing a sample mean derived from one MCMC run. The indeterminacy of the CP decomposition can result in loadings sampled from different MCMC runs representing permuted and sign-switched versions of the same underlying margins. To prevent this phenomenon from distorting our results, we aligned margins across parallel MCMC runs using correlation matching. Specifically, we compute correlations between the (1,1) entry samples shown in Figure 4.3 and the (1, r) entry samples of the same loading in other runs, for $r \in [3]$. The matching process identifies margins by selecting those with the highest absolute correlation, then adjusts them by multiplying with the sign of the correlation. Figure 4.4 demonstrates that the boxplot of coefficient sample mean exhibits lower variation compared to the margin counterparts. Since these sample means serve as plug-in parameters ($\hat{\boldsymbol{\theta}}$) in deviance of posterior mean, DIC^{c,1} yields more reliable results by incorporating coefficient sample means compared to DIC^{c,2} and DIC^{m}, which use margin sample means.

Next, we illustrate that the DIC can identify true model configurations. Table 4.1 presents the confusion matrix of configuration selection. Applying DICs enables the correct identification of model configurations in nearly all cases. TVAR is only selected when it is the true configuration. When the true configuration involves time-varying parameters, only a handful of data sets are misclassified. While Chan and Grant (2016b) showed that the conditional DIC favors complex models in their simulation studies, we do not reach the same conclusion here. For example, TVAR and TVP-TVAR(3,3) are relatively less complex compared to the other two configurations, but the model selection of the former two attains over 90% accuracy.

		Selected			
		TVAR(3)	TVP-TVAR(3,1)	TVP-TVAR(3,2)	TVP-TVAR(3,3)
True	TVAR(3)	96	0	2	2
	TVP-TVAR(3,1)	0	97	2	1
	TVP-TVAR(3,2)	0	8	89	3
	TVP-TVAR(3,3)	0	3	4	93

Table 4.1: Confusion matrix of configuration selection.



(a) With knee point detection.

(b) Without knee point detection.

Figure 4.5: Histograms of selected ranks based on the data sets generated from TVP-TVAR(3,1).

Lastly, we show that the knee point detection improves rank selection. Figure 4.5 depicts the distributions of selected ranks if the data is generated from TVP-TVAR(3,1). We record ranks determined using the knee point detection in the left panel and select the ranks corresponding to the lowest DICs in the right panel. According to the left panel, DICs with knee point detection correctly identify the true rank ($R = 3$) in over 50 data sets, with almost all remaining data sets selecting ranks adjacent to the true rank (i.e., $R = 2$ or $R = 4$). In contrast, ranks selected without the knee point detection span from 2 to 9, with $R = 9$ having the highest frequency. This suggests that knee point detection alleviates the tendency of DICs to favor overfitted models. Similar conclusions emerge from analysis using other model configurations (see figures in Appendix C.2).

4.5 Empirical Results

4.5.1 Data and Implementation

We apply TVP-TVARs to a functional magnetic resonance imaging (fMRI) data set about story understanding (Wehbe et al., 2014) to study the time variation of brain connectivity. The data was collected from 8 subjects when reading chapter 9 of *Harry Potter and the Sorcerer's Stone* (Rowling, 2012). All subjects are native English speakers and right-handed individuals who are familiar with the story in the chapter. The subject read the chapter in rapid serial visual format, i.e., the words were presented one by one for 0.5 seconds each. The chapter contains approximately 5,200 words and was divided into four runs, each lasting approximately 11 minutes. Before and after each run, there would be 20-second and 10-second breaks, respectively, for the subject to stare at a cross on the screen. For each subject in each run, the fMRI data was collected per two seconds from 21764 voxels, corresponding to 117 regions of interest (ROIs).

Following Zhang et al. (2021) and Xiong and Cribben (2023), we select $N = 27$ ROIs which control various cognitive and sensory functions (see the ROIs selected in Appendix C.3). For each ROI, we take the average of the voxel data within this particular ROI to form one time series. We discard the data when subjects took breaks and split the time series according to the runs. This process yields 32 (number of subjects $\times$ number of runs) data sets, with each in the format of $T_{\text{run id}} \times 27$ (the average value of $T_{\text{run id}}$ is 323). We then standardize each data set to avoid any scaling issue.

Following the lag order in Zhang et al. (2021), we apply TVAR(4), TVP-TVAR(4,j), for $j \in [3]$, to the data sets and select the rank from 1 to 10, extending beyond the rank of 8 previously reported as sufficiently large in Zhang et al. (2021). For the margin prior, we set $\sigma^2 = 0.1$, which leads to the variance range of coefficients in $\boldsymbol{A}_p$ in TVAR or $\boldsymbol{A}_{1,p}$ in TVP-TVAR as $\frac{1}{p^2}$×$[10^{-3}, 10^{-2}]$, for $p \in [4]$, given the rank range. The inverse-Wishart prior of the variance-covariance matrix has parameters $\nu = N + 3$ and $\boldsymbol{S} = \mathbf{I}_N$. We set $a_k = b_k = 0.01$ so that non-zero elements in $\boldsymbol{Q}_j$ have non-informative inverse-gamma priors. The burn-in and number of iterations are 1,000 and 10,000, respectively.

The application are implemented with in Intel(R) Xeon(R) Gold 6140 CPU @ 2.30GHz and MATLAB R2021a (9.10). The computing time corresponding to one subject and one run is about 1.5 hours.

4.5.2 Model Selection

We first demonstrate the model selection using one data set corresponding to Subject 1 reading the first part of the chapter. Then, we move to a summary of model selection using all 32 data sets. Table 4.2 presents the DICs computed with a range of model configurations and ranks. Overall, the DICs corresponding to TVP-TVAR(4,1) are lower than those of other configurations with the same rank value. The rank is selected to be 5 or 6 across configurations, as indicated in bold in the table. By comparing these 4 DICs in bold, we select TVP-TVAR(4,1) as the model configuration with a rank of 6. This suggests that Subject 1, when reading the first part of the chapter, had ROIs that processed information from past signals dynamically and maintained a static framework for gathering information from these signals.

Model	$R=1$	$R=2$	$R=3$	$R=4$	$R=5$	$R=6$	$R=7$	$R=8$	$R=9$	$R=10$
TVAR(4)	7290.70	6660.44	6171.06	5705.06	**5300.59**	4977.97	4746.89	4542.20	4375.59	4243.74
TVP-TVAR(4,1)	6645.54	6096.65	5560.65	5145.82	4762.24	**4449.99**	4309.18	4162.92	4031.85	3909.93
TVP-TVAR(4,2)	7280.87	6637.43	6120.45	5690.45	**5276.53**	4941.90	4697.56	4467.36	4291.59	4115.15
TVP-TVAR(4,3)	7335.64	6781.22	6319.24	5880.49	**5529.93**	5299.88	5024.46	4793.37	4657.43	4590.32

Table 4.2: DICs evaluated from the data for which Subject 1 read the first part of the chapter. DICs corresponding to the knee points are in bold.

We repeat the same analysis to all 32 data sets, and summarize the results in Figure 4.6. Three-quarters of the datasets favor TVP-TVAR(4,1), meaning that VAR coefficients exhibit temporal variation due to evolving signal reception patterns across ROIs. The three TVAR selections correspond to Subject 3 or the first part of the chapter, while TVP-TVAR(4,2) is only selected when the subjects read the second and third parts. None of the data sets identifies TVP-TVAR(4,3) as the optimal model fit. Based on these selected ranks, predominantly 4 and 5, Table 4.3 summarizes the average parameter counts for both standard VARs and TVARs. It reveals that both TVAR and TVP-TVARs effectively reduce the number of parameters by over 90%, inducing parsimonious model structures. Due to the high computational cost in standard VARs, we do not fit the data using these models in the application.

4.5.3 Granger Causality Analysis

Granger causality is a prevalent analysis in neuroscience to identify directional connectivity patterns between brain regions (Seth et al., 2015). To demonstrate Granger causality analysis using TVP-TVAR, we use the data set of Subject 1 reading the first part. The model implemented is

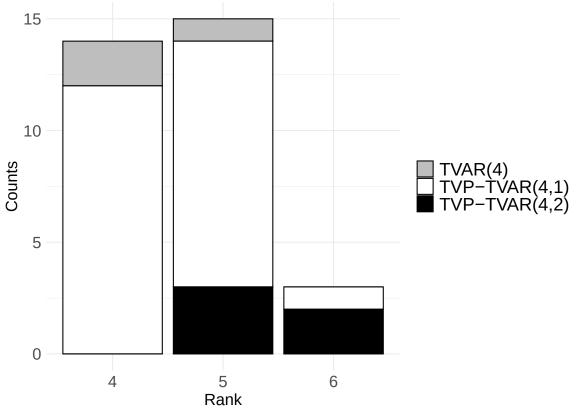


Figure 4.6: Counts of models selected across 32 data sets.

Standard VARs		Tensor VARs	
VAR	2916	TVAR(4)	251
TVP-VAR	941139	TVP-TVAR(4,1)	39718
		TVP-TVAR(4,2)	47224
		TVP-TVAR(4,3)	/

Table 4.3: Averaged number of parameters estimated across all subjects and runs based on different configurations.

TVP-TVAR(4,1) with a rank of 6, selected according to Table 4.2. The results of other data sets are available upon request. The formal definition of Granger causality can be found in Section 2.1.3. Based on the inferential result, we determine a time series m Granger causes another time series n at time t ($m,\ n \in [27]$) if $p_{t,(m\to n)} = p\left(|\boldsymbol{A}_{t,p,(n,m)}| > \delta \text{ for any } p = 1, \ldots, P \mid \boldsymbol{y}_{1:T}\right)$ is higher than a threshold p^*, where we set δ as 0.01 following Fan et al. (2022) and p^* as 99.9% to limit the false positive connections.

Figure 4.7 depicts the number of Granger causalities detected over time. When the subject started reading, there were relatively limited connections between brain regions. Subsequently, the connections increased progressively, maintaining levels above 90 across time points after $t = 100$. This pattern of Granger causality is related to the narrative progression in the first part of the chapter. For example, connectivity peaks during a pivotal scene featuring Harry and his friends participating in their first flying lesson. The number of Granger causalities begins to decline around $t = 225$ as the narrative focus shifts away from the protagonists.

To illustrate the evolving connectivity patterns of ROIs, we rank $p_{t,(m\to n)}$ at a particular time point t in descending order and display the first 50 connections for visualization. Figure 4.8 provides the Granger causality networks at $t = 1$, 171 and 300, corresponding to the start of the chapter, Harry started his flying lesson and the accident of Nievell, respectively. In Figure 4.8a, ST.R (right superior temporal gyrus) emitted the majority of signals. This region plays an important role in receptive language function and social cognition (Bigler et al., 2007), indicating that the subject began to process the textual information of this chapter, which involves

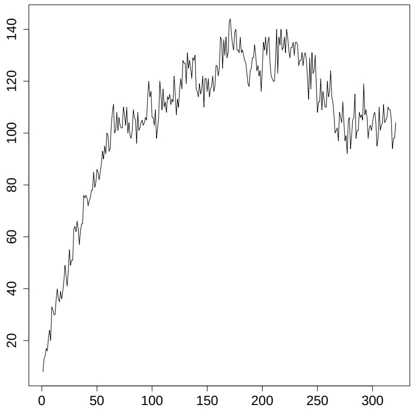
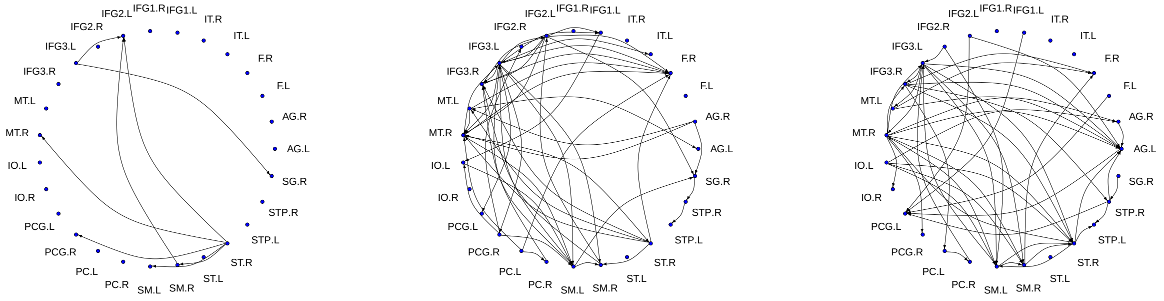


Figure 4.7: Granger causality time series.



(a) $t = 1$ **(b)** $t = 171$. **(c)** $t = 300$

Figure 4.8: Granger causality networks at different time points.

numerous character interactions. The network became denser in Figure 4.8b as the story progressed to the flying lesson. The ROI receiving the highest number of signals (7) from other ROIs is the right fusiform gyrus (F.R), which specializes in high-level visual processing functions, including reading and object recognition (Weiner and Zilles, 2016). Compared to Figure 4.8a, 17 additional connections originated from the inferior frontal gyrus areas (nodes prefixed with IFG), which is a multifunctional region with functionalities including, but not limited to, speech perception, programming sequential order of motor executions and social interaction (Liakakis et al., 2011). This pattern aligns with findings reported in Zhang et al. (2021), who showed similar shifts in Granger causality patterns during another key scene in the chapter. Moving to Figure 4.8c, the left angular gyrus (AG.L) and the right superior temporal gyrus (ST.R), which are both related to auditory processing (Bigler et al., 2007; Seghier, 2013), not only received more signals than other ROIs but also exhibited the most notable changes compared to the previous figure. In particular, AG.L and ST.R received 9 and 6 signals, respectively, increasing from 1 at $t = 171$. A possible explanation for this increase in signals received is that the corresponding story segment involves more auditory elements, such as teaching instructions and laughter, compared to other story segments.

4.6 Conclusion and Discussion

We propose the time-varying parameter tensor vector autoregression with three model configurations and implement a conditional DIC with knee point detection for model selection. The fMRI data application demonstrates that time variation in the response loading is the preferred configuration for most subjects. This finding supports the non-stationarity of fMRI data and reveals time-varying dynamics between the data and its representation of past information.

Our work can be extended in several directions. First, the post-processing procedure described in Section 3.3 could be adapted to identify the margins, potentially improving both the performance of the marginal DIC and the interpretability of the model. Second, more flexible model specifications are worth exploring – for instance, allowing for multiple time-varying loadings. A key challenge in this direction is the potential identifiability issue of the margins, which might hinder sampling and interpretation. Third, since heteroskedasticity is crucial in VAR applications in econometrics (Clark and Mertens, 2023), a natural extension of applying TVP-TVAR in this research field is to model the variance-covariance matrix $\boldsymbol{\Omega}$ as time-varying, such as by incorporating stochastic volatility.

Chapter 5

Time-varying Factor Augmented Vector Autoregression with Grouped Sparse Autoencoder

This chapter moves from the VAR framework to the factor-augmented VAR (FAVAR, see 2.2.1 for the definition), which allows for the analysis of hundreds of macroeconomic time series without encountering the over-parameterization that would occur if all these time series were directly modeled in the VAR. To better capture the time-varying dynamics of monetary policy transmission, heteroskedasticity of economic shocks, and structural breaks introduced by events such as the Global Financial Crisis (GFC) and the COVID-19 pandemic, this chapter explores non-linear extensions of the FAVAR, as discussed in Section 2.2.5. Among these non-linear FAVARs, Klieber (2024) leveraged deep learning techniques, specifically autoencoders, to extract factors, and demonstrated the robustness in handling outliers.

Given the growing popularity of autoencoders in econometric research (Andreini et al., 2020; Cabanilla and Go, 2019; Hauzenberger et al., 2023a), we focus on the FAVAR with an autoencoder and highlight three aspects that require attention and further improvement. Firstly, interpreting the factors and applying them to downstream tasks, such as the impulse response analysis, necessitate identifiable factors; however, factors extracted from a standard autoencoder generally do not satisfy this requirement (Locatello et al., 2019). Secondly, even when the factors are identified, determining their economic meanings remains challenging due to two issues: 1) the black box nature of the autoencoder, and 2) the lack of parsimony. While post-processing interpretation frameworks such as Shapley additive explanations (Strumbelj and Kononenko, 2010) can be applied to the factors extracted from the autoencoder, the complexity of the autoencoder often leads to results suggesting that each factor has a non-negligible impact on most high-dimensional time series. This makes it difficult to discern the specific economic role of individual factors. Lastly, Klieber (2024) assumed a time-invariant VAR, resulting in constant monetary policy effects and homoscedasticity. One can relax this assumption to accommodate time-varying dynamics of the economy.

Our first contribution tackles the identifiability and interpretability issues in the first two aspects through sparsity. Inspired by Moran et al. (2021), we propose a variant of the standard autoencoder, namely the grouped sparse (GS) autoencoder. While a standard autoencoder extracts latent factors using the encoder and reconstructs high-dimensional data with the decoder, our approach introduces an intermediate step. We first group the high-dimensional data, then element-wisely multiply these factors by a set of group-specific parameters before passing the factors to the decoder, so these parameters are the same across variables within each group during reconstruction. We follow Moran et al. (2021) to impose the spike-and-slab lasso (SSL, Ročková and George (2018)) prior on these group-specific parameters. The factors are proved to be identifiable up to an element-wise transformation, given known *anchor groups* – groups of time series reconstructed by only one factor. By exploiting the properties of these element-wise transformations, we determine the decoder architecture and its activation function. For interpretation, the SSL parameters can effectively activate or deactivate each factor when reconstructing a specific group of data, providing clear economic meanings to the factors and eliminating the need for post-processing interpretation approaches. Our second contribution builds upon the third aspect mentioned earlier. Specifically, we extend the non-linearity to the VAR part of the FAVAR by adopting the time-varying parameter VAR (TVP-VAR, Primiceri (2005)), allowing the VAR parameters to evolve as random walks.

In our empirical application to the U.S. economy, we first compare factors extracted by the GS autoencoder with those obtained through PCA. The GS autoencoder factors exhibit superior interpretability due to their parsimonious structure. Examining correlations between factors and high-dimensional time series reveals that each factor from the GS autoencoder shows a stronger correlation with data in its corresponding anchor group compared to non-anchor groups. Assessment of point and density forecasting performance demonstrates that the GS autoencoder combined with the TVP-VAR outperforms models using either linear dimension reduction methods or time-invariant VAR parameters. Our impulse response analysis reveals that the transmission of monetary policy to economic indicators is time-varying, with higher uncertainty observed during the COVID-19 pandemic relative to other periods.

This chapter is organized as follows. Section 5.1 provides the challenges of the standard autoencoder. Section 5.2 introduces the grouped sparse (GS) autoencoder and provides details of the TVP-VAR applied. Section 5.3 illustrates the estimation of unknown parameters. Section 5.4 starts with a description of the data and implementation details, then demonstrates the utility of our proposed model in three areas: factor interpretation, forecasting performance, and impulse response analysis. Section 5.5 concludes the chapter.

5.1 Challenges of Standard Autoencoder

While the standard autoencoder, defined in (2.35) and (2.36) with the MLPs, has been employed to extract latent factors in economic applications, a prerequisite for their use in downstream tasks is the identifiability of these factors. Specifically, if two sets of parameters and factors, $\{\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{f}_t\}$ and $\{\boldsymbol{\theta}^*, \boldsymbol{\phi}^*, \boldsymbol{f}_t^*\}$, give the same $\hat{\boldsymbol{x}}_t$ (i.e., the reconstruction of $\boldsymbol{x}_t$), then $\boldsymbol{f}_t = \boldsymbol{f}_t^*$, for $t \in [T]$[34]. The first challenge of the standard autoencoder is that these factors are generally *not* identified (Locatello et al., 2019). We illustrate this point with two examples.

Example 1. Similar to the rotational invariance in the linear FAVAR, assume $\boldsymbol{f}_t$ and $\boldsymbol{f}_t^* = \boldsymbol{Q}\boldsymbol{f}_t$, for some invertible matrix $\boldsymbol{Q}$, these two sets of factors construct the same $\hat{\boldsymbol{x}}_t$ if the weights in the two autoencoders satisfy $\tilde{\boldsymbol{W}}_1^d = \boldsymbol{Q}^{-1}\boldsymbol{W}_1^d$ and $\tilde{\boldsymbol{W}}_l^d = \boldsymbol{W}_l^d$, for $l = 2, \ldots, L$.

Example 2. If $\boldsymbol{\theta} = \boldsymbol{\theta}^*$ and $\boldsymbol{\phi} \neq \boldsymbol{\phi}^*$, the factors are potentially non-identifiable when the decoder, $g_{\boldsymbol{\theta}}^d(\cdot)$, is non-injective, which arises from two cases: 1) when using non-injective activation functions such as ReLU, 2) when the weights do not have full column ranks.

Identifiable latent factors are useful in practice, because if the underlying high-dimensional data exhibit patterns, identifiable factors can preserve these patterns[35]. However, many standard deep learning implementations can result in non-identifiable latent factors. These implementations introduce randomness to the model itself and/or during the training process to explore the

[34] Recall that $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ denote the parameters of encoder and decoder; $\boldsymbol{f}_t$ is the low-dimensional representation of high-dimensional data $\boldsymbol{x}_t$.

[35] Although the objective function of an autoencoder is generally non-convex, implying that the true global optimum in both parameters and factors is typically unattainable, semi-identifiable factors can still be useful in practice. For instance, factors that are identifiable only up to element-wise transformations preserve key patterns in the data. This phenomenon is illustrated in the simulations of Khemakhem et al. (2020), where the learned factors, despite differing from the true ones, successfully preserve the underlying cluster structure.

parameter space, improve training efficiency, and enhance model expressiveness. Examples include, but are not limited to, initializing parameters randomly, stochastic gradient descent mentioned in Section 2.4.1, and treating latent factors as random variables. Eliminating all these implementations is not ideal because they play a crucial role in deep learning. Thus, many recent efforts in the deep learning literature alleviate the indeterminacy through two streams: 1) modifying the standard decoder structure (Lachapelle et al., 2024; Moran et al., 2021), and 2) imposing well-designed priors on $\boldsymbol{f}_t$ (Khemakhem et al., 2020; Klindt et al., 2020). Both streams aim to identify factors up to trivial transformations, such as element-wise transformations and permutations.

The second challenge of the standard autoencoder lies in its interpretability limitations. Like many dimension reduction methods, a standard autoencoder lacks practical interpretability because factors extracted can be important to multiple data categories. For example, this problem occurred in Klieber (2024), who extracted factors using different dimension reduction methods, such as standard autoencoder and PCA, in an econometrics application. Two prominent and closely related approaches to enhance the interpretability of the autoencoder are: inducing sparsity in the autoencoder components (typically the latent factors and/or the decoder) (Ainsworth et al., 2018; Moran et al., 2021; Tonolini et al., 2020), and developing autoencoder variants which promote disentanglement, where each latent factor represents a distinct meaningful aspect of the high-dimensional data, see Wang et al. (2022b) for a review.

5.2 Methodology

This section describes the non-linearity applied in the two parts of the FAVAR: the factor extraction and VAR parts. Section 5.2.1 introduces a variant of the standard autoencoder, namely the grouped sparse (GS) autoencoder, which can alleviate both the identifiability and interpretability challenges aforementioned. Section 5.2.2 specifies the factor-augmented time-varying VAR.

5.2.1 Grouped Sparse Autoencoder

We aim to construct a variant of the standard autoencoder that enhances both identifiability and interpretability. The sparse autoencoder proposed by Moran et al. (2021) provides a promising foundation, as it identifies latent factors up to element-wise transformation and induces sparsity through the SSL prior to decoder parameters (see its description after (5.2) and (5.3)). To further improve the interpretability of the sparse autoencoder, we extend it to the grouped sparse (GS) autoencoder using group-specific SSL parameters. The grouping effect is justifiable because economic data inherently falls into different categories, with well-established divisions such as labor market, output, and interest rates, among others. In the FAVAR literature, Belviso and Milani (2006) and Korobilis (2013a) divided data into different groups and extracted each factor from one group using the PCA.

The GS autoencoder has the same encoder as in (2.35), the mathematical expression of the decoder in (2.36) changes to

$$\hat{\boldsymbol{x}}_{t,i} = g^d_{i,\boldsymbol{\theta}}\left(\boldsymbol{f}_t \odot \boldsymbol{\beta}_{c_i}\right) = \left(g^d_{i,L} \circ g^d_{L-1} \circ \cdots \circ g^d_1\right)_{\boldsymbol{\theta}}\left(\boldsymbol{f}_t \odot \boldsymbol{\beta}_{c_i}\right), \tag{5.1}$$

where $\hat{\boldsymbol{x}}_t$ is the reconstruction of $\boldsymbol{x}_t$, for $t \in [T]$, $i \in [M]$ is the index of variables in $\boldsymbol{x}_t \in \mathbb{R}^M$, $g^d_{i,\boldsymbol{\theta}}(\cdot) = \left(g^d_{i,L} \circ g^d_{L-1} \circ \cdots \circ g^d_1\right)_{\boldsymbol{\theta}}(\cdot)$ is the decoder that reconstructs $\boldsymbol{x}_{t,i}$ (this decoder takes a K-dimensional input and outputs a scalar), g^d_l, for $l \in [L-1]$, denotes the l-th function in this decoder and is identical across i, $\boldsymbol{\beta}_{c_i} = (\beta_{c_i,1}, \ldots, \beta_{c_i,K}) \in \mathbb{R}^K$ stores sparsity parameters corresponding to the group of the i-th variable, c_i , $\odot$ means element-wise multiplication[36].

The distribution that $\boldsymbol{\beta}_{c_i}$ follows is a SSL, so this autoencoder can turn on or off each factor when reconstructing a group of variables. Specifically,

$$\boldsymbol{\beta}_{c_i,k} \sim \gamma_{c_i,k}\psi_1(\boldsymbol{\beta}_{c_i,k}) + (1-\gamma_{c_i,k})\,\psi_0(\boldsymbol{\beta}_{c_i,k}), \tag{5.2}$$

$$\gamma_{c_i,k} \sim \text{Bernoulli}\,(0.5)\,. \tag{5.3}$$

where $\psi_s(\beta) = \frac{\lambda_s}{2}\exp(-\lambda_s|\beta|)$ for s = 0 or 1, is the Laplace distribution with $\lambda_0 \gg \lambda_1$; $\gamma_{c_i,k}$ is a binary variable that determines the shrinkage level of $\boldsymbol{\beta}_{c_i,k}$, for $k \in [K]$. If $\gamma_{c_i,k} = 0$, the $\boldsymbol{\beta}_{c_i,k}$ is shrunk to zero with a higher probability, and vice versa.

If we allow a sparsity parameter for each variable (i.e., the group size is one) in (5.1)-(5.3), then the model is the sparse autoencoder defined in Moran et al. (2021). The autoencoder in Ainsworth et al. (2018) is also closely related to ours. This autoencoder imposes the Bayesian lasso prior (Park and Casella, 2008) on the weights that produce the first hidden layer in the decoder. The corresponding weight shares the same degree of sparsity for the variables within the same group. The difference between this model and ours is twofold. Firstly, we induce sparsity to $\boldsymbol{\beta}_{c_i}$, for $i \in [M]$, which is different to the weights. We adopt this structure because it facilitates the proof of identifiability that we will discuss later. Secondly, we adopt the SSL

[36] Note that $\odot$ is not the Khatri–Rao product defined related to the tensor decomposition.

because this prior addresses the issue of the Bayesian lasso Bai et al. (2021), which tends to under-regularize large coefficients and over-regularize small coefficients (Ghosh et al., 2015).

Next, we leverage the concept of "anchor" to show that the latent factors in the GS autoencoder are identifiable up to element-wise transformation. This concept originated in identifiable linear models (Arora et al., 2013; Bing et al., 2020a,b). For example, Arora et al. (2013) defined anchor words that anchor the topics of documents – if a document contains an anchor word of a particular topic, this document must be about this topic. Similarly, Moran et al. (2021) posited the existence of "anchor features", which exclusively load on their corresponding factors, enabling factor identification in their sparse autoencoder. Following this approach, we assume that each factor has a known *anchor group* with the following definition.

Definition 1. For $k \in [K]$, a data category c is an anchor group of factor k, if $\boldsymbol{\beta}_{c,k} \neq 0$ and $\boldsymbol{\beta}_{c,k'} = 0$ for all $k' \neq k$.

This definition implies that if the i-th variable is in the anchor group of the k-th factor, then only this factor reconstructs the variable. Mathematically, $g^d_{i,\boldsymbol{\theta}}(\boldsymbol{f}_t \odot \boldsymbol{\beta}_c)$ can be simplified to $\tilde{g}^d_{i,\boldsymbol{\theta}}\left(\boldsymbol{f}_{t,k}\boldsymbol{\beta}_{c,k}\right) = \left(g^d_{i,L} \circ g^d_{L-1} \circ \cdots \circ g^d_2 \circ \tilde{g}^d_{1,k}\right)_{\boldsymbol{\theta}}\left(\boldsymbol{f}_{t,k}\boldsymbol{\beta}_{c,k}\right)$, where $\tilde{g}^d_{1,k}$ is the part of g^d_1 that is associated with the k-th factor. Denote $\boldsymbol{B} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_C)$, for C number of data categories, we have

Theorem 1. Suppose the following assumptions hold:

(1) The decoder follows (5.1) with C number of data categories.

(2) Each factor has a known anchor group.

(3) Within each anchor group, there exists at least one variable, denoted as $\boldsymbol{x}_{t,i}$, such that the associated decoder $\tilde{g}^d_{i,\boldsymbol{\theta}}(\cdot)$ is injective.

If we have two sets of decoder parameters and factors $\{\boldsymbol{\theta}, \boldsymbol{B}, \boldsymbol{f}_t\}$ and $\{\boldsymbol{\theta}^*, \boldsymbol{B}^*, \boldsymbol{f}^*_t\}$, which yield the same reconstructions of $\hat{\boldsymbol{x}}_t$, for $t \in [T]$, then the recovery of $\boldsymbol{f}_{t,k}$ only depends on $\boldsymbol{f}^*_{t,k}$ and parameters learned in the decoders, i.e., $\boldsymbol{f}_{t,k}$ is identified up to element-wise transformations, $h_k(\cdot)$, for $k \in [K]$.

Proof. For $k \in [K]$, suppose that the i-th variable is from the anchor group of the k-th factor and $\tilde{g}^d_{i,\boldsymbol{\theta}}(\cdot)$ is injective. Since the two trained decoders yield the same reconstruction $\hat{\boldsymbol{x}}_{t,i}$, there exists a non-empty set $\mathbb{S} = \left\{ x \in \mathbb{R} : x = \tilde{g}^d_{i,\boldsymbol{\theta}} \left(\boldsymbol{f}_{t,k} \boldsymbol{\beta}_{c_i,k} \right) = \tilde{g}^d_{i,\boldsymbol{\theta}^*} \left(\boldsymbol{f}^*_{t,k} \boldsymbol{\beta}^*_{c_i,k} \right), \text{for } \boldsymbol{f}_{t,k}, \boldsymbol{f}^*_{t,k} \in \mathbb{R} \right\} \subseteq$ $\operatorname{Im}\left(\tilde{g}^d_{i,\boldsymbol{\theta}}\right) \cup \operatorname{Im}\left(\tilde{g}^d_{i,\boldsymbol{\theta}^*}\right)$, where $\operatorname{Im}(f) = \{f(x),\, x \in \mathbb{X}\} \subseteq \mathbb{Y}$ for a function $f : \mathbb{X} \to \mathbb{Y}$. As $\tilde{g}^d_{i,\boldsymbol{\theta}}(\cdot)$ is injective, there is a one-to-one mapping $h_1 : \operatorname{Im}\left(\tilde{g}^d_{i,\boldsymbol{\theta}}\right) \to \mathbb{R}$ such that $h_1\left(\boldsymbol{x}_{t,i}\right) = \boldsymbol{f}_{t,k}$ for all $\boldsymbol{x}_{t,i} \in \operatorname{Im}\left(\tilde{g}^d_{i,\boldsymbol{\theta}}\right)$ and $\boldsymbol{f}_{t,k} \in \mathbb{R}$ satisfying $\tilde{g}^d_{i,\boldsymbol{\theta}}\left(\boldsymbol{f}_{t,k}\boldsymbol{\beta}_{c_i,k}\right) = \boldsymbol{x}_{t,i}$. As $\mathbb{S} \subseteq \operatorname{Im}\left(\tilde{g}^d_{i,\boldsymbol{\theta}}\right)$ and $\mathbb{S} \subseteq \operatorname{Im}\left(\tilde{g}^d_{i,\boldsymbol{\theta}^*}\right)$, $h_1(\cdot)$ and $\tilde{g}^d_{i,\boldsymbol{\theta}^*}(\cdot)$ are well-defined from $\mathbb{S}$ to $\mathbb{R}$ and from $\mathbb{R}$ to $\mathbb{S}$, respectively. Therefore, we can define an element-wise transformation $h_k(\boldsymbol{f}^*_{t,k}) = h_1 \circ \tilde{g}^d_{i_k,\boldsymbol{\theta}^*}\left(\boldsymbol{f}^*_{t,k}\boldsymbol{\beta}^*_{c_i,k}\right) = \boldsymbol{f}_{t,k}$. □

The identifiability in Theorem 1 mitigates rotational invariance and any invariance involving transformations that require multiple factors. Even though the factors are semi-identifiable, which is weaker than the canonical one such that $\boldsymbol{f}_{t,k} = \boldsymbol{f}^*_{t,k}$, for $t \in [T]$ and $k \in [K]$, in practice, we find that the GS autoencoder effectively identifies most factors after standardization and sign switching.

We shed light on the decoder architecture and choice of activation function according to the third assumption of Theorem 1. While this assumption only requires one variable (denoted as i) in each anchor group, such that the corresponding decoder $\tilde{g}^d_{i,\boldsymbol{\theta}}$ is injective, we implement identical architecture and activation functions across all decoders. We consider the decoder to be an MLP due to its simplicity. According to Wang et al. (2021), this decoder is injective if two conditions are satisfied: (1) the activation function is injective, and (2) the weights, $\boldsymbol{W}^d_l \in \mathbb{R}^{D^d_l \times D^d_{l-1}}$ (see (2.31) for the notation) have full column rank, for all $l \in [L]$. To satisfy the first condition, we use 5-fold cross-validation to select an injective activation function between tanh($\cdot$) and leaky ReLU, $g\left(x\right)$ = max(ax, x), where a is a multiplier smaller than 1, due to their popularity in the deep learning literature. This cross-validation will also determine the number of factors and layers. We assume that the second condition regarding full column ranks holds. This assumption is feasible when $D^d_l \geq D^d_{l-l}$, which is the architecture commonly applied in a standard autoencoder. In the grouped sparse autoencoder, this inequality constraint can be directly implemented when $l < L$, but it is not as straightforward when $l = L$, because the corresponding matrix has dimension 1-by-D^d_{L-1}. Incorporating the constraint for $l < L$ results in $D^d_{L-1} \geq D^d_0 = K$, and K is generally higher than 1. To ensure a feasible assumption, we modify this output layer matrix, $\boldsymbol{W}^d_{L,i}$, to $\left((\boldsymbol{W}^d_{L,i})', \boldsymbol{I}'_{D^d_{L-1},(2:D^d_{L-1},\cdot)}\right)'$, where $\boldsymbol{I}_{D^d_{L-1},(2:D^d_{L-1},\cdot)}$ is the second to the last rows of a D_{L-1}-by-D_{L-1} identity matrix. With this modification, the output layer now has dimension D^d_{L-1}, of which the first element is the reconstruction $\hat{\boldsymbol{x}}_{t,i}$ and the remaining ones record $D^d_{L-1} - 1$ elements in the last hidden layer. Thus, based on these constraints and supposing that the two conditions hold, factors are identifiable up to element-wise transformation given these recorded elements in the last hidden layer.

