University of Strathclyde Glasgow

Essays on variational Bayes in Econometrics

PhD Thesis

Paponpat Taveeapiradeecharoen

Research Group

Department of Economics

University of Strathclyde, Glasgow

October 3, 2023

Declaration

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Signed:
Date:

"The greatest glory in living lies not in never falling, but in rising every time we fall." - Nelson Mandela

To my mum, Tipvipha

Abstract

The first essay (Chapter 1) presents a Variational Bayes (Vb) algorithm for Vector Autoregression (reduced-form VAR). The algorithm is derived based on the evidence lower bound, which is demonstrated to be tight, ensuring efficient convergence. The optimization is carried through the Coordinate descent optimization. To validate the proposed method, its accuracy and computational costs are compared with existing Vb approaches that approximate VAR using a one equation at a time technique (Cholesky-transformed VAR), and a more computationally intensive Markov Chain Monte Carlo (MCMC) method using Gibbs sampling. In applications using both US macroeconomic data and artificial data, our Vb for VAR outperforms Vb in Cholesky-transformed VAR in terms of VAR covariance accuracy. Furthermore, compared to the MCMC method, our proposed Vb algorithm for reduced-form VAR achieves comparable accuracy while significantly reducing computation time.

The second essay (Chapter 2) takes the Variational Bayes (Vb) approach to the next level by extending it to the challenging domain of Mixed Frequency Vector Autoregression (MF-VAR) models. These models tackle the complexities of dealing with multiple frequency data in a single estimation, including the issue of missing lower frequency observations in a higher frequency system. To overcome these challenges, we introduce a robust and innovative Vb method known as the Variational Bayes-Expectation Maximization algorithm (Vb-EM). Our Vb-EM algorithm offers several key contributions to approximate Bayesian inference in the MF-VAR model. We derive an evidence lower bound to the log marginal likelihood, accounting for missing observations, and optimize it with respect to the variational parameters. In doing so, we surpass existing Vb methods in the literature by achieving a tighter evidence lower bound, ensuring optimal convergence. To further validate our approach, we compare it to the more computationally demanding Markov Chain Monte Carlo (MCMC) method using Gibbs sampling. Through extensive empirical evaluations and out-of-sample forecasts of eleven US macroeconomic series, we demonstrate that our Vb-EM algorithm performs on par with MCMC in terms of point forecasts. Furthermore, when assessing predictive density, we find no significant empirical evidence to distinguish between the two methods. Notably, our Vb-EM algorithm offers the distinct advantage of significantly lower computational costs, making it an appealing choice for researchers and practitioners alike.

The third essay (Chapter 3) begins by emphasizing that the spike of volatilities of macroeconomic variables during the surge of Covid-19 pandemic, which led to poor performance of the workhorse Bayesian VAR with stochastic volatility in terms of forecasting. This has attracted considerable attention from economists towards alternative models, including non-parametric models such as Gaussian process VAR. The approach to estimate VAR one equation at a time, namely Cholesky-transformed VARs, enables the application of more advanced regression models in VAR. In this chapter I explore several advanced Gaussian process VARs, including GP-VAR, GP-DNN-VAR (which incorporates a deep neural network as the mean function in the GP prior), and Heteroscedastic-GP-VAR (HGP-VAR) where the likelihood variance is assumed to be time-varying and parameterized by another latent-GP function. In this chapter the variational inference is utilized to be the approximating method for HGP-VAR. The forecasting results suggest that during non-pandemic periods, HGP-VAR and GP-VAR perform similarly to BVAR-SV. However, during the Covid-19 pandemic, the advantage of having time-variant likelihood variance in HGP-VAR becomes more pronounced for predicting macroeconomic variables in a highly turbulent period.

Contents

Declaration i

List of Figures viii

List of Tables xii

1 Variational Bayes for Bayesian Vector Autoregression. 2

1.1 Introduction 2

1.2 Variational Bayes 4

1.3 Why approximate reduced-form VAR? 5

1.4 Normal-Wishart conditional on Horseshoe shrinkage prior VAR 7

1.5 Optimal variational parameters 8

1.6 A simulation study 10

1.6.1 Investigate further on VAR covariance. 12

1.6.2 Computational costs 15

1.7 Convergence and in-sample prediction 19

1.8 Forecast Application 21

1.8.1 Investigating the accuracy of tails forecasts 23

1.9 Impulse response functions comparisons 26

1.10 Conclusion 27

1.11 Appendix A: Optimal variational parameters 28

1.12 Appendix B: Evidence Lower Bound 31

1.13 Appendix C: Data 33

1.14 Appendix D: Additional Results 34

2 Variational Bayes-Expectation Maximisation for Mixed Frequency VAR 35

2.1 Introduction 35

2.2 EM-algorithm 39

2.3 Variational Bayes with missing observations. 42

2.3.1 Variational Bayes-Expectation Maximization 44

2.3.2 Variational predictive distribution 47

2.4 Mixed Frequency VAR 47

2.5 Optimal variational parameters in Vb:M-step 50

2.6 Empirical applications 53

2.6.1 Predictive density evaluation scores 53

2.6.2 Pseudo out-of-sample real-time forecasts 56

2.7 Summary of different algorithms to estimate MF-VAR. 59

2.8 Conclusion 62

2.9 Appendix A: Data 64

2.10 Appendix B: Derivation of evidence lower bound to the log marginal likelihood 64

2.11 Appendix C: Kalman filter and smoother recursion, Variance and Covariance matrices of smoothed estimators 70

2.11.1 Kalman filter recursion 70

2.11.2 Kalman Smoother recursion 71

2.11.3 Variance and covariance of smoothed state variables 71

2.12 Appendix D: Lemmas 72

3 Forecasting macroeconomic variables with Gaussian process VAR. 73

3.1 Introduction 73

3.2 Gaussian process regression 75

3.2.1 Kernels 77

3.2.2 Compositional kernels. 79

3.2.3 Marginal likelihood of Gaussian process regression 81

3.3 VAR models 82

3.3.1 Bayesian VAR with SV 82

3.3.2 Gaussian process-VARs 83
3.4 Estimation and predictive density of GP-VARs models 91
3.4.1 Learning each equation in GP-VAR 92
3.4.2 Learning each equation in GP-DNN-VAR. 93
3.4.3 Learning each equation in HGP-VAR. 96
3.4.4 Model summarizations 103
3.5 Full Sample Estimation 105
3.5.1 Capture macroeconomic correlations with Radial Basis Function kernel. 105
3.5.2 Conditional mean 110
3.5.3 Volatilities 115
3.6 Forecasting Results 119
3.6.1 During 2002Q2 through 2019Q1 119
3.6.2 During pandemic 2020Q1 through 2022Q1 124
3.7 Conclusion 127
3.8 Appendix 130
3.8.1 Data 130
3.8.2 Additional Results 130

4 Conclusions **141**

List of Figures

1.1 Computational complexity of VAR models scales with the number of VAR lags (left) and the number of variables (right). 7

1.2 Box plots of Euclidean-norm between true VAR coefficients and approximated ones from four different algorithms, i.e.$\|\widehat{B} - B\|_2^2$ (left-panel), and $\|\widehat{\Sigma} - \Sigma\|_2^2$ (right-panel) 11

1.3 Box plots of Euclidean-norm between actual observations and in-sample VAR predictions, $\|\widehat{Y} - Y\|_2^2$. 12

1.4 Reproducible artificial data simulated with `simulate` function from matlab according to VAR parameters in eq. (1.10). 15

1.5 Heatmap displaying the value of actual and estimated covariance of a VAR model 15

1.6 Heatmap displaying the value of estimated covariance of Minnesota prior VAR model (system-wide estimation) 16

1.7 Matlab code to simulate artificial data. 16

1.8 Euclidean norm distance between actual VAR covariance and posterior mean of different models and algorithms 17

1.9 Computational costs of three different algorithms, scale with number of equations in VAR 18

1.10 Evidence lower bound and vectorized posterior mean of VAR coefficient from Vb algorithm for small-VAR 20

1.11 Evidence lower bound and vectorized posterior mean of VAR coefficient from Vb algorithm for medium-VAR 20

1.12 Evidence lower bound and vectorized posterior mean of VAR coefficient from Vb algorithm for large-VAR 20

1.13 Impulse response (untransformed or cumulated) of selected variables to FEDFUNDS shocks. Conjugate normal-Wishart VAR (top panel), Minnesota VAR using MCMC CCCM algorithm (middle panel) and Horseshoe VAR using Vb algorithm (bottom). 26

1.14 Kernel estimate of posterior distribution from different model. 34

2.1 Evidence lower bound to log marginal likelihood, of Vb-EM algorithm (red-dashed-dot), and literature algorithm (blue-solid-dot), see eq. (2.35) for full derivations 55

2.2 Threshold decomposition mean of CRPS for quarterly one-step-ahead forecast density for entire evaluation period 57

2.3 Threshold decomposition mean of CRPS for monthly variable forecast density 58

3.1 Gaussian process regression (GPR) with US gross domestic product, regressed with its own single lag. 79

3.2 Samples generated from $\mathcal{GP}$ prior distribution using different kernels. . . 80

3.3 Gaussian process regression (GPR) of US GDP observation with five different settings. 85

3.4 Visualization of feed-forward Deep neural networks, where three hidden layers are configured, each has 4 nodes. 88

3.5 Visualization of linear mean function for $\mathcal{GP}$ prior (OLS estimates). . . 88

3.6 Data sets after transformation according to McCracken & Ng (2020). . . 105

3.7 Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US GDP. 107

3.8 Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US inflation rate. 108

3.9 Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged Federal fund rate. 108

3.10 Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US industrial production. 109

3.11 Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US government spending expenditures. 109

3.12 US GDP conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. . 111

3.13 US unemployment rate conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. 112

3.14 US inflation conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. 113

3.15 US industrial production conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. 114

3.16 Volatility of eight variables BVAR-SV (red-solid) and HGP-VAR (black-solid). 117

3.17 VAR-Volatilities of eight variables BVAR-SV (red-solid) and HGP-VAR (black-solid). 117

3.18 Conditional volatilities of HGP-VAR model with different number of lag of explanatory variables. Dash-dotted-line (one lag), dash-line (six lags), and solid-line (nine lags). Grey-vertical-fills (US recession according to NBER). 118

3.19 RMSE of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 1, 2$ 132

3.20 RMSE of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 3, 4$ 133

3.21 CRPS of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 1, 2$ 134

3.22 CRPS of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 3, 4$ 135

3.23 RMSE of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 1, 2$ 136

3.24 RMSE of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 3, 4$ 137

3.25 CRPS of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 1, 2$ 138

3.26 CRPS of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 3, 4$ 139

3.27 Forecast density of one-step-ahead forecast of 2020Q2. 140

3.28 Forecast density (one-step-ahead) of 2020Q3. 140

3.29 Forecast density (one-step-ahead) of 2020Q4. 140

3.30 Forecast density (one-step-ahead) of 2021Q1. 140

List of Tables

1.1 The computational complexity for a draw of the conditional posterior of VAR coefficients in each MCMC iteration. 6

1.2 Computational costs, Euclidean norm distance between observed data and predicted values from Vb-VAR, OLS, and MCMC (CCCM algorithm, see Carriero et al. (2022)). 21

1.3 RMSE and CRPS of Vb system-wide VAR relative to benchmark model (Minnesota prior VAR using MCMC). Color in each block indicate statistical siginificance of the Diebold-Mariano test (two-side) for equal predictive performance at 1 (light green), 5 (yellow), 10 (orange) percent level. 23

1.4 Emphasize of tail, uniform, centre, right, and left of CPIAUCSL predictive distribution obtained from weighted-CRPS, averaged over forecasting evaluation periods of the Vb system-wide VAR compared to the benchmark model (Minnesota prior VAR using MCMC). 24

1.5 Emphasize of tail, uniform, centre, right, and left of UNRATE predictive distribution obtained from weighted-CRPS, averaged over forecasting evaluation periods of the Vb system-wide VAR compared to the benchmark model (Minnesota prior VAR using MCMC). 25

1.6 Emphasize of tail, uniform, centre, right, and left of INDPRO predictive distribution obtained from weighted-CRPS, averaged over forecasting evaluation periods of the Vb system-wide VAR compared to the benchmark model (Minnesota prior VAR using MCMC). 25

1.7 Data used for 3/20/40-variables VAR, and impulse response analysis . . 33

2.1 Dimension of state estimators, and Y, X in eq. (2.18). 49

2.2 Quantile weights. 55

2.3 RMSE of nowcasting each month within quarter. 57

2.4 CRPS of nowcasting each month within quarter. 57

2.5 RMSE, and CRPS over forecasting periods. The last three columns are RMSE and CRPS of Vb-EM relative to MCMC and EM-algorithm. . . 58

2.6 Quantile weighted mean of CRPS of Vb-EM algorithm relative to MCMC (one-step-ahead) predictive forecast of eleven macroeconomic series. . . 58

2.7 Variational Bayes-Expectation Maximization algorithm for MF-VAR. . . 60

2.8 Literature variational Bayesian (Vb) algorithm for MF-VAR. 61

2.9 EM-algorithm for MF-VAR. 62

2.10 Macroeconomic series. 64

3.1 Kernel selection for each equation in GP-VARs models. 91

3.2 Kernels hyper-parameters. 91

3.3 Algorithm to simulate predictive density for GP-VAR. 94

3.4 Algorithm for Gaussian process regression with deep neural network mean function. 95

3.5 Algorithm to simulate predictive density for HGP-VAR. 103

3.6 Acronyms of models, model descriptions, approximation method and parameters. 104

3.7 RMSE of GP-VAR, HGP-VAR, GP-DNN-VAR models relative to BVAR-SV for `h=1,...,h=12`, one to twelve-steps-ahead forecasts during non-pandemic periods. 122

3.8 Quantile Scores during non-pandemic periods (selected variables). . . . 123

3.9 RMSE of GP-VAR, HGP-VAR, GP-DNN-VAR models relative to BVAR-SV for `h=1,...,h=4` during pandemic periods (2020Q1 through 2022Q1). 127

3.10 Quantile scores of GP-VAR, HGP-VAR, GP-DNN-VAR models relative to BVAR-SV for `h=1,...,h=4` during pandemic periods (2020Q1 through 2022Q1). 128
3.11 US macroeconomic variables 130

Acknowledgements

I would like to express my deepest gratitude to a number of individuals and institutions who have been instrumental in my academic journey.