5.2.2 TVP-FAVAR

After training the GS autoencoder using gradient descent, a time series model is required to specify the factor dynamics and facilitate subsequent analysis. Specifically, we adopt a TVP-FAVAR framework defined as follows:

$$\begin{pmatrix} \boldsymbol{x}_t \\ \boldsymbol{y}_t \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Lambda}_{xf} & \boldsymbol{\Lambda}_{xy} \\ \boldsymbol{0} & \boldsymbol{I}_N \end{pmatrix} \begin{pmatrix} \boldsymbol{f}_t \\ \boldsymbol{y}_t \end{pmatrix} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}), \tag{5.4}$$

$$\begin{pmatrix} \boldsymbol{f}_t \\ \boldsymbol{y}_t \end{pmatrix} = \boldsymbol{A}_{t,1} \begin{pmatrix} \boldsymbol{f}_{t-1} \\ \boldsymbol{y}_{t-1} \end{pmatrix} + \cdots + \boldsymbol{A}_{t,P} \begin{pmatrix} \boldsymbol{f}_{t-P} \\ \boldsymbol{y}_{t-P} \end{pmatrix} + \boldsymbol{u}_t, \quad \boldsymbol{u}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Omega}_t), \tag{5.5}$$

where $\boldsymbol{y} \in \mathbb{R}^t$, $\boldsymbol{A}_{t,p} \in \mathbb{R}^{(N+K)\times(N+K)}$ for $t \in [T], p \in [P]$, $\boldsymbol{\Omega}_t = \boldsymbol{H}_t^{-1}\boldsymbol{S}_t(\boldsymbol{H}_t^{-1})'$, $\boldsymbol{H}_t$ is a lower triangular matrix with unit diagonal terms and $\boldsymbol{S}_t$ is a diagonal matrix. Equation (5.5) is a TVP-VAR time-varying parameters following (geometric) random walks defined in Section 2.1.5. For clarity, the random walk formulations are restated below:

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} + \boldsymbol{u}_t^{\alpha}, \quad \boldsymbol{u}_t^{\alpha} \sim \mathcal{N}\left(\boldsymbol{0}, \Sigma_{\alpha}\right), \tag{5.6}$$

$$\boldsymbol{h}_t = \boldsymbol{h}_{t-1} + \boldsymbol{u}_t^{h}, \quad \boldsymbol{u}_t^{h} \sim \mathcal{N}\left(\boldsymbol{0}, \Sigma_{h}\right), \tag{5.7}$$

$$\log \boldsymbol{s}_t = \log \boldsymbol{s}_{t-1} + \boldsymbol{u}_t^{s}, \quad \boldsymbol{u}_t^{s} \sim \mathcal{N}\left(\boldsymbol{0}, \Sigma_{s}\right), \tag{5.8}$$

where $\boldsymbol{\alpha}_t = \mathrm{vec}(\boldsymbol{A}_t')$, $\boldsymbol{h}_t$ denotes all non-zero and non-unit elements in $\boldsymbol{H}_t$, $\boldsymbol{s}_t$ corresponds to diagonal terms in $\boldsymbol{S}_t$. Σ_{α} and Σ_s are unconstrained symmetric positive definite matrices, and Σ_h is block diagonal such that the elements within the same row in $\boldsymbol{H}_t$ are correlated, while elements across different rows are uncorrelated.

The TVP-FAVAR is a suitable model here for two reasons. First, (5.4) approximates a linear relation between $(\boldsymbol{f}_t', \boldsymbol{y}_t')'$ and $\boldsymbol{x}_t$[37], which allows straightforward derivation of impulse responses of $\boldsymbol{x}_t$ from the shocks of $(\boldsymbol{f}_t', \boldsymbol{y}_t')'$ (the details will be introduced in the next para-

[37] This linear approximation seems to contradict with the non-linear dimension reduction from $\boldsymbol{x}_t$ to $\boldsymbol{f}_t$, but it is noteworthy that the goal of non-linear dimension reduction is to extract the factors which are more representative than those extracted from linear models. As we employ a two-step procedure in the TVP-FAVAR, the factors are still extracted from a non-linear model in the first step, and no linear model is involved in this step.

graph). Secondly, as the factors are extracted from $\boldsymbol{x}_t$ via a non-linear autoencoder, it is natural to also model the evolution of factors as non-linear, so a TVP-VAR model is suitable in this sense.

Two kinds of impulse responses can be studied based on this TVP-FAVAR model. The first kind corresponds to $\boldsymbol{y}_{t+h}$. The IRF of the i-th variable in $\boldsymbol{y}_{t+h}$ to the shock of the j-th variable of $\boldsymbol{y}_t$ (h is the horizon, $i, j \in [N]$) is the (i, j) entry of $\boldsymbol{\Psi}_{t,h}$, where $\boldsymbol{\Psi}_{t,h} = \boldsymbol{\Phi}_{t,h}\boldsymbol{H}_t$, and $\boldsymbol{\Phi}_{t,h} = \sum_{h'=1}^{h} \boldsymbol{\Phi}_{t,h-h'}\boldsymbol{A}_{t,h'}$, for $h > 0$.. The second kind corresponds to $\boldsymbol{x}_{t+h}$. For the IRF of the i'-th variable in $\boldsymbol{y}_{t+h}$ to the shock of the j-th variable of $\boldsymbol{y}_t$ is the (i', j) entry of $\boldsymbol{\Lambda\Psi}_{t,h}$, where $\boldsymbol{\Lambda} = \begin{pmatrix} \boldsymbol{\Lambda}_{xf} & \boldsymbol{\Lambda}_{xy} \\ \boldsymbol{0} & \boldsymbol{I}_N \end{pmatrix}$.

5.3 Estimation

As introduced in Section 2.2.2, the inference of a standard FAVAR can follow the one-step or two-step procedures. Given the non-linearity in both the factor extraction and VAR parts, we estimate unknown parameters in the GS autoencoder and the TVP-VAR using the two-step procedure, for which standard implementations of gradient descent and Bayesian inference are sufficient. In contrast, the one-step procedure requires more advanced deep learning frameworks such as dynamical VAE (Giannone et al., 2015) and Bayesian neural networks (Goan and Fookes, 2020), see the references therein, which extend beyond the scope of this study.

The following subsections describe each step in turn. Section 5.3.1 provides details about training the GS autoencoder, which optimizes $\{\boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{B}, \boldsymbol{\Gamma}\}$, where $\boldsymbol{\Gamma} = (\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_C)$ with $\boldsymbol{\gamma}_c = (\gamma_{c,1}, \ldots, \gamma_{c,K})'$. Section 5.3.2 infers the remaining parameters, $\{\boldsymbol{\alpha}_{1:T}, \boldsymbol{h}_{1:T}, \boldsymbol{s}_{1:T}, \boldsymbol{\Sigma}_\alpha, \boldsymbol{\Sigma}_h, \boldsymbol{\Sigma}_s, \boldsymbol{\Lambda}, \boldsymbol{\Sigma}\}$, where the first 6 blocks correspond to the TVP-VAR defined in Section 2.1.5 and the last 2 blocks are the non-zero factor loading and the variance-covariance of the FAVAR, see (2.20).

5.3.1 Training the Deep Learning Model

The training process is essentially to maximize the marginal loglikelihood $\log p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{B}}\left(\boldsymbol{x}_{1:T}\right)$. We assume the distribution of $\boldsymbol{x}_{t,i}$ (for $t \in [T]$ and $i \in [M]$) as $\mathcal{N}\left(\hat{\boldsymbol{x}}_{t,i}, 1\right)$. The fixed variance of 1 is justified as we normalize $\boldsymbol{x}_{1:T}$ to have unit variance, so a variance of 1 is consistent with the scale of the data[38]. Since this likelihood is intractable due to the non-linear activation function,

[38] A more flexible variance choice can be a variable following an inverse-gamma prior or a time-varying one following stochastic volatility.

we maximize an objective function that is an evidence lower bound (ELBO) of the likelihood instead.

The objective function is written as

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{B}) = -\frac{1}{2T} \sum_{t=1}^{T} \text{MSE}(\boldsymbol{x}_t, \hat{\boldsymbol{x}}_t) + \frac{1}{T}\frac{1}{M} \sum_{c=1}^{C} \sum_{k=1}^{K} \left[p_{c,k} \left(\log \psi_1 \left(\boldsymbol{\beta}_{c,k} \right) - \log p_{c,k} \right) \right. \tag{5.9}$$

$$\left. + (1 - p_{c,k}) \left(\log \psi_0 \left(\boldsymbol{\beta}_{c,k} \right) - \log (1 - p_{c,k}) \right) \right], \tag{5.10}$$

where MSE denotes mean squared error, $p_{c,k} = \mathbb{E}[\gamma_{c,k} \mid \boldsymbol{\beta}_{c,k}] = \frac{\psi_1(\boldsymbol{\beta}_{c,k})}{\psi_0(\boldsymbol{\beta}_{c,k}) + \psi_1(\boldsymbol{\beta}_{c,k})}$. The derivation of this objective function is in Appendix D.2.

This ELBO is composed of two parts. The mean squared error part guides the latent factors to form effective low-dimensional representations of $\boldsymbol{x}_{1:T}$. The remaining part regularizes $\boldsymbol{B}$ to follow a SSL prior. Algorithm 7 summarizes the training of all parameters in the GS autoencoder. λ_0 and λ_1 are chosen by the cross-validation. j presents the j-th iteration, which passes all data points in the training process. We follow the deep learning standard to use the Adam defined in Section 2.4.1, and set the epochs and batch size as 200 and 24, respectively.

Algorithm 7 Training the GS autoencoder

```
1: **Input:** $\boldsymbol{x}_{1:T}$, $\lambda_0$ and $\lambda_1$.
2: **Output:** $\boldsymbol{\phi}$, $\boldsymbol{\theta}$, $\boldsymbol{B}$ and $\boldsymbol{\Gamma}$.
3: for $j$ in $1, \ldots,$ epochs do:
4:     for each batch do:
5:         Update $\boldsymbol{\phi}$, $\boldsymbol{\theta}$, $\boldsymbol{B}$ according to $\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{B})$ with Adam.
6:         Update $p_{c,k} = \mathbb{E}[\gamma_{c,k} \mid \boldsymbol{\beta}_{c,k}] = \frac{\psi_1(\boldsymbol{\beta}_{c,k})}{\psi_0(\boldsymbol{\beta}_{c,k}) + \psi_1(\boldsymbol{\beta}_{c,k})}$, for $c \in [C]$ and $k \in [K]$.
7:     end for
8: end for
```

5.3.2 Bayesian Inference

We use Bayesian inference to estimate the remaining parameters in this non-linear FAVAR model. To induce parsimony, we follow Korobilis (2013a) to set a Minnesota prior to $\boldsymbol{\alpha}_0 = \text{vec}\left((\boldsymbol{A}_{0,1}, \ldots, \boldsymbol{A}_{0,P})'\right)$, the initialization of the coefficient matrix. Specifically, we set the hyperparameters in the prior defined in (2.10) as $\lambda_1 = 0.7$ and $\lambda_2 = 0.1$, mitigating explosive draws during the MCMC. The priors of $\boldsymbol{h}_0$ and $\log \boldsymbol{s}_0$ are multivariate normal distributions with zero means and diagonal variance-covariance matrices, where the diagonal terms are 4. Denote the variance-covariance matrix of $\boldsymbol{\alpha}_0$,

$\boldsymbol{h}_{0,m}$ (non-zero entries on the m-th row of $\boldsymbol{H}_0$ for $m = 2, \ldots, N+K$) and $\log \boldsymbol{s}_0$ as $\underline{\boldsymbol{V}}_a$, $\underline{\boldsymbol{V}}_{h,m}$ and $\underline{\boldsymbol{V}}_s$, We specify priors of the variance-covariance matrices corresponding the random walks as $\boldsymbol{\Sigma}_\alpha \sim \mathcal{IW}\,(\text{dim}(\boldsymbol{\alpha}_0) + 1, 0.0001 \times (\text{dim}(\boldsymbol{\alpha}_0) + 1) \times \underline{\boldsymbol{V}}_\alpha)$, $\boldsymbol{\Sigma}_{h,m} \sim \mathcal{IW}\left(\text{dim}(\boldsymbol{h}_{0,m}) + 1, 0.0001 \times (\text{dim}(\boldsymbol{h}_{0,m}) + 1) \times \underline{\boldsymbol{V}}_{h,m}\right)$, then $\boldsymbol{\Sigma}_s \sim \mathcal{IW}\,(\text{dim}(\boldsymbol{s}_0) + 1,$ $0.01 \times (\text{dim}(\boldsymbol{s}_0) + 1) \times \underline{\boldsymbol{V}}_s)$, where $\text{dim}(\boldsymbol{a}_0) = P(N+K)^2$, $\text{dim}(\boldsymbol{h}_{0,m}) = m - 1$ and $\text{dim}(\boldsymbol{s}_0) = N + K$. For the linear approximation of the factor model, $\text{vec}(\boldsymbol{\Lambda}) \sim \mathcal{N}\left(\boldsymbol{0}, 4\boldsymbol{I}_{M(N+K)}\right)$ and $\boldsymbol{\Sigma}_{(i,i)} \sim \mathcal{IG}\,(0.01, 0.01)$ for $i \in [M]$.

We use the standard MCMC algorithm for TVP-VARs specified in Section 2.1.5 to sample $\boldsymbol{\alpha}$, $\boldsymbol{h}_t$, and $\boldsymbol{s}_t$ (for $t \in [T]$). In particular, a forward filtering backward sampling procedure (see the detail in Appendix A.2) updates each set of parameters mentioned. We implemented the algorithm using the R package **bvarsv** (Krueger, 2015).

The full conditional of $\boldsymbol{\Lambda}_{(i,\cdot)}$, the i-th row of Λ, for $i \in [M]$, is $\mathcal{N}\left(\overline{\boldsymbol{m}}_i, \overline{\boldsymbol{V}}_i\right)$, with

$$\overline{\boldsymbol{V}}_i^{-1} = \frac{1}{4}\boldsymbol{I}_{M(N+K)} + \boldsymbol{\Sigma}_{(i,i)}^{-1} \left(\boldsymbol{F}, \boldsymbol{Y}\right)' \left(\boldsymbol{F}, \boldsymbol{Y}\right),$$

$$\overline{\boldsymbol{m}}_i = \boldsymbol{\Sigma}_{(i,i)}^{-1} \overline{\boldsymbol{V}}_i \left(\boldsymbol{F}, \boldsymbol{Y}\right)' \boldsymbol{X},$$

where $\boldsymbol{F} = (\boldsymbol{f}_1, \ldots, \boldsymbol{f}_T)'$, $\boldsymbol{Y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T)'$, $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)'$. The full conditional of $\boldsymbol{\Sigma}_{(i,i)}$ is $\mathcal{IG}(T/2 + 0.005, \boldsymbol{\epsilon}_i'\boldsymbol{\epsilon}_i/2 + 0.005)$, where $\boldsymbol{\epsilon}_i = (\epsilon_{1,i}, \ldots, \epsilon_{T,i})'$ denotes the residual of the factor model. In the real data application, we sample these parameters with 10,000 iterations of burn-in and 100,000 iterations of MCMC sampling.

5.4 Empirical Results

5.4.1 Data and Implementation Detail

We use 168 quarterly U.S. macroeconomic variables in McCracken and Ng (2020) to demonstrate the utility of the proposed model. The data ranges from 1965:Q1 to 2023:Q1, and is divided into 12 groups: (1) national income and product accounts (NIPA), (2) industrial production, (3) earning and productivity, (4) labor market, (5) housing, (6) inventory, orders and sales, (7) prices, (8) interest rate, (9) money and credit, (10) household balance sheets, (11) exchange rates and (12) stock market. All time series are transformed according to McCracken and Ng (2020) and standardized, as is conventional in the FAVAR literature. Readers can refer to Appendix D.5 for more details about the data. For all FAVAR models considered in this application, we follow Korobilis (2013a) to choose the observable factors ($\boldsymbol{y}_t \in \mathbb{R}^3$) as the gross domestic product: implicit price deflator (GDPDEF), unemployment rate (UNRATE) and effective Federal funds rate (FEDFUNDS), which are the proxies of inflation, labor market and interest rate, respectively. The other variables construct the high-dimensional variable $\boldsymbol{x}_t \in \mathbb{R}^{165}$ that we extract latent factors from.

There are 4 dimension reduction methods considered in this study: PCA, standard autoencoder, the GS autoencoders with either identity or non-linear activation functions. The former GS autoencoder is a linear model analogous to the structural FAVAR (Belviso and Milani, 2006), and the latter is non-linear. We present the implementation of the non-linear GS autoencoder, as it is the most general case. Readers can adapt relevant components of this implementation to other dimension reduction methods of interest. In particular, the PCA only requires specifying the number of factors, the standard autoencoder implementation does not involve anchor groups or the SSL, and the linear GS autoencoder replaces the non-linear activation function with an identity function.

According to the cross-validation result, we extract 5 factors from the high-dimensional data, and find that the same number of principal components explain about 67% of the variation. The anchor groups of these factors are: NIPA, labor market, prices, interest rates, money and credit. We choose these 5 groups because they align as closely as possible with those considered in the structural FAVAR of Belviso and Milani (2006), and cover a large proportion of the variables. After setting the anchor groups, we assume that the first 5 rows of $\boldsymbol{\Gamma}$ (the matrix corresponding to $\boldsymbol{B}$ to indicate the spike or slab lasso, see definition in Section 5.3.1) form an identity matrix. For the SSL, the cross-validation suggests that the hyperparameter choices are $\lambda_0 = 1000$ and $\lambda_1 = 1$. As discussed in Ročková and George (2016), setting a very large value to λ_0 (like in this case) allows $\lambda_0 \approx \infty$ in practice, which leads to a Dirac spike at zero in the SSL. Thus, this hyperparameter choice further strengthens our assumption about anchor groups because the top 5 rows of $\boldsymbol{B}$ will be approximately diagonal.

Then we turn to the remaining architecture of the encoder $g^e_{\boldsymbol{\phi}}$ and decoder, $g^d_{i,\boldsymbol{\theta}}$, for $i \in [M]$. Three sets of hyperparameters need to be determined: the number of layers (L), the dimensions of layers in the encoder (D^e_l for $l \in [L]$) and those in the decoder (D^d_l for $l \in [L]$). The cross-validation chooses $L = 3$, then we can evenly downsize the dimensions from $M = 165$ to $K = 5$ and get $(D^e_1, D^e_2, D^e_3) = (111, 58, 5)$. For the layers in the decoder, we adopt a mirror structure of the encoder $\left(D^d_1, D^d_2, D^d_3\right) = (58, 111, 111)$, where the last dimension is 111 due to the modification mentioned in Section 5.2.1. Lastly, we select the activation function as the leaky ReLU with $a = 10^{-16}$ from the cross-validation result if the GS autoencoder is non-linear.

For the evolution of factors, we compare the time-invariant (TIV) and TVP model specifications. We set the lag order (P) to be 2, which is the same as that in Korobilis (2013a). The TIV one imposes a hierarchical Minnesota prior to the coefficient matrix and an inverse-Wishart prior to the variance-covariance matrix, see the definition of these priors in Section 2.1.2 and 2.1.4.

5.4.2 Analysis of Latent Factors

This subsection compares factors extracted via PCA and the non-linear GS autoencoder. While comparing the non-linear GS autoencoder and the standard autoencoder would be natural, we exclude the latter from this analysis due to its non-identifiable factors. Given that PCA factors are identifiable in the FAVAR literature and share similar interpretability limitations discussed in Section 5.1, they serve as suitable alternatives to the standard autoencoder factors for this comparison. The comparison of factors from the linear and non-linear GS autoencoders is in Appendix D.4. To facilitate comparison, we permute the PCA factors to maximize their correlation with their respective GS autoencoder counterparts. These permuted factors explain about 22%, 25%, 7%, 4%, and 8% of the data variation, respectively.

A conventional approach to interpreting FAVAR latent factors involves plotting factor time series alongside the variable in $\boldsymbol{x}_t$ exhibiting the highest correlation with each factor. However, this method overlooks other variables that also demonstrate strong correlations with the factors. Thus, we follow Klieber (2024) to record the variables with the highest 15 correlation magnitudes to each factor. Figure 5.1 - 5.5 present the time series of both PCA and the non-linear GS autoencoder factors, as well as the magnitudes of correlations between these factors and the 15 variables. The factors extracted from these dimension reduction methods show varying degrees of similarity. The first and second factors exhibit stronger similarity, with 7 and 11 common variables, respectively. While the third factors share 3 variables in common, the fourth and fifth factors show no overlap in their recorded variables.

The first factors from both methods capture recession periods, with the GS autoencoder factor showing stronger sensitivity to recent crises like the dot-com bubble, GFC, and COVID-19 pandemic. While the PCA factor correlates with various economic indicators (prices, labor market, NIPA, and industrial production) representing broad real activities, the GS autoencoder factor exhibits stronger correlations specifically with NIPA and industrial production variables, suggesting a more focused representation of these categories.

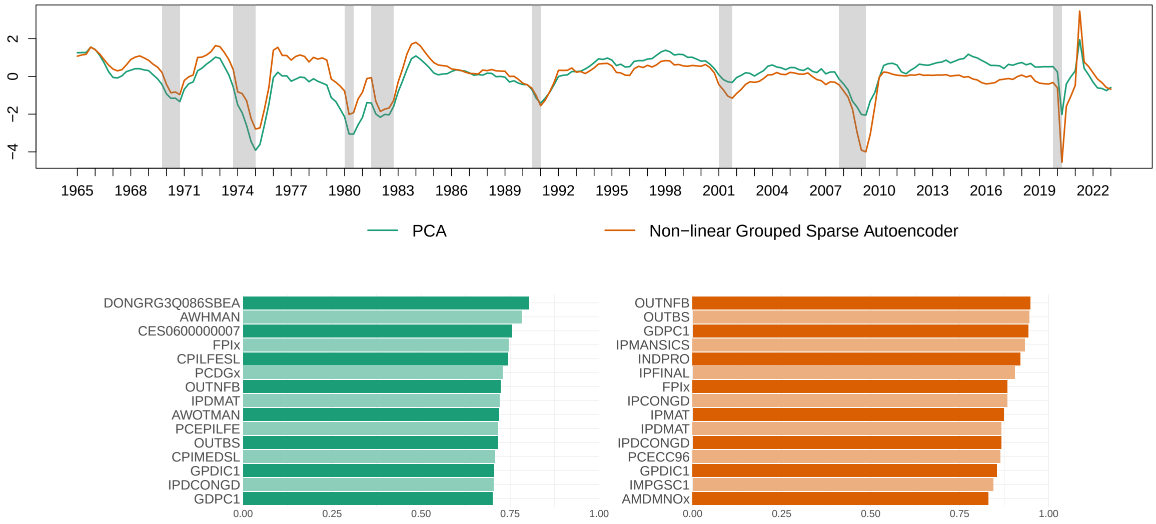


Figure 5.1: The first factor extracted from the PCA and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

The second factors from the two methods have strong correlations with labor market data. Given that both factors show pronounced spikes during recession periods, this factor can be interpreted as a measure of labor market distress. The non-linear factor is smoother than its PCA counterpart, especially post-1990s.

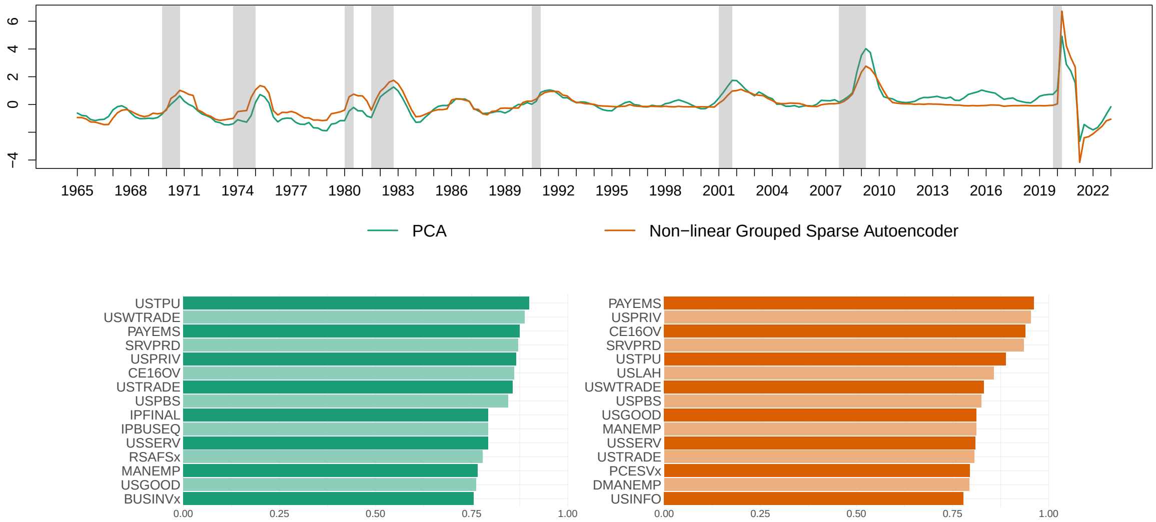


Figure 5.2: The second factor extracted from the PCA and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

The third factors from both methods relate to prices, but differ in their focus and correlation magnitudes. The GS autoencoder factor emphasizes consumption prices, while the PCA factor captures both consumption and producer prices. However, the GS autoencoder factor shows a consistently stronger correlation (>0.75) with its price variables compared to the PCA factor, where only one correlation exceeds 0.7.

The fourth factor extracted from the GS autoencoder clearly represents interest rates, as it captures the major monetary decisions of the Federal Reserve. This factor is strongly correlated with short and long-term interest rates as well as the price variables closely monitored by the Federal Reserve. Unlike the concentration of the GS autoencoder factor, the PCA factor exhibits broad correlations across labor market and price variables (similar to its second and third factors), making its economic interpretation less clear.

Analysis of the fifth factor suggests that they had a similar trend before 1995 and then

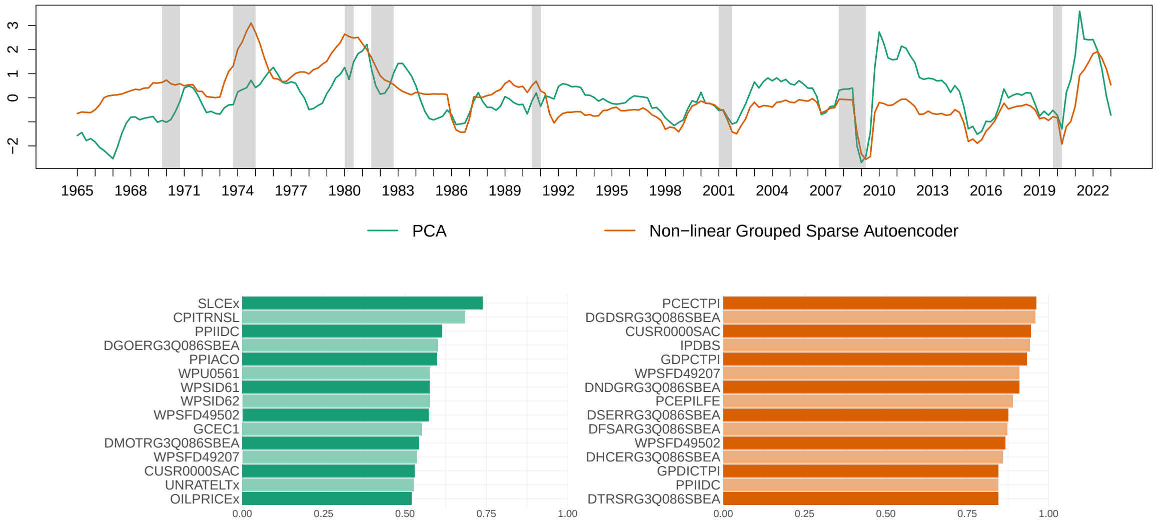


Figure 5.3: The third factor extracted from the PCA and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

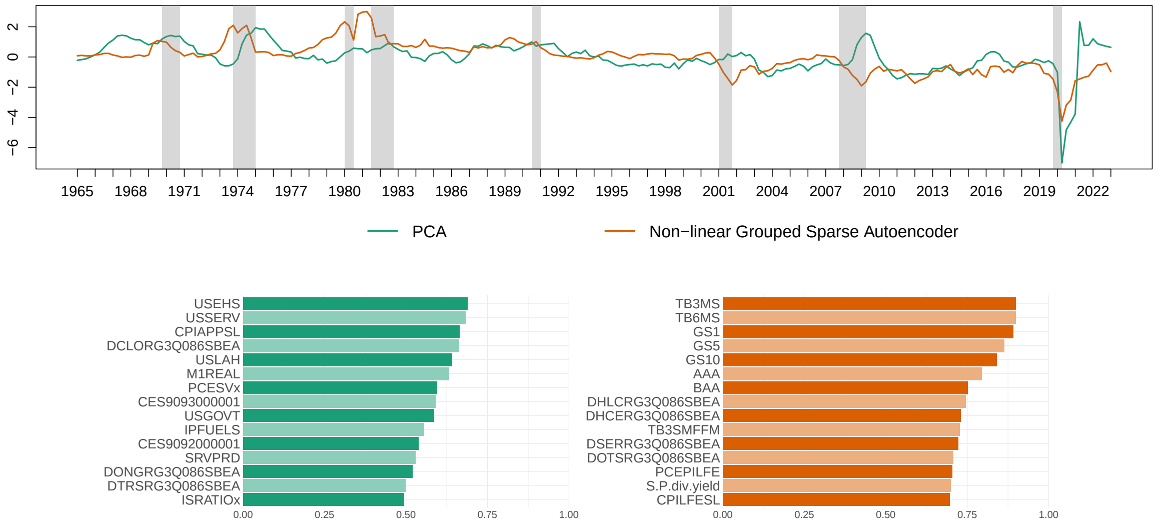


Figure 5.4: The fourth factor extracted from the PCA and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

turned out to be negatively correlated. The two factors represent different effects, as almost half of the variables corresponding to the PCA are about housing, while the non-linear factor is the only factor that reconstructs the money and credit variables, so it emphasizes more on them and those variables known to be related to this category, such as government and corporate yields and prices.

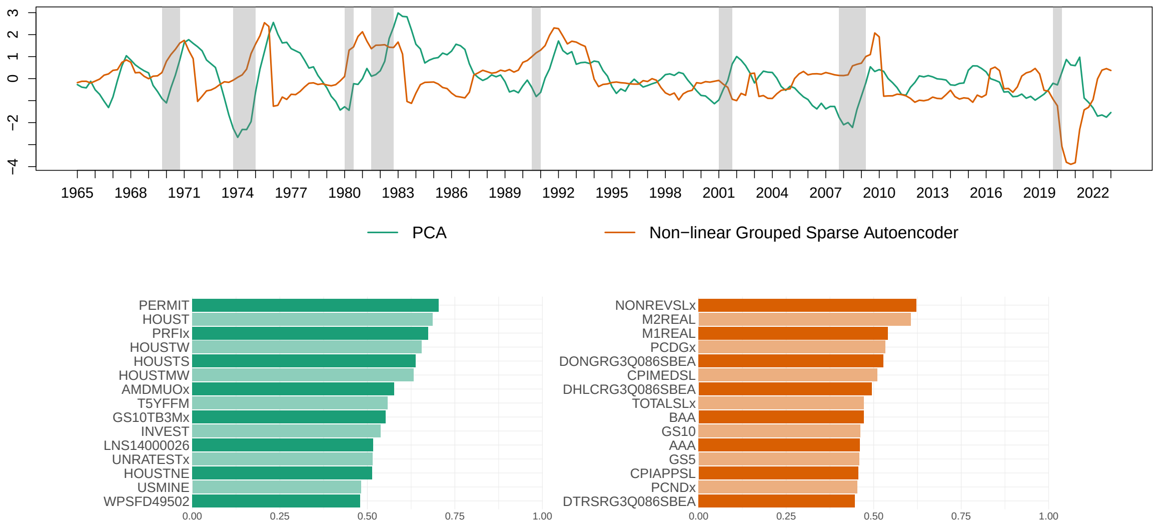
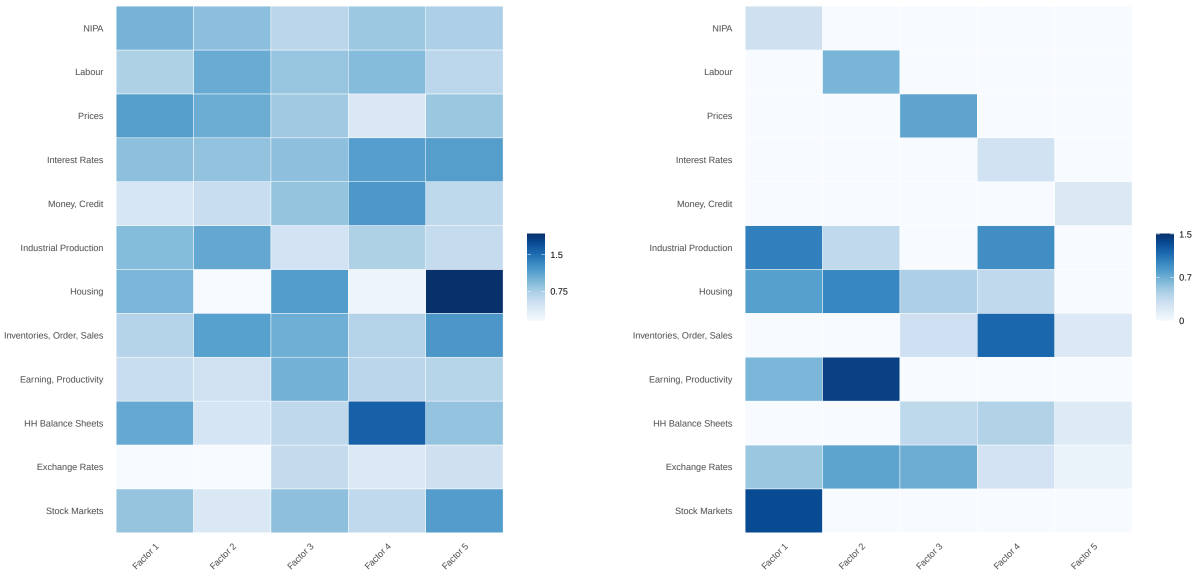


Figure 5.5: The fifth factor extracted from the PCA and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

While Figures 5.1 - 5.5 demonstrate that GS autoencoder factors exhibit stronger group-specific correlations than PCA factors, some factors (particularly the first and second ones) from both methods are highly correlated. However, despite these similarities, the GS autoencoder provides more interpretable relationships between these factors and the high-dimensional data. Figure 5.6 depicts the importance of factors to data categories. The importance measure on the left panel is the averaged PCA loading over these categories, and the right panel uses $\boldsymbol{B}$, the SSL parameters. Overall, the GS autoencoder heat map is sparser than the PCA one, indicating better interpretability from a parsimonious structure. For PCA, the fourth and fifth factors can be primarily attributed to household balance sheets and housing, respectively, but identifying the main drivers of the first three factors is more challenging due to the comparable scales of their importance measures. In contrast, we do not have this issue in the right panel since the first



(a) PCA

(b) Non-linear Grouped sparse autoencoder

Figure 5.6: Importance of factors to different categories. "HH Balance Sheets" means household balance sheets. Factors in the left panel is re-ordered so that each factor has a high correlation with the corresponding one in the right panel.

five categories are anchor groups. Thus, we can name these factors according to their anchor groups. For instance, the first factor is called the NIPA factor. There is also a sparser structure among the non-anchor groups in the right panel, so it is easier to determine the factors that reconstruct these categories. For example, industrial production variables are mainly driven by the NIPA and interest rate factors; the labor market factor is the driving force for reconstructing earning and productivity variables.

5.4.3 Forecasting Performance

We compare the forecasting performance of the FAVARs with 4 dimension reduction methods and 2 VAR specifications: time-invariant (TIV) and time-varying parameters (TVP). The inclusion of the standard autoencoder demonstrates the potential degradation of forecasting power due to non-identifiable factors, and the linear GS autoencoder serves as an approximation of models considered in Belviso and Milani (2006) and Korobilis (2013a).

We use the expanding window procedure to make forecasts. In particular, we first fit a factor extraction model and the VAR model with the data from 1965:Q1 to 1983:Q4, then conduct the 1- to 4-step-ahead point and density forecasts in 1984. We repeat this procedure by adding one more data point to the training set each time until getting the 1-step-ahead forecasts in 2023:Q1.

Table 5.1 presents the forecasting performance of different combinations of dimension reduction methods and model specifications. We use mean absolute error (MAE) and averaged log predictive likelihood (ALPL) to assess the point and density forecasts. We take the TIV-PCA as the benchmark model, with its performance highlighted in grey, and all other evaluations are relative to the benchmark ones. The relative MAE is the ratio between the MAE of a model and the benchmark, so a value smaller than 1 indicates the superior point forecasting compared to the benchmark. Similarly, the relative ALPL is the difference between the ALPL of the model and that of the benchmark, so a value greater than 0 means the model is better in density forecasting.

Forecast metric	MAE				ALPL			
	h=1	h=2	h=3	h=4	h=1	h=2	h=3	h=4
	GDPDEF							
TIV-PCA	0.121	0.224	0.327	0.422	0.431	-0.489	-1.122	-1.992
TVP-PCA	0.821	0.753	0.716	0.708	0.200	0.688	1.044	1.658
TIV-AE	1.069	1.093	1.081	1.064	-0.099	-0.110	-0.284	-0.254
TVP-AE	0.859	0.802	0.759	0.745	0.176	0.645	0.999	1.638
TIV-Linear GS AE	0.941	0.973	0.980	1.013	0.037	0.105	0.174	0.669
TVP-Linear GS AE	0.795	0.739	0.704	0.706	**0.218**	0.709	1.077	**1.701**
TIV-Nonlinear GS AE	0.945	0.972	0.972	0.998	0.059	0.122	0.265	0.436
TVP-Nonlinear GS AE	**0.785**	**0.729**	**0.691**	**0.694**	**0.218**	**0.710**	**1.079**	**1.701**
	UNRATE							
TIV-PCA	0.175	0.280	0.375	0.461	-3.279	-5.616	-6.383	-7.090
TVP-PCA	0.811	0.859	0.846	0.865	3.501	**5.561**	**5.963**	**6.385**
TIV-AE	0.949	0.938	0.943	0.960	-0.211	0.417	1.317	1.044
TVP-AE	**0.807**	0.866	0.849	0.868	**3.716**	5.511	5.945	6.371
TIV-Linear GS AE	1.031	1.008	0.983	0.958	0.091	0.349	0.133	0.947
TVP-Linear GS AE	0.819	0.855	0.847	0.864	3.632	5.292	5.895	6.327
TIV-Nonlinear GS AE	0.973	0.961	0.954	0.924	0.433	0.635	0.763	0.808
TVP-Nonlinear GS AE	0.817	**0.853**	**0.840**	**0.856**	3.645	5.486	5.874	6.335
	FEDFUNDS							
TIV-PCA	0.125	0.228	0.313	0.379	0.195	-0.142	-0.434	-0.655
TVP-PCA	0.623	**0.659**	**0.680**	0.722	0.265	0.118	0.098	0.084
TIV-AE	1.015	1.056	1.066	1.069	-0.066	-0.094	-0.075	-0.061
TVP-AE	0.667	0.738	0.760	0.791	0.243	0.084	0.066	0.060
TIV-Linear GS AE	0.831	0.855	0.865	0.859	-0.008	0.046	0.113	**0.155**
TVP-Linear GS AE	0.621	0.679	0.703	0.737	**0.288**	0.139	0.119	0.112
TIV-Nonlinear GS AE	0.894	0.896	0.910	0.911	-0.020	0.032	0.083	0.117
TVP-Nonlinear GS AE	**0.617**	0.670	0.692	**0.717**	0.285	**0.143**	**0.121**	0.113

Table 5.1: Point and density forecasting performance evaluated by the MAE and ALPL. Values highlighted in grey present the actual MAE and ALPL of the benchmark model, TIV-PCA, and the rest of the values are relative to those of the benchmark model. AE means autoencoder, and GS means grouped sparse. The best-performing model in each horizon and variable has its evaluation in bold.

For the point forecasts, 9 out of 12 cases show that the non-linear GS autoencoder has the best performance, and the TVP outperforms the TIV in all the cases. For density forecasting, half of the evaluations indicate the superior performance of the non-linear GS autoencoder with the TVP. Sparsity yields notable improvements, especially in the density forecasts of GDPDEF and FEDFUNDS, with all evaluations for these two variables showing its advantage. The evaluations of UNRATE forecasts reveal a discrepancy between point and density forecasting performance. The non-linear GS autoencoder with heteroskedasticity predominantly outperforms other models as measured by the MAE, whereas the TVP-PCA model excels in density forecasting as indicated by the ALPL.

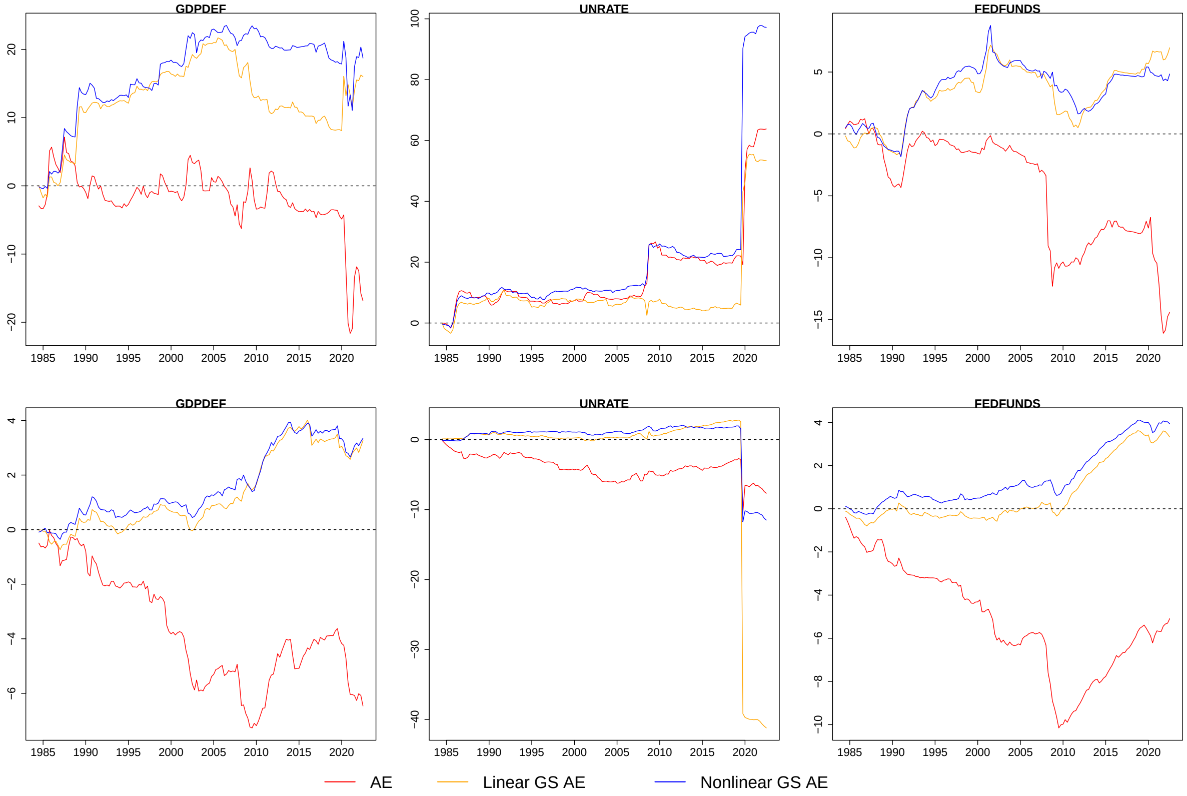


Figure 5.7: Cumulative ALPL (h=2) of models relative to the TIV-PCA (top) and the TVP-PCA (bottom).

While Table 5.1 provides an overview of the forecasting performance, Figure 5.7 gives a more detailed analysis through the cumulative ALPL. This figure depicts the cumulative ALPLs of models with TIV parameters relative to the TIV-PCA and those with TVP relative to the TVP-PCA. A curve above the zero horizontal line means the performance of the associated model is better than its PCA counterpart. The red curves corresponding to the standard autoencoder in most panels are below zero, suggesting the importance of sparsity and factor identification to the downstream forecasting task. The top panels about the TIV models show that the non-linearity from the autoencoder improves the forecasts of GDPDEF and UNRATE. In particular, the curves corresponding to the linear (orange) and non-linear (blue) GS autoencoders start to deviate around the GFC, due to the superior performance of the non-linear model. The curves about the FEDFUNDS do not exhibit such discrepancy. Moving to the TVP models, the performance of the non-linear GS autoencoder is slightly better than its linear counterpart in forecasting GDPDEF, FEDFUNDS, and UNRATE before the COVID-19 pandemic. The less significant discrepancy between the blue and orange curves indicates that the introduction of the TVP structure partially diminishes the contribution of deep learning to the overall model performance. Following the onset of the COVID-19 pandemic, model performance about the UNRATE deteriorates relative to the PCA benchmark, but the non-linear GS autoencoder yields a less pronounced decline compared to its linear counterpart. To conclude, we find that both types of non-linearity (GS autoencoder and TVP) effectively improve the forecasting performance.

5.4.4 Impulse Response Analysis

Since the non-linear GS autoencoder is the best model in most forecasting tasks, we explore the impulse responses (IRF) inferred by this model using the whole data set. Firstly, we consider the responses of three variables included in the VAR model: GDPDEF, UNRATE, and FEDFUNDS, to an expansionary monetary shock, which is a 100 bps decrease of the FEDFUNDS[39]. Figure 5.8 presents the evolution of IRFs over time. The medians of IRFs are in the first column, and we select three time points, 1981:Q3, 2000:Q4, and 2020:Q1, for the remaining columns. These three time points correspond to the representative rate cuts during the chairmanship of Volcker, Greenspan, and Powell, respectively, and are separated by an approximately 20-year interval. We exclude the rate cuts during the chairmanship of Bernanke and Yellen in this analysis for two reasons. Firstly, the IRFs during the GFC, are similar to those in 2000:Q4[40], albeit with greater uncertainty. Secondly, no significant rate cut occurred during the tenure of Yellen. Nevertheless, a comprehensive analysis of IRFs across various Federal Reserve chairmanships remains valuable for a holistic understanding; thus we include a figure with all regimes from Burns to Powell in Appendix D.4.

[39] "bps" stands for "basis points", and 1 basis point equals to 0.01%.

[40] Our result reflects that in Korobilis (2013a) as the IRFs are similar during the chairmanship of Greenspan and Bernanke.

The GDPDEF IRFs exhibit a similar shape over time. These IRFs are typically hump-shaped, starting at zero, reaching a peak at specific horizons, and decaying back to zero. However, we can still find the difference in the transmission of monetary policyinm the next three columns. In the 1981 scenario, the IRF reached its peak 13 quarters after introducing the interest rate shock, while the IRFs for the other two periods peaked earlier at 10 quarters post-shock. The monetary policy had a larger impact on the GDPDEF in 1981 than oin ther time periods, as evidenced by a higher peak response and more persistent effects of the shock over time.

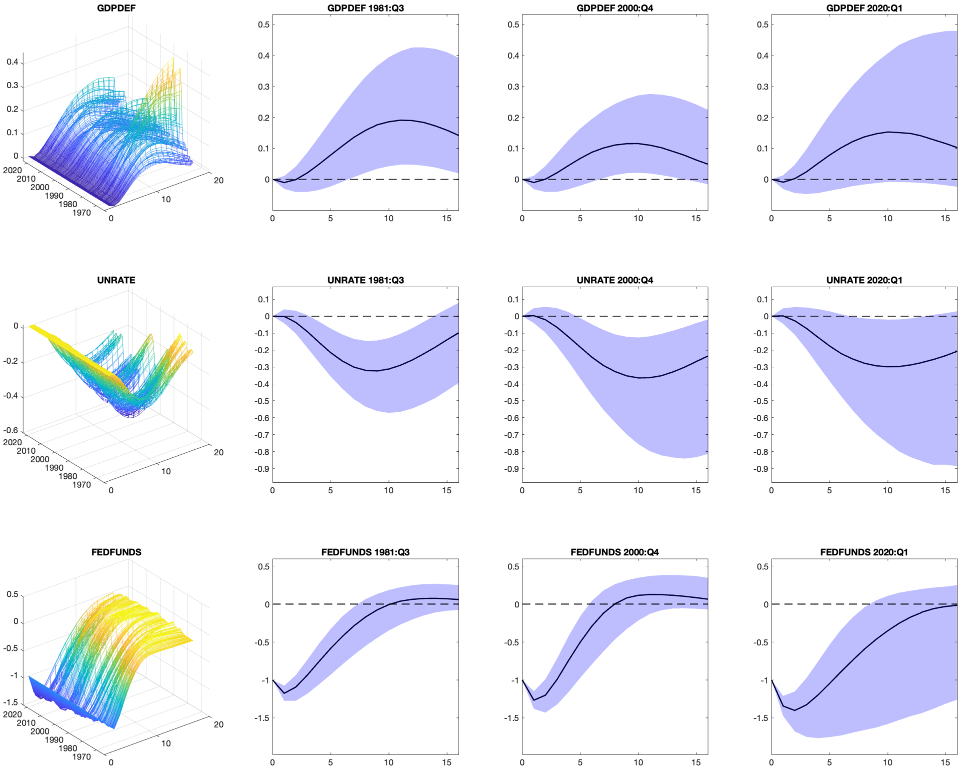


Figure 5.8: Impulse responses of the VAR variables to a 100 bps decrease in FEDFUNDS.The firstt column shows the medians over time. The rest three columns show the IRFs with their 68% credible intervals at 1981:Q3, 2000:Q4, and 2020:Q1, respectively.

The middle left panel in Figure 5.8 reveals time variation in the IRFs of the UNRATE. During the recession periods, the UNRATE responded moderately, characterized by shallower troughs. Examining the three selected time points, we find that the IRFs in 1981 and 2000 gave similar patterns, with the latter showing more uncertainty. In contrast, the IRF for 2020 was not statistically significant from zero for most of the post-shock period.

Regarding the FEDFUNDS, the median IRFs gradually evolved in shape, progressing from a subtle to a more pronounced inverted hump-shaped pattern. The rate at which the responses decayed to zero is notably reduced. While the IRF in 1981 and 2000 crossed the zero line after approximately 8 quarters, the IRF for 2020 required 16 quarters to reach zero. An additional distinction between this IRF and the previous two is the higher uncertainty.

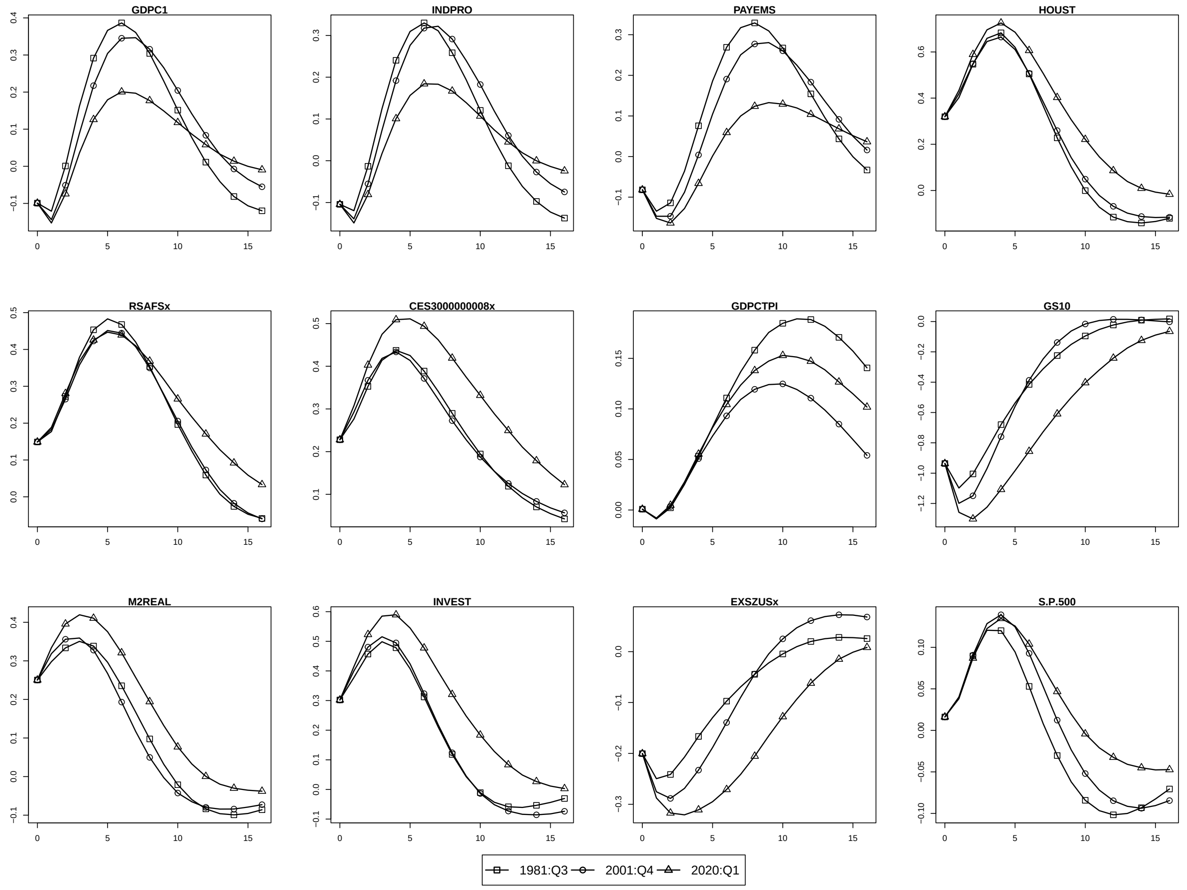


Figure 5.9: Impulse response medians of selected variables to a 100 bps decrease in FEDFUNDS at 1981:Q3, 2000:Q4, and 2020:Q1.

Figure 5.9 depicts the IRFs of a selection of variables at the three time points considered. In general, the shapes and signs align with the results in Bernanke et al. (2005). Unlike the finding in Korobilis (2013a) that some IRFs are time-varying and others are not, all IRFs show a certain degree of time variation in our case. A possible explanation is that we study a data set with a longer period, and the choices of time points are different. Most IRFs were similar in 1981 and 2000, yet they manifested a divergence in 2020. For example, the peaks of IRFs of the GDP, industrial production (INDPRO) and non-farm payroll employment (PAYEMS) were lower in 2020, compared to the ones for the other two time points, but the 2020 IRFs surpassed the others 16 quarters after the shock. The exception is the price index GDPCPTI, of which the IRFs had overlapping trajectories in the first 4 quarters, followed by divergent paths in the subsequent quarters.

5.5 Conclusion and Discussion

In this chapter, we extend the FAVAR with an autoencoder by proposing a more interpretable variant called the grouped sparse autoencoder, which identifies the factors up to element-wise transformation. To apply non-linearity to both the factor extraction and VAR parts of the FAVAR, we also adopt the TVP-VAR to model a time-varying evolution of factors. The empirical results suggest the model proposed has better interpretability and forecasting performance than the FAVARs with either a linear dimension reduction method or a TIV-VAR. This model also captures time variation in the variable responses to the monetary policy shocks.

Compared to the time-invariant study in Klieber (2024), the impulse responses regarding prices, unemployment, and interest rates have narrower credible intervals than those in Figure 5.8. While the TVP-VARs do not show signs of overfitting, as the time-varying models generally outperforms the time-invariant ones in Table 5.1, it is worth investigating the sample variances of parameters from the MCMC algorithm and how the probability for the VAR coefficient matrix lying in the stationary region changes over time. If the sample variances are high and/or the probability is low across time, the impulse responses will have high credible intervals, and corresponding solutions need to be propsoed.

The current FAVAR framework can be extended in several directions. One direction is to relax the full column rank condition of the injective decoder by exploring alternative invertible or injective decoder architectures. Another enhancement would be replacing the TVP-VAR with neural networks to potentially achieve better expressiveness. Developing a one-step procedure for simultaneous factor extraction and parameter estimation across the entire FAVAR system presents another interesting direction. From an application perspective, exploring asymmetric impulse responses in the high-dimensional data merits investigation, as the current framework still assumes symmetric responses from the high-dimensional data to expansionary and recessionary shocks.

Chapter 6

General Conclusions and Discussion

This thesis investigates multivariate time series models in high-dimensional settings, with a particular focus on vector autoregressions (VARs) and factor-augmented VARs (FAVARs), both of which are originally formulated as linear models. In light of the increasing complexity and scale of modern datasets, there is a growing need to adapt these frameworks towards parsimonious and non-linear settings. Building on existing literature, this thesis develops models which alleviate over-parameterization through shrinkage priors and dimension reduction techniques, and extends them to accommodate non-linearity via specifying time-varying coefficients, dynamic variance-covariance structures, and non-linear factor extraction. Parameter estimation is conducted using Bayesian inference, enabling flexible model specification and uncertainty quantification.

In Chapter 3, we focus on the VAR with a time-invariant coefficient matrix and stochastic volatility. Following the work of Wang et al. (2022a) about tensor VAR (TVAR), we model the coefficient matrix as a third-order tensor and employ the tensor decomposition as a dimension reduction technique. To facilitate the Bayesian inference for this model, we propose an MCMC scheme that infers the decomposition rank adaptively and provides interpretable inferential results. Using macroeconomic data from the U.S. economy, this chapter evaluates the forecasting performance of the proposed TVAR model. The results demonstrate that the TVARs, particularly the variant with an additional own-lag matrix, are competitive with standard VARs with shrinkage priors.

Chapter 4 extends the previous chapter to investigate the utility of the TVAR when the coefficient matrix is time-varying. This extension contributes to the non-linear VAR literature, as it can be viewed as a dimension-reduced variant of the widely used time-varying parameter VAR (TVP-VAR) Primiceri (2005). While parameter inference continues to rely on MCMC methods, the rank of the tensor decomposition is selected using a performance-based criterion, the Deviance Information Criterion (DIC), in contrast to the model-based approach adopted in Chapter 3. We apply the proposed model to fMRI data, demonstrating both the presence of time variation in Granger causality and the dynamic evolution of these causal relationships over time.

While the previous two chapters focus on the VAR framework, Chapter 5 shifts to the FAVAR, incorporating a deep learning model to extract factors. Extending the work of Klieber (2024), we introduce the grouped sparse autoencoder, which enables semi-identifiability and interpretability of the factors. In addition to the autoencoder, we embed a TVP-VAR to capture the dynamic evolution of these latent factors. Unlike the applications in the previous two chapters, which use data sets with 20-40 variables, this chapter leverages the FAVAR structure to handle hundreds of variables. Beyond demonstrating the superior forecasting performance of our model, we also analyze impulse response functions across different U.S. interest rate cut periods and show different effects of monetary policy.

A common feature across the three preceding chapters is the use of Minnesota-type priors, which have shown robust forecasting performance both in Chapter 3 and in prior studies such as Cross et al. (2020) and Gruber and Kastner (2025). Motivated by this, one extension is to integrate a Minnesota-type prior within the TVAR[41]. Specifically, the CP decomposition in (3.2) can be modified to

$$\boldsymbol{\mathcal{A}}_{(i,j,p)} = \sum_{r=1}^{R} \boldsymbol{\mathcal{A}}^{(r)}_{(i,j,p)} = \sum_{r=1}^{R} \lambda_{i,j} \boldsymbol{\beta}^{(r)}_{1,i} \boldsymbol{\beta}^{(r)}_{2,j} \boldsymbol{\beta}^{(r)}_{3,p},$$

where $\boldsymbol{\mathcal{A}}_{(i,j,p)}$ denotes an element in $\boldsymbol{\mathcal{A}}$, for $i,\ j \in [N]$ and $p \in [P]$, $\lambda_{i,i} > \lambda_{i,i'}$ for $i' \neq i,\ i,\ i' \in [N]$, which allows the own-lag coefficients to have high magnitudes with high probability than the cross-lag ones. The prior variance of $\beta^{(r)}_{3,p}$ can be specified to decay as p increases, which suggests the shorter lags are more informative than longer lags.

From a data perspective, the methods introduced in this thesis can be extended in two directions. The first pertains to the frequency of economic data. This thesis focuses exclusively on quarterly data; however, a rich body of literature addresses the modeling of mixed-frequency data (see the survey by Foroni and Marcellino (2013)). Both VARs and FAVARs are well-suited

[41] The additional own-lag matrix in Chapter 3 is one approach, but the extension presented in this chapter does not require additional parameters.

for adaptation to mixed-frequency settings, which can facilitate applications such as nowcasting. Moreover, deep learning models are useful in handling asynchronous time series (Lin and Michailidis, 2024), suggesting potential for improving the grouped sparse autoencoder. The second extension relates to the fMRI data set. As mentioned in Section 4.5, the fMRI data set contains 8 subjects. While Chapter 4 applies the TVP-TVAR to all of these subjects separately, Fan et al. (2022) shows the feasibility of applying the TVAR to multiple subjects simultaneously. To enable this objective, we can specify the tensor corresponding to the coefficient matrix as $\boldsymbol{\mathcal{A}}_{t,\text{fix}} + \boldsymbol{\mathcal{A}}_{t,\text{subject}}$, where $\boldsymbol{\mathcal{A}}_{t,\text{fix}}$ captures common patterns across subjects, and $\boldsymbol{\mathcal{A}}_{t,\text{subject}}$ encodes subject-specific dynamics. This decomposition allows for more efficient sharing of patterns across subjects while preserving individual variability.

Beyond the MCMC methods employed throughout this thesis, a natural extension lies in exploring alternative Bayesian computational techniques. For example, Gefang et al. (2023), Chan and Yu (2022), and Loaiza-Maya and Nibbering (2022) implemented variational inference to estimate VAR parameters; when Mørup and Hansen (2009) and Takayama et al. (2022) imposed shrinkage priors on the margins in the tensor decompositions, their posterior computation used variational inference. Combining the above two types of examples, TVARs can also be inferred via variational posteriors to avoid the time-consuming simulation of the MCMC.

The application of Bayesian inference to time series models with tensor decompositions extends beyond the multivariate frameworks examined in this thesis. As discussed in Section 2.3.4, tensor decompositions have also been employed in modeling matrix- or tensor-valued time series. While most existing studies adopt frequentist approaches to parameter estimation[42], there remains the potential for Bayesian methods to be further explored in this domain, following a trajectory similar to their development in multivariate time series analysis. We anticipate further progress in this direction and hope that the contributions of this thesis serve as a foundation for future developments in Bayesian tensor-based time series modeling.

[42] An exception is the Bayesian dynamic tensor regression proposed by Billio et al. (2023).

Appendix A

Supplementary Material of Chapter 2

A.1 Distributions

This section provides descriptions of the distributions mentioned in the thesis. Table A.1 includes the distributions and the corresponding PDF, and Table A.2 shows how the $\log \chi_1^2$ distribution is approximated by a mixture of normal distributions.

Distribution	Notation	Probability Density Function	Support
Inverse-Wishart	$\boldsymbol{X} \sim \mathcal{IW}(\nu, \mathbf{S})$	$\dfrac{\lvert\mathbf{S}\rvert^{\nu/2}}{2^{\nu N/2}\Gamma_N(\nu/2)}\lvert\mathbf{X}\rvert^{-(\nu+p+1)/2}\exp\left(-\frac{1}{2}\mathrm{tr}(\mathbf{S}\mathbf{X}^{-1})\right)$	$N \times N$ positive definite matrix
Multivariate-t	$x \sim t_\nu(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	$\dfrac{\lvert\boldsymbol{\Sigma}\rvert^{-1/2}}{C(\nu, N)\pi^{N/2}}\left[1+\frac{1}{\nu}(\mathbf{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]^{-(\nu+N)/2}$	$\mathbb{R}^N$
Location-scale t	$x \sim t_\nu(\mu, \sigma)$	$\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma\sqrt{\nu\pi}\,\Gamma\left(\frac{\nu}{2}\right)}\left[1+\frac{1}{\nu}\left(\frac{x-\mu}{\sigma}\right)^2\right]^{-\frac{\nu+1}{2}}$	$\mathbb{R}$
Double Exponential	$x \sim \mathcal{DE}(\mu, \beta)$	$\dfrac{1}{2\beta}\exp\left(-\dfrac{\lvert x-\mu\rvert}{\beta}\right)$	$\mathbb{R}$
Dirichlet	$\boldsymbol{x} \sim \mathcal{D}(\boldsymbol{\alpha})$	$\dfrac{1}{B(\boldsymbol{\alpha})}\prod_{i=1}^N x_i^{\alpha_i-1}$	N-dimensional simplex
Half-Cauchy	$x \sim \mathcal{C}^+(\mu, \sigma)$	$\dfrac{2}{\pi\sigma}\left(1+\left(\dfrac{x-\mu}{\sigma}\right)^2\right)^{-1}$	$\mathbb{R}_{\geq\mu}$
Inverse-Gamma	$x \sim \mathcal{G}^{-1}(\alpha, \beta)$	$\dfrac{\beta^\alpha}{\Gamma(\alpha)}x^{-\alpha-1}\exp\left(-\dfrac{\beta}{x}\right)$	$\mathbb{R}_{>0}$
Generalized Inverse Gaussia	$x \sim \mathcal{GIG}\left(\lambda, \chi, \psi\right)$	$\frac{(\psi/\chi)^\lambda}{2K_\lambda(\sqrt{\chi\psi})}x^{\lambda-1}\exp\left(-\left(\chi/x+\psi x\right)/2\right)$	$\mathbb{R}_{>0}$

Table A.1: Distributions mentioned in the thesis. $\Gamma_N(\nu/2) = \pi^{N(N-1)/4}\prod_{j=1}^N \Gamma(a+(1-j)/2)$ is the multivariate gamma function. $C(\nu, N) = \frac{\Gamma(\nu/2)\nu^{p/2}}{\Gamma((\nu+p)/2)}$ and $B(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^N \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^N \alpha_i)}$. $K_\lambda(z) = \left(\frac{\pi}{2}\right)\frac{I_{-\lambda}(z)-I_\lambda(z)}{\sin(\lambda\pi)}$ is a modified Bessel function of the second kind, where $I_\lambda(z) = \left(\frac{z}{2}\right)^\lambda \sum_{k=0}^\infty \frac{\left(\frac{z^2}{4}\right)^k}{k!\,\Gamma(\nu+k+1)}$.

A.2 Forward Filtering Backward Sampling

The objective of the forward filtering backward sampling (FFBS) is to sample the states given the observations and other parameters in the linear Gaussian state-space model defined in (2.18) and (2.19). Specifically, sampling $\boldsymbol{x}_{1:T}$ from the full conditional $p(\boldsymbol{x}_{1:T} \mid$

j	p_j	m_j	v_j^2
1	0.00730	-10.12999	5.79596
2	0.10556	-3.97281	2.61369
3	0.00002	-8.56686	5.17950
4	0.04395	2.77786	0.16735
5	0.34001	0.61942	0.64009
6	0.24566	1.79518	0.34023
7	0.25750	-1.08819	1.26261

Table A.2: Hyperparameters of the mixture of normal distributions, which approximates $\log \chi_1^2$.

$\boldsymbol{y}_{1:T}, \boldsymbol{G}_{1:T}, \boldsymbol{Q}_{1:T}, \boldsymbol{M}_{1:T}, \boldsymbol{R}_{1:T})$. Define $\boldsymbol{x}_{t|s} = \mathbb{E}\left[\boldsymbol{x}_t \mid \boldsymbol{y}_{1:s}, \boldsymbol{G}_{1:s}, \boldsymbol{Q}_{1:s}, \boldsymbol{M}_{1:s}, \boldsymbol{R}_{1:s}\right]$, $\boldsymbol{V}_{t|s} = \text{Var}\left[\boldsymbol{x}_t \mid \boldsymbol{y}_{1:s}, \boldsymbol{G}_{1:s}, \boldsymbol{Q}_{1:s}, \boldsymbol{M}_{1:s}, \boldsymbol{R}_{1:s}\right]$ as the mean and variance-covariance matrix of the Gaussian conditional distributions (for $t,\ s \in [T]$), then the FFBS conducts the Kalman filter by iterating the following steps for $t \in [T]$

$$\begin{aligned}
\boldsymbol{x}_{t|t-1} &= \boldsymbol{G}_t \boldsymbol{x}_{t-1|t-1}, \\
\boldsymbol{V}_{t|t-1} &= \boldsymbol{G}_t \boldsymbol{V}_{t-1|t-1} \boldsymbol{G}_t' + \boldsymbol{Q}_t, \\
\boldsymbol{K}_t &= \boldsymbol{V}_{t|t-1} \boldsymbol{M}_t' \left(\boldsymbol{M}_t \boldsymbol{V}_{t|t-1} \boldsymbol{M}_t' + \boldsymbol{R}_t\right)^{-1}, \\
\boldsymbol{x}_{t|t} &= \boldsymbol{x}_{t|t-1} + \boldsymbol{K}_t \left(\boldsymbol{Y}_t - \boldsymbol{M}_t \boldsymbol{x}_{t|t-1}\right), \\
\boldsymbol{V}_{t|t} &= \boldsymbol{V}_{t|t-1} - \boldsymbol{K}_t \boldsymbol{M}_t \boldsymbol{V}_{t|t-1}.
\end{aligned}$$

After obtaining $\boldsymbol{x}_{T|T}$ and $\boldsymbol{V}_{T|T}$, the FFBS first samples $\boldsymbol{x}_T$ corresponding to the above conditional mean and variance, and then computes the Gaussian conditional distributions $p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:T}, \boldsymbol{G}_{1:T}, \boldsymbol{Q}_{1:T}, \boldsymbol{M}_{1:T}, \boldsymbol{R}_{1:T})$ with the following mean and variance-covariance matrix

$$\begin{aligned}
\boldsymbol{x}_{t|t+1} &= \boldsymbol{x}_{t|t} + \boldsymbol{V}_{t|t} \boldsymbol{G}_t' \boldsymbol{V}_{t+1|t}^{-1} (\boldsymbol{x}_{t+1} - \boldsymbol{G}_t \boldsymbol{x}_{t|t}), \\
\boldsymbol{V}_{t|t+1} &= \boldsymbol{V}_{t|t} - \boldsymbol{V}_{t|t} \boldsymbol{G}_t' \boldsymbol{V}_{t+1|t}^{-1} \boldsymbol{G}_t \boldsymbol{V}_{t|t}.
\end{aligned}$$

Appendix B

Supplementary Material of Chapter 3

B.1 Bayesian Inference

B.1.1 Prior Setting

Each non-zero off-diagonal entry in $\boldsymbol{H}$ follows

$$\boldsymbol{H}_{i,j} \sim \mathcal{N}\left(0, \left(2/\lambda_h^2\right) \psi_h^{(i,j)}\right), \; \psi_h^{(i,j)} \sim \text{Gamma}(a_h, a_h), \text{ for } i \in [N] \text{ and } j < i,$$

where λ_h^2 is the global parameter, which controls the overall shrinkage and follows $\mathcal{G}$(0.01,0.01) prior, $\psi_h^{(i,j)}$ allows flexibility locally, and hyperparameter a_h follows an exponential prior with parameter 1. Hyperpriors corresponding to the stochastic volatilities are the same as those in Kastner and Frühwirth-Schnatter (2014). We impose $\mathcal{N}(0, 100)$ to μ_i, $\mathcal{B}(5, 1.5)$ to $\frac{1+\psi_i}{2}$ and $\mathcal{G}(1/2, 1/2)$ to σ_i^2, for $i \in [N]$.