I am deeply grateful to my supervisor, Gary Koop, for providing me with the opportunity to pursue research even though I had little knowledge of Bayesian econometrics, and for his constant support, guidance, and motivation throughout my research journey. I would also like to extend my thanks to Stuart McIntyre, who also served as my supervisor, for his unwavering support and prompt advice whenever I needed it. Their guidance and feedback have been priceless in shaping my research and assisting me in overcoming the numerous obstacles that came my way.

I am also grateful to my colleagues at Wonlope Khumpradith, Aubrey Poon, and my fellow Ph.D. colleagues at the department conferences who have provided a stimulating and supportive academic environment throughout my studies. Their friendship, encouragement, and intellectual curiosity have been a constant source of inspiration and motivation.

I would like to acknowledge the financial support provided by Office of Education Affairs, The Royal Thai Embassy, which has enabled me to pursue my research and make important contributions to my field.

Last but not least, I would like to thank my family and friends for their unwavering love and support throughout my academic journey. Their encouragement, understanding, and patience have been a constant source of strength and motivation, and I am forever grateful for their presence in my life.

Acknowledgements

Chapter 1

Variational Bayes for Bayesian Vector Autoregression.

1.1 Introduction

In recent years, there has been a growing body of literature on the use of large Bayesian Vector Autoregressive (VAR) models. These models have been widely used in macroeconomics and finance, with many papers published on the subject, for instance Koop & Korobilis (2013), Carriero et al. (2009, 2012). However, one of the main challenges with VAR models is their computational complexity, which increases with the number of variables and lags. To address this issue, some researchers have proposed methods to estimate VAR models one equation at a time. Carriero et al. (2019, 2021) have developed Gibbs sampling algorithms to estimate VAR models in this fashion, while other researchers have used approximating methods such as Variational Bayes (Vb) to estimate one equation at a time in VAR models. However, these methods have limitations and researchers have started to question the necessity of estimating VAR models in this way. For example Bognanni (2022) comment on the per equation algorithm from Carriero et al. (2019) (labelled as CCM algorithm hereafter), proving that such algorithm is theoretically incorrect but empirically and practically implementable. The results produced by such algorithm is quite reliable which we will show in one of our empirical results on artificial data. Then Carriero et al. (2022) response to the comment from Bognanni (2022) by providing more theoretical proofs and slightly modify the position of lower-triangular A.

The Cholesky decomposition method has been criticized by numerous scholars for its tendency to produce order-dependent results, meaning that posterior inference can be affected by the ordering of the variables in a VAR model. In a recent study, Arias et al. (2022) highlight the significance of the ordering issue in Cholesky-transformed VARs through both theoretical and empirical analysis. Their findings reveal that while point forecasts are generally robust to variable ordering, predictive standard deviations can be significantly influenced. Chan et al. (2021) support this notion, demonstrating that the order invariance problem intensifies with the dimensionality of the VAR, exacerbating the issue in precisely the situation where the Cholesky transformation is most needed.

In this chapter, a new Vb algorithm for VAR models, called Vb-VAR, is proposed. The algorithm approximates the parameters of the VAR model by maximizing the Evidence Lower Bound (ELBO) with respect to an approximate distribution. The proposed Vb-VAR algorithm is shown to converge by demonstrating an increase in the ELBO as the number of optimization iterations grows. It is worth nothing that our Vb algorithm is approximating system-wide VAR, meaning that we do not rely on any per equation algorithm. In other words our Vb approximate Covariance of VAR entirely. Not being recovered from lower-triangular A matrix as in Carriero et al. (2019, 2022) or Gefang et al. (2023, 2020).

To validate our proposed algorithm, we compare it to other methods such as CCM, Carriero et al. (2019), and CCCM Carriero et al. (2022), using simulated artificial data. We also investigate how each algorithm performs when the size of VAR covariance is larger and compare their computational costs. We then evaluate the accuracy of real-world macroeconomic monthly data out-of-sample forecasts using both point and density forecasts. We compare our results with other methods such as Ordinary Least Squares, MCMC, and Vb approximating one equation at a time. We use root mean square error (RMSE)and cumulative ranked probabilistic scores (CRPS) to evaluate both point and density forecasts, and compare tail forecasts of our Vb-VAR algorithm to MCMC using weighted CRPS. The results show that our proposed algorithm is accurate relative to MCMC and has better computational efficiency. We conclude that the proposed algorithm is a promising approach for large-scale macroeconomic forecasting.

Finally it is noteworthy that the proposed Vb-VAR algorithm will be extended for MF-VAR in the next chapter. For now the roadmap of this chapter is categorized as followed: A brief introduction to Vb in section 1.2. Next VAR model and its prior in section 1.4. Then section 1.5 presents optimal variational parameters for Vb-VAR algorithm. A simulation study of real macroeconomic data is investigated, which is presented in section 1.6. section 1.7. Furthermore we also provide the impulse response analysis in section 1.9. Finally the conclusion is drawn in section 1.10.

1.2 Variational Bayes

In a Bayesian model, the goal of variational Bayes computational methods is to approximate the posterior distribution of the model's parameters (denoted as $p(\theta|y)$ with a simpler, more manageable distribution from a specified family of distributions (denoted as $q(\theta)$). This approximation is achieved by minimizing the Kullback-Leibler (KL) divergence between the posterior and the approximating distribution. These methods have been widely studied in the literature, with references Ormerod & Wand (2010), Blei et al. (2017) being notable examples and comprehensive reviews. The KL divergence can be simply derived as followed:

$$\log p(y) = \int \log\left(\frac{p(\theta)p(y|\theta)}{q(\theta)}\right) q(\theta)d\theta + \int \log\left(\frac{q(\theta)}{p(\theta|y)}\right) q(\theta)d\theta. \tag{1.1}$$

Or equivalently:

$$\log p(\mathbf{y}) \geq \int q(\theta) \log \frac{p(y,\theta)}{q(\theta)} d\theta.$$

The exact KL-divergence is almost always intractable, but with Jensen's inequality, it is a fact that maximizing lower bound to the log marginal likelihood is equivalent to minimizing KL-divergence. From eq. (1.1) above the evidence lower bound can be written as:

$$\mathcal{L} = \int q(\theta) \log \frac{p(y,\theta)}{q(\theta)} d\theta.$$

Suppose that the model's parameters θ can be divided into J sub-vectors $\theta_1, ..., \theta_J$. The approximating distribution $q(\theta)$ for the posterior distribution $p(\theta|y)$ is assumed to factorize as $q(\theta) = \prod_j^J q(\theta_j)$. To maximize the lower bound on the log marginal likelihood, each of these sub-vectors is updated in turn, holding the others fixed, using the following update rule:

$$\widehat{q}(\theta_j) \propto \exp\{\mathbb{E}_{-\theta_j}[\log p(y|\theta)p(\theta)]\}. \tag{1.2}$$

where $\mathbb{E}_{-\theta_j}[\cdot]$ denotes the expectation with respect to the other parameters apart from θ_j. Equation (1.2) is the update rule for a block-wise gradient descent algorithm (aka coordinate gradient descent) that is used to maximize the lower bound on the log marginal likelihood $p(y)$. In this algorithm, an initial choice for the sub-vectors of the approximating distribution $q(\theta)$ are made, and then each sub-vector is updated in turn, holding the others fixed at their current values, using the update rule given by eq. (1.2). This process is repeated until convergence is achieved. For Bayesian linear regression tutorial, readers are referred to Fox & Roberts (2012).

1.3 Why approximate reduced-form VAR?

It is no surprise that computational complexity of VAR model is massive due to the requirement of inverse covariance of residual in VAR during the each sampling iteration in MCMC (Gibbs sampling for instance). In this sub-section we briefly provide details on computational complexity of VAR scales poorly to both number of variables and lags. Suppose we are dealing with simple VAR with noisy observations. The system can be formulated as followed:

$$y_t = x_t\beta + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, \Sigma) \tag{1.3}$$

where subscription $t = 1, ..., T$ denotes the time index of observations, $y_t \in \mathbb{R}^N$. N is number of variables in VAR system. If p is number of total lag in VAR then x_t are predictors with $K = N \times p$ dimension. The parameters to be estimated are VAR coefficients β, and covariance Σ. Suppose further that we estimate this system using sampling-based method (Gibbs sampling) say the very popular among econometricians, Minnesota prior, the conditional posterior of VAR coefficient is, see Dieppe et al. (2016) and among others.

$$
\begin{aligned}
p(\beta|Y) &\propto \exp\left[-\frac{1}{2}\left\{(\beta_0 - \overline{\beta})'\overline{\Omega}^{-1}(\beta_0 - \overline{\beta})\right\}\right], \\
\overline{\Omega} &= \left[\Omega_0^{-1} + \Sigma^{-1} \otimes X'X\right]^{-1}, \\
\overline{\beta} &= \overline{\Omega}\left[\Omega_0^{-1}\beta_0 + (\Sigma^{-1} \otimes X')Y\right].
\end{aligned}
$$

where $Y = \{y_t\}_{t=1}^T$, and $X = \{x_t\}_{t=1}^T$. β_0, Ω_0 are prior for VAR coefficient and covariance. It is very obvious that the computational complexity of estimating whole VAR system scales not very well to number of variables N and number of lags p. Here I break down one by one of very high complexity in table 1.1. Since the number of observations in macroeconomic literature is quite few so I ignore the number of observations for simplicity.

terms	**computational complexity**
$(\Sigma^{-1} \otimes \mathbf{x}'\mathbf{x})$	$\mathcal{O}(N^2p^2 + N^3p)$
$(\Omega_0^{-1} + \Sigma^{-1} \otimes \mathbf{x}'\mathbf{x})^{-1}$	$\mathcal{O}(N^5p^3)$

Table 1.1: The computational complexity for a draw of the conditional posterior of VAR coefficients in each MCMC iteration.

The term Ω_0^{-1} is prior and can be computed preliminarily. In fact the second term in table 1.1 can be reduced to $\mathcal{O}(N^4p^2)$ if the linear solver is implemented. For example the backslash operator in Matlab program. The second term also dominates the first term therefore estimating whole VAR system has complexity scale with both number of variable and number of lags. To make it crystal, fig. 1.1 illustrates the complexity as the number of lag increase (left-panel), and number of variables in VAR grows (right-panel). Such limitation is a main obstacle for econometricians to perform a large VAR.

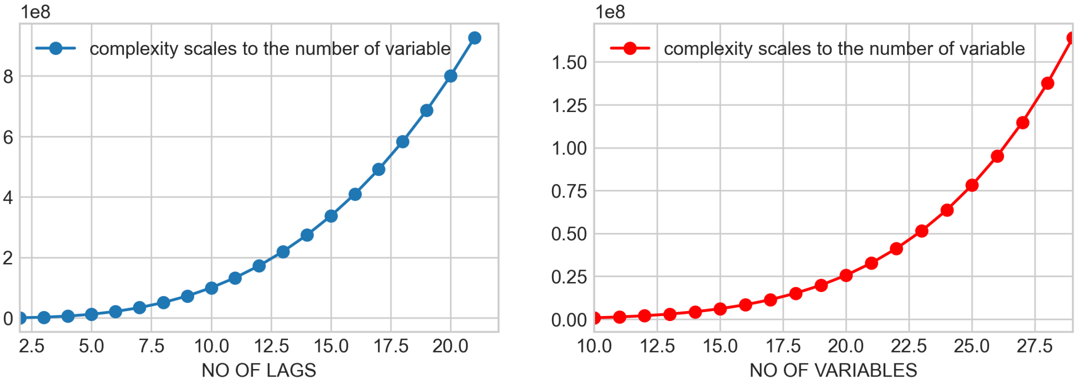


Figure 1.1: Computational complexity of VAR models scales with the number of VAR lags (left) and the number of variables (right).

The examples can be seen by multiple popular literature such as Bańbura et al. (2010), Carriero et al. (2015), Giannone et al. (2015), Koop & Korobilis (2013), where they include 20 US macroeconomic variables and has been reused in many papers afterward.

1.4 Normal-Wishart conditional on Horseshoe shrinkage prior VAR

Let N, T be numbers of equations in VAR and total observations, respectively. K be dimension of covariate at each time t for $t = 1, ..., T$. Then $Y_t = N$ dimensional vector of responsive variable at time t, $B = N \times K$, $X_t = K$-dimensional covariate vector at time t. We primarily focus on one of the most famous global-local shrinkage priors, which do not require predetermined hyperparameters. This prior automatically shrinks the VAR coefficients towards zero as soon as it is realized that the signal contains no predictive information and vice versa. Our conjugate normal-Wishart conditional on horseshoe shrinkage prior VAR can be expressed as:

$$\begin{aligned}
Y_t|B,\Sigma &\sim \mathcal{N}(BX_t,\Sigma^{-1}),\\
B|\Sigma,\lambda,\tau &\sim \mathcal{N}(0,(\Sigma\otimes\lambda\tau)^{-1}),\\
\lambda|\vartheta &\sim \mathcal{G}(1/2,\vartheta),\\
\tau|\xi &\sim \mathcal{G}(1/2,\xi),\\
\vartheta &\sim \mathcal{G}(1/2,1),\\
\xi &\sim \mathcal{G}(1/2,1),\\
\Sigma &\sim \mathcal{W}(S_0,\nu_0).
\end{aligned} \tag{1.4}$$

where $\sim \mathcal{N}(\cdot,\cdot')$ is multivariate normal distribution with $\cdot$ mean vector and $\cdot'$ covariance. $\sim \mathcal{G}(\cdot,\cdot')$ is gamma distribution with shape and rate $\cdot$, $\cdot'$, respectively. Finally the $\sim \mathcal{W}(\cdot,\cdot')$ is wishart distribution with $\cdot$ being scale matrix, and $\cdot'$ degree of freedom (real value). The local shrinkage parameter is $\lambda = \text{diag}(\lambda_1,...,\lambda_K)$, and $\vartheta = (\vartheta_1,...,\vartheta_K)$. Such VAR formulation above is equivalent to VAR with normal-Wishart prior with local shrinkage parameters λ, and global shrinkage parameter τ. The former controls weights of covariate in each VAR equation that being projected to responsive variables. The latter, on the other hand, is global shrinkage parameter, controlling how tight overall VAR coefficients B should be. Finally the probability density function of $p(B|\Sigma,\lambda,\tau) \sim \mathcal{N}(0,(\Sigma\otimes\lambda\tau)^{-1})$ takes the form:

$$\begin{aligned}
p(B|\Sigma,\lambda,\tau) &= (2\pi)^{\frac{-NK}{2}}|\Sigma|^{\frac{K}{2}}|\lambda\tau|^{\frac{N}{2}}\exp\left(\text{Tr}\left\{-\frac{1}{2}\Sigma B(\lambda\tau)B'\right\}\right),\\
&= (2\pi)^{\frac{-NK}{2}}|\Sigma|^{\frac{K}{2}}|\lambda\tau|^{\frac{N}{2}}\exp\left(\text{Tr}\left\{-\frac{1}{2}(\lambda\tau)B'\Sigma B\right\}\right).
\end{aligned}$$

1.5 Optimal variational parameters

We work with model eq. (1.4), and shall begin by denoting the joint distribution as:

$$p(Y,X,B,\Sigma,\lambda,\tau) = p(Y|B,\Sigma)p(B|\Sigma,\lambda)p(\Sigma)p(\lambda)p(\tau). \tag{1.5}$$

The approximate distribution is defined as followed:

$$q(\theta) = q(B,\Sigma)q(\lambda)q(\tau)q(\vartheta)q(\xi). \tag{1.6}$$

With the advantage from conjugate prior of B and Σ (conditional on Horseshoe shrinkage parameters) to the posterior one can write $q(B,\Sigma) = q(B|\Sigma)q(\Sigma)$. The optimal variational parameters of $q(\theta)$ can be written as followed:

$$\begin{aligned}
\widehat{q}(B|\Sigma) &\sim \mathcal{N}_{N\times K}(\widehat{B}, \Sigma^{-1} \otimes \Phi^{-1}), \\
\widehat{q}(\Sigma) &\sim \mathcal{W}(\overline{S}, \overline{\nu}), \\
\widehat{q}(\lambda) &\sim \mathcal{G}(\overline{a}, \overline{b}), \\
\widehat{q}(\tau) &\sim \mathcal{G}(\overline{c}, \overline{d}), \\
\widehat{q}(\vartheta) &\sim \mathcal{G}(\overline{e}, \overline{f}), \\
\widehat{q}(\xi) &\sim \mathcal{G}(\overline{g}, \overline{h}).
\end{aligned} \tag{1.7}$$

The mean of each parameters will be denoted as $\widehat{X}$. For example, if $X \sim \mathcal{G}(a,b)$ then the mean of $\mathbb{E}[X] = \widehat{X} = a/b$. Next is the mean of Wishart distribution, if $\widehat{q}(\Sigma) \sim \mathcal{W}(\overline{S}, \overline{\nu})$ then $\mathbb{E}[\Sigma] = \overline{S}\overline{\nu}$. The optimal variational parameters of each parameters can be described as followed:

$$
\begin{aligned}
\widehat{B} &= \Phi^{-1}(XY'), \\
\Phi &= \widehat{\lambda}\widehat{\tau} + XX', \\
\overline{S}^{-1} &= S_0 + YY' - \widehat{B}\Phi\widehat{B}', \\
\overline{\nu} &= \nu_0 + T, \\
\overline{a} &= 1/2 + N/2, \\
\overline{b} &= \widehat{\vartheta} + 1/2\widehat{\tau}\text{diag}(\widehat{B}'\widehat{\Sigma}\widehat{B} + N\Phi^{-1}), \\
\overline{c} &= 1/2 + NK/2, \\
\overline{d} &= \widehat{\xi} + 1/2\text{Tr}\left\{\widehat{\lambda}\left(N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B}\right)\right\}, \\
\overline{e} &= 1, \\
\overline{f} &= \widehat{\lambda} + 1, \\
\overline{g} &= 1, \\
\overline{h} &= \widehat{\tau} + 1.
\end{aligned}
\tag{1.8}
$$

By iteratively computing the variational parameters of each parameters, the coordinate descent optimizes evidence lower bound one block at a time while holding other fixed, thus each parameter's KL will be minimized. Finally, how these optimal variational parameters, and evidence lower bound are obtained, can be found in the appendix of section 1.11.

1.6 A simulation study

To prove that our Vb for VAR algorithm is legitimate, we do provide estimation results with the artificial data simulated using Matlab toolbox to simulate VAR in eq. (1.4) with $N = 5, T = 754, p = 10$ where p is number of lag. Four algorithms are investigated, which are, first our proposed Vb for VAR, Ordinary least square method, Vb for regression using CCM algorithm (recently proven to be theoretically incorrect but very appealing and practically accurate which will be shown shortly after, Carriero et al. (2019)), and Vb for regression using CCCM algorithm (The new proven to be theoretically correct recently, Carriero et al. (2022)), labelled, Vb-VAR, OLS, Vb-CCM, and Vb-CCCM, respectively. It is noteworthy that the hierarchical horseshoe prior is employed and derived for optimal variational parameters for the last two algorithms.

We simulate in total of 5000 data sets with code to reproduce the similar results plotted in fig. 1.2, and fig. 1.2. Our simulation begins by first sample true VAR coefficients from $\mathcal{N}_{N,K}(0,1)$, where 30% of which is zero (represent sparsity). Then for VAR covariance we sample $U = \mathcal{U}_{N,N}(0,1)$, where $\sim \mathcal{U}(\cdot, \cdot')$ is standard uniform distribution on the open interval $(0,1)$. To guarantee the positive definite of covariance we set $\Sigma = U \times U'$. Given those two parameters we leave the rest to Matlab toolbox do the rest. Two primary functions are being used, which are, `varm`, and `simulate`. The reproducibility is carried by `rng` function in Matlab.

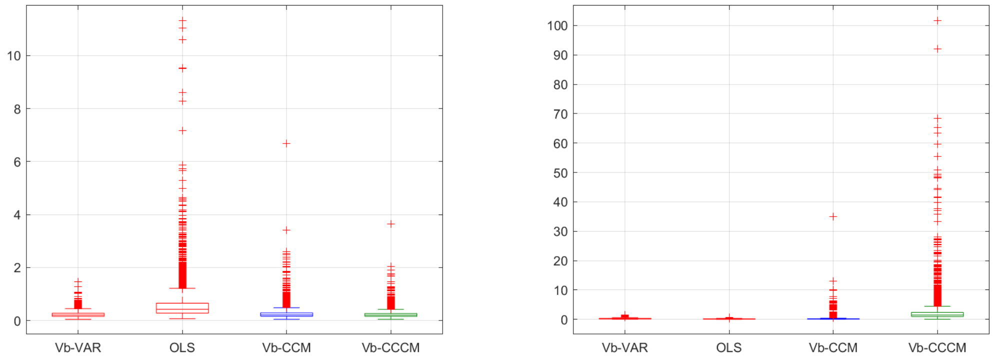


Figure 1.2: Box plots of Euclidean-norm between true VAR coefficients and approximated ones from four different algorithms, i.e.$\|\widehat{B} - B\|_2^2$ (left-panel), and $\|\widehat{\Sigma} - \Sigma\|_2^2$ (right-panel)

We illustrate boxplot of Euclidean-norm (L2-norm) between true VAR coefficients (left-panel) and covariance (right-panel) between four differen algorithms in fig. 1.2. With 30% of VAR coefficients being zero (representing sparsity), the OLS method seems to perform worst with artificial data as described above. VAR covariance, on the other hand, all algorithms are almost identical, with the exception of Vb-CCCM approach, where there are some small sample that actually spike up to 100. Despite some small difference between these four methods, fig. 1.3 suggests that all in-sample forecast mean are almost indistinguishable.

The findings in fig. 1.2 are intriguing, particularly the precision of the posterior mean of the VAR covariance estimated using the Vb-CCM and Vb-CCCM algorithms. As a

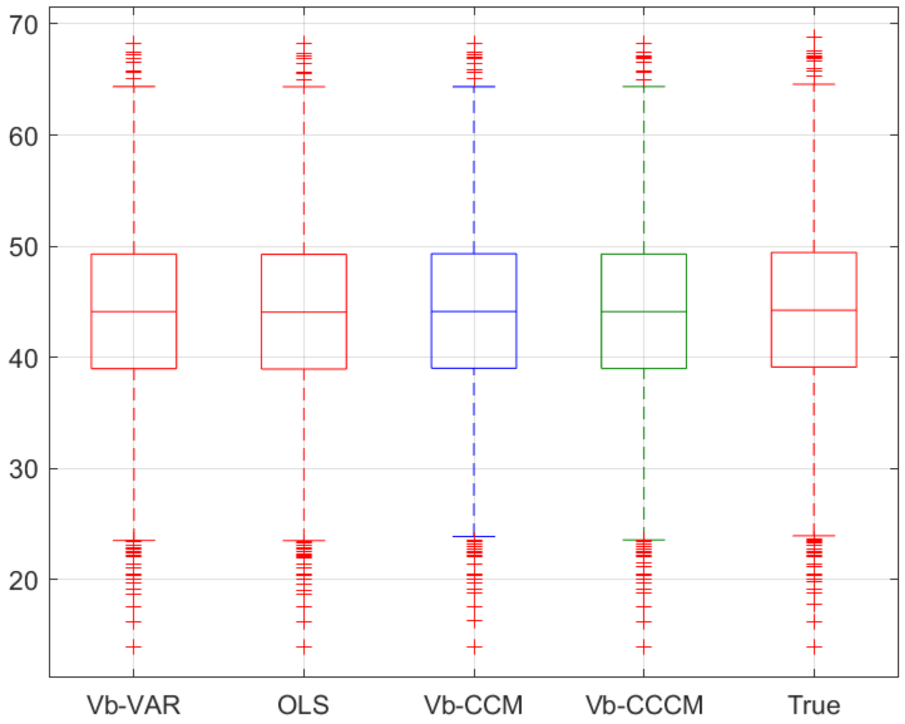


Figure 1.3: Box plots of Euclidean-norm between actual observations and in-sample VAR predictions, $\|\widehat{Y} - Y\|_2^2$.

result, we delve deeper into the accuracy of the VAR covariance estimate in the next subsection. Since two approaches recovers VAR covariance using lower-triangular A. However, as the size of the VAR covariance matrix increases, there may be limitations to such approaches.