B.1.2 Full Conditionals of $\boldsymbol{B}_1$, $\boldsymbol{B}_2$, $\boldsymbol{B}_3$ and $\boldsymbol{D}$

Recall a TVAR in terms of $\boldsymbol{B}_1$, $\boldsymbol{B}_2$ and $\boldsymbol{B}_3$ is as follows

$$\begin{aligned}\boldsymbol{y}_t^* &= \left(\boldsymbol{x}_t' \left(\boldsymbol{B}_3 \otimes \boldsymbol{B}_2\right) \boldsymbol{\mathcal{I}}_{(1)}' \otimes \boldsymbol{I}_N\right) \text{vec}(\boldsymbol{B}_1) + \boldsymbol{\epsilon}_t \\ &= \boldsymbol{B}_1 \boldsymbol{\mathcal{I}}_{(1)} \left(\left(\boldsymbol{B}_3' \boldsymbol{X}_t'\right) \otimes \boldsymbol{I}_R\right) \text{vec}(\boldsymbol{B}_2') + \boldsymbol{\epsilon}_t \\ &= \boldsymbol{B}_1 \boldsymbol{\mathcal{I}}_{(1)} \left(\boldsymbol{I}_R \otimes \left(\boldsymbol{B}_2' \boldsymbol{X}_t\right)\right) \text{vec}(\boldsymbol{B}_3) + \boldsymbol{\epsilon}_t.\end{aligned}$$

We denote the terms before vectorizations of $\boldsymbol{B}_1$, $\boldsymbol{B}_2'$ and $\boldsymbol{B}_3$ as $\boldsymbol{Z}_{t,1}$, $\boldsymbol{Z}_{t,2}$ and $\boldsymbol{Z}_{t,3}$, respectively. Given other parameters, the full conditional of vec$(\boldsymbol{B}_j)$ for $j = 1,\ 3$ or that of vec$(\boldsymbol{B}_j')$ for $j = 2$ is $\mathcal{N}\left(\overline{\boldsymbol{\mu}}_j, \overline{\boldsymbol{\Sigma}}_j\right)$ with

$$\overline{\boldsymbol{\Sigma}}_j^{-1} = \underline{\boldsymbol{\Sigma}}_j^{-1} + \sum_{t=1}^{T} \boldsymbol{Z}_{t,j}' \boldsymbol{H}' \boldsymbol{S}_t^{-1} \boldsymbol{H} \boldsymbol{Z}_{t,j},$$

$$\overline{\boldsymbol{\mu}}_j = \overline{\boldsymbol{\Sigma}}_j \sum_{t=1}^{T} \boldsymbol{Z}_{t,j}' \boldsymbol{H}' \boldsymbol{S}_t^{-1} \tilde{\boldsymbol{y}}_t^*,$$

where $\tilde{\boldsymbol{y}}_t^* = \boldsymbol{H}\boldsymbol{y}_t^* = \boldsymbol{H}(\boldsymbol{y}_t - \boldsymbol{D}\boldsymbol{x}_t)$, $\underline{\boldsymbol{\Sigma}}_j$ is the prior covariance matrix of the corresponding vector.

Given $\boldsymbol{B}_1$, $\boldsymbol{B}_2$, $\boldsymbol{B}_3$ and other parameters, we can infer $\boldsymbol{D}$ in a similar way as in Carriero et al. (2022). Assume that $\boldsymbol{D} = (\text{diag}\,(d_{1,1}, \ldots, d_{N,1}), \ldots, \text{diag}\,(d_{1,P}, \ldots, d_{N,P}))$, and $\boldsymbol{y}_t^{**} = \boldsymbol{y}_t - \boldsymbol{\mathcal{A}}_{(1)}\boldsymbol{x}_t = \boldsymbol{D}\boldsymbol{x}_t + \boldsymbol{\epsilon}_t$, if we multiply both sides of the equation aforementioned with $\boldsymbol{H}$, we get $\tilde{\boldsymbol{y}}_t^{**} = \boldsymbol{H}\boldsymbol{y}_t^{**} = \boldsymbol{H}\boldsymbol{D}\boldsymbol{x}_t + \boldsymbol{u}_t$, where $\boldsymbol{u}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{S}_t)$. This equation can be expanded to

$$
\begin{aligned}
\tilde{\boldsymbol{y}}_{t,1}^{**} &= \left(\boldsymbol{y}_t^{(1)}\right)' \boldsymbol{d}_1 + \boldsymbol{u}_{t,1}, \\
\tilde{\boldsymbol{y}}_{t,2}^{**} &= h_{2,1}\left(\boldsymbol{y}_t^{(1)}\right)' \boldsymbol{d}_1 + \left(\boldsymbol{y}_t^{(2)}\right)' \boldsymbol{d}_2 + \boldsymbol{u}_{t,2}, \\
&\vdots \\
\tilde{\boldsymbol{y}}_{t,N}^{**} &= h_{N,1}\left(\boldsymbol{y}_t^{(1)}\right)' \boldsymbol{d}_1 + \cdots + h_{N,N-1}\left(\boldsymbol{y}_t^{(N-1)}\right)' \boldsymbol{d}_{N-1} + \left(\boldsymbol{y}_t^{(N)}\right)' \boldsymbol{d}_N + \boldsymbol{u}_{t,N},
\end{aligned}
\tag{B.1}
$$

where $\boldsymbol{y}_t^{(j)}$ is a vector that contains the P lagged values of $\boldsymbol{y}_{t,j}$, $\boldsymbol{d}_j = (d_{j,1}, \ldots, d_{j,P})'$, for $j \in [N]$, $h_{i,j}$ is the (i, j) entry of $\boldsymbol{H}$.

It is noteworthy that (B.1) is similar to Equation (12) in Carriero et al. (2022). An important difference is that they multiplied the same $\boldsymbol{x}_t$ to each row of the coefficient matrix, whereas we multiply $\left(\boldsymbol{y}_t^{(j)}\right)'$ to each $\boldsymbol{d}_j$. After slightly modifying Equations (13) - (15) in Carriero et al. (2022), we get the full conditional posterior $\boldsymbol{d}_j \mid \boldsymbol{B}_1, \boldsymbol{B}_2, \boldsymbol{B}_3, \boldsymbol{d}_{-j}, \boldsymbol{H}, \boldsymbol{S}_{1:T} \sim \mathcal{N}\left(\overline{\boldsymbol{\mu}}_{\boldsymbol{d}_j}, \overline{\boldsymbol{\Sigma}}_{\boldsymbol{d}_j}\right)$, with

$$
\overline{\boldsymbol{\Sigma}}_{\boldsymbol{d}_j}^{-1} = \underline{\boldsymbol{\Sigma}}_{\boldsymbol{d}_j}^{-1} + \sum_{i=j}^{N} h_{i,j}^2 \sum_{t=1}^{T} s_{t,i}^{-1} \boldsymbol{y}_t^{(j)} \left(\boldsymbol{y}_t^{(j)}\right)', \tag{B.2}
$$

$$
\overline{\boldsymbol{\mu}}_{\boldsymbol{d}_j} = \overline{\boldsymbol{\Sigma}}_{\boldsymbol{d}_j} \left(\sum_{i=j}^{N} h_{i,j}^2 \sum_{t=1}^{T} s_{t,i}^{-1} \boldsymbol{z}_{t,i}^{(j)} \boldsymbol{y}_t^{(j)}\right), \tag{B.3}
$$

where $\boldsymbol{z}_{t,j+l}^{(j)} = \tilde{\boldsymbol{y}}_{t,j+l}^{**} - \sum_{i \neq j, i=1}^{j+l} h_{j+l,i}(\boldsymbol{y}_t^{(i)})'\boldsymbol{d}_i$, $\boldsymbol{d}_{-j}$ represents $\boldsymbol{D}$ without $\boldsymbol{d}_j$, $\underline{\boldsymbol{\Sigma}}_{\boldsymbol{d}_j}$ is the prior covariance matrix of $\boldsymbol{d}_j$.

A more efficient way is to rewrite the system in (B.1) as

$$
\left(\boldsymbol{Y}^{**} - \boldsymbol{X}\left(\boldsymbol{D}^{[j=0]}\right)'\right) \boldsymbol{H}'_{j:N,1:N} = \boldsymbol{Y}^{(j)} \boldsymbol{d}_j \boldsymbol{H}'_{j:N,k} + \boldsymbol{U}_{j:N}, \tag{B.4}
$$

where $\boldsymbol{Y}^{**} = (\boldsymbol{y}_1^{**}, \ldots, \boldsymbol{y}_T^{**})'$, $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)'$, $\boldsymbol{H}_{j:N,1:N}$ is the block of $\boldsymbol{H}$ composed of j- to N-th rows and all N columns, $\boldsymbol{Y}^{(j)} = (\boldsymbol{y}_1^{(j)}, \ldots, \boldsymbol{y}_T^{(j)})'$, $\boldsymbol{D}^{[j=0]}$ is the same as $\boldsymbol{D}$ except the j-th row as zeros, $\boldsymbol{U}_{j:N} = (\boldsymbol{u}_{1,j:N}, \ldots, \boldsymbol{u}_{T,j:N})'$. If we vectorize both sides of (B.4), the new equation is

$$\text{vec}\left(\left(\boldsymbol{Y}^{**}-\boldsymbol{X}\left(\boldsymbol{D}^{[j=0]}\right)'\right)\boldsymbol{H}'_{j:N,1:N}\right)=\left(\boldsymbol{H}_{j:N,j}\otimes\boldsymbol{Y}^{(j)}\right)\boldsymbol{d}_j+\text{vec}\left(\boldsymbol{U}_{j:N}\right).$$

Let

$$\begin{aligned}\tilde{\boldsymbol{Y}}^{(j)}&=\text{vec}\left(\left(\boldsymbol{Y}^{**}-\boldsymbol{X}\left(\boldsymbol{D}^{[j=0]}\right)'\right)\boldsymbol{H}'_{j:N,1:N}\right)./\text{vec}\left(\boldsymbol{S}^{1/2}_{1:T,j:N}\right),\\ \tilde{\boldsymbol{X}}^{(j)}&=\left(\boldsymbol{H}_{j:N,j}\otimes\boldsymbol{Y}^{(j)}\right)./\text{vec}\left(\boldsymbol{S}^{1/2}_{1:T,j:N}\right),\end{aligned}$$

where $./$ is Matlab element-by-element division operation, $\boldsymbol{S}_{1:T,j:N}$ is a T-by-$(N-j+1)$ matrix with the t-th row has entries $(s_{t,j},\dots,s_{t,N})$. Then (B.2) and (B.3) are simplified to

$$\overline{\boldsymbol{\Sigma}}^{-1}_{\boldsymbol{d}_j}=\underline{\boldsymbol{\Sigma}}^{-1}_{\boldsymbol{d}_j}+\left(\tilde{\boldsymbol{X}}^{(j)}\right)'\tilde{\boldsymbol{X}}^{(j)},\tag{B.5}$$

$$\overline{\boldsymbol{\mu}}_{\boldsymbol{d}_j}=\overline{\boldsymbol{\Sigma}}_{\boldsymbol{d}_j}\left(\tilde{\boldsymbol{X}}^{(j)}\right)'\tilde{\boldsymbol{Y}}^{(j)}.\tag{B.6}$$

B.1.3 Full Conditionals Related to Multiplicative Gamma Prior

Posteriors of hyperparameters in the MGP are similar to those in Bhattacharya and Dunson (2011). Since $\phi_{(r,j,i_j)}$ is a local hyperparameter of $\boldsymbol{\beta}^{(r)}_{j,i_j}$, the derivation of its conditional posterior given $\boldsymbol{\beta}^{(r)}_{j,i_j}$ and τ_r is

$$\begin{aligned}p\left(\phi_{(r,j,i_j)}\mid\boldsymbol{\beta}^{(r)}_{j,i_j},\tau_r\right)&\propto\left(\phi^{-1}_{(r,j,i_j)}\right)^{-1/2}\exp\left(-\frac{\left(\boldsymbol{\beta}^{(r)}_{j,i_j}\right)^2}{2\phi^{-1}_{(r,j,i_j)}\tau_2^{-1}}\right)\left(\phi_{(r,j,i_j)}\right)^{\frac{\nu}{2}-1}\exp\left(-\frac{\nu}{2}\phi_{(r,j,i_j)}\right)\\&=\left(\phi_{(r,j,i_j)}\right)^{\frac{\nu+1}{2}-1}\exp\left(-\frac{\tau_r\left(\boldsymbol{\beta}^{(r)}_{j,i_j}\right)^2+\nu}{2}\phi_{(r,j,i_j)}\right).\end{aligned}$$

Thus, the conditional posterior of $\phi_{(r,j,i_j)}$ is a Gamma distribution

$$\phi_{(r,j,i_j)}\mid\boldsymbol{\beta}^{(r)}_{j,i_j},\tau_r\sim\mathcal{G}\left(\frac{\nu+1}{2},\frac{\nu+\tau_r\left(\boldsymbol{\beta}^{(r)}_{j,i_j}\right)^2}{2}\right).$$

δ_1 involves in all τ_r's, for $r\in[R]$, so the sampling δ_1 is conditional to all margins and corresponding hyperparameters, denoted as $\cdot$. Combining the likelihood and prior, we get

$$p(\delta_1\mid\cdot)\propto\delta_1^{a_1-1}\exp(-\delta_1)\prod_{r=1}^{R}\prod_{j=1}^{3}\prod_{i_j=1}^{I_j}\delta_1^{\frac{1}{2}}\exp\left(-\frac{\phi_{(r,j,k)}\delta_1\tau_r^{(1)}\left(\boldsymbol{\beta}^{(r)}_{j,i_j}\right)^2}{2}\right)\tag{B.7}$$

$$=\delta_1^{a_1+\frac{(2N+P)R}{2}-1}\exp\left(-\left(1+\sum_{r=1}^{R}\tau_r^{(1)}\sum_{j=1}^{3}\sum_{i_j=1}^{I_j}\frac{\phi_{(r,j,i_j)}\left(\boldsymbol{\beta}^{(r)}_{j,i_j}\right)^2}{2}\right)\delta_1\right),$$

where $\tau_{r'}^{(r)} = \prod_{i=1, i \neq r}^{r'} \delta_i$. The derivation leads to a Gamma conditional posterior of δ_1,

$$\delta_1 \mid \cdot \sim \mathcal{G}\left(a_1 + \frac{(2N+P)R}{2}, 1 + \frac{1}{2}\sum_{r'=1}^{R} \tau_{r'}^{(1)} \sum_{j=1}^{3}\sum_{i_j=1}^{I_j} \phi_{(r,j,i_j)} \left(\boldsymbol{\beta}_{j,i_j}^{(r')}\right)^2\right).$$

The derivation of the conditional posterior of δ_r, for $r > 1$, is similar to the above derivation, but the prior and likelihood are slightly different. We first need to change a_1 in B.7 to a_2, and since δ_r is only related to $\beta_{j,i_j}^{(r')}$'s and their corresponding hyperparameters, where $r' \geq r$, the starting value of r' is r rather than 1, and we amend $\tau_{r'}^{(1)}$ to $\tau_{r'}^{(r)}$. This results in a Gamma conditional posterior of δ_r is

$$\delta_r \mid \cdot \sim \mathcal{G}\left(a_2 + \frac{(2N+P)(R-r+1)}{2}, 1 + \frac{1}{2}\sum_{r'=r}^{R} \tau_{r'}^{(r)} \sum_{j=1}^{3}\sum_{i_j=1}^{I_j} \phi_{(r,j,i_j)} \left(\boldsymbol{\beta}_{j,i_j}^{(r')}\right)^2\right),$$

where we keep $\cdot$ as conditions for brevity. τ_r is updated as the product of $\delta_1, \ldots, \delta_r$.

B.1.4 Details for Other Full Conditionals

Conditional posteriors related to the normal-gamma prior (hyperparameters of $\boldsymbol{D}$ and $\boldsymbol{H}$) are almost identical to those in Huber and Feldkircher (2019). The only difference is that these posteriors are conditional on entries of $\boldsymbol{D}$ and $\boldsymbol{H}$, instead of the coefficient matrix.

The conditional posterior of $\boldsymbol{H}$ can also be found in Huber and Feldkircher (2019). For stochastic volatility, we use an ASIS algorithm proposed in Kastner and Frühwirth-Schnatter (2014) and implement it with an R package called **stochvol** (Kastner, 2016).

B.1.5 Algorithms

Algorithm 8 Full interweaving algorithm.

```
Step (a): Update $\boldsymbol{B}_1^{\mathrm{old}}$ under the base parameterization.

Step (b*): Store the first row of $\boldsymbol{B}_1^{\mathrm{old}}$ into $\boldsymbol{D}_1$ and determine $\boldsymbol{B}_1^*, \boldsymbol{B}_2^*$.

Step (b**): Sample $\left(\boldsymbol{\beta}_{1,1}^{\mathrm{new}(r)}\right)^2$ for $r \in [R]$ using the second parameterization and store corresponding values in $\boldsymbol{D}_1$.

Step (b***): Update $\boldsymbol{B}_1^{\mathrm{new}}$ and $\boldsymbol{B}_2^{\mathrm{n\tilde{e}w}}$ with transformation
        $\boldsymbol{B}_1^{\mathrm{new}} = \boldsymbol{B}_1^* \boldsymbol{D}_1,\ \boldsymbol{B}_2^{\mathrm{n\tilde{e}w}} = \boldsymbol{B}_2^* \boldsymbol{D}_1^{-1}.$

Step (c): Update $\boldsymbol{B}_2^{\mathrm{old}}$ under the base parameterization.
```

Step (d*): Store the first row of $\boldsymbol{B}_2^{\text{old}}$ into $\boldsymbol{D}_2$ and determine $\boldsymbol{B}_2^{**}$, $\boldsymbol{B}_3^{**}$.

Step (d**): Sample $\left(\boldsymbol{\beta}_{2,1}^{\text{new}(r)}\right)^2$ for $r \in [R]$ using the third parameterization and store corresponding values in $\boldsymbol{D}_2$.

Step (d***): Update $\boldsymbol{B}_2^{\text{new}}$ and $\boldsymbol{B}_3^{\mathrm{n\tilde{e}w}}$ with transformation

$$\boldsymbol{B}_2^{\text{new}} = \boldsymbol{B}_2^{**}\boldsymbol{D}_2,\ \boldsymbol{B}_3^{\mathrm{n\tilde{e}w}} = \boldsymbol{B}_3^{**}\boldsymbol{D}_2^{-1}.$$

Step (e): Update $\boldsymbol{B}_3^{\text{old}}$ under the base parameterization.

Step (f*): Store the first row of $\boldsymbol{B}_3^{\text{old}}$ into $\boldsymbol{D}_3$ and determine $\boldsymbol{B}_3^{***}$, $\boldsymbol{B}_1^{***}$.

Step (f**): Sample $\left(\boldsymbol{\beta}_{3,1}^{\text{new}(r)}\right)^2$ for $r \in [R]$ using the fourth parameterization and store corresponding values in $\boldsymbol{D}_3$.

Step (f***): Update $\boldsymbol{B}_3^{\text{new}}$ and $\boldsymbol{B}_1^{\mathrm{n\tilde{e}w}}$ with transformation

$$\boldsymbol{B}_3^{\text{new}} = \boldsymbol{B}_3^{***}\boldsymbol{D}_3,\ \boldsymbol{B}_1^{\mathrm{n\tilde{e}w}} = \boldsymbol{B}_1^{***}\boldsymbol{D}_3^{-1}.$$

Step (g): Sample other unknown parameters from their full conditionals.

B.2 Additional Results

B.2.1 Additional Results in Simulation Study

The section contains the following tables and figures based on the simulation study in Section 3.4:

- Table B.1 provides the sensitivity test to select thresholds γ_1 and γ_2.
- Table B.2 shows the convergence diagnostic based on the Geweke diagnostic (Geweke, 1991).
- Figure B.1 demonstrates trace plots of a margin based on different pivots described in Section 3.3.
- Figure B.2 presents the inefficiency factors of coefficients.

To choose γ_1 and γ_2, we use γ_1 from a range of values close to 0, $\{10^{-4}, 10^{-3}, 5{\times}10^{-4}, 10^{-3}\}$, and γ_2 from values below and close to 1, $\{0.85, 0.9, 0.95\}$, to the simulation study of the

$(N, R) = (10, 3)$ scenario. Table B.1 provides the inferential results based on different combinations of γ_1 and γ_2. The inference of coefficients is not sensitive to the combination, but the rank inferred is. We choose $\gamma_1 = 0.001$ and $\gamma_2 = 0.9$ because this combination leads to the lowest rank value and narrowest 90% credible interval.

(γ_1, γ_2)	MSE	R	ESS	Running Time (hr)
(0.0001,0.85)	0.011 (0.003,0.037)	5.4 (3.6,9.4)	4199.721 (2641.187,7027.556)	0.387 (0.344,0.445)
(0.0001,0.9)	0.011 (0.003,0.037)	5.28 (3,9.4)	4203.355 (2668.541,6850.881)	0.397 (0.336,0.471)
(0.0001,0.95)	0.011 (0.003,0.037)	6.16 (3.6,11.4)	4186.997 (2629.508,7058.076)	0.42 (0.379,0.505)
(0.0005,0.85)	0.011 (0.003,0.037)	4.24 (3,6.4)	4168.846 (2739.378,6953.491)	0.369 (0.328,0.4)
(0.0005,0.9)	0.011 (0.003,0.037)	4.4 (3,6.4)	4228.833 (2624.555,7160.893)	0.368 (0.286,0.4)
(0.0005,0.95)	0.011 (0.003,0.037)	3.96 (3,5.8)	4222.431 (2645.81,7013.176)	0.372 (0.326,0.403)
(0.001,0.85)	0.011 (0.003,0.037)	3.96 (3,6)	4181.059 (2722.343,6912.24)	0.412 (0.366,0.445)
(0.001,0.9)	0.011 (0.003,0.038)	3.84 (3,5)	4128.954 (2603.196,6871.378)	0.429 (0.361,0.504)
(0.001,0.95)	0.011 (0.003,0.037)	3.84 (3,5.8)	4143.039 (2707.385,6898.21)	0.404 (0.347,0.509)

Table B.1: Sensitivity table of the MGP with different threshold (γ_1) and proportion (γ_2). Each cell corresponds to an averaged value over 25 simulations with a 90% credible interval in the parentheses.

Figure B.1 suggests using $\boldsymbol{B}_3$ as candidates for the pivot matrix, rather than $\boldsymbol{B}_1$, $\boldsymbol{B}_2$ and $\boldsymbol{B}$ because the result corresponding to $\boldsymbol{B}_3$ has the best mixing. Note that numbers of rows in $\boldsymbol{B}_1$, $\boldsymbol{B}_2$, $\boldsymbol{B}_3$ and $\boldsymbol{B}$ are N, N, P, $2N+P$, respectively. One possible reason for this best performance of using $\boldsymbol{B}_3$ is that N and $2N+P$ are higher than P in our simulation and real data experiments, so it is easier to correctly match columns in $\boldsymbol{B}_3$ to those in $\boldsymbol{B}_3^{\text{pivot}}$, compared to similar procedures using $\boldsymbol{B}_1$, $\boldsymbol{B}_2$ and $\boldsymbol{B}$.

N=10, R=3	Interweaving	Non-interwoven	**N=20, R=5**	Interweaving	Non-interwoven	**N=50, R=10**	Interweaving	Non-interwoven
Normal	0.950	0.940	Normal	0.910	0.921	Normal	0.913	0.911
MGP	0.665	0.752	MGP	0.627	0.615	MGP	0.636	0.665
MDGDP	0.912	0.680	MDGDP	0.881	0.760	MDGDP	0.813	0.787

Table B.2: Averaged proportions of margins which are convergent according to Geweke's Diagnostics.

B.2.2 Additional Descriptions and Results about Forecasting

This subsection includes supplementary materials of Section 3.5.2:

- Table B.3 and Table B.4: MSFE of the medium and large data sets.
- Table B.5 and Table B.6: MAE of the medium and large data sets.
- Figure B.3: Cumulative marginal ALPL from the large data set.

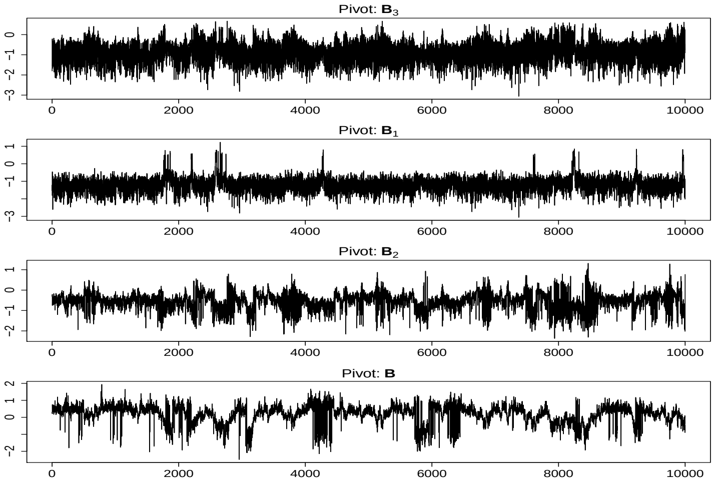


Figure B.1: Trace plots of the first 10,000 draws of $\boldsymbol{\beta}_{1,1}^{(1)}$ in $N = 10$, $R = 3$ scenario after burn-in period. The inferential scheme adopts a standard normal prior with the interweaving strategy, and the post-processing procedure in each panel chooses a different pivot indicated in the title.

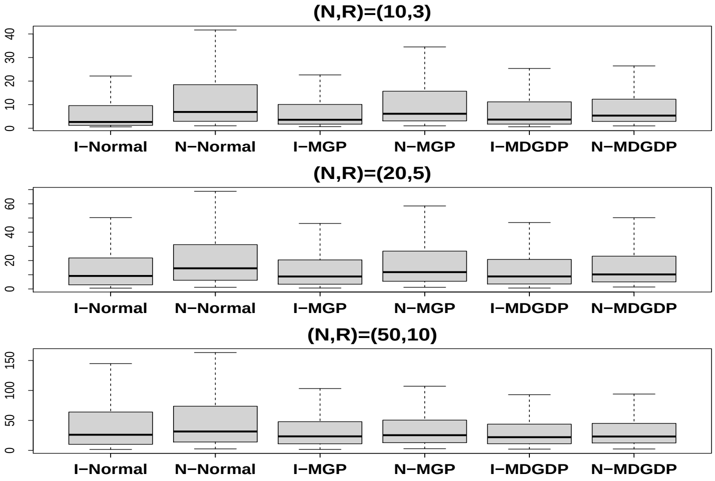


Figure B.2: Boxplots of inefficiency factors of coefficient matrices from different scenarios: $(N, R) = (10, 3)$ (top), $(N, R) = (20, 5)$ (middle), and $(N, R) = (50, 10)$ (bottom). Inferential schemes with and without interweaving are represented as “I-” and “N-”, respectively, followed by a prior setting. Outliers are discarded.

- Table B.7, B.8, and B.9: MSFE, MAE and ALPL from the alternative medium data set.
- Table B.10, B.11, and B.12: MSFE, MAE, and ALPL from the order-invariant model fitted from the medium data set.
- Table B.13, B.14 and B.15: MSFE, MAE, and ALPL from the order-invariant model fitted from the large data set.
- Figure B.4: Trace plots of margins using the order-invariant model.

We use mean squared forecast error (MSFE), mean absolute error (MAE), and average log predictive likelihood (ALPL) to assess the point and density forecasting performance. Both joint and marginal forecasting performance are evaluated. The joint MSFE is the averaged MSFE over the 20 and 40 time series for medium- and large-scale data sets, respectively

$$\text{MSFE}_{M,\,\text{joint}} = \frac{1}{N}\sum_{i=1}^{N}\text{MSFE}_{M,i},\text{ with }\text{MSFE}_{M,i} = \frac{1}{\overline{T}-h-T+1}\sum_{t=T}^{\overline{T}-h}\left(\boldsymbol{y}_{t+h,i} - \mathbb{E}\left(\boldsymbol{y}_{t+h,i} \mid \boldsymbol{y}_{1:t}, M\right)\right)^2,$$

where M denotes a model index, $\overline{T}$ is the number of time points in the data set, and h is the horizon. $\mathbb{E}(\boldsymbol{y}_{t+h,i} \mid \boldsymbol{y}_{1:t}, m)$ is the Monte Carlo estimate of posterior predictive mean.

Similarly, the joint MAE is written as

$$\text{MAE}_{M,\,\text{joint}} = \frac{1}{N}\sum_{i=1}^{N}\text{MAE}_{M,i},\text{ with }\text{MAE}_{M,i} = \frac{1}{\overline{T}-h-T+1}\sum_{t=T}^{\overline{T}-h}|\boldsymbol{y}_{t+h,i} - \mathbb{E}\left(\boldsymbol{y}_{t+h,i} \mid \boldsymbol{y}_{1:t}, M\right)|,$$

We follow Billio et al. (2023) to approximate the joint ALPL, $\text{ALPL}_{M,\text{joint}}$, by its Monte Carlo estimate in terms of stochastic volatility and lower triangular matrix sampled over the L iterations ($L = 10,000$ in this case),

$$\text{ALPL}_{M,\,\text{joint}} \approx \frac{1}{\overline{T}-h-T+1}\sum_{t=T}^{\overline{T}-h}\log\left(\frac{1}{L}\sum_{l=1}^{L}p\left(\boldsymbol{y}_{t+h} \mid \boldsymbol{y}_{1:t}, \boldsymbol{A}^{(l)}, \boldsymbol{S}_t^{(l)}, \boldsymbol{H}^{(l)}, M\right)\right).$$

Given the sample $\boldsymbol{S}_t^{(l)}$, we simulate $\boldsymbol{S}_{(t+1):(t+h)}^{(l)}$ based on the stochastic volatility specification in Equation (3.6), so $p\left(\boldsymbol{y}_{t+h} \mid \boldsymbol{y}_{1:t}, \boldsymbol{A}^{(l)}, \boldsymbol{S}_t^{(l)}, \boldsymbol{H}^{(l)}, M\right) \approx \mathcal{N}\left(\boldsymbol{y}_{t+h}; \boldsymbol{A}^{(l)}\boldsymbol{x}_{t+1}, (\boldsymbol{H}^{(l)})^{-1}\boldsymbol{S}_{t+h}^{(l)}((\boldsymbol{H}^{(l)})^{-1})'\right)$, where $\boldsymbol{x}_{t+1} = (\boldsymbol{y}_t', \ldots, \boldsymbol{y}_{t-P+1}')'$.

Marginal MSFE and MAE for the i-th variable from model M relative to the benchmark are defined as

$$\text{RMSFE}_{M,i} = \frac{\text{MSFE}_{M,i}}{\text{MSFE}_{\text{benchmark},i}},\ \text{RMAE}_{M,i} = \frac{\text{MAE}_{M,i}}{\text{MAE}_{\text{benchmark},i}}.$$

Similarly, the relative ALPL for the i-th variable from model M is

$$\text{RALPL}_{M,i} = \text{ALPL}_{M,i} - \text{ALPL}_{\text{benchmark},i},$$

where $\text{ALPL}_{M,i}$ is also approximated by its Monte Carlo estimate.

Table B.3 and Table B.4 show the performance of joint and marginal point forecasts inferred from data sets with different sizes. Overall, TVARs achieve better joint performance than standard VARs. For the marginal performance, TVARs outperform standard VARs in 11 and 12 out of 21 cases for the two data sets, respectively. Forecasts of PAYEMS, UNRATE, and GDP are more favorable when using TVARs, while standard VARs have better performance in forecasting CPIAUCSL, FEDFUNDS, and GS10. For the point forecasts evaluated by MAE (see Table B.5 and Table B.6), most results align with the MSFE. One notable difference is that TVARs are better than standard VARs in forecasting FEDFUNDS and GS10.

Comparing the difference between point and density forecasting performance, we notice that the best model for forecasting PAYEMS in longer horizons (h = 2 or 4) changes from TVARs to standard VARs, if one considers density forecasts rather than the point ones. We inspect marginal density forecasts of PAYEMS by looking at the cumulative log predictive likelihood shown in Figure B.3. A potential explanation for the inferior performance of TVARs compared to standard VARs in forecasting PAYEMS is attributed to the volatile economic data before the Great Moderation because the slopes of cumulative ALPLs corresponding to standard VARs are steeper than the TVAR counterparts from 1985 to 1990, and they share a similar trend afterward.

To check the robustness of forecasting performance across different variable choices in the medium data set, we construct an alternative medium data set with variables selected in Appendix B.3. Experimental results from this data set are available in Table B.7, B.8, and B.9. These results lead to a similar conclusion as the ones from the medium data set: TVARs are better than standard VARs in joint point (especially for MAE) and density forecasting. Marginal performance in TVARs is also competitive. These results mirror the findings from the original medium data set: 1) TVARs yield better density forecasts of CPIAUSL compared to point forecasts; 2) standard VARs outperform TVARs in forecasting FEDFUNDS when the evaluation metric is MSFE, yet TVARs exhibit superior forecasts when the metric changes to ALPL and MAE.

The last consideration about forecasting is the ordering issue due to the decomposition of $\mathbf{\Omega}_t$, the variance-covariance matrix. The Cholesky decomposition of $\mathbf{\Omega}_t$ might affect the

Model	Horizon	MSFE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	0.714	0.624	1.020	0.336	0.639	0.735	0.869	0.759
	2	0.776	0.663	0.974	0.374	0.596	0.690	0.741	0.823
	4	0.853	0.678	0.988	0.477	0.639	0.680	0.827	0.898
Tensor MPG Own-lag	1	0.703	**0.588**	1.178	0.323	**0.608**	**0.683**	0.846	0.750
	2	**0.773**	**0.653**	0.991	0.361	**0.594**	**0.677**	**0.737**	0.818
	4	**0.852**	**0.656**	1.008	0.442	**0.623**	**0.663**	**0.821**	0.892
Minnesota	1	**0.689**	0.696	**0.903**	0.292	0.697	0.725	**0.717**	**0.735**
	2	0.774	0.682	0.968	0.362	0.653	0.721	0.743	**0.810**
	4	0.904	0.675	**0.983**	0.457	0.727	0.725	0.853	0.902
NG	1	0.710	0.719	0.965	**0.285**	0.740	0.770	0.795	0.757
	2	0.784	0.758	**0.955**	**0.349**	0.700	0.780	0.744	0.814
	4	0.859	0.718	0.988	**0.437**	0.656	0.757	0.836	0.888
Horseshoe	1	0.790	1.621	1.178	0.323	0.780	1.004	0.817	0.780
	2	0.825	1.009	0.991	0.361	0.716	0.867	0.742	0.822
	4	0.873	0.800	1.008	0.442	0.667	0.762	0.839	**0.850**

Table B.3: MSFE of joint and marginal variables using the medium-scale data set. The best forecasts are in bold.

Model	Horizon	MSFE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	0.720	0.856	0.984	0.333	0.711	0.835	0.887	**0.767**
	2	0.757	0.658	0.946	0.352	**0.636**	0.708	0.740	0.835
	4	0.810	**0.623**	1.006	0.442	0.682	**0.648**	0.820	0.903
Tensor MPG Own-lag	1	**0.690**	**0.794**	0.961	0.301	**0.697**	**0.738**	0.860	0.770
	2	**0.748**	**0.656**	**0.935**	0.347	0.644	**0.678**	0.736	0.832
	4	0.810	0.630	1.001	0.436	0.693	**0.648**	**0.818**	**0.897**
Minnesota	1	0.726	1.247	0.994	0.362	0.790	1.007	**0.844**	0.777
	2	0.763	0.847	0.936	0.363	0.669	0.822	**0.728**	**0.817**
	4	0.811	0.703	**0.994**	0.446	**0.634**	0.738	0.825	0.898
NG	1	0.691	0.923	**0.949**	**0.295**	0.820	0.782	0.845	0.778
	2	0.754	0.794	0.952	0.343	0.736	0.729	0.745	0.829
	4	**0.809**	0.698	0.999	**0.429**	0.673	0.701	0.824	0.904
Horseshoe	1	0.712	0.980	0.973	0.300	0.822	0.842	0.869	0.772
	2	0.759	0.806	0.939	**0.339**	0.708	0.748	0.738	0.819
	4	0.810	0.698	0.998	0.433	0.665	0.703	0.824	0.898

Table B.4: MSFE of joint and marginal variables using the large-scale data set. The best forecasts are in bold.

Model	Horizon	MAE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
	1	0.599	0.955	1.021	0.583	0.811	0.945	0.927	0.880
Tensor MGP	2	0.627	0.860	0.956	0.582	**0.794**	0.881	**0.883**	0.895
	4	0.656	0.832	**0.981**	0.630	0.810	0.896	0.898	0.945
	1	**0.592**	**0.911**	1.012	**0.559**	**0.803**	**0.912**	0.923	**0.874**
Tensor MPG Own-lag	2	**0.622**	**0.827**	**0.951**	**0.567**	**0.794**	**0.863**	**0.883**	**0.892**
	4	**0.651**	**0.779**	0.989	**0.605**	**0.797**	**0.876**	**0.894**	0.937
	1	0.632	0.968	**0.999**	0.771	0.943	0.941	0.938	0.930
Minnesota	2	0.673	0.874	0.997	0.795	0.902	0.881	0.942	0.943
	4	0.717	0.835	1.032	0.818	0.910	0.927	0.952	1.004
	1	0.618	0.986	1.029	0.636	0.908	0.957	0.954	0.921
NG	2	0.647	0.866	0.979	0.650	0.871	0.887	0.928	0.9239
	4	0.662	0.783	1.001	0.655	0.812	0.892	0.926	0.955
	1	0.635	1.214	1.069	0.580	0.879	1.019	**0.917**	0.915
Horseshoe	2	0.652	0.949	0.978	0.584	0.850	0.922	0.912	0.909
	4	0.662	0.836	0.994	0.623	0.811	0.887	0.917	**0.928**

Table B.5: MAE of joint and marginal variables using the medium-scale data set. The best forecasts are in bold.