Next sub-section we investigate further on how accurate of each algorithm on estimating VAR covariance.

1.6.1 Investigate further on VAR covariance.

In fig. 1.3, we observe that the prediction mean/median from our algorithm and other methods are quite similar. However, we will now investigate how different algorithms perform in terms of estimating VAR covariance. Specifically, we want to highlight that there are certain circumstances in which per equation algorithm (Cholesky-transformed VARs) appears to perform poorly, especially in terms of Covariance accuracy, see Arias et al. (2022), Chan et al. (2021). To begin, we will express the VAR in compact form when the per equation algorithm is utilized.

$$y_t = B_0 + B_1 y_{t-1} + B_2 y_{t-2} + A^{-1}\sigma^{1/2}E, \quad \text{where } E \sim \mathcal{N}(0, I_N). \tag{1.9}$$

The notation of the equation above is similar to the one described in eq. (1.4), where y_t is the responsive variable at time t, B_0, B_1, and B_2 are the intercept, coefficient of lag 1, and lag 2, respectively. The additional $\sigma = \text{diag}(\sigma, ..., \sigma_N)$ represents the stacking regression residuals' variance that arises from estimating VAR one equation at a time Carriero et al. (2022). The lower triangular element in A^{-1} is treated as VAR parameters, so it is necessary to assign a prior. Different priors, such as the Horseshoe prior or Minnesota prior, can theoretically lead to different results of the VAR covariance matrix since the covariance is recovered by simply computing $\Sigma = A^{-1}\sigma A^{-1'}$. To concretely prove this idea, we simulate artificial data and estimate it using different algorithms, similar to what is described in the beginning of this section. We simulate VAR data with $T = 754, N = 3$, and $p = 2$ with the following coefficients and covariance.

$$\begin{gathered} B_0 = \begin{bmatrix} 1 \\ .5 \\ -.5 \end{bmatrix} B_1 = \begin{bmatrix} .3 & -.1 & .05 \\ .1 & .2 & .1 \\ -.1 & .2 & .4 \end{bmatrix} B_2 = \begin{bmatrix} .1 & .06 & .001 \\ .001 & .1 & .01 \\ -.01 & -.01 & .2 \end{bmatrix}, \\ \Sigma = \begin{bmatrix} .373 & .139 & .615 \\ .139 & 1.53 & 1.123 \\ .615 & 1.123 & 1.655 \end{bmatrix}. \end{gathered} \tag{1.10}$$

The simulated data according to VAR parameters above eq. (1.10) can be plotted in fig. 1.4. Since our objective in this subsection is to investigate VAR covariance, we first illustrate heatmap in fig. 1.5, the figure shows the posterior mean of covariance produced by proposed VB with Horseshoe prior (top-right), MCMC-CCCM algorithm with Minnesota prior (bottom-left), and MCMC-CCCM algorithm with Horseshoe prior (bottom-right).

To aid interpretation we wish to highlight that our Vb algorithm performs better than both MCMC-CCCM-Minnesota/Horseshoe prior. This is due to the fact that the VAR covariance is actually approximated entirely, not being recovered using lower triangular A^{-1} from per equation VAR eq. (1.9). The pattern we found from result from fig. 1.5 is that the per equation algorithm Carriero et al. (2022) both using CCCM-Minnesota and CCCM-Horseshoe seem to overestimate VAR covariance. If we compare with the actual Covariance (top-left). Next we illustrate the result of Minnesota prior VAR but this time it is system-wide estimation, see fig. 1.6. Obviously the system-wide VAR is able to capture the actual covariance quite well. In fact the results are almost identical to our system-wide Vb with Horseshoe prior.

The overestimation from per equation algorithm can be larger as size of residual is increased. In other words the element in VAR covariance is large. To prove such point we simulate 100 data sets with similar VAR coefficients as described in eq. (1.10) but this time we time original actual VAR covariance by the following:

$$\begin{aligned} \text{loop} \quad & j = 1, 2, ..., 100, \\ \text{set} \quad & \Sigma_j = \Sigma \times \frac{j}{2}. \end{aligned}$$

For clarity we show the matlab code in the box below. Again we compare the posterior mean results of Euclidean norm (L2-norm) to measure the distance between actual VAR covariance in each loop. The results are demonstrated in fig. 1.8. According to fig. 1.8, it is quite obvious that as the actual VAR covariance is larger, the larger distance between actual covariance and posterior mean of those produced by per equation algorithm. Our Vb-system-wide algorithm, however, performs as well as MCMC-system-wide with Minnesota prior. Additionally the slope of Euclidean norm is tremendously lower than per equation algorithm. This lead us to conclude that although our Vb algorithm has completely different prior to MCMC, the algorithm is able to manage to be as accurate as MCMC.

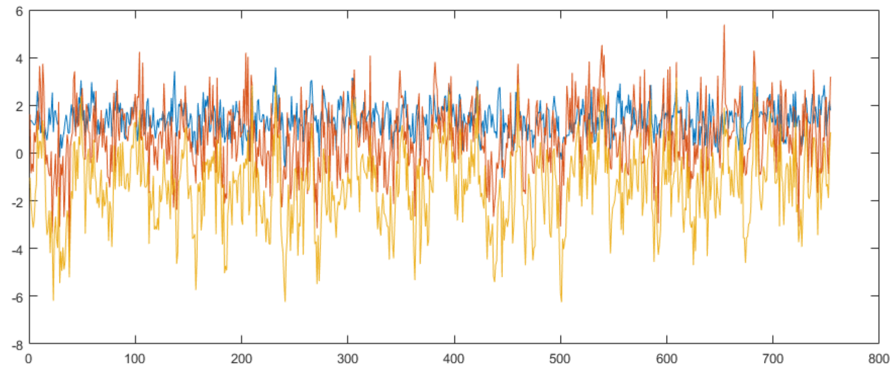


Figure 1.4: Reproducible artificial data simulated with `simulate` function from matlab according to VAR parameters in eq. (1.10).

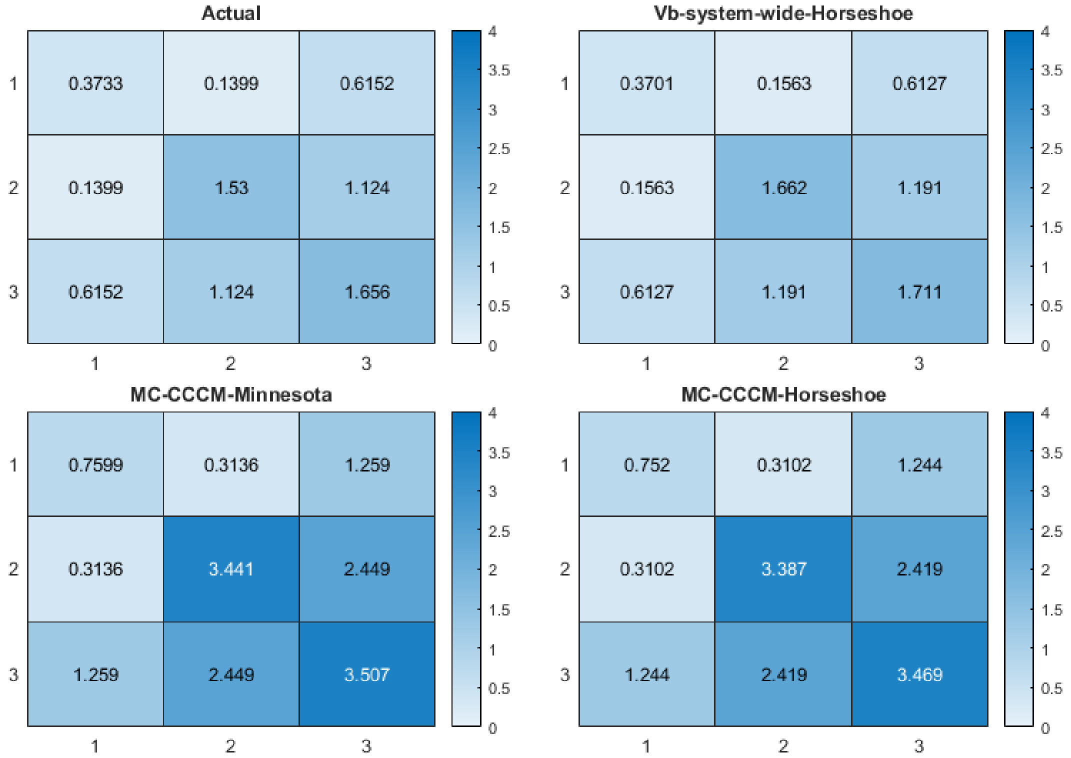


Figure 1.5: Heatmap displaying the actual and estimated covariance of a VAR model: Actual covariance from eq. (1.10) (top-left), our Vb-system-wide with Horseshoe prior (top-right), MCMC per equation algorithm Carriero et al. (2022) with Minnesota prior (bottom-left), and MCMC per equation algorithm with Horseshoe prior (bottom-right).

1.6.2 Computational costs

We also examine how the proposed Vb-VAR algorithm performs in terms of computational costs as the size of cross-section N in the VAR model increases. We simulate the

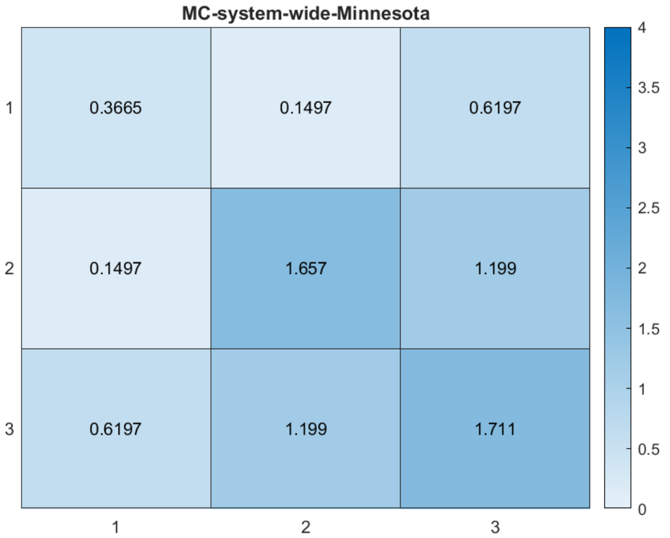


Figure 1.6: Heatmap displaying the estimated covariance of a Minnesota prior VAR model (system-wide estimation) using artificial data simulated from eq. (1.10).

```
for j=1:number_of_test
Constant = [1; 0.5; -0.5]; % intercept term
AR1 = [0.3 -0.1 0.05; 0.1 0.2 0.1; -0.1 0.2 0.4]; % VAR coefficient for lag 1
AR2 = [0.1 0.05 0.001; 0.001 0.1 0.01; -0.01 -0.01 0.2]; % VAR coefficient for lag 2
Trend = zeros(3,1); % no-trend
Sigma = [.373, .139, .615;
                        .139, 1.53, 1.123;
                                .615, 1.123, 1.655];
Sigma = Sigma*j/2; % maniupate the actual VAR covariance.
TrueMdl = varm('Constant',Constant,'AR',{AR1 AR2},'Trend',Trend,'Covariance',Sigma); % assemble VAR parameters for matlab simulate function.
rng(1228) % for reproducibility
data = simulate(TrueMdl,n); % simulate data from VAR parameters above.
end
```

Figure 1.7: Matlab code to simulate artificial data.

VAR model in a similar way to what is described in section 1.6, but with only a single lag ($p = 1$) to avoid high-dimensional issues as number of equation in VAR grows. We compare the computational costs to those of Vb-CCM and Vb-CCCM and plot the results of total time of three different algorithms, labelled Vb-VAR, Vb-CCM, and Vb-CCCM in fig. 1.9 (on 4.8 GHz AMD Ryzen 5900x machine). The x-axis represents

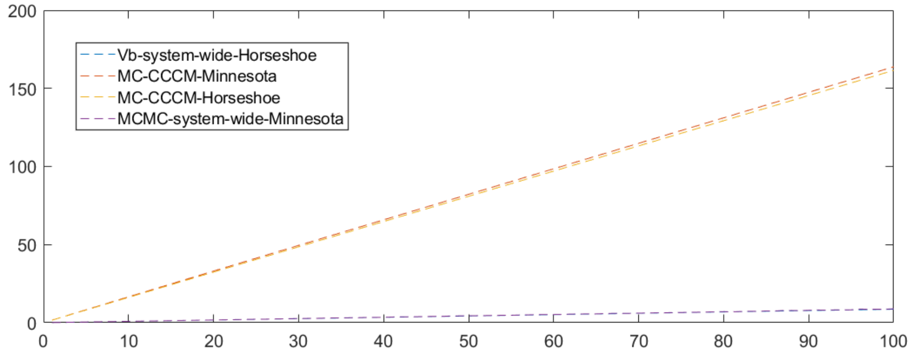


Figure 1.8: Euclidean norm distance between actual VAR covariance and posterior mean of different models and algorithms: $x-$axis is the number of simulation test, $y-$axis is the Euclidean distance (higher means worse in terms of accuracy).

size of cross-section N, and y-axis is time in seconds. As shown in the figure, the proposed algorithm surprisingly scales very well with the number of equations in the VAR model. In contrast, the computational time of both Vb-CCM and Vb-CCCM increases quickly as the number of VAR equations increases, while the proposed algorithm's computational time remains relatively stable.

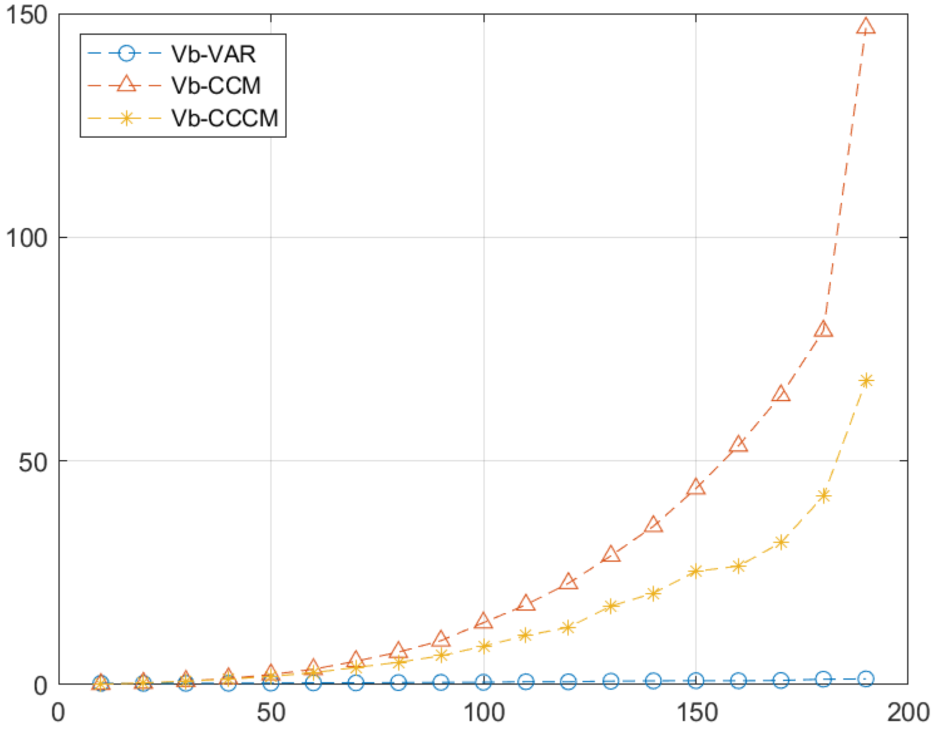


Figure 1.9: Computational costs of three different algorithms, scale with number of equations in VAR. $x-$axis is number of equations in VAR, and $y-$axis is computational time (in seconds).

1.7 Convergence and in-sample prediction

We demonstrate the proposed variational Bayes for Vector Autoregression via empirical US monthly macroeconomic applications. The method is tested using small (3 variables), medium (20 variables), and large (40 variables) datasets, spanning from 1960 to 2022.[1] The efficiency of the Vb algorithm is demonstrated by showing the convergence of the evidence lower bound (ELBO), see left-panel of figs. 1.10 to 1.12. The $x-$axis is the number of Vb-iterations, and $y-$axis is the ELBO As expected as the number of variables in VAR grows the ELBO also decreases. Next we demonstrate how our Vb-VAR algorithm with Horseshoe prior can effectively shrink the VAR coefficients and compare them to those obtained using Ordinary Least Squares (OLS). The coefficients are plotted in figs. 1.10 and 1.12, and it is evident that the Vb-VAR algorithm with Horseshoe prior can effectively shrink the VAR coefficients compared to the OLS estimates (right-panel of figs. 1.10 to 1.12). Additionally, we computed the Euclidean norm of observed data to in-sample prediction means to test the accuracy of the VAR parameters obtained from the Vb algorithm, OLS, and MCMC. The results in table 1.2 indicate that despite the VAR coefficients being shrunk, the Vb-VAR algorithm maintains the in-sample prediction error well and is as accurate as more computationally demanding methods such as MCMC. Furthermore, the computational costs of the proposed algorithm scale well with the number of variables in the VAR.

[1] Data is collected online via *fredapi* with the January 2023 vintage date.

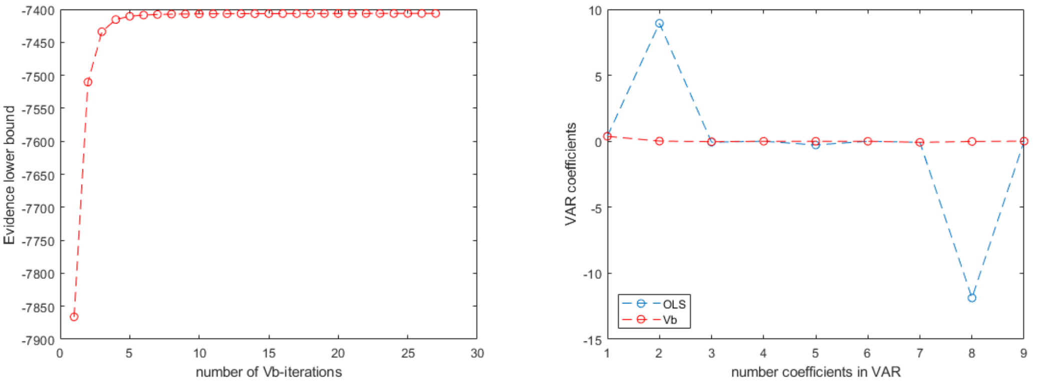


Figure 1.10: Evidence lower bound of the Vb algorithm is shown in the left panel. The vectorized VAR coefficients estimated using Vb are indicated by the red dashed-dot line, while the OLS VAR coefficients are represented by the blue dashed-dot line (Small VAR).

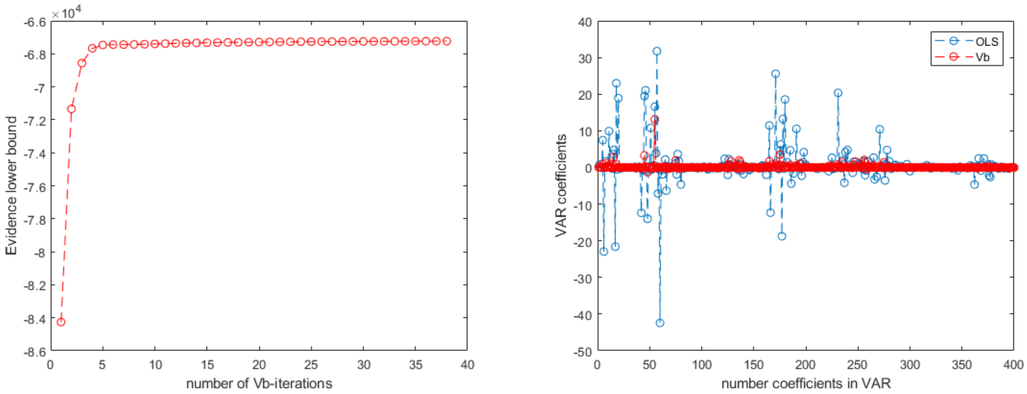


Figure 1.11: Evidence lower bound of the Vb algorithm is shown in the left panel. The vectorized VAR coefficients estimated using Vb are indicated by the red dashed-dot line, while the OLS VAR coefficients are represented by the blue dashed-dot line (Medium VAR).

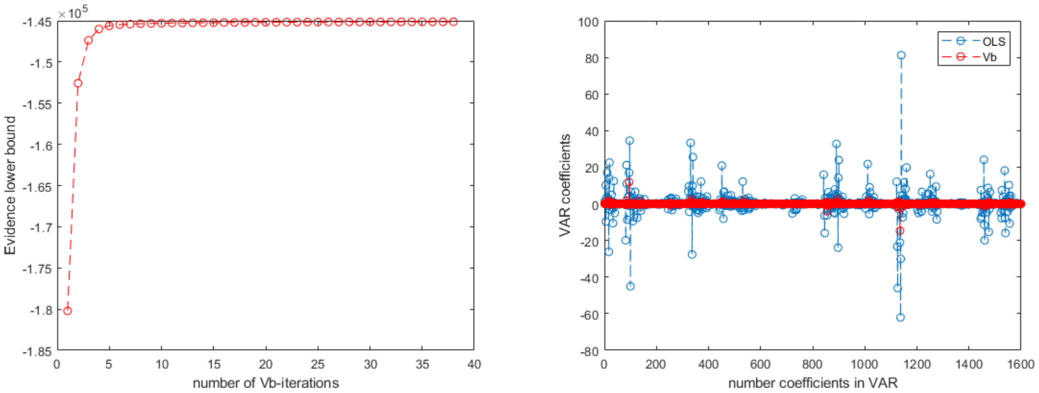


Figure 1.12: Evidence lower bound of the Vb algorithm is shown in the left panel. The vectorized VAR coefficients estimated using Vb are indicated by the red dashed-dot line, while the OLS VAR coefficients are represented by the blue dashed-dot line (Large VAR).

	Computational costs (seconds)		**Euclidean norm**$\|\widehat{Y}-Y\|_2^2$		
Number of VAR variables	**Vb**	**MCMC**	**Vb**	**OLS**	**MCMC**
3	0.10	9.4	13.0	13.0	13.0
20	0.18	580.8	15.5	15.0	15.0
40	0.19	3023.6	22.0	21.0	21.5

Table 1.2: Computational costs, Euclidean norm distance between observed data and predicted values from Vb-VAR, OLS, and MCMC (CCCM algorithm, see Carriero et al. (2022)).

1.8 Forecast Application

In this section, we utilize our proposed approach to large models to demonstrate its effectiveness. Our objective is to investigate how a large information set can influence the accuracy of out-of-sample forecasts from Vector Autoregressions (VARs). We conduct an out-of-sample exercise recursively, starting with a 20-year monthly data sample from 1960:1 to 1979:1, and ending with a sample from 1960:1 to 2021:10. We generate iterative 12-step-ahead forecasts, resulting in a total of 502 sets of forecasts that cover the period from 1980:1 to 2022:10.

As our algorithm is considerably faster than MCMC, we compare its forecasting performance to that of a widely-used benchmark: the Minnesota prior VAR. We chose this prior because it is popular among economists, and recent research Cross et al. (2019) investigate the recursive out-of-sample forecasts of such prior, compared to multiple priors, including global-local-prior. The reports from their work show that the parameters from VARs are dense rather than sparse, resulting in Minnesota prior VAR outperforms most of competitive models, with only three focused variables: US inflation ($\Delta^2 \log(CPIAUCSL)$), unemployment rate ($\Delta\, UNRATE$), and industrial production ($\Delta \log(INDPRO)$). We first compare Vb with exact three variables then gradually increase the number of variables to 20 and eventually 40 variables. Both point and density forecasts are evaluated via RMSE and CRPS, with respect to the benchmark model. This allows us to see the effects of whether increasing variables in VAR improve the forecasts. The macroeconomic variables are listed in table 1.7, and have been shown to be popular among economists in forecasting purposes in previous studies such as Bańbura et al. (2010), Koop (2013), Koop & Korobilis (2013), Giannone et al. (2015), Carriero et al. (2015). To gauge the statistic test of difference between predictive density we provide Diebold & Mariano (1995) (two-sided DM-test), representing in color light green for 1%, yellow for 5%, and orange for 10%.

A priori, we expect the inclusion of more variables in VAR to improve point forecasts via a better specification of the conditional mean, while the use of a larger data set might result in slightly worse density forecast because higher probability of error variance. However, this is not the whole story, as shown in table 1.3, which displays RMSE, and CRPS relative to the benchmark (3 variables Minnesota VAR using MCMC) of $N = 3, 20, 40$ variables in VAR using Vb, so that the value below 1 denotes a model outperforming the benchmark and vice versa.

For $N = 3$, point forecast of three selected variables, Vb performs almost identical to the benchmark at all horizons, proving that our algorithm is competitive to MCMC. For accuracy of forecast density, at one-step-ahead forecast, we found that Vb is outperformed by the benchmark 15% in forecasting US industrial production. Nonetheless, Vb's accuracy is still competitive with the benchmark for both RMSE and CRPS in forecasting US inflation and unemployment rate.