Model	Horizon	MAE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
	1	0.607	1.105	0.995	0.571	0.861	0.925	0.943	**0.889**
Tensor MGP	2	0.623	0.877	0.939	**0.556**	0.823	0.852	0.889	0.903
	4	0.640	0.814	1.006	0.597	0.842	**0.834**	0.894	0.943
	1	**0.595**	**1.064**	0.994	**0.549**	0.872	**0.897**	0.937	0.893
Tensor MPG Own-lag	2	0.620	0.874	0.941	0.559	0.834	**0.841**	0.889	**0.901**
	4	0.640	0.814	1.001	0.593	0.859	0.839	**0.893**	**0.938**
	1	0.606	1.072	**0.993**	0.618	**0.838**	0.988	**0.922**	0.917
Minnesota	2	**0.618**	**0.828**	**0.931**	0.582	**0.788**	0.869	**0.875**	0.906
	4	**0.629**	**0.730**	**0.977**	**0.590**	**0.760**	0.871	0.892	0.946
	1	0.642	1.152	1.049	0.757	0.965	0.982	0.983	0.982
NG	2	0.660	0.934	0.988	0.729	0.906	0.880	0.935	0.966
	4	0.658	0.803	1.012	0.703	0.833	0.870	0.927	0.983
	1	0.622	1.084	1.030	0.651	0.909	0.954	0.960	0.935
Horseshoe	2	0.635	0.871	0.958	0.631	0.847	0.849	0.906	0.924
	4	0.639	0.764	0.994	0.631	0.797	0.849	0.909	0.950

Table B.6: MAE of joint and marginal variables using the large-scale data set. The best forecasts are in bold.

inference of parameters in (Tensor) VARs, as discussed in many papers - Arias et al. (2023); Carriero et al. (2019); Chan and Qi (2024), among others. The basic idea is that the prior of each element in the variance-covariance matrix $\boldsymbol{\Omega}_t$ depends on the ordering of variables. For example, denote $\boldsymbol{\Omega}_t$ as the variance-covariance matrix corresponding to a particular variable order and $\tilde{\boldsymbol{\Omega}}_t$ as the one corresponding to switching the first and second variables in the original order. Then the prior of the $(1,1)$ entry of $\boldsymbol{\Omega}_t$ is not equivalent to the prior of the 2-2 entry of $\tilde{\boldsymbol{\Omega}}_t$. Motivated by the ordering issue, we applied the non-restrictive $\boldsymbol{H}$ proposed in Chan et al. (2024) to model $\boldsymbol{\Omega}_t$. The prior of each non-zero element in $\boldsymbol{H}$ is a standard normal distribution.

Model	Horizon	MSFE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	0.594	0.693	0.956	0.337	0.709	0.958	0.887	0.769
	2	0.641	0.653	0.950	0.349	0.627	0.837	0.745	0.825
	4	0.689	0.673	0.994	0.493	0.641	0.796	0.828	0.900
Tensor MPG Own-lag	1	0.564	**0.617**	0.870	0.311	0.642	0.780	0.854	0.759
	2	0.629	**0.615**	0.928	**0.333**	**0.600**	0.781	0.735	0.825
	4	0.852	0.656	1.008	0.442	**0.623**	**0.663**	**0.821**	**0.892**
Minnesota	1	**0.546**	0.649	**0.860**	0.309	**0.633**	**0.702**	**0.732**	**0.729**
	2	**0.628**	0.638	0.957	0.366	0.625	**0.702**	0.741	**0.822**
	4	**0.670**	**0.647**	0.996	0.442	0.632	0.719	0.835	0.914
NG	1	0.567	0.684	0.893	**0.246**	0.745	0.734	0.791	0.774
	2	0.637	0.702	0.942	0.337	0.713	0.736	0.741	0.844
	4	0.681	0.696	**0.993**	**0.435**	0.678	0.766	0.824	0.914
Horseshoe	1	0.584	1.621	0.926	0.270	0.734	0.876	0.798	0.752
	2	0.644	0.899	**0.921**	0.334	0.682	0.742	**0.731**	0.830
	4	0.691	0.739	1.016	**0.435**	0.675	0.771	0.823	0.895

Table B.7: MSFE of joint and marginal variables using an alternative medium-scale data set. The best forecasts are in bold.

Model	Horizon	MAE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	0.535	0.959	0.998	0.567	0.841	1.011	0.929	0.882
	2	0.562	0.871	0.941	0.564	0.798	0.949	0.882	**0.897**
	4	0.584	0.863	**0.994**	0.649	0.821	0.961	0.896	0.943
Tensor MPG Own-lag	1	**0.524**	0.909	0.973	**0.538**	0.825	0.941	0.923	**0.871**
	2	**0.554**	0.820	**0.933**	**0.546**	**0.789**	0.922	**0.877**	**0.897**
	4	0.572	0.805	0.995	**0.609**	0.811	0.944	**0.886**	**0.935**
Minnesota	1	0.527	**0.875**	**0.942**	0.638	**0.815**	**0.895**	**0.911**	0.911
	2	0.563	**0.779**	0.959	0.649	**0.789**	**0.829**	0.909	0.930
	4	**0.569**	**0.739**	0.999	0.634	**0.775**	**0.864**	0.914	0.967
NG	1	0.541	0.950	0.978	0.578	0.885	0.932	0.940	0.934
	2	0.570	0.832	0.962	0.627	0.857	0.861	0.916	0.939
	4	0.576	0.772	0.996	0.643	0.810	0.898	0.912	0.968
Horseshoe	1	0.543	1.236	0.978	0.547	0.866	0.983	0.946	0.905
	2	0.566	0.952	0.933	0.582	0.811	0.868	0.899	0.911
	4	0.577	0.815	1.003	0.623	0.797	0.889	0.907	0.960

Table B.8: MAE of joint and marginal variables using an alternative medium-scale data set. The best forecasts are in bold.

The inference of margins in TVARs does not require any amendment, and the inference of $\boldsymbol{H}$ can be found in the original paper. Table B.10 - B.15 give the point and density forecasting performance using medium- and large-data sets. The same conclusion can be found from these tables: TVARs outperform standard VARs in both point and density forecasts.

We do not replace the results using the Cholesky decomposition to Ω_t by the ones using the order-invariant decomposition because some Markov chains of margins do not exhibit good mixing if we estimate parameters with the latter decomposition. For example, Figure B.4 presents the trace plots of the $(27, 2)$ entry of the tensor matrix $\boldsymbol{B}$ (corresponding to the effect

Model	Horizon	ALPL							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
	1	-12.857	0.162	0.144	0.667	0.153	0.049	0.121	0.171
Tensor MGP	2	-14.551	0.433	0.240	0.663	0.217	0.195	0.140	0.151
	4	**-15.728**	0.682	**0.182**	0.472	0.188	0.203	0.109	0.099
	1	**-12.471**	**0.193**	0.186	0.711	**0.184**	0.118	0.130	0.179
Tensor MPG Own-lag	2	**-14.384**	0.460	**0.254**	**0.691**	**0.242**	0.223	**0.146**	0.147
	4	-15.734	0.692	0.180	**0.507**	**0.198**	0.202	**0.112**	0.103
	1	-13.033	0.156	**0.203**	0.585	0.176	**0.170**	**0.173**	**0.200**
Minnesota	2	-15.146	**0.483**	0.220	0.566	0.221	**0.309**	0.139	**0.152**
	4	-16.326	**0.814**	0.143	0.454	0.187	**0.289**	0.097	0.091
	1	-13.447	0.168	0.174	**0.719**	0.138	0.166	0.157	0.176
NG	2	-15.612	0.471	0.218	0.629	0.181	0.304	0.140	0.142
	4	-16.826	0.795	0.149	0.475	0.165	0.266	0.109	0.091
	1	-13.918	-0.136	0.143	0.665	0.106	0.064	0.127	0.186
Horseshoe	2	-15.666	0.244	0.227	0.622	0.164	0.232	0.125	**0.152**
	4	-16.742	0.635	0.142	0.496	0.141	0.192	0.103	**0.104**

Table B.9: ALPL of joint and marginal variables using an alternative medium-scale data set. The best forecasts are in bold.

of the past economy to the M2 money supply). The margin inferred from the order-invariant model has bad mixing issue that affects the interpretation of $\boldsymbol{B}$.

Model	Horizon	MSFE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
	1	0.700	0.629	**0.849**	0.354	0.661	0.740	0.860	0.769
Tensor MGP	2	**0.765**	0.626	0.954	0.358	**0.613**	0.675	**0.736**	0.839
	4	**0.844**	0.668	0.999	0.475	0.636	**0.681**	0.827	0.909
	1	**0.670**	**0.583**	0.855	**0.285**	**0.638**	**0.664**	**0.692**	**0.752**
Tensor MPG Own-lag	2	0.772	**0.609**	0.945	0.357	0.640	**0.666**	0.737	0.825
	4	0.849	**0.666**	**0.993**	0.457	**0.625**	0.717	0.834	0.901
	1	0.799	1.531	1.090	0.345	0.845	1.088	0.819	0.768
Minnesota	2	0.816	0.859	**0.941**	0.374	0.674	0.863	**0.736**	**0.816**
	4	0.862	0.715	1.018	0.436	0.649	0.749	0.827	**0.895**
	1	0.700	0.675	0.918	0.298	0.721	0.796	0.773	0.763
NG	2	0.782	0.705	0.957	**0.342**	0.679	0.809	0.750	0.828
	4	0.859	0.701	0.997	0.437	0.650	0.764	0.831	0.896
	1	0.717	0.967	0.946	0.303	0.765	0.881	0.803	0.787
Horseshoe	2	0.800	0.811	0.983	0.356	0.686	0.870	0.761	0.838
	4	0.862	0.741	1.028	**0.433**	0.649	0.776	**0.820**	**0.895**

Table B.10: MSFE of joint and marginal variables using the medium-scale data set. The models applied are order-invariant. The best forecasts are in bold.

Model	Horizon	MAE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	0.594	0.934	0.950	0.590	**0.824**	0.942	0.941	0.892
	2	0.623	0.831	0.957	0.573	0.802	0.853	0.889	0.905
	4	0.652	0.805	1.006	0.623	0.802	0.878	0.903	0.947
Tensor MPG Own-lag	1	**0.573**	**0.864**	**0.932**	0.537	0.843	**0.889**	**0.876**	**0.885**
	2	**0.619**	**0.798**	0.947	**0.569**	0.833	**0.840**	0.891	0.897
	4	0.648	0.778	0.991	0.598	0.801	0.889	0.905	0.941
Minnesota	1	0.628	1.166	1.042	0.583	0.867	1.026	0.932	0.898
	2	0.633	0.831	**0.943**	0.570	**0.790**	0.886	**0.880**	**0.895**
	4	**0.644**	**0.738**	**0.988**	**0.573**	**0.765**	**0.872**	**0.896**	**0.939**
NG	1	0.595	0.926	0.986	0.590	0.873	0.949	0.941	0.912
	2	0.630	0.809	0.983	0.593	0.842	0.883	0.931	0.917
	4	0.651	0.756	1.013	0.612	0.796	0.886	0.924	0.952
Horseshoe	1	0.598	0.988	0.998	0.561	0.858	0.964	0.948	0.920
	2	0.632	0.839	0.981	0.572	0.830	0.912	0.919	0.921
	4	0.649	0.773	1.014	0.588	0.783	0.892	0.912	0.944

Table B.11: MAE of joint and marginal variables using the medium-scale data set. The models applied are order-invariant. The best forecasts are in bold.

Model	Horizon	ALPL							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	-16.338	0.095	**0.161**	0.519	0.110	0.098	0.067	0.125
	2	**-18.191**	0.371	0.195	0.538	**0.163**	0.270	0.097	0.091
	4	**-19.604**	0.616	0.135	0.375	0.130	**0.276**	**0.056**	0.051
Tensor MPG Own-lag	1	**-15.844**	**0.140**	0.155	**0.584**	**0.123**	**0.147**	**0.136**	**0.139**
	2	-18.405	**0.410**	0.201	**0.556**	**0.163**	**0.294**	**0.100**	**0.100**
	4	-20.173	**0.640**	0.151	**0.404**	**0.143**	0.265	**0.056**	0.058
Minnesota	1	-17.936	-0.266	0.040	0.405	-0.023	-0.058	0.051	0.125
	2	-18.884	0.169	**0.211**	0.434	0.054	0.153	0.070	**0.100**
	4	-20.065	0.509	**0.163**	0.308	0.024	0.190	0.032	0.051
NG	1	-18.505	0.082	0.112	0.512	0.073	0.099	0.095	0.135
	2	-21.045	0.358	0.169	0.503	0.118	0.245	0.095	**0.100**
	4	-22.763	0.605	0.119	0.369	0.099	0.258	0.054	0.055
Horseshoe	1	-18.359	0.001	0.090	0.496	0.049	0.058	0.074	0.123
	2	-21.059	0.316	0.158	0.498	0.091	0.209	0.076	0.099
	4	-22.015	0.590	0.132	0.357	0.090	0.259	0.053	**0.059**

Table B.12: ALPL of joint and marginal variables using the medium-scale data set. The models applied are order-invariant. The best forecasts are in bold.

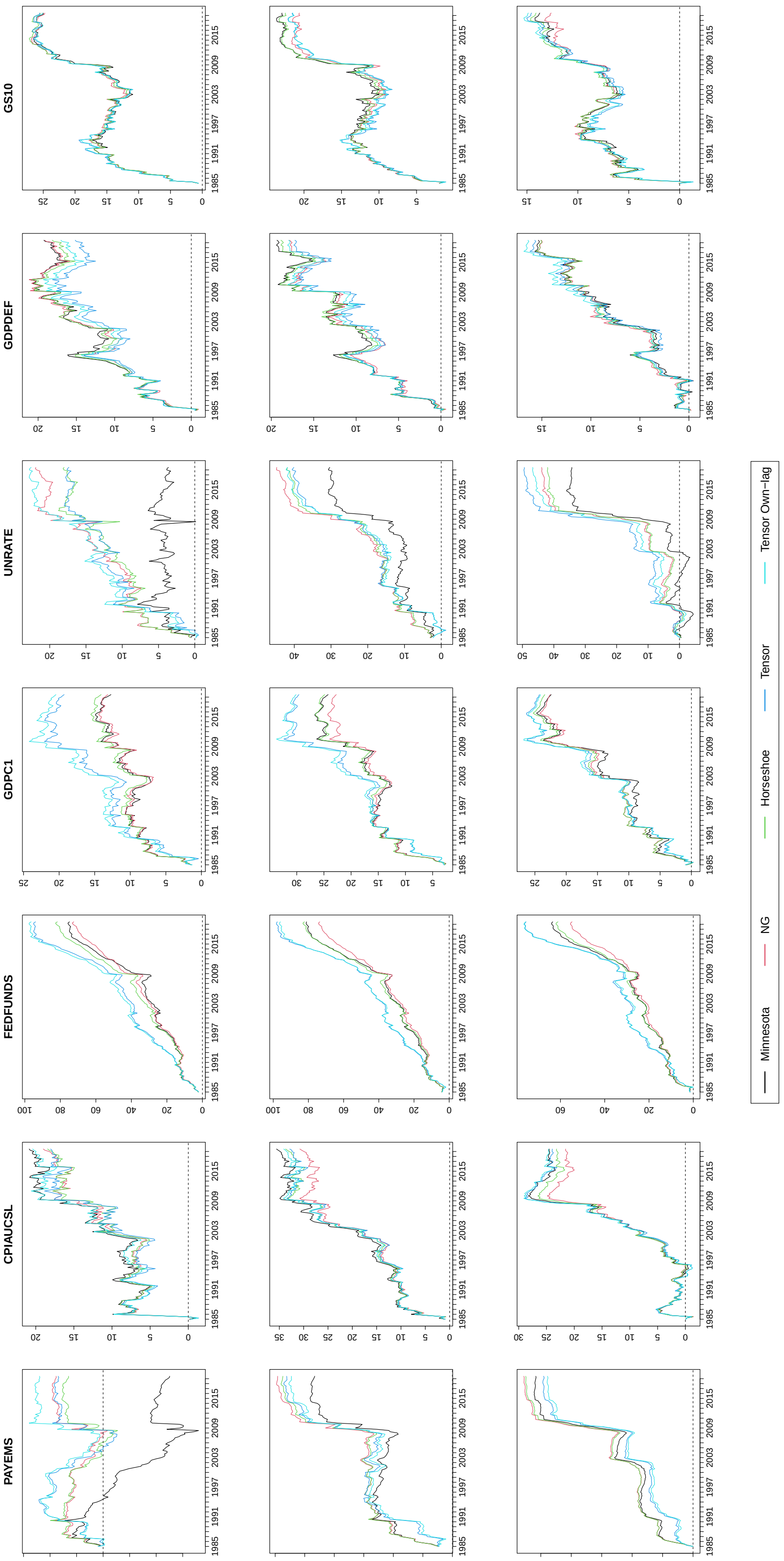


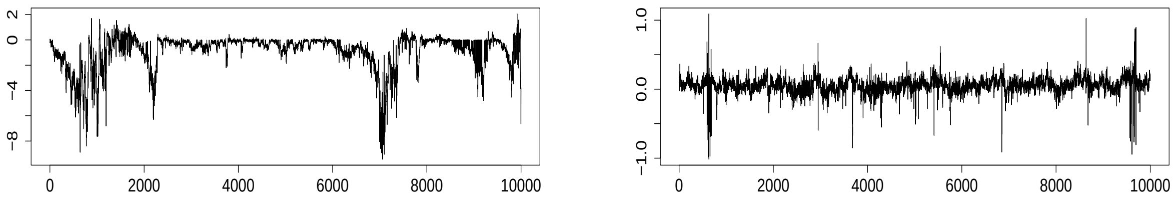
Figure B.3: Cumulative ALPL relative to the flat prior benchmark.

Model	Horizon	MSFE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
	1	0.724	1.158	0.928	0.341	0.723	0.920	0.881	**0.765**
Tensor MGP	2	0.764	0.809	0.940	0.356	0.619	0.744	0.737	**0.815**
	4	0.814	0.714	1.007	0.473	0.642	0.677	**0.820**	0.895
	1	**0.650**	**0.696**	**0.830**	**0.294**	**0.717**	**0.670**	**0.711**	0.770
Tensor MPG Own-lag	2	**0.742**	**0.609**	0.937	**0.336**	**0.599**	**0.649**	0.733	0.817
	4	**0.803**	**0.625**	**0.997**	**0.436**	**0.626**	**0.665**	0.833	0.905
	1	0.747	1.621	0.995	0.374	0.808	1.049	0.818	0.791
Minnesota	2	0.775	0.910	0.988	0.378	0.683	0.838	0.762	**0.815**
	4	0.819	0.747	1.001	0.446	0.648	0.741	0.822	**0.886**
	1	0.728	1.097	0.952	0.360	0.820	0.975	0.881	0.798
NG	2	0.769	0.819	0.958	0.372	0.674	0.807	0.741	0.827
	4	0.813	0.732	1.012	0.462	0.653	0.731	0.830	0.921
	1	0.712	0.943	0.916	0.339	0.789	0.906	0.893	0.782
Horseshoe	2	0.760	0.776	**0.936**	0.362	0.677	0.786	**0.726**	0.846
	4	0.810	0.717	1.007	0.470	0.647	0.719	0.836	0.896

Table B.13: MSFE of joint and marginal variables using the large-scale data set. The models applied are order-invariant. The best forecasts are in bold.

Model	Horizon	MAE							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
	1	0.605	1.173	0.957	0.580	**0.826**	0.900	0.942	**0.891**
Tensor MGP	2	0.619	0.906	**0.937**	0.566	**0.778**	0.822	**0.885**	**0.891**
	4	0.637	0.807	1.007	0.619	0.778	**0.823**	**0.892**	0.938
	1	**0.578**	**0.902**	**0.934**	**0.548**	0.891	**0.863**	**0.894**	0.892
Tensor MPG Own-lag	2	**0.614**	**0.794**	0.954	**0.550**	0.803	**0.806**	0.895	0.892
	4	0.640	0.779	1.003	0.585	0.819	0.856	0.905	0.942
	1	0.620	1.248	1.017	0.590	0.875	1.025	0.953	0.904
Minnesota	2	0.625	0.891	0.963	0.560	0.809	0.880	0.914	0.889
	4	0.633	0.776	**0.974**	**0.569**	0.778	0.872	0.918	**0.930**
	1	0.612	1.070	0.996	0.610	0.886	0.983	0.968	0.912
NG	2	0.625	0.844	0.970	0.592	0.812	0.859	0.908	0.893
	4	0.634	**0.756**	1.001	0.623	0.784	0.863	0.909	0.943
	1	0.605	1.025	0.970	0.606	0.874	0.954	0.980	0.911
Horseshoe	2	0.620	0.826	0.946	0.586	0.811	0.844	0.901	0.904
	4	**0.632**	0.757	0.995	0.618	**0.776**	0.845	0.921	0.939

Table B.14: MAE of joint and marginal variables using the large-scale data set. The models applied are order-invariant. The best forecasts are in bold.



(a) Non-restrictive $\boldsymbol{H}$

(b) Lower-triangular $\boldsymbol{H}$

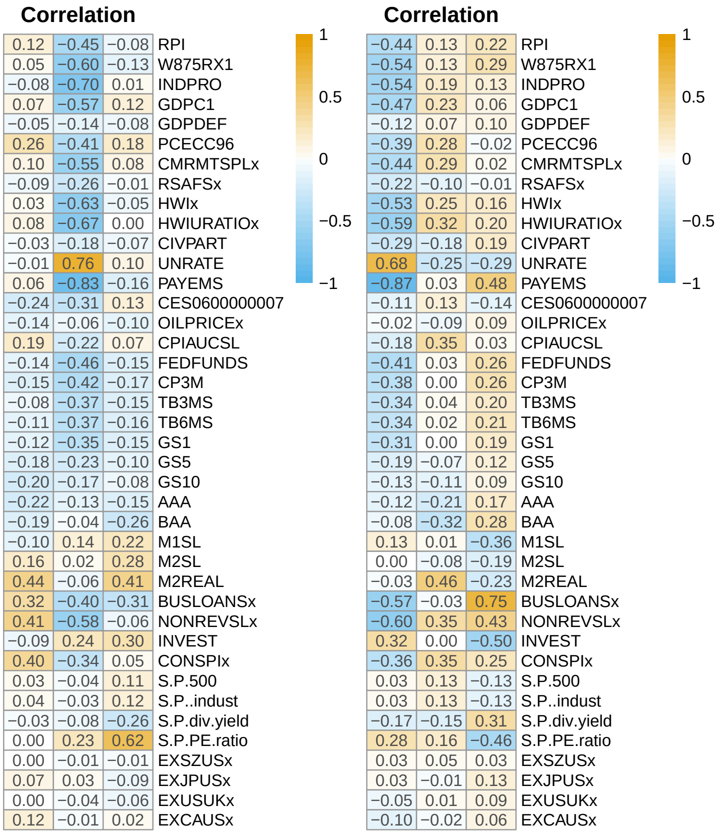
Figure B.4: Trace plots of the (27,2) entry of the tensor matrix $\boldsymbol{B}$.

Model	Horizon	ALPL							
		Joint	PAYEMS	CPIAUCSL	FEDFUNDS	GDP	UNRATE	GDPDEFL	GS10
Tensor MGP	1	-29.578	-0.141	0.096	0.416	0.054	0.093	0.004	0.116
	2	**-38.282**	0.235	0.203	0.449	0.116	0.269	0.044	0.077
	4	**-44.550**	0.528	0.156	0.267	0.076	0.275	0.004	0.013
Tensor MPG Own-lag	1	**-27.850**	**0.059**	**0.162**	**0.429**	**0.062**	**0.144**	**0.086**	**0.121**
	2	-41.932	**0.384**	**0.224**	**0.452**	**0.134**	**0.306**	**0.066**	**0.089**
	4	-54.636	**0.632**	**0.167**	**0.317**	**0.093**	**0.294**	**0.019**	**0.027**
Minnesota	1	-39.395	-0.276	0.089	0.300	0.008	-0.040	0.027	0.106
	2	-45.602	0.157	0.199	0.350	0.070	0.168	0.042	0.079
	4	-52.200	0.495	0.157	0.220	0.040	0.189	0.009	0.017
NG	1	-71.183	-0.134	0.082	0.244	-0.028	-0.009	-0.029	0.074
	2	-84.193	0.240	0.190	0.295	0.050	0.194	0.016	0.040
	4	-92.085	0.549	0.152	0.163	0.022	0.231	-0.021	-0.024
Horseshoe	1	-74.687	-0.081	0.096	0.282	-0.017	0.020	-0.028	0.071
	2	-86.357	0.280	0.202	0.317	0.054	0.213	0.025	0.036
	4	-100.722	0.570	0.148	0.182	0.028	0.239	-0.023	-0.020

Table B.15: ALPL of joint and marginal variables using the large-scale data set. The models applied are order-invariant. The best forecasts are in bold.

B.2.3 Additional Results about Interpretation

Figure B.5 shows the correlation between factors and variables.



0.12	−0.45	−0.08	RPI
0.05	−0.60	−0.13	W875RX1
−0.08	−0.70	0.01	INDPRO
0.07	−0.57	0.12	GDPC1
−0.05	−0.14	−0.08	GDPDEF
0.26	−0.41	0.18	PCECC96
0.10	−0.55	0.08	CMRMTSPLx
−0.09	−0.26	−0.01	RSAFSx
0.03	−0.63	−0.05	HWIx
0.08	−0.67	0.00	HWIURATIOx
−0.03	−0.18	−0.07	CIVPART
−0.01	0.76	0.10	UNRATE
0.06	−0.83	−0.16	PAYEMS
−0.24	−0.31	0.13	CES0600000007
−0.14	−0.06	−0.10	OILPRICEx
0.19	−0.22	0.07	CPIAUCSL
−0.14	−0.46	−0.15	FEDFUNDS
−0.15	−0.42	−0.17	CP3M
−0.08	−0.37	−0.15	TB3MS
−0.11	−0.37	−0.16	TB6MS
−0.12	−0.35	−0.15	GS1
−0.18	−0.23	−0.10	GS5
−0.20	−0.17	−0.08	GS10
−0.22	−0.13	−0.15	AAA
−0.19	−0.04	−0.26	BAA
−0.10	0.14	0.22	M1SL
0.16	0.02	0.28	M2SL
0.44	−0.06	0.41	M2REAL
0.32	−0.40	−0.31	BUSLOANSx
0.41	−0.58	−0.06	NONREVSLx
−0.09	0.24	0.30	INVEST
0.40	−0.34	0.05	CONSPIx
0.03	−0.04	0.11	S.P.500
0.04	−0.03	0.12	S.P..indust
−0.03	−0.08	−0.26	S.P.div.yield
0.00	0.23	0.62	S.P.PE.ratio
0.00	−0.01	−0.01	EXSZUSx
0.07	0.03	−0.09	EXJPUSx
0.00	−0.04	−0.06	EXUSUKx
0.12	−0.01	0.02	EXCAUSx

(a) Tensor MGP Own-lag.

−0.44	0.13	0.22	RPI
−0.54	0.13	0.29	W875RX1
−0.54	0.19	0.13	INDPRO
−0.47	0.23	0.06	GDPC1
−0.12	0.07	0.10	GDPDEF
−0.39	0.28	−0.02	PCECC96
−0.44	0.29	0.02	CMRMTSPLx
−0.22	−0.10	−0.01	RSAFSx
−0.53	0.25	0.16	HWIx
−0.59	0.32	0.20	HWIURATIOx
−0.29	−0.18	0.19	CIVPART
0.68	−0.25	−0.29	UNRATE
−0.87	0.03	0.48	PAYEMS
−0.11	0.13	−0.14	CES0600000007
−0.02	−0.09	0.09	OILPRICEx
−0.18	0.35	0.03	CPIAUCSL
−0.41	0.03	0.26	FEDFUNDS
−0.38	0.00	0.26	CP3M
−0.34	0.04	0.20	TB3MS
−0.34	0.02	0.21	TB6MS
−0.31	0.00	0.19	GS1
−0.19	−0.07	0.12	GS5
−0.13	−0.11	0.09	GS10
−0.12	−0.21	0.17	AAA
−0.08	−0.32	0.28	BAA
0.13	0.01	−0.36	M1SL
0.00	−0.08	−0.19	M2SL
−0.03	0.46	−0.23	M2REAL
−0.57	−0.03	0.75	BUSLOANSx
−0.60	0.35	0.43	NONREVSLx
0.32	0.00	−0.50	INVEST
−0.36	0.35	0.25	CONSPIx
0.03	0.13	−0.13	S.P.500
0.03	0.13	−0.13	S.P..indust
−0.17	−0.15	0.31	S.P.div.yield
0.28	0.16	−0.46	S.P.PE.ratio
0.03	0.05	0.03	EXSZUSx
0.03	−0.01	0.13	EXJPUSx
−0.05	0.01	0.09	EXUSUKx
−0.10	−0.02	0.06	EXCAUSx

(b) Tensor MGP

Figure B.5: Correlation between variables (rows) and 3 factors (columns).

B.3 Data

Slow Variables

	Name	Description	Medium	Medium (Alternative)	Large	Category	Code
1	RPI	Real Personal Income	x	x	x	1	5
2	W875RX1	RPI ex. Transfers	x		x	1	5
3	INDPRO	IP Index		x	x	1	5
4	GDP	Real Gross Domestic Product	x	x	x	1	5
5	GDPDEFL	GDP deflator	x	x	x	1	6
6	PCECC96	Real PCE	x	x	x	2	5
7	CMRMTSPLx	Real M& T Sales	x	x	x	2	5
8	RSAFSx	Retail and Food Services Sales	x	x	x	2	5
9	HWI	Help-Wanted Index for US			x	3	2
10	HWIURATIO	Help Wanted to Unemployed ratio			x	3	2

11	CIVPART	Civilian Labor Force			x	3	5
12	UNRATE	Civilian Unemployment Rate	x	x	x	3	2
13	PAYEMS	All Employees: Total nonfarm	x	x	x	3	5
14	CES0600000007	Hours: Goods-Producing			x	3	5
15	OILPRICEx	Crude Oil Prices: WTI		x	x	4	5
16	CPIAUCSL	CPI: All Items	x	x	x	4	6

Transformation code: 2 - first differences; 5 - first differences of logarithms; 6 - second differences of logarithms.

Table B.16: Description of slow variables.

Fast Variables

	Name	Description	Medium	Medium (Alternative)	Large	Category	Code
1	FEDFUNDS	Effective Federal Funds Rate	x	x	x	5	2
2	CP3Mx	3-Month AA Comm. Paper Rate	x		x	5	2
3	TB3MS	3-Month T-bill		x	x	5	2
4	TB6MS	6-Month T-bill		x	x	5	2
5	GS1	1-Year T-bond		x	x	5	2
6	GS5	5-Year T-bond			x	5	2
7	GS10	10-Year T-bond	x	x	x	5	2
8	AAA	Aaa Corporate Bond Yield			x	5	2
9	BAA	Baa Corporate Bond Yield			x	5	2
10	M1SL	M1 Money Stock			x	6	5
11	M2SL	M2 Money Stock			x	6	5
12	M2REAL	Real M2 Money Stock		x	x	6	5
13	BUSLOANS	Commercial and Industrial Loans	x		x	6	5
14	NONREVSL	Total Nonrevolving Credit	x		x	6	5
15	INVEST	Securities in Bank Credit			x	6	5
16	CONSPI	Credit to PI ratio	x	x	x	6	2
17	S&P 500	S&P 500		x	x	7	5
18	S&P: indust	S&P Industrial			x	7	5
19	S&P div yield	S&P Divident yield			x	7	2
20	S&P PE ratio	S&P Price/Earnings ratio			x	7	5
21	EXSZUSx	Switzerland / U.S. FX Rate	x		x	8	5
22	EXJPUSx	Japan / U.S. FX Rate	x		x	8	5
23	EXUSUKx	U.S. / U.K. FX Rate	x	x	x	8	5
24	EXCAUSx	Canada / U.S. FX Rate	x		x	8	5

Transformation code: 2 - first differences; 5 - first differences of logarithms; 6 - second differences of logarithms.

Table B.17: Description of fast variables.

Appendix C

Supplementary Material of Chapter 4

C.1 Marginal DIC in TVP-TVAR

The key term in the marginal DIC is $p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{b}_{j'}, \boldsymbol{\Omega}, \boldsymbol{Q}_j\right)$, which marginalizes $\boldsymbol{b}_{1:T,j}$

$$\int p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{b}_{1:T,j}, \boldsymbol{b}_{j'}, \boldsymbol{\Omega},\right) p\left(\boldsymbol{b}_{1:T,j} \mid \boldsymbol{Q}_j\right) d\boldsymbol{b}_{1:T,j}. \tag{C.1}$$

Analogous to Chan and Eisenstat (2018), $p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{b}_{j'}, \boldsymbol{\Omega}, \boldsymbol{Q}_j\right)$ has a closed form because of the state-space representations described in Section 4.2. For $j = 1$ or 3, $\boldsymbol{y} \mid \boldsymbol{b}_{1:T,j}, \boldsymbol{b}_{j'}, \boldsymbol{\Omega} \sim \mathcal{N}\left(\boldsymbol{Z}_j \boldsymbol{b}_j, \mathrm{I}_T \otimes \boldsymbol{\Omega}\right)$, where $\boldsymbol{y} = \left(\boldsymbol{y}_1', \ldots, \boldsymbol{y}_T'\right)'$, $\boldsymbol{b}_j = \left(\boldsymbol{b}_{1,j}', \ldots, \boldsymbol{b}_{T,j}'\right)'$, $\boldsymbol{Z}_j = \operatorname{diag}\left(\boldsymbol{Z}_{1,j}, \ldots, \boldsymbol{Z}_{T,j}\right)$; $\boldsymbol{b}_j \mid \boldsymbol{Q}_j \sim \mathcal{N}\left(0, \boldsymbol{H}^{-1} \boldsymbol{S}_j \left(\boldsymbol{H}^{-1}\right)'\right)$, where $\boldsymbol{H} = \begin{pmatrix} \mathbf{I}_{I_jR} & \mathbf{0} & \ldots & \mathbf{0} \\ -\mathbf{I}_{I_jR} & \mathbf{I}_{I_jR} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \ldots & -\mathbf{I}_{I_jR} & \mathbf{I}_{I_jR} \end{pmatrix}$, $\boldsymbol{S}_j = \operatorname{diag}\left(\boldsymbol{\Sigma}_j, \boldsymbol{Q}_j, \ldots, \boldsymbol{Q}_j\right)$. By plugging in the probability density functions of the above two distributions to (C.1), we can get

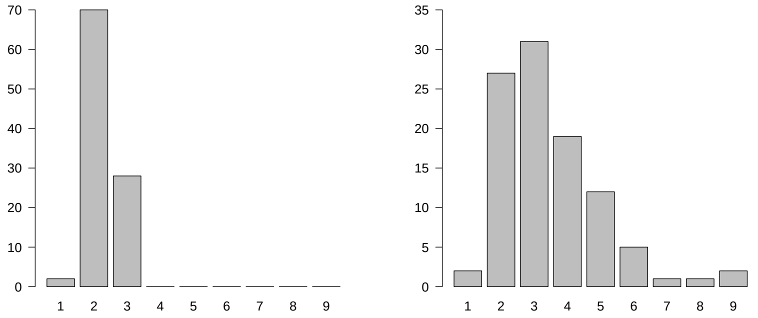
$$p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{b}_{j'}, \boldsymbol{\Omega}, \boldsymbol{Q}_j\right) = (2\pi)^{-\frac{TN}{2}} |\boldsymbol{\Omega}|^{-\frac{T}{2}} |\boldsymbol{Q}_j|^{-\frac{T-1}{2}} |\boldsymbol{\Sigma}_j|^{-\frac{1}{2}} |\boldsymbol{V}_j|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\left[\boldsymbol{y}'\left(\mathrm{I}_T \otimes \boldsymbol{\Omega}\right)^{-1} \boldsymbol{y} - \mathrm{m}_j' \boldsymbol{V}_j^{-1} \boldsymbol{m}_j\right]\right\},$$

where $\boldsymbol{V}_j = \boldsymbol{Z}_j' \left(\mathrm{I}_T \otimes \boldsymbol{\Omega}\right)^{-1} \boldsymbol{Z}_j + \boldsymbol{H}' \boldsymbol{S}_j^{-1} \boldsymbol{H}, \boldsymbol{m}_j = \boldsymbol{Z}_j' \left(\mathrm{I}_T \otimes \boldsymbol{\Omega}\right)^{-1} \boldsymbol{y}$. Thus, $\log p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{b}_{j'}, \boldsymbol{\Omega}, \boldsymbol{Q}_j\right)$ is written as

$$-\frac{TN}{2}\log 2\pi - \frac{T}{2}\log|\boldsymbol{\Omega}| - \frac{T-1}{2}\log|\boldsymbol{Q}_j| - \frac{1}{2}\log|\boldsymbol{\Sigma}_j| - \frac{1}{2}\log|\boldsymbol{V}_j| - \frac{1}{2}\left[\boldsymbol{y}'\left(\mathrm{I}_T \otimes \boldsymbol{\Omega}\right)^{-1} \boldsymbol{y} - \mathrm{m}_j' \boldsymbol{V}_j^{-1} \boldsymbol{m}_j\right].$$

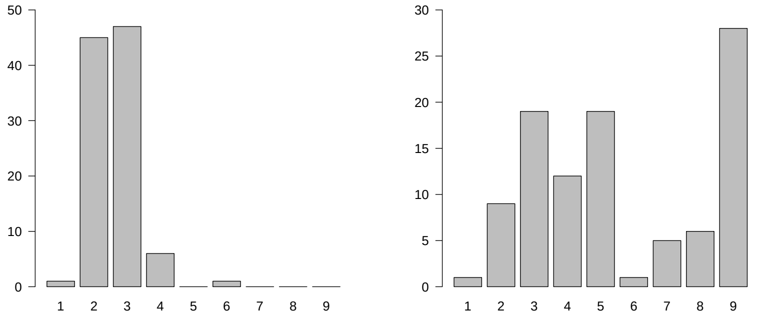
For $j = 2$, the expression of $\log p\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{b}_{j'}, \boldsymbol{\Omega}, \boldsymbol{Q}_j\right)$ is the same after modifying $\boldsymbol{Q}_2$ according to the order of elements in $\boldsymbol{b}_{t,2}^*$.