Moving to $N = 20$ VAR results from Vb approach. We start to see point forecast gain for industrial production at $h = 1, 4, 8$. As discuss in priori expectation about larger variable in VAR, the improvement on point forecast is quite obvious for INDPRO variable but the CRPS is still identical to the benchmark. For US inflation, we found that the one-step-ahead forecast performs slightly worse than the benchmark by approximately 16 percentage points. Additionally, when comparing between $N = 20$ and $N = 3$, $N = 20$ also underperforms by roughly 6 percentage points. We suspect that such results are from the additional predictors lead to over or under predictive mean. However predictive density seems to be improved from $N = 3$ where now at $h = 4, 8, 12$ it is able to compete with the benchmark whereas Vb $N = 3$ moderately worse than the benchmark approximately 2%.

It appears that the inclusion of more variables in VAR using the Vb approach provides an advantage, particularly in forecasting US industrial production at $N = 40$, where the one-step-ahead forecast is improved up to 24%. However, as expected, including more variables may result in slightly worse density forecast accuracy, with the CRPS being 4% outperformed by the benchmark. It is worth noting that the inflation and unemployment rate forecasts are still identical to the benchmark. Therefore, it can be concluded that the Vb algorithm is as good as a more computationally demanding MCMC approach while enjoying exceptionally low computational cost.

		N=3							
		CPIAUCSL	**UNRATE**	**INDPRO**			**CPIAUCSL**	**UNRATE**	**INDPRO**
RMSE	h=1	1.10	0.99	0.99	**CRPS**	h=1	1.09	0.97	1.15
	h=4	1.02	0.97	1.00		h=4	1.02	1.03	1.04
	h=8	1.02	1.01	1.03		h=8	1.02	1.04	1.05
	h=12	1.02	1.01	1.03		h=12	1.02	1.04	1.06
		N=20							
		CPIAUCSL	**UNRATE**	**INDPRO**			**CPIAUCSL**	**UNRATE**	**INDPRO**
RMSE	h=1	1.16	0.99	0.81	**CRPS**	h=1	1.12	1.11	1.00
	h=4	1.00	0.97	0.97		h=4	1.00	0.99	0.99
	h=8	1.00	1.00	0.99		h=8	1.00	1.00	1.00
	h=12	1.00	1.00	1.00		h=12	1.00	1.00	1.01
		N=40							
		CPIAUCSL	**UNRATE**	**INDPRO**			**CPIAUCSL**	**UNRATE**	**INDPRO**
RMSE	h=1	1.11	0.97	0.76	**CRPS**	h=1	1.14	1.08	1.04
	h=4	1.00	0.96	0.96		h=4	1.02	0.97	0.99
	h=8	1.00	1.00	0.99		h=8	1.02	1.00	1.01
	h=12	1.00	1.00	1.00		h=12	1.02	1.00	1.02

Table 1.3: RMSE and CRPS of Vb system-wide VAR relative to benchmark model (Minnesota prior VAR using MCMC). Color in each block indicate statistical siginificance of the Diebold-Mariano test (two-side) for equal predictive performance at 1 (light green), 5 (yellow), 10 (orange) percent level.

1.8.1 Investigating the accuracy of tails forecasts

There are concerns in the economics literature that the Vb approach to predictive forecasts can be weak because the posterior distribution of parameters and forecasts tends to be centered around the mean and median, which may ignore many possible parameter and forecast distributions. To address these concerns and demonstrate the reliability of our Vb approach, we report an emphasized-CRPS score that focuses on five regions of out-of-sample forecast density: both-sided tails, uniform (no weights augmented), center (using median/mean as weights), left tail, and right tail. We present the scores in ratio to the benchmark for inflation, unemployment rate, and industrial production in table 1.4, table 1.5, and table 1.6, respectively. Specifically, we aimed to show that our Vb approach produces scores that are as close to one as possible, indicating that the posterior predictive distribution is nearly identical to those produced by the Minnesota prior VAR using MCMC.

Our study shows that the Vb approach is highly reliable and accurate compared to the MCMC approach, despite being less computationally demanding. This proves that Vb is a viable alternative that saves computational resources without sacrificing accuracy.

N=3

CPIAUCSL	*tail*	*uniform*	*centre*	*right*	*left*
h=1	1.07	1.09	1.09	1.09	1.08
h=4	1.02	1.02	1.03	1.04	1.01
h=8	1.01	1.02	1.02	1.03	1.01
h=12	1.02	1.02	1.03	1.03	1.01

N=20

CPIAUCSL	*tail*	*uniform*	*centre*	*right*	*left*
h=1	1.10	1.12	1.13	1.12	1.12
h=4	1.01	1.01	1.01	1.00	1.01
h=8	1.00	1.01	1.01	1.00	1.01
h=12	1.00	1.01	1.01	1.00	1.01

N=40

CPIAUCSL	*tail*	*uniform*	*centre*	*right*	*left*
h=1	1.10	1.14	1.15	1.12	1.14
h=4	1.03	1.02	1.02	1.01	1.03
h=8	1.03	1.02	1.02	1.01	1.03
h=12	1.03	1.02	1.02	1.02	1.03

Table 1.4: Emphasize of tail, uniform, centre, right, and left of CPIAUCSL predictive distribution obtained from weighted-CRPS, averaged over forecasting evaluation periods of the Vb system-wide VAR compared to the benchmark model (Minnesota prior VAR using MCMC).

N=3					
UNRATE	*tail*	*uniform*	*centre*	*right*	*left*
h=1	0.96	0.97	0.97	0.99	0.94
h=4	1.02	1.03	1.03	1.01	1.04
h=8	1.04	1.04	1.04	1.01	1.06
h=12	1.04	1.04	1.04	1.02	1.06

N=20					
UNRATE	*tail*	*uniform*	*centre*	*right*	*left*
h=1	1.09	1.11	1.12	1.11	1.10
h=4	0.98	0.99	0.99	1.00	0.97
h=8	1.00	1.00	1.00	1.01	1.00
h=12	1.01	1.00	1.00	1.01	1.00

N=40					
UNRATE	*tail*	*uniform*	*centre*	*right*	*left*
h=1	1.06	1.08	1.09	1.09	1.06
h=4	0.97	0.97	0.98	0.99	0.95
h=8	1.00	1.00	1.00	1.00	1.00
h=12	1.00	1.00	1.00	1.00	1.00

Table 1.5: Emphasize of tail, uniform, centre, right, and left of UNRATE predictive distribution obtained from weighted-CRPS, averaged over forecasting evaluation periods of the Vb system-wide VAR compared to the benchmark model (Minnesota prior VAR using MCMC).

N=3					
INDPRO	*tail*	*uniform*	*centre*	*right*	*left*
h=1	1.14	1.15	1.16	1.06	1.24
h=4	1.03	1.04	1.04	1.06	1.03
h=8	1.04	1.05	1.05	1.08	1.02
h=12	1.05	1.06	1.06	1.09	1.02

N=20					
INDPRO	*tail*	*uniform*	*centre*	*right*	*left*
h=1	0.94	1.01	1.03	0.95	1.04
h=4	1.00	0.99	0.99	0.98	1.00
h=8	1.01	1.00	1.00	1.00	1.00
h=12	1.02	1.01	1.00	1.01	1.01

N=40					
INDPRO	*tail*	*uniform*	*centre*	*right*	*left*
h=1	0.99	1.05	1.06	0.98	1.09
h=4	0.99	0.99	0.99	1.00	0.00
h=8	1.01	1.01	1.02	1.01	0.00
h=12	1.04	1.02	1.01	1.03	1.01

Table 1.6: Emphasize of tail, uniform, centre, right, and left of INDPRO predictive distribution obtained from weighted-CRPS, averaged over forecasting evaluation periods of the Vb system-wide VAR compared to the benchmark model (Minnesota prior VAR using MCMC).

1.9 Impulse response functions comparisons

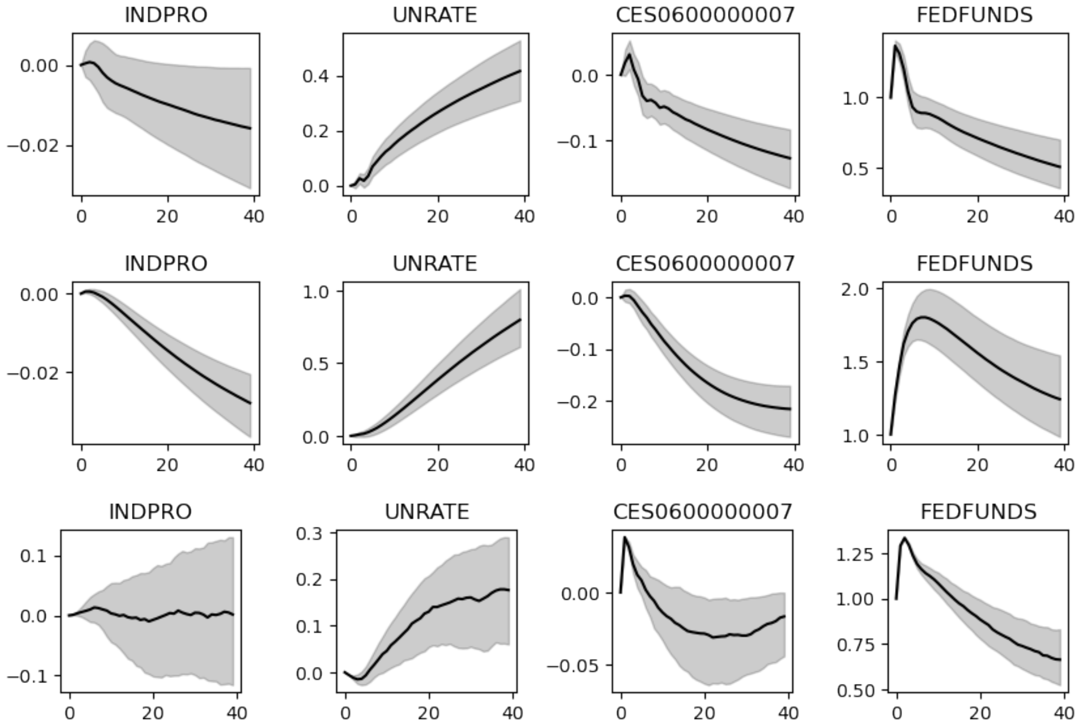


Figure 1.13: Impulse response (untransformed or cumulated) of selected variables to FEDFUNDS shocks. Conjugate normal-Wishart VAR (top panel), Minnesota VAR using MCMC CCCM algorithm (middle panel) and Horseshoe VAR using Vb algorithm (bottom).

1.10 Conclusion

We have developed a new method called Variational Bayes for Vector Autoregression (Vb-VAR) which uses a Horseshoe prior, and approximate system-wide VAR (not one equation at a time). To the best of our knowledge, this method has not been previously studied in literature. We first investigate our approach with simulated artificial data, in comparison with per equation algorithm. We found that our Vb approach is better in terms of in-sample prediction, and is able beat estimating one equation at a time method in terms of approximating VAR covariance.

Then we also test the method on real-world US macroeconomic data, and found that it performs well, producing VAR parameters that are as accurate as more demanding MCMC methods, while also enjoying fast computation. Finally, we demonstrate that our method converges properly by maximizing the evidence lower bound to the log marginal likelihood. Our approach scales considerably well with the number of variables in VAR in terms of computational efficiency.