C.2 Additional Results in the Monte Carlo Study



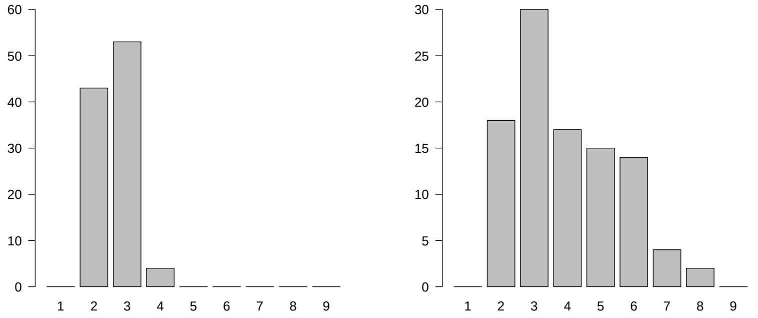
(a) With knee point detection. **(b)** Without knee point detection.

Figure C.1: Histograms of selected ranks based on data sets generated from TVAR(3).



(a) With knee point detection. **(b)** Without knee point detection.

Figure C.2: Histograms of selected ranks based on data sets generated from TVP-TVAR(3,2).



(a) With knee point detection. **(b)** Without knee point detection.

Figure C.3: Histograms of selected ranks based on data sets generated from TVP-TVAR(3,3).

C.3 Data

	ROI	Label
1	Angular gyrus	AG
2	Fusiform gyrus	F
3	Inferior temporal gyrus	IT
4	Inferior frontal gyrus, opercular part	IFG 1
5	Inferior frontal gyrus, orbital part	IFG 2
6	Inferior frontal gyrus, triangular part	IFG 3
7	Middle temporal gyrus	MT
8	Inferior occipital gyrus	IO
9	Precental gyrus	PCG
10	Precuneus	PC
11	Supplementary motor area	SM
12	Superior temporal gyrus	ST
13	Superior Temporal pole	STP
14	Supramarginal gyrus	SG

Table C.1: 27 regions of interest from both right and left cerebral hemispheres, except the supramarginal gyrus for which only the right hemisphere was considered.

Appendix D

Supplementary Material of Chapter 5

D.1 Further Discussion about Theorem 1

We demonstrate that if the third assumption of Theorem 1 does not hold, the element-wise transformation, $h_k(\cdot)$ is not a well-defined function. Note that a function is well-defined if it is a one- or many-to-one mapping. When $\tilde{g}^d_{i,\boldsymbol{\theta}^*}$ is not injective for all variable i within an anchor group, the mapping $h_1 : \mathrm{Im}\left(\tilde{g}^d_{i,\boldsymbol{\theta}^*}\right) \to \mathbb{R}$ is a one-to-many mapping, which is not a well-defined function. As a result, $h_1 \circ \tilde{g}^d_{i,\boldsymbol{\theta}^*}$ is not well-defined either.

D.2 Derivation of the ELBO

$$\begin{aligned}
\log p_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{B}}\left(\boldsymbol{x}_{1:T}\right) &= \log \int\int p_{\boldsymbol{\theta},\boldsymbol{\phi}}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right) p\left(\boldsymbol{B} \mid \boldsymbol{\Gamma}\right) p\left(\boldsymbol{\Gamma}\right)\, d\boldsymbol{\Gamma} d\boldsymbol{B} \\
&= \log \int p_{\boldsymbol{\theta},\boldsymbol{\phi}}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right) \int p\left(\boldsymbol{B} \mid \boldsymbol{\Gamma}\right) p\left(\boldsymbol{\Gamma}\right)\, d\boldsymbol{\Gamma} d\boldsymbol{B} \\
&= \log \int p_{\boldsymbol{\theta},\boldsymbol{\phi}}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right) \mathbb{E}_{p(\boldsymbol{\Gamma}\mid\boldsymbol{B})}\left[\frac{p\left(\boldsymbol{B} \mid \boldsymbol{\Gamma}\right) p\left(\boldsymbol{\Gamma}\right)}{p\left(\boldsymbol{\Gamma} \mid \boldsymbol{B}\right)}\right] d\boldsymbol{B} \\
&= \log \mathbb{E}_{q(\boldsymbol{B}\mid\boldsymbol{x}_{1:T})}\left[\frac{p\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right)}{q\left(\boldsymbol{B} \mid \boldsymbol{x}_{1:T}\right)} \mathbb{E}_{p(\boldsymbol{\Gamma}\mid\boldsymbol{B})}\left[\frac{p\left(\boldsymbol{B} \mid \boldsymbol{\Gamma}\right) p\left(\boldsymbol{\Gamma}\right)}{p\left(\boldsymbol{\Gamma} \mid \boldsymbol{B}\right)}\right]\right] \\
&\geq \mathbb{E}_{q(\boldsymbol{B}\mid\boldsymbol{x}_{1:T})}\left[\log p_{\boldsymbol{\theta},\boldsymbol{\phi}}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right) + \log \mathbb{E}_{p(\boldsymbol{\Gamma}\mid\boldsymbol{B})}\left[\frac{p\left(\boldsymbol{B} \mid \boldsymbol{\Gamma}\right) p\left(\boldsymbol{\Gamma}\right)}{p\left(\boldsymbol{\Gamma} \mid \boldsymbol{B}\right)}\right]\right] \\
&\geq \mathbb{E}_{q(\boldsymbol{B}\mid\boldsymbol{x}_{1:T})}\left[\log p_{\boldsymbol{\theta},\boldsymbol{\phi}}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right) + \mathbb{E}_{p(\boldsymbol{\Gamma}\mid\boldsymbol{B})}\left[\log \frac{p\left(\boldsymbol{B} \mid \boldsymbol{\Gamma}\right) p\left(\boldsymbol{\Gamma}\right)}{p\left(\boldsymbol{\Gamma} \mid \boldsymbol{B}\right)}\right]\right] \\
&= \underbrace{\mathbb{E}_{q(\boldsymbol{B}\mid\boldsymbol{x}_{1:T})}\left[\log p_{\boldsymbol{\theta},\boldsymbol{\phi}}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right)\right]}_{\text{Part 1}} + \underbrace{\mathbb{E}_{q(\boldsymbol{B}\mid\boldsymbol{x}_{1:T})}\left[\mathbb{E}_{p(\boldsymbol{\Gamma}\mid\boldsymbol{B})}\left[\log \frac{p\left(\boldsymbol{B} \mid \boldsymbol{\Gamma}\right) p\left(\boldsymbol{\Gamma}\right)}{p\left(\boldsymbol{\Gamma} \mid \boldsymbol{B}\right)}\right]\right]}_{\text{Part 2}},
\end{aligned} \tag{D.1}$$

where the inequality is because the logarithm is concave. In the j-th update of parameters, $q\left(\boldsymbol{B} \mid \boldsymbol{x}_{1:T}\right) = 1$ when $\boldsymbol{B} = \boldsymbol{B}^{(j)}$ and 0 otherwise.

Since we assume $\boldsymbol{x}_{1:T}$ is i.i.d across time and variables, the first part is

$$\mathbb{E}_{q(\boldsymbol{B}|\boldsymbol{x}_{1:T})}\left[\log p_{\boldsymbol{\theta},\boldsymbol{\phi}}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{B}\right)\right] = -\frac{1}{2}\sum_{t=1}^{T}\sum_{i=1}^{M}\left[\left(\boldsymbol{x}_{t,i} - \hat{\boldsymbol{x}}_{t,i}\right)^2 - \log 2\pi\right].$$

Denote $p_{c,k} = \mathbb{E}[\gamma_{c,k} \mid \boldsymbol{\beta}_{c,k}] = \frac{\psi_1(\boldsymbol{\beta}_{c,k})}{\psi_0(\boldsymbol{\beta}_{c,k})+\psi_1(\boldsymbol{\beta}_{c,k})}$, then we can write the second part as

$$\sum_{c=1}^{C}\sum_{k=1}^{K}\mathbb{E}_{p\left(\gamma_{c,k}|\boldsymbol{\beta}_{c,k}\right)}\left[\log\frac{p\left(\boldsymbol{\beta}_{c,k} \mid \gamma_{c,k}\right)p\left(\gamma_{c,k}\right)}{p\left(\gamma_{c,k} \mid \boldsymbol{\beta}_{c,k}\right)}\right]$$

$$= \sum_{c=1}^{C}\sum_{k=1}^{K} p_{c,k}\left(\log\psi_1\left(\boldsymbol{\beta}_{c,k}\right) - \log p_{c,k}\right) + (1-p_{c,k})\left(\log\psi_0\left(\boldsymbol{\beta}_{c,k}\right) - \log(1-p_{c,k})\right) - \log 2.$$

Take out the constant terms and scale the two parts by T and M, we can get the objective function in (5.10).

D.3 Cross Validation

We split the cross-validation to two parts to determine: 1) the numbers of factors and hidden layers (L), and activation function, and 2) λ_0 and λ_1. We use 5-fold cross-validation to standard autoencoders to determine the first set of hyperparameters, then apply the same technique to the grouped sparse autoencoder with the first set of hyperparameters determined to find the second set of hyperparameters. This cross-validation allows us to make the standard and grouped sparse autoencoder comparable and save computational time.

We choose the number of factors from 2 to 5 and L from 2 to 6. The activation function are selected from tanh and Leaky ReLU with the multiplier as 0.01 or 10^{-16} (which mimic the ReLU but retain an injective activation function). Table D.1 records the averaged loss from each model setting, and suggests that the best-performed model is the one with 5 factors, 3 layers, and Leaky ReLU ($a = 10^{-16}$). Then, we use this architecture to determine the values of λ_0 and λ_1, from $\{100, 500, 1000\}$ and $\{1, 0.1, 0.01, 0.001\}$, respectively. Table D.2 shows that $\lambda_0 = 1000$ and $\lambda_1 = 1$ achieves the lowest validation loss.

D.4 Additional Results of Real Data Application

Figure D.1 shows the importance measures of factors extracted from the linear grouped sparse autoencoder to different categories. The importance measures have the similar extent of sparsity like the one from the non-linear model. The factors can also be named according to the anchor groups. However, the way that factors reconstruct the high-dimensional data is different. For example, the labor market factor is relatively less important than other factors due to low mea-

Tanh				
	K=2	K=3	K=4	K=5
L=2	0.531	0.432	0.387	0.309
L=3	0.510	0.429	0.364	0.300
L=4	0.536	0.403	0.331	0.283
L=5	0.501	0.380	0.341	0.278
L=6	0.494	0.396	0.394	0.353
Leaky ReLU ($a = 0.01$)				
	K=2	K=3	K=4	K=5
L=2	0.480	0.399	0.318	0.267
L=3	0.465	0.364	0.284	0.246
L=4	0.441	0.392	0.290	0.254
L=5	0.429	0.329	0.290	0.282
L=6	0.464	0.360	0.336	0.302
Leaky ReLU ($a = 10^{-16}$)				
	K=2	K=3	K=4	K=5
L=2	0.507	0.405	0.332	0.271
L=3	0.477	0.361	0.286	0.244
L=4	0.477	0.398	0.271	0.253
L=5	0.444	0.360	0.280	0.265
L=6	0.455	0.342	0.321	0.324

Table D.1: Cross validation results from different combinations of activation function and hyperparameters.

	λ_0 = 100	λ_0 = 500	λ_0 = 1000
λ_1 = 1	0.195	0.200	0.187
λ_1 = 0.1	0.227	0.217	0.209
λ_1 = 0.01	0.237	0.238	0.230
λ_1 = 0.001	0.268	0.250	0.246

Table D.2: Cross validation results from different combination of λ_0 and λ_1.

sures in Figure D.1, but the least important factor changes to the money and credit factor if we consider the non-linear method. Moving to different groups, the linear model suggests that main driver of housing is the prices factor, while NIPA and labour market factors are the ones inferred from the non-linear model. Similar difference can also be found in other categories.

The first 4 factors from linear and non-linear grouped sparse autoencoder are highly correlated, so the corresponding variables are almost the same up to permutation, but there are a few differences. For the labor market factor, the non-linear factor is a smoother version of the linear one with more weight to the COVID-19 pandemic. The troughs around 1986 and the GFC are lower in the non-linear price factors than the linear one. Unlike the labor market factors, the non-linear interest rates factor fluctuates more than its linear counterpart, with more pronounced downturns during the three most recent recession periods. Similar to the difference in Figure 5.5, the trends of money and credit factors deviate after 1995 from positive to negative

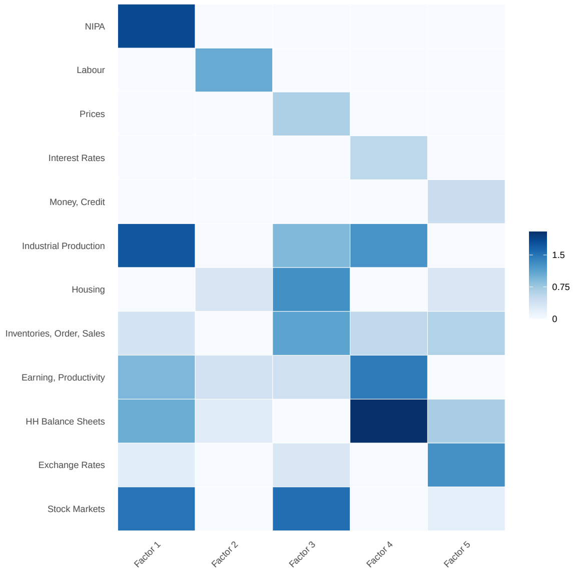


Figure D.1: Importance of factors to different categories. "HH Balance Sheets" means household balance sheets. The importance measures correspond to elements of $\boldsymbol{B}$.

correlation.

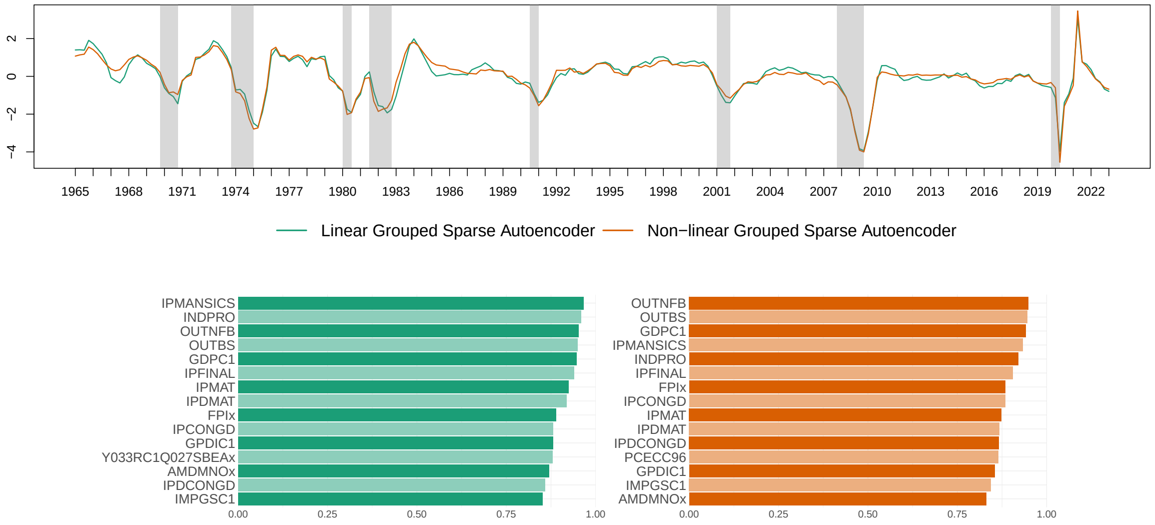


Figure D.2: The first factor extracted from the linear and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

The difference between the linear and non-linear models can also be found in the forecasting performance, as in Section 5.4.3, and the loss curves as shown in Figure D.7. This

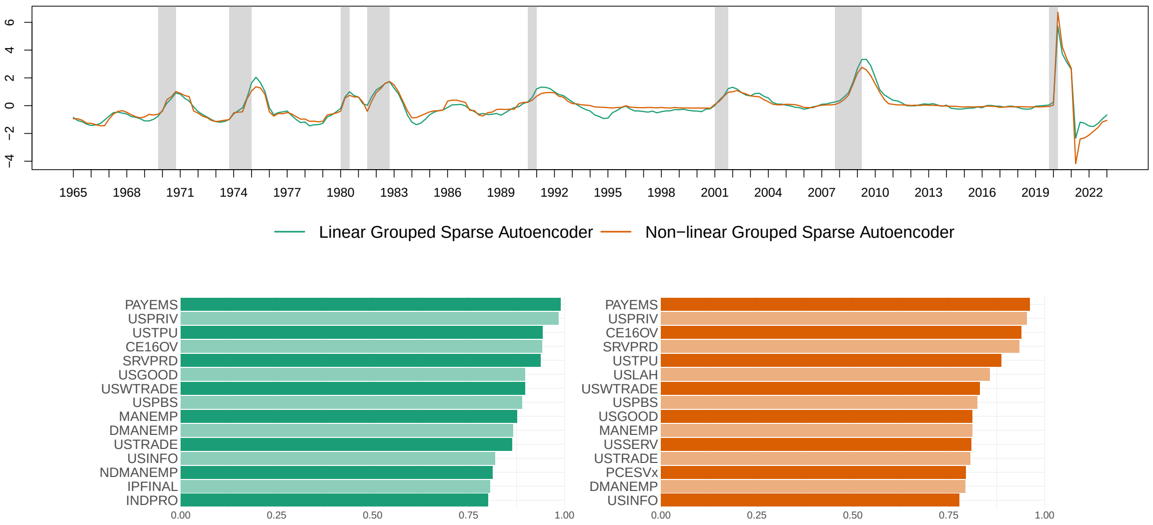


Figure D.3: The second factor extracted from the linear and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

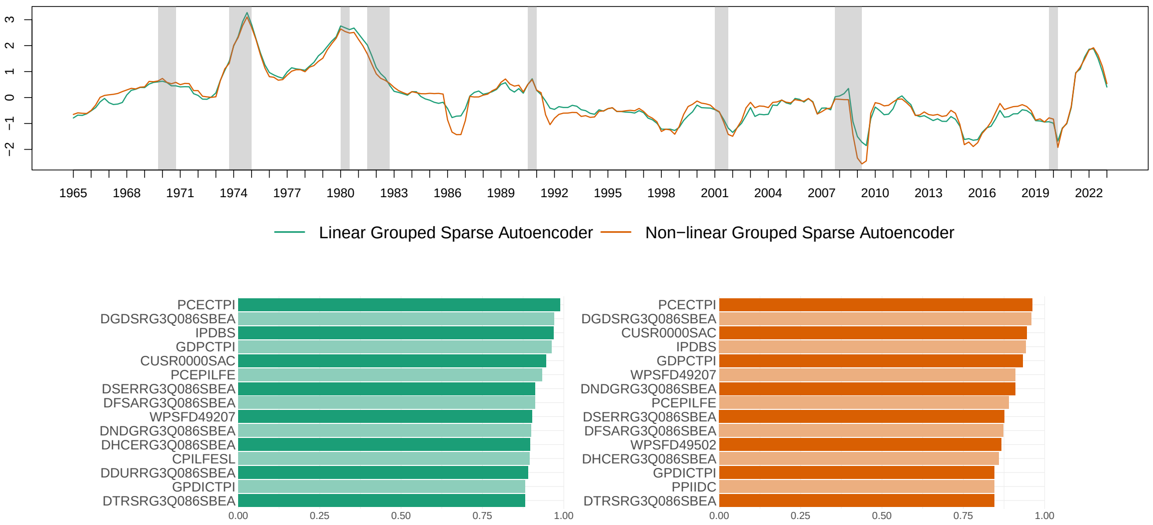


Figure D.4: The third factor extracted from the linear and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

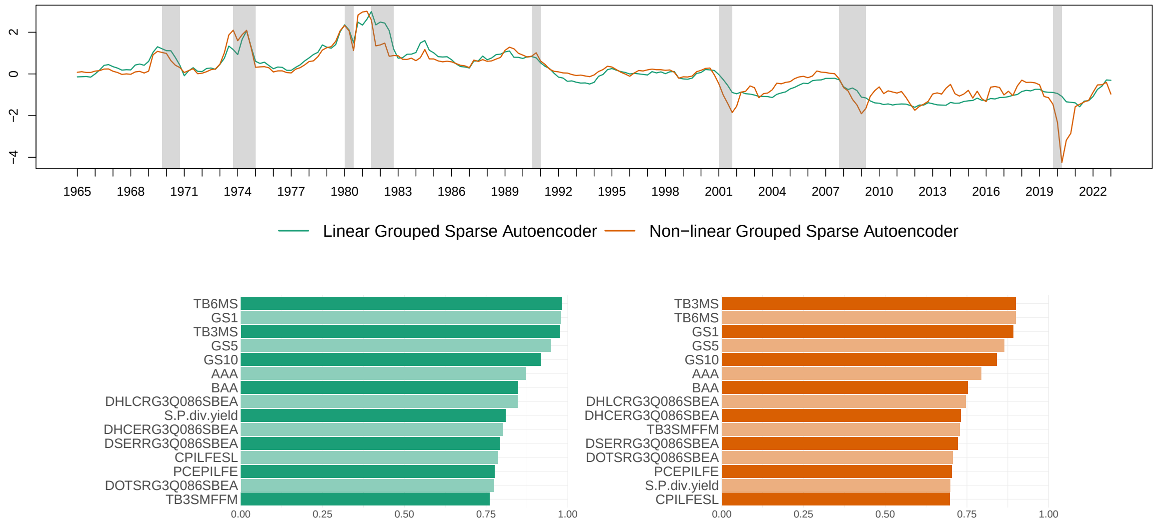


Figure D.5: The fourth factor extracted from the linear and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

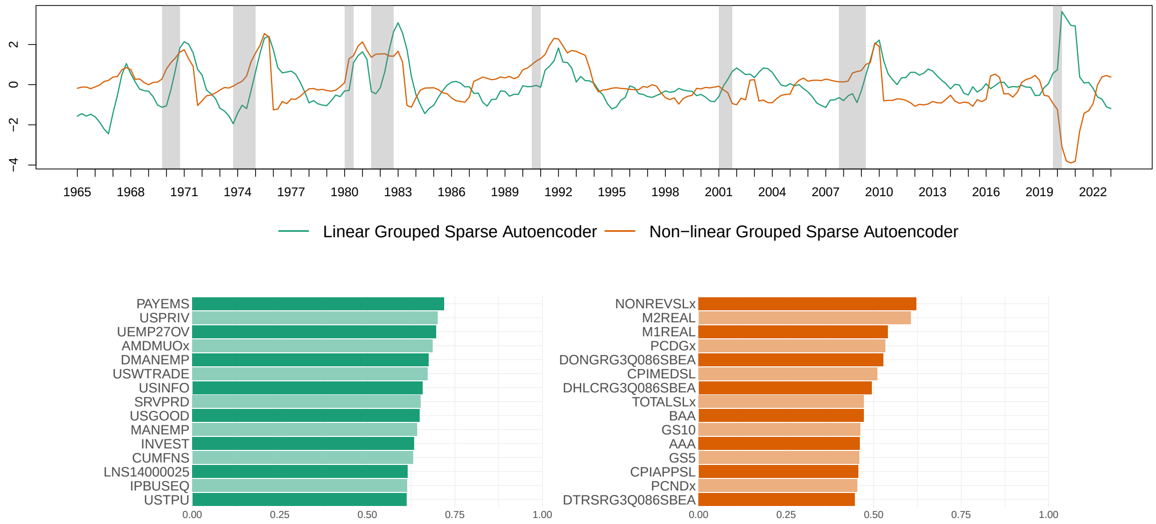


Figure D.6: The fifth factor extracted from the linear and non-linear GS autoencoder (top panel), and variables with the 15 highest correlation magnitudes with the corresponding factors (bottom panel). The time series are standardized to have zero mean and variance one. The grey bands highlight the recession periods.

figure shows the total loss on the left panel and its decomposition: reconstruction loss on the middle panel and regularization loss on the right panel. To ensure that the model performance is not influenced by a particular set of parameter initializations, we train the models 100 times with different initializations and average the losses before generating the plots. The total loss is lower in the non-linear model than the linear one, and this discrepancy is from the reconstruction loss, since the regularization counterparts are similar. Thus, there exists non-linearity in the economic model which can be captured by the non-linear model.

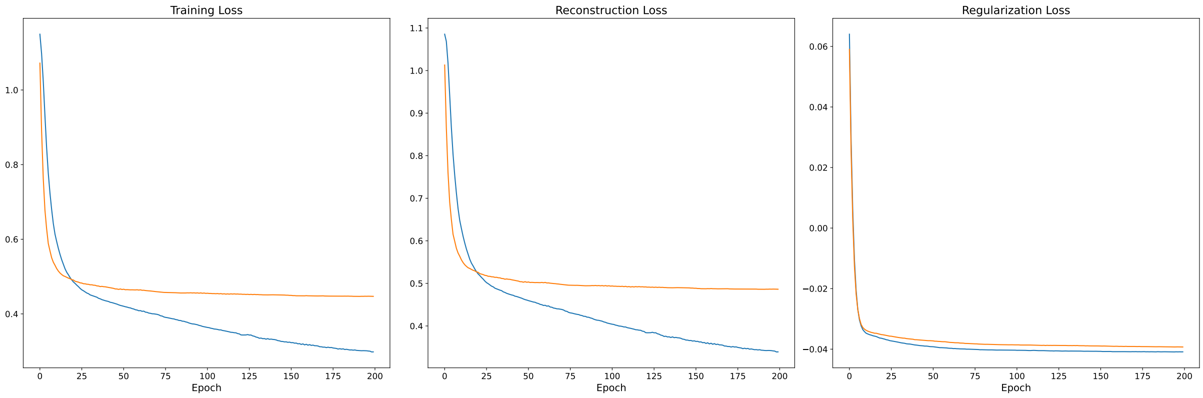


Figure D.7: Loss curves against epochs from the non-linear (blue) and linear (orange) grouped sparse autoencoder

Figure D.8 and Figure D.9 report the quarterly and yearly density forecasting performance with similar findings from the quarterly one in Figure 5.7. The overall findings in Section 5.4.3 still hold, but we observe two differences. Firstly, as the forecasting horizon increases, the linear grouped sparse autoencoder becomes the best model in forecasting real activities without heteroskedasticity, while the PCA gets worse in forecasting FEDFUNDS. Secondly, the GS autoencoder with the TVP got better performance in forecasting UNRATE before the COVID-19 pandemic, but its performance deteriorates afterward.

Figure D.10 provides 4 more time points to study the IRFs in the VAR variables. We include those studied in Korobilis (2013a), 1975:Q1 and 1996:Q1, and two additional time points corresponding to the chairmanship of Bernanke and Yellen. Overall, we find the transmission of monetary policy shifts gradually. For GDPDEF, the responses peaked earlier as time passed. The UNRATE responded with different degree of uncertainty. The rate at which the FEDFUNDS responses reached to zero decreased over time.

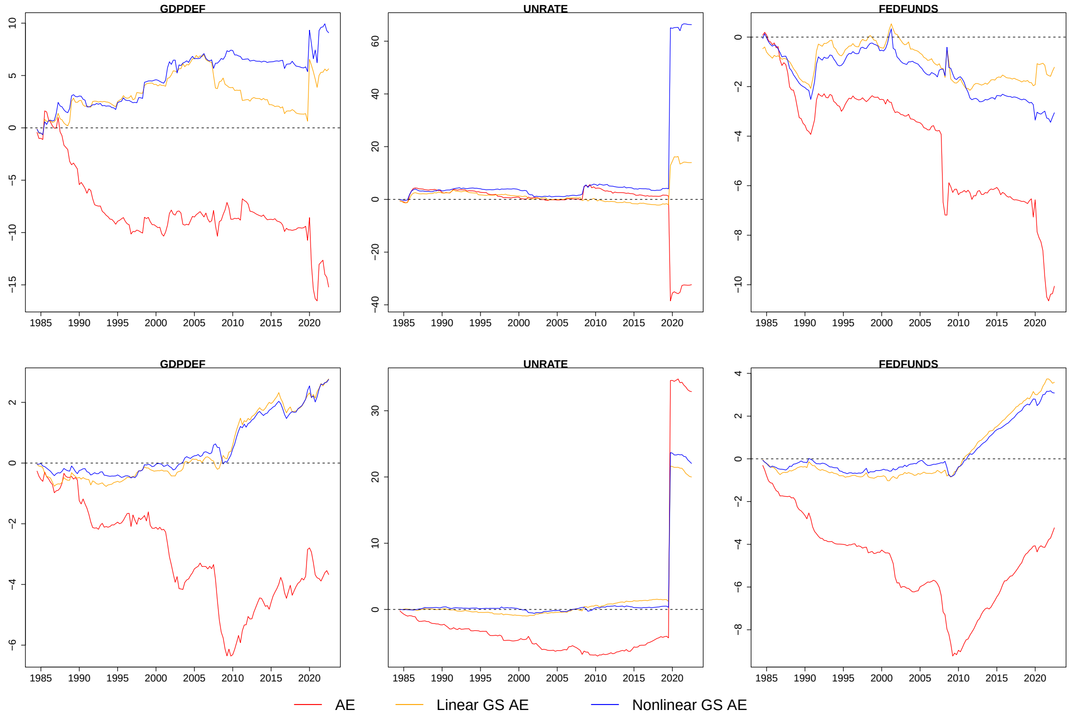


Figure D.8: Cumulative ALPL (h=1) of models relative to their PCA counterparts. The top panels consider TIV models and the bottom ones consider TVP models.

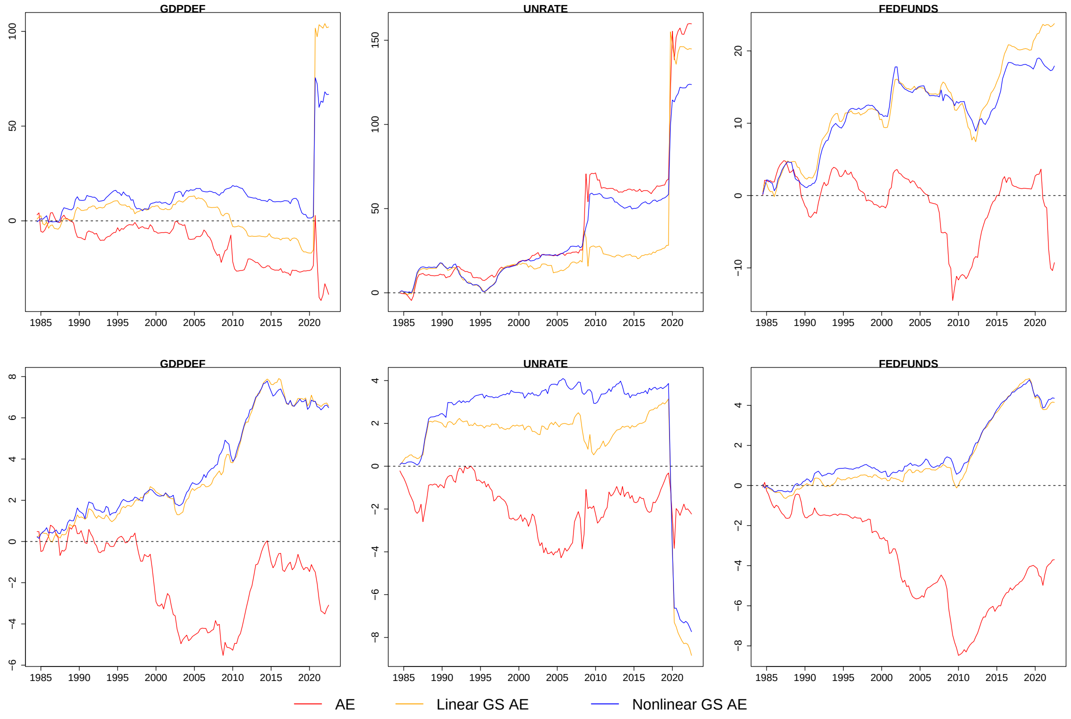


Figure D.9: Cumulative ALPL (h=4) of models relative to their PCA counterparts. The top panels consider TIV models and the bottom ones consider TVP models.

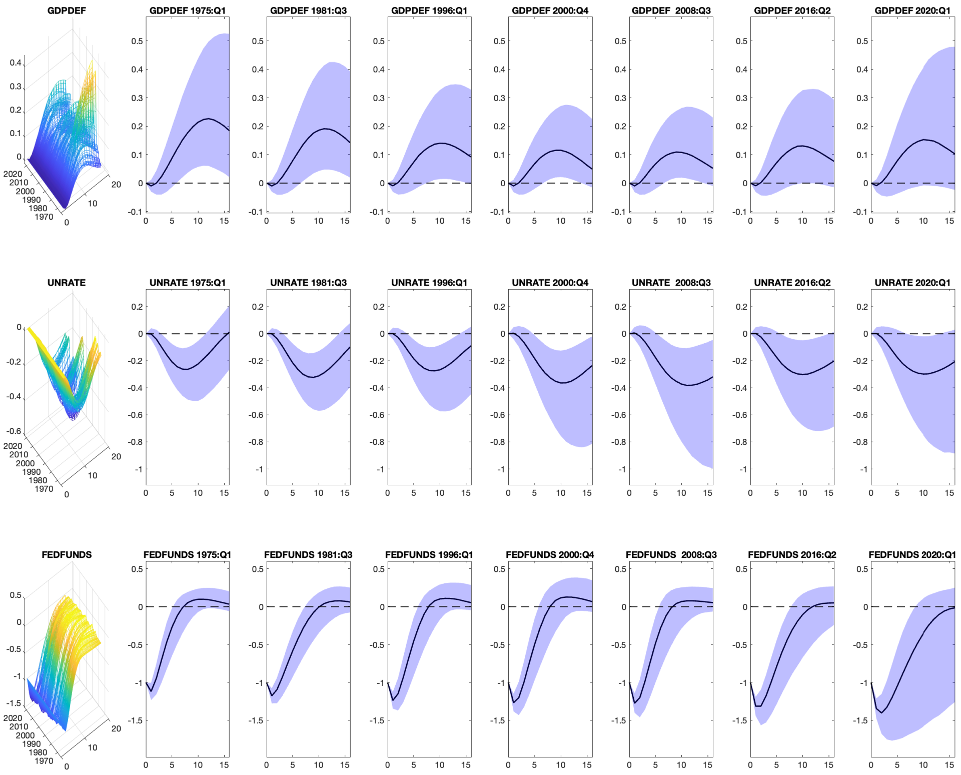


Figure D.10: Impulse responses of the VAR variables to a 100 bps decrease in FEDFUNDS. First column shows the medians over time. The rest three columns show the IRFs with their 68% credible intervals at 1975:Q1, 1981:Q3, 1996:Q1, 2000:Q4, 2008:Q3, 2016:Q2 and 2020:Q1.

D.5 Data

Slow Variables

	Name	**Description**	**Category**	**Code**
1	GDPC1	Real Gross Domestic Product	1	50
2	PCECC96	Real Personal Consumption Expenditures	1	50
3	PCDGx	Real personal consumption expenditures: Durable goods	1	50
4	PCESVx	Real Personal Consumption Expenditures: Services	1	50
5	PCNDx	Real Personal Consumption Expenditures: Nondurable Goods	1	50
6	GPDIC1	Real Gross Private Domestic Investment	1	50
7	FPIx	Real private fixed investment	1	50
8	Y033RC1Q027SBEAx	Real Gross Private Domestic Investment: Fixed Investment: Nonresidential Equip	1	50
9	PNFIx	Real private fixed investment: Nonresidential	1	50
10	PRFIx	Real private fixed investment: Residential	1	50
11	A014RE1Q156NBEA	Shares of gross domestic product: Change in private inventories in private inventories	1	1
12	GCEC1	Real Government Consumption Expenditures & Gross Investment	1	50
13	A823RL1Q225SBEA	Real Government Consumption Expenditures and Gross Investment: Federal	1	1

14	FGRECPTx	Real Federal Government Current Receipts	1	50
15	SLCEx	Real government state and local consumption expenditures	1	50
16	EXPGSC1	Real Exports of Goods & Services, 3 Decimal	1	50
17	IMPGSC1	Real Imports of Goods & Services	1	50
18	DPIC96	Real Disposable Personal Income	1	50
19	OUTNFB	Nonfarm Business Sector: Real Output	1	50
20	OUTBS	Business Sector: Real Output	1	50
21	INDPRO	Industrial Production Index	2	50
22	IPFINAL	Industrial Production: Final Products	2	50
23	IPCONGD	Industrial Production: Consumer Goods	2	50
24	IPMAT	Industrial Production: Materials	2	50
25	IPDMAT	Industrial Production: Durable Materials	2	50
26	IPNMAT	Industrial Production: Nondurable Materials	2	50
27	IPDCONGD	Industrial Production: Durable Consumer Good	2	50
28	IPB51110SQ	Industrial Production: Durable Goods: Automotive products	2	50
29	IPNCONGD	Industrial Production: Nondurable Consumer Goods	2	50
30	IPBUSEQ	Industrial Production: Business Equipment	2	50
31	IPB51220SQ	Industrial Production: Consumer energy products	2	50
32	CUMFNS	Capacity Utilization: Manufacturing	2	1
33	IPMANSICS	Industrial Production: Manufacturing	2	50
34	IPB51222S	Industrial Production: Residential Utilities	2	50
35	IPFUELS	Industrial Production: Fuel	2	50
36	PAYEMS	All Employees: Total nonfarm	3	50
37	USPRIV	All Employees: Total Private Industries	3	50
38	MANEMP	All Employees: Manufacturing	3	50
39	SRVPRD	All Employees: Service-Providing Industries	3	50
40	USGOOD	All Employees: Goods-Producing Industries	3	50
41	DMANEMP	All Employees: Durable goods	3	50
42	NDMANEMP	All Employees: Nondurable goods	3	50
43	USCONS	All Employees: Construction	3	50
44	USEHS	All Employees: Financial Activities	3	50
45	USFIRE	All Employees: Financial Activities	3	50
46	USINFO	All Employees: Information Services	3	50
47	USPBS	All Employees: Professional & Business Services	3	50
48	USLAH	All Employees: Leisure & Hospitality	3	50
49	USSERV	All Employees: Other Services	3	50
50	USMINE	All Employees: Mining and logging	3	50
51	USTPU	All Employees: Trade, Transportation & Utilities	3	50
52	USGOVT	All Employees: Government	3	50
53	USTRADE	All Employees: Retail Trade	3	50
54	USWTRADE	All Employees: Wholesale Trade	3	50
55	CES9091000001	All Employees: Government: Federal	3	50
56	CES9092000001	All Employees: Government: State Government	3	50
57	CES9093000001	All Employees: Government: Local Government	3	50
58	CE16OV	Civilian Employment	3	50
59	CIVPART	Civilian Labor Force Participation Rate	3	1
60	UNRATE	Civilian Unemployment Rate	3	1
61	UNRATESTx	Unemployment Rate less than 27 weeks	3	1
62	UNRATELTx	Unemployment Rate for more than 27 week	3	1

63	LNS14000012	Unemployment Rate - 16 to 19 years	3	1
64	LNS14000025	Unemployment Rate - 20 years and over, Men	3	1
65	LNS14000026	Unemployment Rate - 20 years and over, Women	3	1
66	UEMPLT5	Number of Civilians Unemployed - Less Than 5 Weeks	3	50
67	UEMP5TO14	Number of Civilians Unemployed for 5 to 14 Weeks	3	50
68	UEMP15T26	Number of Civilians Unemployed for 15 to 26 Weeks	3	50
69	UEMP27OV	Number of Civilians Unemployed for 27 Weeks and Over	3	50
70	AWHMAN	Average Weekly Hours of Prod and Nonsuperv Employees: Manufacturing	3	1
71	AWOTMAN	Avg Weekly Overtime Hours of Prod and Nonsuperv Employees: Manufacturing	3	1
72	HWIx	Help-Wanted Index	3	1
73	CES0600000007	Average Weekly Hours of Production and Nonsupervisory Employees: Goods-Producing	3	1
74	CLAIMSx	Initial Claims	3	50
75	HOUST	Housing Starts: Total: New Privately Owned Housing Units Started	4	50
76	HOUST5F	Privately Owned Housing Starts: 5-Unit Structures or More	4	50
77	PERMIT	New Private Housing Units Authorized by Building Permits	4	50
78	HOUSTMW	Housing Starts in Midwest Census Region	4	50
79	HOUSTNE	Housing Starts in Northeast Census Region	4	50
80	HOUSTS	Housing Starts in South Census Region	4	50
81	HOUSTW	Housing Starts in West Census Region	4	50
82	RSAFSx	Real Retail and Food Services Sales	5	50
83	AMDMNOx	Real Manufacturers’ New Orders: Durable Goods	5	50
84	AMDMUOx	Real Value of Manufacturers Unfilled Orders for Durable Goods Industries	5	50
85	BUSINVx	Total Business Inventories	5	50
86	ISRATIOx	Total Business: Inventories to Sales Ratio	5	1
87	GDPDEF	Gross Domestic Product: Implicit Price Deflator	6	1
88	PCECTPI	Pers Cons Ex: Chain-type Price Index	6	50
89	PCEPILFE	Pers Cons Exp Excluding Food and Energy	6	50
90	GDPCTPI	Gross Domestic Product: Chain-type Price Index	6	50
91	GPDICTPI	Gross Private Domestic Investment: Chain-type Price Index	6	50
92	IPDBS	Business Sector: Implicit Price Deflator	6	50
93	DGDSRG3Q086SBEA	Pers Cons Exp: Goods	6	50
94	DDURRG3Q086SBEA	Pers Cons Exp: Durable goods	6	50
95	DSERRG3Q086SBEA	Pers Cons Exp: Services	6	50
96	DNDGRG3Q086SBEA	Pers Cons Exp: Nondurable goods	6	50
97	DHCERG3Q086SBEA	Pers Cons Exp: Services: Household consumption expenditures	6	50
98	DMOTRG3Q086SBEA	Pers Cons Exp: Durable goods: Motor vehicles and parts	6	50
99	DFDHRG3Q086SBEA	Pers Cons Exp: Durable goods: Furnishings and durable household equipment	6	50
100	DREQRG3Q086SBEA	Pers Cons Exp: Durable goods: Recreational goods and vehicles	6	50
101	DODGRG3Q086SBEA	Pers Cons Exp: Durable goods: Other durable goods	6	50
102	DFXARG3Q086SBEA	Pers Cons Exp: Food and beverages for off-premises cons	6	50
103	DCLORG3Q086SBEA	Pers Cons Exp: Nondurable goods: Clothing and footwear	6	50
104	DGOERG3Q086SBEA	Pers Cons Exp: Nondurable goods: Gasoline and other energy goods	6	50
105	DONGRG3Q086SBEA	Pers Cons Exp: Nondurable goods: Other nondurable goods	6	50
106	DHUTRG3Q086SBEA	Pers Cons Exp: Services: Housing and utilities	6	50
107	DHLCRG3Q086SBEA	Pers Cons Exp: Services: Health care	6	50
108	DTRSRG3Q086SBEA	Pers Cons Exp: Transportation services	6	50
109	DRCARG3Q086SBEA	Pers Cons Exp: Recreation services	6	50
110	DFSARG3Q086SBEA	Pers Cons Exp: Recreation services	6	50
111	DIFSRG3Q086SBEA	Pers Cons Exp: Services: Food services and accommodations	6	50

112	DOTSRG3Q086SBEA	Pers Cons Exp: Financial services and insurance	6	50
113	CPIAUCSL	Consumer Price Index for All Urban Consumers: All Items	6	50
114	CPILFESL	Consumer Price Index for All Urban Consumers: All Items Less Food & Energy	6	50
115	WPSFD49207	Producer Price Index by Commodity for Finished Goods	6	50
116	PPIACO	Producer Price Index for All Commodities	6	50
117	WPSFD49502	Producer Price Index by Commodity for Finished Consumer Goods	6	50
118	WPSFD4111	Producer Price Index by Commodity for Finished Consumer Foods	6	50
119	PPIIDC	Producer Price Index by Commodity Industrial Commodities	6	50
120	WPSID61	PPI by Commodity Intermediate Materials: Supplies & Components	6	50
121	WPU0561	Producer Price Index by Commodity for Fuels and Related Products and Power	6	50
122	OILPRICEx	Real Crude Oil Prices: West Texas Intermediate (WTI) - Cushing, Oklahoma	6	50
123	WPSID62	Producer Price Index: Crude Materials for Further Processing	6	50
124	PPICMM	PPI: Commodities: Metals and metal products: Primary nonferrous metals	6	50
125	CPIAPPSL	Consumer Price Index for All Urban Consumers: Apparel	6	50
126	CPITRNSL	Consumer Price Index for All Urban Consumers: Transportation	6	50
127	CPIMEDSL	Consumer Price Index for All Urban Consumers: Medical Care	6	50
128	CUSR0000SAC	Consumer Price Index for All Urban Consumers: Commodities	6	50
129	CES2000000008x	Real Average Hourly Earnings of Prod and Nonsuperv Employees: Construction	7	50
130	CES3000000008x	Real Average Hourly Earnings of Prod and Nonsuperv Employees: Manufacturing	7	50
131	COMPRNFB	Nonfarm Business Sector: Real Compensation Per Hour (Index 2012=100)	7	50
132	CES0600000008	Average Hourly Earnings of Production and Nonsupervisory Employees	7	50

Table D.3: Description of slow variables. Transformation code: 1 - level; 5 - first differences of logarithms; 7 - $\Delta(x_t/x_{t-1} - 1)$; 50 - year-over-year log difference.

Fast Variables

	Name	Description	Category	Code
1	FEDFUNDS	Effective Federal Funds Rate	8	1
2	TB3MS	3-Month Treasury Bill: Secondary Market Rate	8	1
3	TB6MS	6-Month Treasury Bill: Secondary Market Rate	8	1
4	GS1	1-Year Treasury Constant Maturity Rate	8	1
5	GS10	10-Year Treasury Constant Maturity Rate	8	1
6	AAA	Moodys Seasoned Aaa Corporate Bond Yield	8	1
7	BAA	Moodys Seasoned Baa Corporate Bond Yield	8	1
8	BAA10YM	Moodys Seasoned Baa Corporate Bond Yield Rel. to Yield on 10-Year Treasury	8	1
9	TB6M3Mx	6-Month Treasury Bill Minus 3-Month Treasury Bill, secondary market	8	1
10	GS1TB3Mx	1-Year Treasury Constant Maturity Minus 3-Month Treasury Bill, second market	8	1
11	GS10TB3Mx	10-Year Treasury Constant Maturity Minus 3-Month Treasury Bill, second market	8	1
12	CPF3MTB3Mx	3-Month Commercial Paper Minus 3-Month Treasury Bill, second market	8	1
13	GS5	5-Year Treasury Constant Maturity Rate	8	1
14	TB3SMFFM	3-Month Treasury Constant Maturity Minus Federal Funds Rate	8	1
15	T5YFFM	5-Year Treasury Constant Maturity Minus Federal Funds Rate	8	1
16	AAAFFM	Moodys Seasoned Aaa Corporate Bond Minus Federal Funds Rate	8	1
17	M1REAL	Real M1 Money Stock	9	50
18	M2REAL	Real M2 Money Stock	9	50

19	BUSLOANSx	Real Commercial and Industrial Loans, All Commercial Banks	9	50
20	CONSUMERx	Real Consumer Loans at All Commercial Banks	9	50
21	NONREVSLx	Total Real Nonrevolving Credit Owned and Securitized, Outstanding	9	50
22	REALLNx	Real Real Estate Loans, All Commercial Banks	9	50
23	TOTALSLx	Total Consumer Credit Outstanding	9	50
24	TOTRESNS	Total Reserves of Depository Institutions	9	50
25	NONBORRES	Reserves Of Depository Institutions, Nonborrowed	9	7
26	DTCOLNVHFNM	Consumer Motor Vehicle Loans Outstanding Owned by Finance Companies	9	50
27	DTCTHFNM	Total Consumer Loans and Leases Outstanding Owned and Sec by Finance Comp	9	50
28	INVEST	Securities in Bank Credit at All Commercial Banks	9	50
29	TABSHNOx	Real Total Assets of Households and Nonprofit Organizations	10	50
30	EXSZUSx	Switzerland / U.S. Foreign Exchange Rate	11	50
31	EXJPUSx	Japan / U.S. Foreign Exchange Rate	11	50
32	EXUSUKx	U.S. / U.K. Foreign Exchange Rate	11	50
33	EXCAUSx	Canada / U.S. Foreign Exchange Rate	11	50
34	S&P 500	S&Ps Common Stock Price Index: Composite	12	5
35	S&P: indust	S&Ps Common Stock Price Index: Industrials	12	50
36	S&P div yield	S&Ps Composite Common Stock: Dividend Yield	12	1

Table D.4: Description of fast variables. Transformation code: 1 - level; 5 - first differences of logarithms; 7 - $\Delta(x_t/x_{t-1} - 1)$; 50 - year-over-year log difference.

Bibliography

Abbate, A., Eickmeier, S., Lemke, W., and Marcellino, M. (2016). The changing international transmission of financial shocks: evidence from a classical time-varying FAVAR. *Journal of Money, Credit and Banking*, 48(4):573–601.

Achinstein, A. (1961). Economic fluctuations. *American Economic History*, pages 162–80.

Ahelegbey, D. F., Billio, M., and Casarin, R. (2016a). Bayesian graphical models for structural vector autoregressive processes. *Journal of Applied Econometrics*, 31(2):357–386.

Ahelegbey, D. F., Billio, M., and Casarin, R. (2016b). Sparse graphical vector autoregression: A bayesian approach. *Annals of Economics and Statistics/Annales d'Économie et de Statistique*, (123/124):333–361.

Ainsworth, S. K., Foti, N. J., Lee, A. K., and Fox, E. B. (2018). oi-VAE: Output interpretable VAEs for nonlinear group factor analysis. In *International Conference on Machine Learning*, pages 119–128. PMLR.

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected Papers of Hirotugu Akaike*, pages 199–213. Springer.

Anderson, H. M. and Vahid, F. (1998). Testing multiple equation systems for common nonlinear components. *Journal of Econometrics*, 84(1):1–36.

Andreini, P., Izzo, C., and Ricco, G. (2020). Deep dynamic factor models. *arXiv preprint arXiv:2007.11887*.

Antonakakis, N., Chatziantoniou, I., and Gabauer, D. (2020). Refined measures of dynamic connectedness based on time-varying parameter vector autoregressions. *Journal of Risk and Financial Management*, 13(4):84.

Arias, J. E., Rubio-Ramirez, J. F., and Shin, M. (2023). Macroeconomic forecasting and variable ordering in multivariate stochastic volatility models. *Journal of Econometrics*, 235(2):1054–1086.

Ariyo, O., Lesaffre, E., Verbeke, G., and Quintero, A. (2022). Model selection for Bayesian linear mixed models with longitudinal data: sensitivity to the choice of priors. *Communications in Statistics-Simulation and Computation*, 51(4):1591–1615.

Ariyo, O., Quintero, A., Muñoz, J., Verbeke, G., and Lesaffre, E. (2020). Bayesian model selection in linear mixed models for longitudinal data. *Journal of Applied Statistics*, 47(5):890–913.

Arora, S., Ge, R., Halpern, Y., Mimno, D., Moitra, A., Sontag, D., Wu, Y., and Zhu, M. (2013). A practical algorithm for topic modeling with provable guarantees. In *International Conference on Machine Learning*, pages 280–288. PMLR.

Ashraf, M., Anowar, F., Setu, J. H., Chowdhury, A. I., Ahmed, E., Islam, A., and Al-Mamun, A. (2023). A survey on dimensionality reduction techniques for time-series data. *IEEE Access*, 11:42909–42923.

Babii, A., Ghysels, E., and Pan, J. (2022). Tensor principal component analysis. *arXiv preprint arXiv:2212.12981*.

Bacciu, D. and Mandic, D. P. (2020). Tensor decompositions in deep learning. *arXiv preprint arXiv:2002.11835*.

Bai, J., Li, K., and Lu, L. (2016). Estimation and inference of FAVAR models. *Journal of Business & Economic Statistics*, 34(4):620–641.

Bai, J. and Ng, S. (2002). Determining the number of factors in approximate factor models. *Econometrica*, 70(1):191–221.

Bai, J. and Wang, P. (2015). Identification and Bayesian estimation of dynamic factor models. *Journal of Business & Economic Statistics*, 33(2):221–240.

Bai, R., Ročková, V., and George, E. I. (2021). Spike-and-slab meets LASSO: A review of the spike-and-slab LASSO. *Handbook of Bayesian variable selection*, pages 81–108.

Baldi, P. and Hornik, K. (1989). Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58.

Ballarin, G. (2025). Ridge regularized estimation of VAR models for inference. *Journal of Time Series Analysis*, 46(2):235–257.

Bańbura, M., Giannone, D., and Reichlin, L. (2010). Large Bayesian vector auto regressions. *Journal of Applied Econometrics*, 25(1):71–92.

Banerjee, A., Marcellino, M., and Masten, I. (2008). Forecasting macroeconomic variables using diffusion indexes in short samples with structural change. In *Forecasting in the Presence of Structural Breaks and Model Uncertainty*, volume 3, pages 149–194. Emerald Group Publishing Limited.

Barigozzi, M. (2018). Dynamic factor models. *Lecture notes. London School of Economics*.

Basu, S. and Matteson, D. S. (2021). A survey of estimation methods for sparse high-dimensional time series models. *arXiv preprint arXiv:2107.14754*.

Baumeister, C., Liu, P., and Mumtaz, H. (2010). Changes in the transmission of monetary policy: Evidence from a time-varying factor-augmented VAR.

Bazerque, J. A., Mateos, G., and Giannakis, G. B. (2013). Rank regularization and Bayesian inference for tensor completion and extrapolation. *IEEE Transactions on Signal Processing*, 61(22):5689–5703.

Bazzi, M., Blasques, F., Koopman, S. J., and Lucas, A. (2017). Time-varying transition probabilities for markov regime switching models. *Journal of Time Series Analysis*, 38(3):458–478.

Belke, A. and Osowski, T. (2019). International effects of Euro area versus US policy uncertainty: A FAVAR approach. *Economic inquiry*, 57(1):453–481.

Belmonte, M. A., Koop, G., and Korobilis, D. (2014). Hierarchical shrinkage in time-varying parameter models. *Journal of Forecasting*, 33(1):80–94.

Belviso, F. and Milani, F. (2006). Structural factor-augmented VARs (SFAVARs) and the effects of monetary policy. *Topics in Macroeconomics*, 6(3).

Bernanke, B. S., Boivin, J., and Eliasz, P. (2005). Measuring the effects of monetary policy: a factor-augmented vector autoregressive (FAVAR) approach. *The Quarterly Journal of Economics*, 120(1):387–422.

Beyeler, S. and Kaufmann, S. (2021). Reduced-form factor augmented VAR—exploiting sparsity to include meaningful factors. *Journal of Applied Econometrics*, 36(7):989–1012.

Bhattacharya, A., Chakraborty, A., and Mallick, B. K. (2016). Fast sampling with gaussian scale mixture priors in high-dimensional regression. *Biometrika*, 103(4):985–991.

Bhattacharya, A. and Dunson, D. B. (2011). Sparse Bayesian infinite factor models. *Biometrika*, 98(2):291–306.

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.

Bi, X., Tang, X., Yuan, Y., Zhang, Y., and Qu, A. (2021). Tensors in statistics. *Annual Review of Statistics and Its Application*, 8(1):345–368.

Bianchi, F., Mumtaz, H., and Surico, P. (2009). Dynamics of the term structure of UK interest rates. Working Paper 363, Bank of England.

Bigler, E. D., Mortensen, S., Neeley, E. S., Ozonoff, S., Krasny, L., Johnson, M., Lu, J., Provencal, S. L., McMahon, W., and Lainhart, J. E. (2007). Superior temporal gyrus, language function, and autism. *Developmental Neuropsychology*, 31(2):217–238.

Billio, M., Casarin, R., Iacopini, M., and Kaufmann, S. (2023). Bayesian dynamic tensor regression. *Journal of Business & Economic Statistics*, 41(2):429–439.

Bing, X., Bunea, F., Ning, Y., and Wegkamp, M. (2020a). Adaptive estimation in structured factor models with applications to overlapping clustering. *The Annals of Statistics*, 48(4):2055 – 2081.

Bing, X., Bunea, F., and Wegkamp, M. (2020b). A fast algorithm with minimax optimal guarantees for topic models with an unknown number of topics. *Bernoulli*, 26(3):1765 – 1796.

Binks, R. L., Heaps, S. E., Panagiotopoulou, M., Wang, Y., and Wilkinson, D. J. (2024). Bayesian inference on the order of stationary vector autoregressions. *Bayesian Analysis*, 1(1):1–22.

Boivin, J., Giannoni, M. P., and Stevanovic, D. (2010). Monetary transmission in a small open economy: More data, fewer puzzles. *Manuscript, HEC Montreal, Erişim Tarihi*, 17(2015):10–2139.

Boivin, J., Giannoni, M. P., and Stevanović, D. (2020). Dynamic effects of credit shocks in a data-rich environment. *Journal of Business & Economic Statistics*, 38(2):272–284.

Bok, B., Caratelli, D., Giannone, D., Sbordone, A. M., and Tambalotti, A. (2018). *Annual Review of Economics*, 10(1):615–643.

Breitung, J. and Eickmeier, S. (2011). Testing for structural breaks in dynamic factor models. *Journal of Econometrics*, 163(1):71–84.

Bro, R. and Kiers, H. A. (2003). A new efficient method for determining the number of components in PARAFAC models. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 17(5):274–286.

Brune, B., Scherrer, W., and Bura, E. (2022). A state-space approach to time-varying reduced-rank regression. *Econometric Reviews*, 41(8):895–917.

Bruns, M. and Piffer, M. (2024). Tractable Bayesian estimation of smooth transition vector autoregressive models. *The Econometrics Journal*, 27(3):343–361.

Cabanilla, K. I. and Go, K. T. (2019). Forecasting, causality, and impulse response with neural vector autoregressions. *arXiv preprint arXiv:1903.09395*.

Camba-Mendez, G., Kapetanios, G., Smith, R. J., and Weale, M. R. (2003). Tests of rank in reduced rank regression models. *Journal of Business & Economic Statistics*, 21(1):145–155.

Canova, F. (1993). Modelling and forecasting exchange rates with a Bayesian time-varying coefficient model. *Journal of Economic Dynamics and Control*, 17(1-2):233–261.

Carriero, A., Chan, J., Clark, T. E., and Marcellino, M. (2022). Corrigendum to “Large Bayesian vector autoregressions with stochastic volatility and non-conjugate priors”[j. Econometrics 212 (1)(2019) 137–154]. *Journal of Econometrics*, 227(2):506–512.

Carriero, A., Clark, T. E., and Marcellino, M. (2015). Bayesian VARs: specification choices and forecast accuracy. *Journal of Applied Econometrics*, 30(1):46–73.

Carriero, A., Clark, T. E., and Marcellino, M. (2016a). Common drifting volatility in large Bayesian VARs. *Journal of Business & Economic Statistics*, 34(3):375–390.

Carriero, A., Clark, T. E., and Marcellino, M. (2019). Large Bayesian vector autoregressions with stochastic volatility and non-conjugate priors. *Journal of Econometrics*, 212(1):137–154.

Carriero, A., Kapetanios, G., and Marcellino, M. (2011). Forecasting large datasets with Bayesian reduced rank multivariate models. *Journal of Applied Econometrics*, 26(5):735–761.

Carriero, A., Kapetanios, G., and Marcellino, M. (2012). Forecasting government bond yields with large Bayesian vector autoregressions. *Journal of Banking & Finance*, 36(7):2026–2047.

Carriero, A., Kapetanios, G., and Marcellino, M. (2016b). Structural analysis with multivariate autoregressive index models. *Journal of Econometrics*, 192(2):332–348.

Carroll, J. D. and Chang, J.-J. (1970). Analysis of individual differences in multidimensional scaling via an N-way generalization of “Eckart-Young” decomposition. *Psychometrika*, 35(3):283–319.

Carter, C. K. and Kohn, R. (1994). On Gibbs sampling for state space models. *Biometrika*, 81(3):541–553.

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2009). Handling sparsity via the horseshoe. In *Artificial Intelligence and Statistics*, pages 73–80. PMLR.

Cauchy, A. et al. (1847). Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538.

Celeux, G., Forbes, F., Robert, C. P., and Titterington, D. M. (2006). Deviance information criteria for missing data models. *Bayesian Analysis*, 1(4):651 – 673.

Cen, Z. and Lam, C. (2025). Tensor time series imputation through tensor factor modelling. *Journal of Econometrics*, 249:105974.

Ceulemans, E. and Kiers, H. A. (2006). Selecting among three-mode principal component models of different types and complexities: A numerical convex hull based method. *British Journal of Mathematical and Statistical Psychology*, 59(1):133–150.

Chan, J. C. (2023). Large hybrid time-varying parameter VARs. *Journal of Business & Economic Statistics*, 41(3):890–905.

Chan, J. C. (2024). BVARs and stochastic volatility. In *Handbook of Research Methods and Applications in Macroeconomic Forecasting*, pages 43–67. Edward Elgar Publishing.

Chan, J. C. and Eisenstat, E. (2015). Marginal likelihood estimation with the cross-entropy method. *Econometric Reviews*, 34(3):256–285.

Chan, J. C. and Eisenstat, E. (2018). Bayesian model comparison for time-varying parameter VARs with stochastic volatility. *Journal of Applied Econometrics*, 33(4):509–532.

Chan, J. C., Eisenstat, E., and Strachan, R. W. (2020). Reducing the state space dimension in a large TVP-VAR. *Journal of Econometrics*, 218(1):105–118.

Chan, J. C. and Grant, A. L. (2016a). Fast computation of the deviance information criterion for latent variable models. *Computational Statistics & Data Analysis*, 100:847–859.

Chan, J. C. and Grant, A. L. (2016b). On the observed-data deviance information criterion for volatility modeling. *Journal of Financial Econometrics*, 14(4):772–802.

Chan, J. C., Jacobi, L., and Zhu, D. (2019). How sensitive are VAR forecasts to prior hyperparameters? an automated sensitivity analysis. In *Topics in Identification, Limited Dependent Variables, Partial Observability, Experimentation, and Flexible Modeling: Part A*, volume 40, pages 229–248. Emerald Publishing Limited.

Chan, J. C., Koop, G., and Yu, X. (2024). Large order-invariant Bayesian VARs with stochastic volatility. *Journal of Business & Economic Statistics*, 42(2):825–837.

Chan, J. C. and Qi, Y. (2024). Large Bayesian tensor VARs with stochastic volatility. *arXiv preprint arXiv:2409.16132*.

Chan, J. C. and Yu, X. (2022). Fast and accurate variational inference for large Bayesian VARs with stochastic volatility. *Journal of Economic Dynamics and Control*, 143:104505.

Chang, J., He, J., Yang, L., and Yao, Q. (2023). Modelling matrix time series via a tensor CP-decomposition. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 85(1):127–148.

Chen, R., Xiao, H., and Yang, D. (2021). Autoregressive models for matrix-valued time series. *Journal of Econometrics*, 222(1):539–560.

Chen, R., Yang, D., and Zhang, C.-H. (2022). Factor models for high-dimensional tensor time series. *Journal of the American Statistical Association*, 117(537):94–116.

Chen, W. (2024). *Factor modelling for tensor time series*. PhD thesis, London School of Economics and Political Science.

Chen, X., Zhang, C., Chen, X., Saunier, N., and Sun, L. (2023). Discovering dynamic patterns from spatiotemporal data with time-varying low-rank autoregression. *IEEE Transactions on Knowledge and Data Engineering*, 36(2):504–517.

Cheng, C., Sa-Ngasoongsong, A., Beyca, O., Le, T., Yang, H., Kong, Z., and Bukkapatnam, S. T. (2015). Time series forecasting for nonlinear and non-stationary processes: a review and comparative study. *Iie Transactions*, 47(10):1053–1071.

Chipman, H. A., George, E. I., and McCulloch, R. E. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298.

Chiu, C.-W. J., Mumtaz, H., and Pinter, G. (2017). Forecasting with VAR models: Fat tails and stochastic volatility. *International Journal of Forecasting*, 33(4):1124–1143.

Christiano, L. J., Eichenbaum, M., and Evans, C. L. (1999). Monetary shocks: What have we learned and to what end? In Taylor, J. B. and Woodford, M., editors, *Handbook of Macroeconomics*, volume 1A, pages 65–148. Elsevier Science, North-Holland, New York.

Clark, T. E. (2011). Real-time density forecasts from Bayesian vector autoregressions with stochastic volatility. *Journal of Business & Economic Statistics*, 29(3):327–341.

Clark, T. E. and Mertens, E. (2023). Stochastic volatility in Bayesian vector autoregressions. In *Oxford Research Encyclopedia of Economics and Finance*. Oxford University Press.

Cogley, T. and Sargent, T. J. (2001). Evolving post-world war II US inflation dynamics. *NBER Macroeconomics Annual*, 16:331–373.

Cogley, T. and Sargent, T. J. (2005). Drifts and volatilities: monetary policies and outcomes in the post WWII US. *Review of Economic Dynamics*, 8(2):262–302.

Corander, J. and Villani, M. (2006). A bayesian approach to modelling graphical vector autoregressions. *Journal of Time Series Analysis*, 27(1):141–156.

Corrado, L., Grassi, S., and Minnella, E. (2021). The Transmission Mechanism of Quantitative Easing: A Markov-Switching FAVAR Approach. CEIS Research Paper 520, Tor Vergata University, CEIS.

Cross, J. L., Hou, C., and Poon, A. (2020). Macroeconomic forecasting with large Bayesian VARs: Global-local priors and the illusion of sparsity. *International Journal of Forecasting*, 36(3):899–915.

Cubadda, G., Grassi, S., and Guardabascio, B. (2025). The time-varying multivariate autoregressive index model. *International Journal of Forecasting*, 41(1):175–190.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314.

D'Agostino, A., Gambetti, L., and Giannone, D. (2013). Macroeconomic forecasting and structural change. *Journal of Applied Econometrics*, 28(1):82–101.

Dai, H., Tian, Y., Dai, B., Skiena, S., and Song, L. (2018). Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786*.

Daniele, M. and Schnaitmann, J. (2019). A regularized factor-augmented vector autoregressive model. *arXiv preprint arXiv:1912.06049*.

Datta, J. and Ghosh, J. K. (2013). Asymptotic Properties of Bayes Risk for the Horseshoe Prior. *Bayesian Analysis*, 8(1):111 – 132.

Davis, R. A., Zang, P., and Zheng, T. (2016). Sparse vector autoregressive modeling. *Journal of Computational and Graphical Statistics*, 25(4):1077–1096.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000a). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000b). On the best rank-1 and rank-(r_1, r_2,. . . , r_n) approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342.

Del Negro, M. and Otrok, C. (2008). Dynamic factor models with time-varying parameters: measuring changes in international business cycles. Staff Reports 326, Federal Reserve Bank of New York.

Del Negro, M. and Primiceri, G. E. (2015). Time varying structural vector autoregressions and monetary policy: a corrigendum. *The Review of Economic Studies*, 82(4):1342–1345.

Diebold, F. X. and Yilmaz, K. (2012). Better to give than to receive: Predictive directional measurement of volatility spillovers. *International Journal of Forecasting*, 28(1):57–66.

Dijk, D. v., Teräsvirta, T., and Franses, P. H. (2002). Smooth transition autoregressive models—a survey of recent developments. *Econometric Reviews*, 21(1):1–47.

Doan, T., Litterman, R., and Sims, C. (1984). Forecasting and conditional projection using realistic prior distributions. *Econometric Reviews*, 3(1):1–100.

Doz, C. and Fuleky, P. (2020). Dynamic factor models. *Macroeconomic Forecasting in the Era of Big Data: Theory and Practice*, pages 27–64.

Duan, L. L., Yuwen, Z., Michailidis, G., and Zhang, Z. (2023). Low tree-rank bayesian vector autoregression models. *Journal of Machine Learning Research*, 24(286):1–35.

Dufour, J.-M. and Stevanović, D. (2013). Factor-augmented VARMA models with macroeconomic applications. *Journal of Business & Economic Statistics*, 31(4):491–506.

Düker, M.-C., Matteson, D. S., Tsay, R. S., and Wilms, I. (2025). Vector autoregressive moving average models: A review. *Wiley Interdisciplinary Reviews: Computational Statistics*, 17(1):e70009.

Durante, D. (2017). A note on the multiplicative gamma process. *Statistics & Probability Letters*, 122:198–204.

Eichler, M. (2012). Graphical modelling of multivariate time series. *Probability Theory and Related Fields*, 153(1):233–268.

Eickmeier, S., Lemke, W., and Marcellino, M. (2015). Classical time varying factor-augmented vector auto-regressive models—estimation, forecasting and structural analysis. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 178(3):493–533.

Eisenstat, E., Chan, J. C., and Strachan, R. W. (2016). Stochastic model specification search for time-varying parameter VARs. *Econometric Reviews*, 35(8-10):1638–1665.

Ellis, C., Mumtaz, H., and Zabczyk, P. (2014). What lies beneath? a time-varying FAVAR model for the UK transmission mechanism. *The Economic Journal*, 124(576):668–699.

Eltoft, T., Kim, T., and Lee, T.-W. (2006). Multivariate scale mixture of Gaussians modeling. In *International Conference on Independent Component Analysis and Signal Separation*, pages 799–806. Springer.

Erhan, D., Courville, A., Bengio, Y., and Vincent, P. (2010). Why does unsupervised pre-training help deep learning? In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 201–208. JMLR Workshop and Conference Proceedings.