1.11 Appendix A: Optimal variational parameters

To begin with the optimal variational parameters of B, Σ. Let $\mathbb{E}_q[\bullet]$ denotes the expectation of $[\bullet]$ with respect to approximate distribution $q(\theta)$.

$$\widehat{q}(B, \Sigma) = \mathbb{E}_q \left[\log p(Y|B, \Sigma) + \log p(B|\Sigma, \lambda, \tau) + \log p(\Sigma)\right], \tag{1.11}$$

We first derive the last term in eq. (1.11).

$$\mathbb{E}[\log p(\Sigma)] = \frac{\nu_0 - N - 1}{2}\mathbb{E}_q[\log|\Sigma|] - \frac{1}{2}\mathbb{E}_q\left[\text{Tr}\{S_0^{-1}\Sigma\}\right].$$

Next is the middle term.

$$\mathbb{E}[\log p(B|\Sigma, \lambda, \tau)] = \frac{K}{2}\mathbb{E}_q[\log|\Sigma|] + \frac{N}{2}\mathbb{E}_q[\log|\lambda\tau|] - \frac{1}{2}\mathbb{E}_q\left[\text{Tr}\big\{(\lambda\tau)B'\Sigma B\big\}\right].$$

Lastly the term

$$\mathbb{E}[\log p(Y|B, \Sigma)] = \frac{T}{2}\mathbb{E}_q[\log|\Sigma|] - \frac{1}{2}\mathbb{E}_q\left[\text{Tr}\{(Y - BX)'\Sigma(Y - BX)\}\right].$$

Re-arrange all terms then we have:

$$\begin{aligned}\log\widehat{q}(B, \Sigma) = &\left(\frac{\nu_0 - N - 1 + K + T}{2}\right)\log|\Sigma| - \frac{1}{2}\text{Tr}\{S_0^{-1}\Sigma\} \\ &- \frac{1}{2}\text{Tr}\{(\lambda\tau)B'\Sigma B\} - \frac{1}{2}\text{Tr}\{(Y - BX)'\Sigma(Y - BX)\}.\end{aligned}$$

After quite a bit of manipulation then we have:

$$\begin{aligned}\widehat{q}(B, \Sigma) = &\left(\frac{\nu_0 - N - 1 + K + T}{2}\right)\log|\Sigma| - \frac{1}{2}\text{Tr}\{\Sigma(B - \widehat{B})\Phi(B - \widehat{B})'\} \\ &\frac{1}{2}\text{Tr}\left\{\Sigma\left(S_0 + YY' - \widehat{B}\Phi\widehat{B}'\right)\right\}.\end{aligned}$$

where $\widehat{B} = \Phi^{-1}(XY')$, and $\Phi = \widehat{\lambda}\widehat{\tau} + XX'$. Such derivation is exactly to the condition posterior distribution in the case of normal-Wishart VAR in Gibb-sampling algorithm, see Koop (2003), and Dieppe et al. (2016) eq.3.4.15 for clarity . The above expression is then recognized as normal-Wishart conjugate posterior distribution with the form of $q(B, \Sigma)$ being $q(\Sigma)q(B|\Sigma)$.

Next is Horseshoe local shrinkage parameters λ where:

$$\widehat{q}(\lambda) = \mathbb{E}_q[\log p(B|\Sigma, \lambda, \tau) + \log p(\lambda|\vartheta)].$$

First derive the last term of eq above, where the following is up to its additive constant:

$$\mathbb{E}_q[\log p(\lambda|\vartheta)] = -1/2 \log \lambda - \widehat{\vartheta}\lambda.$$

Deriving another term where it is up to additive constant apart from λ then we have:

$$\mathbb{E}_q[\log p(B|\Sigma, \lambda, \tau)] = \frac{N}{2}\mathbb{E}_q\left[\sum_{j=1}^{K} \log \lambda_j\right] - \frac{1}{2}\mathbb{E}_q\left[\mathrm{Tr}\{(\lambda\tau)B'\Sigma B\}\right].$$

Lemma 1. *Suppose that* $X \sim \mathcal{N}_{N,K}(M, \Delta \otimes \Omega)$, *and let* A be $N \times N$ *matrix then the expectation of* X *follows immediately, see* Gupta & Nagar (2018) *page 60.*

$$\mathbb{E}[X'AX] = \mathrm{Tr}\{\Delta A'\}\Omega + M'AM.$$

Using Lemma 1 and some simple algebra then we have:

$$\widehat{q}(\lambda) = \left(\frac{N}{2} + \frac{1}{2} - 1\right)\log \lambda - \lambda\left(\frac{\widehat{\tau}}{2}\mathrm{diag}\left\{\left[N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B}\right]\right\} + \overline{\vartheta}\right).$$

The eq above is recognized as Gamma distribution:

$$\begin{aligned} \widehat{q}(\lambda) &\sim \mathcal{G}(\overline{a}, \overline{b}), \qquad \text{where,} \\ \overline{a} &= \frac{1+N}{2}, \\ \overline{b} &= \widehat{\vartheta} + 1/2\widehat{\tau}\mathrm{diag}(\widehat{B}'\widehat{\Sigma}\widehat{B} + N\Phi^{-1}). \end{aligned}$$

Next is the global shrinkage parameter τ:

$$\widehat{q}(\tau) = \mathbb{E}_q \left[\log p(\tau|\xi) + \log p(B|\Sigma, \lambda, \tau) \right].$$

Again up to additive constant that is independent from τ we have:

$$\widehat{q}(\tau) = \left(\frac{NK}{2} + \frac{1}{2} - 1 \right) \log \tau - \tau \left(\widehat{\xi} + \frac{1}{2} \mathrm{Tr} \left\{ \widehat{\lambda} \big(N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B} \big) \right\} \right).$$

The eq above is recognized as Gamma distribution:

$$\begin{aligned} \widehat{q}(\tau) &\sim \mathcal{G}(\overline{c}, \overline{d}), \qquad \text{where,} \\ \overline{c} &= \frac{1 + NK}{2}, \\ \overline{d} &= \widehat{\xi} + 1/2 \mathrm{Tr} \left\{ \widehat{\lambda} \left(\widehat{B}'\widehat{\Sigma}\widehat{B} + N\Phi^{-1} \right) \right\}. \end{aligned}$$

Next is hierarchical local and global shrinkage ϑ, ξ.

$$\widehat{q}(\xi) = (1 - 1) \log \xi - \xi(1 + \overline{\tau})$$

$$\widehat{q}(\vartheta_j) = (1 - 1) \log \vartheta_j - \vartheta_j(1 + \widehat{\lambda}_j).$$

$$\begin{aligned} \widehat{q}(\vartheta) &\sim \mathcal{G}(\overline{e}, \overline{f}), \qquad \text{where,} \\ \overline{e} &= 1, \\ \overline{f} &= \widehat{\lambda} + 1. \end{aligned}$$

Finally

$$\begin{aligned} \widehat{q}(\xi) &\sim \mathcal{G}(\overline{g}, \overline{h}), \qquad \text{where,} \\ \overline{g} &= 1, \\ \overline{h} &= \widehat{\tau} + 1. \end{aligned}$$

1.12 Appendix B: Evidence Lower Bound

Evidence lower bound

$$
\begin{aligned}
\mathcal{L} = \mathbb{E}_q\big[&\log p(Y|B,\Sigma) + \log p(B|\Sigma,\lambda) + \log p(\Sigma) + \log p(\lambda) \\
&+ \log p(\tau) - \log q(B,\Sigma,\lambda,\vartheta)\big].
\end{aligned} \tag{1.12}
$$

The first term of eq. (1.12):

$$
\begin{aligned}
\mathbb{E}_q[\log p(Y|B,\Sigma)] &= -\frac{TN}{2}\log(2\pi) + \frac{T}{2}\mathbb{E}_q[\log|\Sigma|] \\
&\quad - \frac{1}{2}\mathbb{E}_q\left[\mathrm{Tr}\left\{(Y-BX)'\Sigma(Y-BX)\right\}\right], \\
&= -\frac{TN}{2}\log(2\pi) + \frac{T}{2}\left(\psi_N\left(\frac{\overline{\nu}}{2}\right) + N\log(2) + \log|\overline{S}|\right) \\
&\quad - \frac{1}{2}\mathrm{Tr}\left\{(Y-\widehat{B}X)'\widehat{\Sigma}(Y-\widehat{B}X)\right\} - \frac{1}{2}\mathrm{Tr}\{NX'\Phi^{-1}X\}.
\end{aligned}
$$

where $\psi_N(X)$ is multivariate Gamma function. The second term of eq. (1.12):

$$
\begin{aligned}
\mathbb{E}_q[\log p(B|\Sigma,\lambda,\tau)] &= -\frac{NK}{2}\log(2\pi) + \frac{K}{2}\mathbb{E}_q\left[\log|\Sigma|\right] + \frac{N}{2}\mathbb{E}_q\left[\log|\lambda\tau|\right], \\
&\quad - \frac{1}{2}\mathbb{E}_q\left[\mathrm{Tr}\left\{\Sigma B(\lambda\tau)B'\right\}\right]. \\
&= -\frac{NK}{2}\log(2\pi) + \frac{K}{2}\left(\psi_N\left(\frac{\overline{\nu}}{2}\right) + N\log(2) + \log|\overline{S}|\right) \\
&\quad + \frac{NK}{2}\left(\psi(\overline{c}) - \log(\overline{d})\right) + \frac{N}{2}\left(\psi(\overline{a}) - \log(\overline{b})\right) \\
&\quad - \frac{1}{2}\mathrm{Tr}\left\{NI_K + \widehat{B}'\widehat{\Sigma}\widehat{B}\right\}.
\end{aligned}
$$

where I_K is $K \times K$ dimension of identity matrix. Next the third term of eq. (1.12):

$$
\begin{aligned}
\mathbb{E}_q[\log p(\Sigma)] &= \left(\frac{\nu_0 - N - 1}{2}\right)\mathbb{E}_q[\log|\Sigma|] - \frac{1}{2}\mathrm{Tr}\{S_0^{-1}\widehat{\Sigma}\} - \frac{\nu_0 N}{2} - \frac{\nu_0}{2}\log|S_0| - \log\Gamma(\nu_0/2), \\
&= \left(\frac{\nu_0 - N - 1}{2}\right)\left(\psi_N\left(\frac{\overline{\nu}}{2}\right) + N\log(2) + \log|\overline{S}|\right) - \frac{1}{2}\mathrm{Tr}\{S_0^{-1}\widehat{\Sigma}\} - \frac{\nu_0 N}{2} \\
&\quad - \frac{\nu_0}{2}\log|S_0| - \log\Gamma(\nu_0/2).
\end{aligned}
$$

Next is the term $\log p(\lambda), \log p(\tau), \log$ in eq. (1.12):

$$\begin{aligned}
\mathbb{E}_q[\log p(\lambda|\vartheta)] &= 1/2\mathbb{E}_q[\log \vartheta] - 1/2\big(\psi(\overline{a}) - \log(\overline{b})\big) - \widehat{\vartheta}\widehat{\lambda} - \log\Gamma(1/2). \\
\mathbb{E}_q[\log p(\tau|\xi)] &= 1/2\mathbb{E}_q[\log \xi] - 1/2\big(\psi(\overline{c}) - \log(\overline{d})\big) - \widehat{\xi}\widehat{\tau} - \log\Gamma(1/2), \\
\mathbb{E}_q[\log p(\vartheta)] &= -1/2\big(\psi(\overline{e}) - \log(\overline{f})\big) - \widehat{\vartheta} - \log\Gamma(1/2). \\
\mathbb{E}_q[\log p(\xi)] &= -1/2\big(\psi(\overline{g}) - \log(\overline{h})\big) - \widehat{\xi} - \log\Gamma(1/2).
\end{aligned}$$

Entropy

$$\begin{aligned}
-\mathbb{E}[q(B|\Sigma)] &= \frac{NK}{2}\log(2\pi) - \frac{K}{2}\left(\frac{\psi(\overline{\nu})}{2} + N\log(2) + \log|\overline{S}|\right) - \frac{N}{2}\left(\psi(\overline{a}) - \log(\overline{b})\right), \\
-\mathbb{E}[q(\Sigma)] &= \frac{N+1}{2}\log|\overline{S}| + \frac{1}{2}N(N+1)\log(2) + \log\Gamma\left(\frac{\overline{\nu}}{2}\right) - \frac{\overline{\nu}-N-1}{2}\psi_N\left(\frac{\overline{\nu}}{2}\right) + \frac{\overline{\nu}N}{2}, \\
-\mathbb{E}[q(\lambda)] &= \overline{a} - \log(\overline{b}) + \log\Gamma(\overline{a}) - (\overline{a}-1)\psi(\overline{a}), \\
-\mathbb{E}[q(\tau)] &= \overline{c} - \log(\overline{d}) + \log\Gamma(\overline{c}) - (\overline{c}-1)\psi(\overline{c}).
\end{aligned}$$