Faber, N. K. M., Bro, R., and Hopke, P. K. (2003). Recent developments in CANDECOMP/PARAFAC algorithms: a critical review. *Chemometrics and Intelligent Laboratory Systems*, 65(1):119–137.

Fan, J. (1996). *Local Polynomial Modelling and Its Applications: Monographs on Statistics and Applied Probability*. Routledge, 1st edition.

Fan, J., Sitek, K., Chandrasekaran, B., and Sarkar, A. (2022). Bayesian tensor factorized mixed effects vector autoregressive processes for inferring Granger causality patterns from high-dimensional neuroimage data. *arXiv preprint arXiv:2206.10757*.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, pages 209–230.

Fischer, M. M., Hauzenberger, N., Huber, F., and Pfarrhofer, M. (2023). General Bayesian time-varying parameter vector autoregressions for modeling government bond yields. *Journal of Applied Econometrics*, 38(1):69–87.

Follett, L. and Yu, C. (2019). Achieving parsimony in Bayesian vector autoregressions with the horseshoe prior. *Econometrics and Statistics*, 11:130–144.

Foroni, C. and Marcellino, M. G. (2013). A survey of econometric methods for mixed-frequency data. *Available at SSRN 2268912*.

Frühwirth-Schnatter, S. (1994). Data augmentation and dynamic linear models. *Journal of Time Series Analysis*, 15(2):183–202.

Frühwirth-Schnatter, S. and Wagner, H. (2010). Stochastic model specification search for Gaussian and partial non-Gaussian state space models. *Journal of Econometrics*, 154(1):85–100.

Fry-Mckibbin, R. and Zheng, J. (2016). Effects of the US monetary policy shocks during financial crises–a threshold vector autoregression approach. *Applied Economics*, 48(59):5802–5823.

Fu, Z., Su, L., and Wang, X. (2024). Estimation and inference on time-varying FAVAR models. *Journal of Business & Economic Statistics*, 42(2):533–547.

Galvao, A. B. and Marcellino, M. (2014). The effects of the monetary policy stance on the transmission mechanism. *Studies in Nonlinear Dynamics & Econometrics*, 18(3):217–236.

Gaschler-Markefski, B., Baumgart, F., Tempelmann, C., Schindler, F., Stiller, D., Heinze, H.-J., and Scheich, H. (1997). Statistical methods in functional magnetic resonance imaging with respect to nonstationary time-series: auditory cortex activity. *Magnetic Resonance in Medicine*, 38(5):811–820.

Gefang, D. (2014). Bayesian doubly adaptive elastic-net Lasso for VAR shrinkage. *International Journal of Forecasting*, 30(1):1–11.

Gefang, D., Koop, G., and Poon, A. (2023). Forecasting using variational Bayesian inference in large vector autoregressions with hierarchical shrinkage. *International Journal of Forecasting*, 39(1):346–363.

Gefang, D. and Strachan, R. (2009). Nonlinear impacts of international business cycles on the UK–a Bayesian smooth transition var approach. *Studies in Nonlinear Dynamics & Econometrics*, 14(1).

Gelman, A. and Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical science*, 7(4):457–472.

George, E. I. and McCulloch, R. E. (1993). Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889.

George, E. I., Sun, D., and Ni, S. (2008). Bayesian stochastic search for VAR model restrictions. *Journal of Econometrics*, 142(1):553–580.

Geweke, J. (1977). The dynamic factor analysis of economic time series. In Aigner, D. J. and Goldberger, A. S., editors, *Latent Variables in Socio-Economic Models*. North-Holland, Amsterdam.

Geweke, J. (1991). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. Technical report, Federal Reserve Bank of Minneapolis.

Geweke, J. (1996). Bayesian reduced rank regression in econometrics. *Journal of Econometrics*, 75(1):121–146.

Ghosh, P., Tang, X., Ghosh, M., and Chakrabarti, A. (2015). Asymptotic properties of Bayes risk of a general class of shrinkage priors in multiple hypothesis testing under sparsity. *Bayesian Analysis*, 11.

Ghosh, S., Khare, K., and Michailidis, G. (2019). High-dimensional posterior consistency in Bayesian vector autoregressive models. *Journal of the American Statistical Association*, 114(526):735–748.

Ghosh, S., Khare, K., and Michailidis, G. (2021). Strong selection consistency of Bayesian vector autoregressive models based on a pseudo-likelihood approach. *The Annals of Statistics*, 49(3):1267–1299.

Giannone, D., Lenza, M., Momferatou, D., and Onorante, L. (2014). Short-term inflation projections: A Bayesian vector autoregressive approach. *International Journal of Forecasting*, 30(3):635–644.

Giannone, D., Lenza, M., and Primiceri, G. E. (2015). Prior selection for vector autoregressions. *Review of Economics and Statistics*, 97(2):436–451.

Giannone, D., Lenza, M., and Primiceri, G. E. (2021). Economic predictions with big data: The illusion of sparsity. *Econometrica*, 89(5):2409–2437.

Goan, E. and Fookes, C. (2020). Bayesian neural networks: An introduction and survey. *Case Studies in Applied Bayesian Data Science: CIRM Jean-Morlet Chair, Fall 2018*, pages 45–87.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. (2018). Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT Press Cambridge.

Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society*, pages 424–438.

Griffin, J. and Brown, P. (2010). Inference with normal-gamma prior distributions in regression problems. *Bayesian Analysis*, 5:171–188.

Gruber, L. and Kastner, G. (2025). Forecasting macroeconomic data with Bayesian VARs: Sparse or dense? it depends! *International Journal of Forecasting*. In press.

Guhaniyogi, R., Qamar, S., and Dunson, D. B. (2017). Bayesian tensor regression. *Journal of Machine Learning Research*, 18(79):1–31.

Guhaniyogi, R. and Spencer, D. (2021). Bayesian tensor response regression with an application to brain activation studies. *Bayesian Analysis*, 16(4):1221–1249.

Gupta, R., Marco Lau, C. K., Plakandaras, V., and Wong, W.-K. (2019). The role of housing sentiment in forecasting us home sales growth: evidence from a Bayesian compressed vector autoregressive model. *Economic Research-Ekonomska Istraživanja*, 32(1):2554–2567.

Hacioglu, S. and Tuzcuoglu, K. (2016). Interpreting the latent dynamic factors by threshold FAVAR model. Working paper 622, Bank of England.

Hajargasht, G. and Woźniak, T. (2018). Accurate computation of marginal data densities using variational Bayes. *arXiv preprint arXiv:1805.10036*.

Han, Y., Chen, R., and Zhang, C.-H. (2022). Rank determination in tensor factor model. *Electronic Journal of Statistics*, 16(1):1726–1803.

Han, Y., Yang, D., Zhang, C.-H., and Chen, R. (2024). CP factor model for dynamic tensors. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(5):1383–1413.

Hannan, E. J. (1979). The statistical theory of linear systems. In *Developments in Statistics*, volume 2, pages 83–121. Elsevier.

Härdle, W., Tsybakov, A., and Yang, L. (1998). Nonparametric vector autoregression. *Journal of Statistical Planning and Inference*, 68(2):221–245.

Harris, K. D., Aravkin, A., Rao, R., and Brunton, B. W. (2021). Time-varying autoregression with low-rank tensors. *SIAM Journal on Applied Dynamical Systems*, 20(4):2335–2358.

Harshman, R. A. et al. (1970). Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16(1):84.

Harvey, A. C. (1990). *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press.

Håstad, J. (1989). Tensor rank is NP-complete. In *Automata, Languages and Programming: 16th International Colloquium Stresa, Italy, July 11–15, 1989 Proceedings 16*, pages 451–460. Springer.

Hauzenberger, N., Huber, F., and Klieber, K. (2023a). Real-time inflation forecasting using non-linear dimension reduction techniques. *International Journal of Forecasting*, 39(2):901–921.

Hauzenberger, N., Huber, F., Koop, G., and Mitchell, J. (2022). Bayesian modeling of TVP-VARs using regression trees. *arXiv preprint arXiv:2209.11970*.

Hauzenberger, N., Huber, F., Koop, G., and Mitchell, J. (2023b). Bayesian modeling of time-varying parameters using regression trees. Working Paper 23-05, Federal Reserve Bank of Cleveland.

Hauzenberger, N., Huber, F., Marcellino, M., and Petz, N. (2025). Gaussian process vector autoregressions and macroeconomic uncertainty. *Journal of Business & Economic Statistics*, 43(1):27–43.

Havlicek, M., Jan, J., Brazdil, M., and Calhoun, V. D. (2010). Dynamic Granger causality based on Kalman filter for evaluation of functional network connectivity in fMRI data. *Neuroimage*, 53(1):65–77.

Hecq, A., Ricardo, I., and Wilms, I. (2024). Reduced-rank matrix autoregressive models: A medium n approach. *arXiv preprint arXiv:2407.07973*.

Hewamalage, H., Bergmeir, C., and Bandara, K. (2021). Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, 37(1):388–427.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507.

Hitchcock, F. L. (1927). The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1-4):164–189.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Hsu, N.-J., Hung, H.-L., and Chang, Y.-M. (2008). Subset selection for vector autoregressive processes using lasso. *Computational Statistics & Data Analysis*, 52(7):3645–3657.

Huang, F., Lu, K., and Zheng, Y. (2024). SARMA: Scalable low-rank high-dimensional autoregressive moving averages via tensor decomposition. *arXiv preprint arXiv:2405.00626*.

Huber, F. (2014). Forecasting exchange rates using Bayesian threshold vector autoregressions. *Economics Bulletin*, 34(3):1687–1695.

Huber, F. and Feldkircher, M. (2019). Adaptive shrinkage in Bayesian vector autoregressive models. *Journal of Business & Economic Statistics*, 37(1):27–39.

Huber, F. and Fischer, M. M. (2018). A markov switching factor-augmented VAR model for analyzing US business cycles and monetary policy. *Oxford Bulletin of Economics and Statistics*, 80(3):575–604.

Huber, F., Koop, G., and Onorante, L. (2021). Inducing sparsity and shrinkage in time-varying parameter models. *Journal of Business & Economic Statistics*, 39(3):669–683.

Huber, F. and Rossini, L. (2022). Inference in Bayesian additive vector autoregressive tree models. *The Annals of Applied Statistics*, 16(1):104–123.

Hubrich, K. and Teräsvirta, T. (2013). Thresholds and smooth transitions in vector autoregressive models. In *VAR Models in Macroeconomics—New Developments and Applications: Essays in Honor of Christopher A. Sims*, volume 32 of *Advances in Econometrics*. Emerald Group Publishing Limited.

Inayati, S., Iriawan, N., and Irhamah (2024). A markov switching autoregressive model with time-varying parameters. *Forecasting*, 6(3):568–590.

Izenman, A. J. (2012). Introduction to manifold learning. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(5):439–446.

Jeffreys, H. (1935). Some tests of significance, treated by the theory of probability. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 31, pages 203–222. Cambridge University Press.

Ji, Y., Wang, Q., Li, X., and Liu, J. (2019). A survey on tensor techniques and applications in machine learning. *IEEE Access*, 7:162950–162990.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine learning*, 37:183–233.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589.

Juselius, K. (2006). *The cointegrated VAR model: methodology and applications*. Oxford University Press.

Kadiyala, K. R. and Karlsson, S. (1997). Numerical methods for estimation and inference in Bayesian VAR-models. *Journal of Applied Econometrics*, 12(2):99–132.

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23(3):187–200.

Kalli, M. and Griffin, J. E. (2018). Bayesian nonparametric vector autoregressive models. *Journal of Econometrics*, 203(2):267–282.

Karim, R. G., Guo, G., Yan, D., and Navasca, C. (2020). Accurate tensor decomposition with simultaneous rank approximation for surveillance videos. In *2020 54th Asilomar Conference on Signals, Systems, and Computers*, pages 842–846. IEEE.

Karlsson, S. (2013). Forecasting with Bayesian vector autoregression. *Handbook of Economic Forecasting*, 2:791–897.

Kastner, G. (2016). Dealing with stochastic volatility in time series using the R package stochvol. *Journal of Statistical Software*, 69(5):1–30.

Kastner, G. and Frühwirth-Schnatter, S. (2014). Ancillarity-sufficiency interweaving strategy (ASIS) for boosting MCMC estimation of stochastic volatility models. *Computational Statistics & Data Analysis*, 76:408–423.

Kastner, G., Frühwirth-Schnatter, S., and Lopes, H. F. (2017). Efficient Bayesian inference for multivariate factor stochastic volatility models. *Journal of Computational and Graphical Statistics*, 26(4):905–917.

Kastner, G. and Huber, F. (2020). Sparse Bayesian vector autoregressions in huge dimensions. *Journal of Forecasting*, 39(7):1142–1165.

Kaufmann, S. and Schumacher, C. (2019). Bayesian estimation of sparse dynamic factor models with order-independent and ex-post mode identification. *Journal of Econometrics*, 210(1):116–134.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. (2020). Variational autoencoders and nonlinear ICA: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR.

Khozeimeh, F., Sharifrazi, D., Izadi, N. H., Joloudari, J. H., Shoeibi, A., Alizadehsani, R., Gorriz, J. M., Hussain, S., Sani, Z. A., Moosaei, H., et al. (2021). Combining a convolutional neural network with autoencoders to predict the survival chance of COVID-19 patients. *Scientific Reports*, 11(1):15343.

Kiefer, J. and Wolfowitz, J. (1952). Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, pages 462–466.

Kiers, H. A. (1997). Three-mode orthomax rotation. *Psychometrika*, 62(4):579–598.

Kiers, H. A. (1998). Joint orthomax rotation of the core and component matrices resulting from three-mode principal components analysis. *Journal of Classification*, 15(2):245–263.

Kiers, H. A. (2000). Towards a standardized notation and terminology in multiway analysis. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 14(3):105–122.

Kiers, H. A. and Der Kinderen, A. (2003). A fast method for choosing the numbers of components in Tucker3 analysis. *British Journal of Mathematical and Statistical Psychology*, 56(1):119–125.

Kilian, L. and Ivanov, V. (2001). A practitioner's guide to lag-order selection for vector autoregressions. Discussion Paper 2685, Centre for Economic Policy Research (CEPR). CEPR Press, Paris & London.

Kilian, L. and Lütkepohl, H. (2017). *Structural vector autoregressive analysis*. Cambridge University Press.

Kim, S., Shephard, N., and Chib, S. (1998). Stochastic volatility: likelihood inference and comparison with ARCH models. *The Review of Economic Studies*, 65(3):361–393.

Kingma, D. P. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational Bayes. In *International Conference on Learning Representations (ICLR)*.

Kitagawa, G. and Gersch, W. (2012). *Smoothness priors analysis of time series*, volume 116. Springer Science & Business Media.

Klieber, K. (2024). Non-linear dimension reduction in factor-augmented vector autoregressions. *Journal of Economic Dynamics and Control*, 159:104800.

Klindt, D., Schott, L., Sharma, Y., Ustyuzhaninov, I., Brendel, W., Bethge, M., and Paiton, D. (2020). Towards nonlinear disentanglement in natural data with temporal sparse coding. *arXiv preprint arXiv:2007.10930*.

Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review*, 51(3):455–500.

Komunjer, I. and Ng, S. (2011). Dynamic identification of dynamic stochastic general equilibrium models. *Econometrica*, 79(6):1995–2032.

Koop, G. and Korobilis, D. (2013). Large time-varying parameter VARs. *Journal of Econometrics*, 177(2):185–198.

Koop, G. and Korobilis, D. (2014). A new index of financial conditions. *European Economic Review*, 71:101–116.

Koop, G., Korobilis, D., and Pettenuzzo, D. (2019). Bayesian compressed vector autoregressions. *Journal of Econometrics*, 210(1):135–154.

Koop, G., Leon-Gonzalez, R., and Strachan, R. W. (2009). On the evolution of the monetary policy transmission mechanism. *Journal of Economic Dynamics and Control*, 33(4):997–1017.

Koop, G., Pesaran, M. H., and Potter, S. M. (1996). Impulse response analysis in nonlinear multivariate models. *Journal of econometrics*, 74(1):119–147.

Koop, G. M. (2013). Forecasting with medium and large Bayesian VARs. *Journal of Applied Econometrics*, 28(2):177–203.

Korobilis, D. (2008). Forecasting in vector autoregressions with many predictors. In *Bayesian Econometrics*, pages 403–431. Emerald Group Publishing Limited.

Korobilis, D. (2013a). Assessing the transmission of monetary policy using time-varying parameter dynamic factor models. *Oxford Bulletin of Economics and Statistics*, 75(2):157–179.

Korobilis, D. (2013b). VAR forecasting using Bayesian variable selection. *Journal of Applied Econometrics*, 28(2):204–230.

Korobilis, D. and Pettenuzzo, D. (2019). Adaptive hierarchical priors for high-dimensional vector autoregressions. *Journal of Econometrics*, 212(1):241–271.

Korobilis, D., Shimizu, K., et al. (2022). Bayesian approaches to shrinkage and sparse estimation. *Foundations and Trends® in Econometrics*, 11(4):230–354.

Krolzig, H.-M. and Krolzig, H.-M. (1997). *The markov-switching vector autoregressive model.* Springer.

Krueger, F. (2015). bvarsv: Bayesian analysis of a vector autoregressive model with stochastic volatility and time-varying parameters. *R package version*, 1.

Kusner, M. J., Paige, B., and Hernández-Lobato, J. M. (2017). Grammar variational autoencoder. In *International Conference on Machine Learning*, pages 1945–1954. PMLR.

Kwon, Y., Bozdogan, H., and Bensmail, H. (2008). Performance of model selection criteria in Bayesian threshold VAR (TVAR) models. *Econometric Reviews*, 28(1-3):83–101.

Lachapelle, S., Mahajan, D., Mitliagkas, I., and Lacoste-Julien, S. (2024). Additive decoders for latent variables identification and cartesian-product extrapolation. *Advances in Neural Information Processing Systems*, 36.

LeCun, Y. (1987). *Modeles connexionnistes de l'apprentissage (connectionist learning models).* PhD thesis, Université P. et M. Curie (Paris 6).

Ledermann, W. (1937). On the rank of the reduced correlational matrix in multiple-factor analysis. *Psychometrika*, 2(2):85–93.

Legramanti, S., Durante, D., and Dunson, D. B. (2020). Bayesian cumulative shrinkage for infinite factorizations. *Biometrika*, 107(3):745–752.

Li, L. and Zhang, X. (2017). Parsimonious tensor response regression. *Journal of the American Statistical Association*, 112(519):1131–1146.

Li, X., Safikhani, A., and Shojaie, A. (2022). Estimation of high-dimensional Markov-switching VAR models with an approximate EM algorithm. *arXiv preprint arXiv:2210.07456*.

Li, X., Xu, D., Zhou, H., and Li, L. (2018). Tucker tensor regression and neuroimaging analysis. *Statistics in Biosciences*, 10(3):520–545.

Li, Y., Yu, J., and Zeng, T. (2020). Deviance information criterion for latent variable models and misspecified models. *Journal of Econometrics*, 216(2):450–493.

Li, Z. and Xiao, H. (2021). Multi-linear tensor autoregressive models. *arXiv preprint arXiv:2110.00928*.

Liakakis, G., Nickel, J., and Seitz, R. (2011). Diversity of the inferior frontal gyrus—a meta-analysis of neuroimaging studies. *Behavioural Brain Research*, 225(1):341–347.

Lin, J. and Michailidis, G. (2020). Regularized estimation of high-dimensional factor-augmented vector autoregressive (FAVAR) models. *Journal of Machine Learning Research*, 21(117):1–51.

Lin, J. and Michailidis, G. (2024). A multi-task encoder-dual-decoder framework for mixed frequency data prediction. *International Journal of Forecasting*, 40(3):942–957.

Litterman, R. B. (1979). Techniques of forecasting using vector autoregressions. Working Paper 115, Federal Reserve Bank of Minneapolis.

Litterman, R. B. (1986). Forecasting with Bayesian vector autoregressions—five years of experience. *Journal of Business & Economic Statistics*, 4(1):25–38.

Liu, J., Musialski, P., Wonka, P., and Ye, J. (2012). Tensor completion for estimating missing values in visual data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):208–220.

Liu, P., Mumtaz, H., and Theophilopoulou, A. (2011). International transmission of shocks: A time-varying factor-augmented VAR approach to the open economy. Technical report, Bank of England.

Liu, X. and Chen, R. (2020). Threshold factor models for high-dimensional time series. *Journal of Econometrics*, 216(1):53–70.

Loaiza-Maya, R. and Nibbering, D. (2022). Efficient variational approximations for state space models. *arXiv preprint arXiv:2210.11010*.

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pages 4114–4124. PMLR.

Lozano, A., Abe, N., Liu, Y., and Rosset, S. (2009). Grouped graphical granger modeling for gene expression regulatory network discovery. *Bioinformatics*, 25:i110–8.

Lu, Y. and Zhu, D. (2023). Modelling mortality: A Bayesian factor-augmented VAR (FAVAR) approach. *ASTIN Bulletin: The Journal of the IAA*, 53(1):29–61.

Lütkepohl, H. (2013). *Introduction to multiple time series analysis*. Springer Science & Business Media.

Maity, A. K., Basu, S., and Ghosh, S. (2021). Bayesian criterion-based variable selection. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 70(4):835–857.

Majumdar, A. and Tripathi, A. (2017). Asymmetric stacked autoencoder. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 911–918. IEEE.

Maung, K. (2021). Estimating high-dimensional Markov-switching VARs. *arXiv preprint arXiv:2107.12552*.

McCracken, M. and Ng, S. (2020). FRED-QD: A quarterly database for macroeconomic research. Working paper, National Bureau of Economic Research.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5:115–133.

Merkle, E. C., Furr, D., and Rabe-Hesketh, S. (2019). Bayesian comparison of latent variable models: Conditional versus marginal likelihoods. *Psychometrika*, 84(3):802–829.

Mitchell, T. J. and Beauchamp, J. J. (1988). Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032.

Moran, G. E., Sridhar, D., Wang, Y., and Blei, D. M. (2021). Identifiable deep generative models via sparse decoding. *arXiv preprint arXiv:2110.10804*.

Mørup, M. and Hansen, L. K. (2009). Automatic relevance determination for multi-way models. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 23(7-8):352–363.

Nakajima, J. (2011). Time-Varying Parameter VAR Model with Stochastic Volatility: An Overview of Methodology and Empirical Applications. IMES Discussion Paper Series 11-E-09, Institute for Monetary and Economic Studies, Bank of Japan.

Neuhaus, J. O. and Wrigley, C. (1954). The quartimax method: An analytic approach to orthogonal simple structure 1. *British Journal of Statistical Psychology*, 7(2):81–91.

Ng, A. (2011). Sparse autoencoder. http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders/. UFLDL Tutorial, Stanford University.

Nicholson, W. B., Wilms, I., Bien, J., and Matteson, D. S. (2020). High dimensional forecasting via interpretable vector autoregression. *Journal of Machine Learning Research*, 21(166):1–52.

O'Hara, R. B. and Sillanpää, M. J. (2009). A review of Bayesian variable selection methods: what, how and which. *Bayesian Analysis*, 4(1):85 – 117.

Otter, D. W., Medina, J. R., and Kalita, J. K. (2020). A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):604–624.

Park, T. and Casella, G. (2008). The Bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686.

Peña, D. and Tsay, R. S. (2021). *Statistical learning for big dependent data*. John Wiley & Sons.

Pinkus, A. (1999). Approximation theory of the MLP model in neural networks. *Acta Numerica*, 8:143–195.

Pitt, M. K. and Shephard, N. (1999). Time varying covariances: a factor stochastic volatility approach. *Bayesian Statistics*, 6:547–570.

Polson, N. G. and Scott, J. G. (2010). Shrink globally, act locally: Sparse Bayesian regularization and prediction. *Bayesian Statistics*, 9(501-538):105.

Polyak, B. T. (1964). Some methods of speeding up the convergence of iteration methods. *Ussr Computational Mathematics and Mathematical Physics*, 4(5):1–17.

Poworoznek, E., Ferrari, F., and Dunson, D. (2021). Efficiently resolving rotational ambiguity in Bayesian matrix sampling with matching. *arXiv preprint arXiv:2107.13783*.

Primiceri, G. E. (2005). Time varying structural vector autoregressions and monetary policy. *The Review of Economic Studies*, 72(3):821–852.

Prüser, J. (2021). The horseshoe prior for time-varying parameter VARs and monetary policy. *Journal of Economic Dynamics and Control*, 129:104188.

Prüser, J. and Schlösser, A. (2020). The effects of economic policy uncertainty on European economies: evidence from a TVP-FAVAR. *Empirical Economics*, 58(6):2889–2910.

Raftery, A. E., Kárnỳ, M., and Ettler, P. (2010). Online prediction under model uncertainty via dynamic model averaging: Application to a cold rolling mill. *Technometrics*, 52(1):52–66.

Raftery, A. E., Madigan, D., and Hoeting, J. A. (1997). Bayesian model averaging for linear regression models. *Journal of the American Statistical Association*, 92(437):179–191.

Rai, P., Wang, Y., Guo, S., Chen, G., Dunson, D., and Carin, L. (2014). Scalable Bayesian low-rank decomposition of incomplete multiway tensors. In *International Conference on Machine Learning*, pages 1800–1808. PMLR.

Reinsel, G. (1983). Some results on multivariate autoregressive index models. *Biometrika*, 70(1):145–156.

Ren, Y., Guo, Q., Zhu, H., and Ying, W. (2020). The effects of economic policy uncertainty on China's economy: evidence from time-varying parameter FAVAR. *Applied Economics*, 52(29):3167–3185.

Rifai, S., Mesnil, G., Vincent, P., Muller, X., Bengio, Y., Dauphin, Y., and Glorot, X. (2011a). Higher order contractive auto-encoder. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2011, Athens, Greece, September 5-9, 2011, Proceedings, Part II 22*, pages 645–660. Springer.

Rifai, S., Vincent, P., Muller, X., Glorot, X., and Bengio, Y. (2011b). Contractive auto-encoders: explicit invariance during feature extraction. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 833–840. Omnipress.

Robert, C. P. (2014). On the Jeffreys-Lindley paradox. *Philosophy of Science*, 81(2):216–232.

Ročková, V. and George, E. I. (2016). Fast Bayesian factor analysis via automatic rotations to sparsity. *Journal of the American Statistical Association*, 111(516):1608–1622.

Ročková, V. and George, E. I. (2018). The spike-and-slab lasso. *Journal of the American Statistical Association*, 113(521):431–444.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326.

Rowling, J. (2012). *Harry Potter and the Sorcerer's Stone*. Pottermore Limited.

Rubio-Ramirez, J. F., Waggoner, D. F., and Zha, T. (2005). Markov-switching structural vector autoregressions: theory and application. FRB Atlanta Working Paper 2005-27, Federal Reserve Bank of Atlanta.

Rumelhart, D., Hinton, G., and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.

Satopaa, V., Albrecht, J., Irwin, D., and Raghavan, B. (2011). Finding a“ kneedle” in a haystack: Detecting knee points in system behavior. In *2011 31st International Conference on Distributed Computing Systems Workshops*, pages 166–171. IEEE.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, pages 461–464.

Seghier, M. L. (2013). The angular gyrus: multiple functions and multiple subdivisions. *The Neuroscientist*, 19(1):43–61.

Seth, A. K., Barrett, A. B., and Barnett, L. (2015). Granger causality analysis in neuroscience and neuroimaging. *Journal of Neuroscience*, 35(8):3293–3297.

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica*, pages 639–650.

Shi, Q., Lu, H., and Cheung, Y.-m. (2017). Tensor rank estimation and completion via CP-based nuclear norm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 949–958.

Shojaie, A. and Michailidis, G. (2010). Discovering graphical Granger causality using the truncating lasso penalty. *Bioinformatics*, 26(18):i517–i523.

Sidiropoulos, N. D. and Bro, R. (2000). On the uniqueness of multilinear decomposition of N-way arrays. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 14(3):229–239.

Sims, C. A. (1980). Macroeconomics and reality. *Econometrica: Journal of the Econometric Society*, pages 1–48.

Sims, C. A., Waggoner, D. F., and Zha, T. (2008). Methods for inference in large multiple-equation Markov-switching models. *Journal of Econometrics*, 146(2):255–274.

Sims, C. A. and Zha, T. (2006). Were there regime switches in US monetary policy? *American Economic Review*, 96(1):54–81.

Song, C., Liu, F., Huang, Y., Wang, L., and Tan, T. (2013). Auto-encoder based data clustering. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 18th Iberoamerican Congress, CIARP 2013, Havana, Cuba, November 20-23, 2013, Proceedings, Part I 18*, pages 117–124. Springer.

Song, S. and Bickel, P. J. (2011). Large vector auto regressions. *arXiv preprint arXiv:1106.3915*.

Soori, M., Arezoo, B., and Dastres, R. (2023). Artificial intelligence, machine learning and deep learning in advanced robotics, a review. *Cognitive Robotics*, 3:54–70.

Spencer, D., Guhaniyogi, R., and Prado, R. (2022). Parsimonious Bayesian sparse tensor regression using the Tucker decomposition. *arXiv preprint arXiv:2203.04733*.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):583–639.

Stenvall, D. (2024). The heterogeneous effects of financial shocks on labor markets: National, sectoral, and regional dynamics in Sweden. *Available at SSRN 4983750*.

Stock, J. and Watson, M. (2010). *Dynamic Factor Models*. Oxford University Press, Oxford.

Stock, J. H. and Watson, M. (2009). Forecasting in dynamic factor models subject to structural instability. *The Methodology and Practice of Econometrics. A Festschrift in Honour of David F. Hendry*, 173:205.

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.

Stock, J. H. and Watson, M. W. (2016). Dynamic factor models, factor-augmented vector autoregressions, and structural vector autoregressions in macroeconomics. In *Handbook of Macroeconomics*, volume 2, pages 415–525. Elsevier.

Strumbelj, E. and Kononenko, I. (2010). An efficient explanation of individual classifications using game theory. *The Journal of Machine Learning Research*, 11:1–18.

Su, L. and Wang, X. (2017). On time-varying factor models: Estimation and testing. *Journal of Econometrics*, 198(1):84–101.

Sun, W. W. and Li, L. (2017). STORE: sparse tensor response regression and neuroimaging analysis. *The Journal of Machine Learning Research*, 18(1):4908–4944.

Takayama, H., Zhao, Q., Hontani, H., and Yokota, T. (2022). Bayesian tensor completion and decomposition with automatic CP rank determination using MGP shrinkage prior. *SN Computer Science*, 3(3):225.

Tank, A. et al. (2021). Neural Granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4267–4279.

Taveeapiradeecharoen, P., Chamnongthai, K., and Aunsri, N. (2019). Bayesian compressed vector autoregression for financial time-series analysis and forecasting. *IEEE Access*, 7:16777–16786.

Teräsvirta, T., Tjøstheim, D., and Granger, C. W. (2010). *Modelling nonlinear economic time series*. Oxford University Press.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288.

Tieleman, T. and Hinton, G. (2012). Lecture 6.5—RMSprop: Divide the gradient by a running average of its recent magnitude. Coursera: Neural Networks for Machine Learning. Lecture notes.

Timmerman, M. E. and Kiers, H. A. (2000). Three-mode principal components analysis: Choosing the numbers of components and sensitivity to local optima. *British Journal of Mathematical and Statistical Psychology*, 53(1):1–16.

Tonolini, F., Jensen, B. S., and Murray-Smith, R. (2020). Variational sparse coding. In *Uncertainty in Artificial Intelligence*, pages 690–700. PMLR.

Tsay, R. S. (1998). Testing and modeling multivariate threshold models. *Journal of the American Statistical Association*, 93(443):1188–1202.

Tsay, R. S. (2013). *Multivariate time series analysis: with R and financial applications*. John Wiley & Sons.

Tucker, L. R. (1963). Implications of factor analysis of three-way matrices for measurement of change. *Problems in Measuring Change*, 15(122-137):3.

Uhlenbeck, G. E. and Ornstein, L. S. (1930). On the theory of the Brownian motion. *Physical review*, 36(5):823.

Uhlig, H. (2005). What are the effects of monetary policy on output? results from an agnostic identification procedure. *Journal of Monetary Economics*, 52(2):381–419.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Vats, D. and Knudson, C. (2021). Revisiting the Gelman–Rubin diagnostic. *Statistical Science*, 36(4):518–529.

Velu, R. P., Reinsel, G. C., and Wichern, D. W. (1986). Reduced rank models for multiple time series. *Biometrika*, 73(1):105–118.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1096–1103.

Voulodimos, A., Doulamis, N., Doulamis, A., and Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018(1):7068349.

Wang, D., Zheng, Y., and Li, G. (2024a). High-dimensional low-rank tensor autoregressive time series modeling. *Journal of Econometrics*, 238(1):105544.

Wang, D., Zheng, Y., Lian, H., and Li, G. (2022a). High-dimensional vector autoregressive time series modeling via tensor decomposition. *Journal of the American Statistical Association*, 117(539):1338–1356.

Wang, X., Chen, H., Tang, S., Wu, Z., and Zhu, W. (2022b). Disentangled representation learning. *arXiv preprint arXiv:2211.11695*.

Wang, Y., Blei, D., and Cunningham, J. P. (2021). Posterior collapse and latent variable non-identifiability. *Advances in Neural Information Processing Systems*, 34:5443–5455.

Wang, Y., Wu, H., Dong, J., Liu, Y., Long, M., and Wang, J. (2024b). Deep time series models: A comprehensive survey and benchmark. *arXiv preprint arXiv:2407.13278*.

Wang, Y., Yao, H., and Zhao, S. (2016). Auto-encoder based dimensionality reduction. *Neurocomputing*, 184:232–242.

Wehbe, L., Murphy, B., Talukdar, P., Fyshe, A., Ramdas, A., and Mitchell, T. (2014). Simultaneously uncovering the patterns of brain regions involved in different story reading subprocesses. *PloS One*, 9(11):e112575.

Weiner, K. S. and Zilles, K. (2016). The anatomical and functional specialization of the fusiform gyrus. *Neuropsychologia*, 83:48–62.

West, M. and Harrison, J. (1997). *Bayesian Forecasting and Dynamic Models (Springer Series in Statistics)*. Springer-Verlag.

Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

Wold, H. (1938). *A study in the analysis of stationary time series*. PhD thesis, Almqvist & Wiksell.

Xiong, X. and Cribben, I. (2023). Beyond linear dynamic functional connectivity: a vine copula change point model. *Journal of Computational and Graphical Statistics*, 32(3):853–872.

Yu, R., Zheng, S., Anandkumar, A., and Yue, Y. (2019). Long-term forecasting using higher order tensor RNNs. *arXiv preprint arXiv:1711.00073*.

Yu, Y. and Meng, X.-L. (2011). To center or not to center: That is not the question—an Ancillarity–Sufficiency Interweaving Strategy (ASIS) for boosting MCMC efficiency. *Journal of Computational and Graphical Statistics*, 20(3):531–570.

Zhang, A., Lipton, Z. C., Li, M., and Smola, A. J. (2023). *Dive into deep learning*. Cambridge University Press.

Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, 50:159–175.

Zhang, K., Safikhani, A., Tank, A., and Shojaie, A. (2022a). Penalized estimation of threshold auto-regressive models with many components and thresholds. *Electronic Journal of Statistics*, 16(1):1891.

Zhang, W., Cribben, I., Guindani, M., et al. (2021). Bayesian time-varying tensor vector autoregressive models for dynamic effective connectivity. *arXiv preprint arXiv:2106.14083*.

Zhang, Y. D., Naughton, B. P., Bondell, H. D., and Reich, B. J. (2022b). Bayesian regression using a prior on the model fit: The r2-d2 shrinkage prior. *Journal of the American Statistical Association*, 117(538):862–874.

Zhao, Q., Hautamaki, V., and Fränti, P. (2008). Knee point detection in BIC for detecting the number of clusters. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 664–673. Springer.

Zhao, Q., Zhang, L., and Cichocki, A. (2015a). Bayesian CP factorization of incomplete tensors with automatic rank determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1751–1763.

Zhao, Q., Zhou, G., Zhang, L., Cichocki, A., and Amari, S.-I. (2015b). Bayesian robust tensor factorization for incomplete multiway data. *IEEE Transactions on Neural Networks and Learning Systems*, 27(4):736–748.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. (2023). A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2).

Zhou, H., Li, L., and Zhu, H. (2013). Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association*, 108(502):540–552.