1.13 Appendix C: Data

no.	FRED Acronyms	description	tcode	VAR-3	VAR-20	VAR-40	IrF-analysis
1	FEDFUNDS	Federal Funds Effective Rate	2	x	x	x	x
2	CPIAUCSL	Consumer Price Index for All Urban Consumers: All Items in U.S. City Average	6	x	x	x	
3	UNRATE	Unemployment Rate	2	x	x	x	x
4	M1SL	M1	6		x	x	
5	M2SL	M2	6		x	x	
6	PAYEMS	All Employees, Total Nonfarm	5		x	x	x
7	HOUST	New Privately-Owned Housing Units Started: Total Units	4		x	x	
8	INDPRO	Industrial Production: Total Index	5		x	x	x
9	TB3MS	3-Month Treasury Bill Secondary Market Rate, Discount Basis	2		x	x	
10	AAA	Moody's Seasoned Aaa Corporate Bond Yield	2		x	x	
11	PCEPI	Personal Consumption Expenditures: Chain-type Price Index	6		x	x	x
12	GS10	Market Yield on U.S. Treasury Securities at 10-Year Constant Maturity, Quoted on an Investment Basis	2		x	x	
13	PERMIT	New Privately-Owned Housing Units Authorized in Permit-Issuing Places: Total Units	4		x	x	
14	BAA	Moody's Seasoned Baa Corporate Bond Yield	2		x	x	
15	BUSLOANS	Commercial and Industrial Loans, All Commercial Banks	6		x	x	
16	M2REAL	Real M2 Money Stock	5		x	x	
17	CLF16OV	Civilian Labor Force Level	5		x	x	
18	MANEMP	All Employees, Manufacturing	5		x	x	
19	TOTRESNS	Reserves of Depository Institutions: Total	6		x	x	
20	CE16OV	Employment Level	5		x	x	
21	CPIMEDSL	Consumer Price Index for All Urban Consumers: Medical Care in U.S. City Average	6			x	
22	AWHMAN	Average Weekly Hours of Production and Nonsupervisory Employees, Manufacturing	1			x	
23	GS1	Market Yield on U.S. Treasury Securities at 1-Year Constant Maturity, Quoted on an Investment Basis	2			x	
24	CPIAPPSL	Consumer Price Index for All Urban Consumers: Apparel in U.S. City Average	6			x	
25	RPI	Real Personal Income	5			x	x
26	GS5	Market Yield on U.S. Treasury Securities at 5-Year Constant Maturity, Quoted on an Investment Basis	2			x	
27	USCONS	All Employees, Construction	5			x	
28	CES3000000008	Average Hourly Earnings of Production and Nonsupervisory Employees, Manufacturing	6			x	x
29	UEMPMEAN	Average Weeks Unemployed	2			x	
30	W875RX1	Real personal income excluding current transfer receipts	5			x	
31	UEMP27OV	Number Unemployed for 27 Weeks & over	5			x	
32	TB3SMFFM	3-Month Treasury Bill Minus Federal Funds Rate	1			x	
33	USGOVT	All Employees, Government	5			x	
34	REALLN	Real Estate Loans, All Commercial Banks	6			x	
35	CPITRNSL	Consumer Price Index for All Urban Consumers: Transportation in U.S. City Average	6			x	
36	CUSR0000SAS	Consumer Price Index for All Urban Consumers: Services in U.S. City Average	6			x	
37	BAAFFM	Moody's Seasoned Baa Corporate Bond Minus Federal Funds Rate	1			x	
38	USTRADE	All Employees, Retail Trade	5			x	
39	T10YFFM	10-Year Treasury Constant Maturity Minus Federal Funds Rate	1			x	
40	IPMANSICS	Industrial Production: Manufacturing (SIC)	5			x	
41	CES0600000007	Average Weekly Hours of Production and Nonsupervisory Employees, Goods-Producing	1				x
43	CMRMTSPLx	Real Manu. and Trade Industries Sales	5				x
44	DPCERA3M086SBEA	Real personal consumption expenditures (chain-type quantity index)	5				x
45	PPICMM	Producer Price Index by Commodity: Metals and Metal Products: Primary Nonferrous Metals	6				x
46	PPIFGS	Producer Price Index by Commodity for Finished Goods	6				x
47	CUMFNS	Capacity Utilization: Manufacturing (SIC)	2				x

Table 1.7: Data used for 3/20/40-variables VAR, and impulse response analysis. The column tcode denotes the following transformation for series y: (1) no transformation; (2) Δy_t; (3) $\Delta^2 y_t$; (4)$\log(y_t)$; (5)$\Delta \log(y_t)$; (6) $\Delta^2 \log(y_t)$; (7)$\Delta(y_t/y_{t-1} - 1)$.

1.14 Appendix D: Additional Results

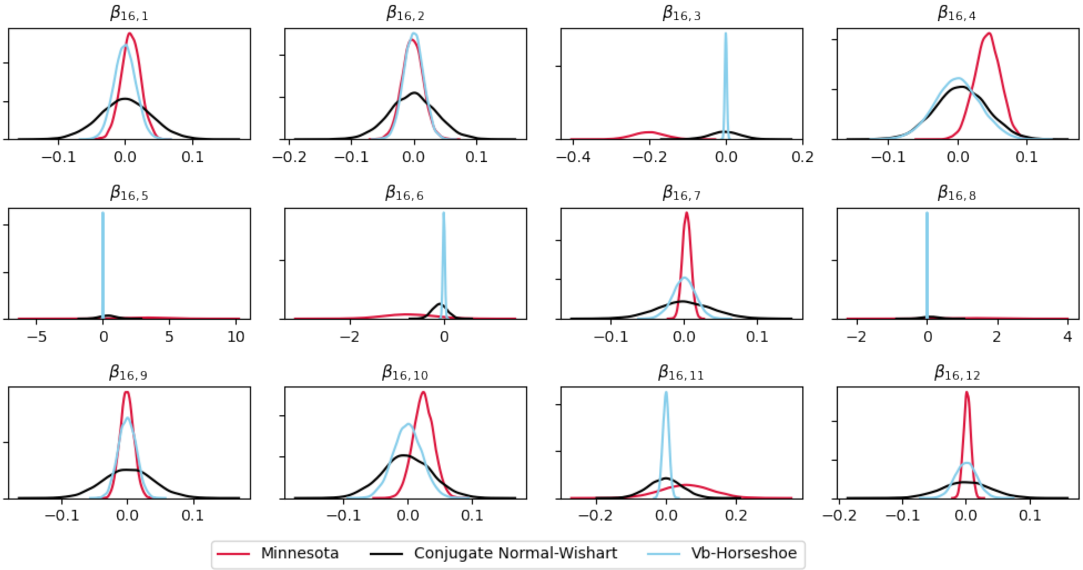


Figure 1.14: Kernel estimate of posterior distribution from different model.

Chapter 2

Variational Bayes-Expectation Maximisation for Mixed Frequency VAR

2.1 Introduction

Mixed frequency models have been beneficial for economist to do real-time forecast of future low-frequency observation by utilizing the predictive information of high-frequency data. From regression point of view the method of mixed-data sampling (MIDAS) is introduced. For Vector Auto-regression (VAR), on the other hand, the method to handle different observed data is from state-space model. This chapter of thesis focuses mainly on the latter model. Through the incorporation of high-frequency data, such as monthly observations, alongside lower-frequency data (quarterly observations), MF-VAR models frequently produce forecasts that are not only more precise but also more timely. This is especially advantageous in the realm of economic policy makings and decision-makings.

In the original MF-VAR model, as outlined by Schorfheide & Song (2015), the approach is straightforward: include GDP as the low-frequency data and then add various monthly macroeconomic variables into the mix. Thanks to the early availability of these monthly variables, you can create a monthly estimate of GDP. This concept gives rise to the idea of real-time nowcasting, allowing us to gauge the current state of GDP as it unfolds.

Another advantages of MF-VAR model that is worth mentioning is one can also incorporate data from multiple frequencies (more than two frequencies). The most common practice in the econometric literature with MF-VAR models involves two frequencies, specifically monthly and quarterly observations.

In practice, the MF-VAR framework can be extended to encompass more than two frequencies, provided that these mixed frequencies remain consistent. For example, it is feasible to combine data at the minute, hourly, and daily levels. This flexibility arises from the constant relationships among these units of time - an hour always consists of 60 minutes, and a day comprises 24 hours. This approach is elucidated in recent research focused on estimating MF-VAR models using the EM algorithm, as demonstrated by Brave et al. (2020). However, when mixed-frequency observations lack consistency, manual data manipulation becomes necessary. For instance, it is impractical to mix daily, monthly, and quarterly data due to the varying number of days in each month, not to mention the occurrence of a 29-day February every four years. This serves as an example of the inconsistency that can arise in mixed-frequency observations. We highlight that our purposed algorithm can approximate multiple frequencies as long as it suffices the mentioned conditions above.

Since the pioneer of MF-VAR from Schorfheide & Song (2015), there has been considerable growth in related studies. Example includes Ghysels & Ozkan (2015), Wohlrabe (2009), Marcellino et al. (2016), Carriero et al. (2020), Koop, McIntyre & Mitchell (2020), Huber et al. (2020), Koop, McIntyre, Mitchell & Poon (2020), Clark et al. (2023), Koop, McIntyre, Mitchell, Poon et al. (2020), Brave et al. (2016). These growing mentioned mixed frequency articles share two things in common. First is the method to handle the different frequencies time series data using state-space model and the global parameters are estimated by Markov Chain Monte Carlo (MCMC) method (Gibbs sampling). Although Gibbs sampling method has strong statistical guarantee approximation but the computational cost is often high and sometimes suffer from the slow convergence even for medium size VAR such as US eleven macroeconomic, see Schorfheide & Song (2015). One alternative to Gibbs sampling is to estimate via EM-algorithm, see Brave et al. (2020). This method is fast since there is no sampling-involved in the algorithm. The approach is carried through two steps. E-step does exact posterior (analytical expression is available) of state parameters conditional on global parameters using Kalman filter and Smoother. Then treating those state variables as complete data, and maximize likelihood function of VAR using modified ordinary least square. The modified OLS is the maximum likelihood OLS conditional on estimators from E-step.

We highlight that this method, EM-algorithm, is not Bayesian inference and may not be suitable for high-dimensional data, and is unable to provide uncertainty of forecasts. Another option which is most related to our work is to approximate with variational Bayes (Vb). There are quite few literature on specific model such as MF-VAR, to our knowledge Gefang et al. (2020) for example, introduce an algorithm that modify the M-step. They first generate state variables similarly to the E-step in EM-algorithm. Then conditional on state variable, they approximate VAR parameters with Vb. Noteworthy the Vb is performed by per equation algorithm.

This algorithm is computationally cheap with one shortcoming. To be more specific, the derivation of Vb in their M-step has no expectation with respect to the estimates produced by the E-step. In other words, an integral of state variables from E-step are ignored. Consequently the evidence lower bound of the marginal likelihood is not maximized, which is a crucial key, indicating the convergence of an algorithm for Vb in any model. Moreover EM-algorithm must yield non-decreasing log-likelihood as the EM-iteration move forward. We prove that literature algorithm does not satisfy this property.

This chapter develops an algorithm so-called Variational Bayes-Expectation Maximization (Vb-EM). It should be pointed out that this is not the first derivation of such algorithm. The first Vb-EM is introduced date back to Bernardo et al. (2003), where they develop Vb-EM algorithm for finite mixture of exponential family models such as scoring discrete directed graphical model structures (Bayesian networks). Our method, however, is derived specifically for state-space MF-VAR model. The difference between our algorithm and Gefang et al. (2020) lies solely on M-step where Vb is derived to approximate VAR parameters conditional on estimates obtained from the E-step. Thus yielding a tighter evidence lower bound to log marginal likelihood. The primary contributions are to provide an alternative algorithm to approximate MF-VAR model that is computationally cheap, more robust relatively to existing literature and as accurate as a more computational demanding method such as MCMC as possible.

We demonstrate robustness and accuracy of the Vb-EM algorithm by using real world macroeconomic data application of pseudo real-time out-of-sample forecasts, compare against MCMC method and possibly EM-algorithm. Out-of-sample forecasting exercises of eleven US macroeconomic variables are evaluated to examine the robustness and accuracy. The data transformation and the series are identical to Schorfheide & Song (2015). To control the tightness of VAR coefficients, we employ the Horseshoe prior. It is worth noting that our Vb is not based on per equation algorithm. It is Vb for system-wide/reduced-form VAR as described in the previous chapter.

Roadmap for this chapter are sectioned as. To clearly show the distinction between our proposed Vb-EM algorithm, we first demonstrate how EM-algorithm is derived in section 2.2. Then section 2.3 offers details of proposed Vb-EM method. Section 2.4 details MF-VAR model and prior being implemented. Optimal variational parameters is presented in section 2.5. This section is quite similar to the previous chapter with slightly modification on optimal variational parameters that involve using state variables. The approach we employ in this study differs in the sense that it involves integrating out the state variables. Specifically, we express the integral as an expectation with respect to the state variables obtained from the Variational Bayes E-step (Vb:E-step). This integration allows us to account for the uncertainty associated with the state variables and incorporate their influence on the overall model estimation and inference. Next is section 2.6, where it shows real-time nowcasting performance, compare against MF-VAR using Gibbs sampling method, as well as real-world data applications to assess the robustness of Vb-EM in comparison to the literature. Finally the derivation of evidence lower bound to the log marginal likelihood are given in section 2.10.

2.2 EM-algorithm

Before we move to explaining how Vb-EM algorithm works for MF-VAR, we first briefly introduce the EM-algorithm, and some useful notations and symbols. To begin with $\mathbf{y}$ is the observed variables, which apparently not entirely observed since we mixed between two different frequency of data. Next is $\mathbf{z}$ is hidden/state variables, the role of state variables is to help us impute the missing observation in $\mathbf{y}$ so that one can derive the global parameters, which we denoted by θ.

Expectation-maximization (EM) algorithm is a computational method used to estimate the parameters of statistical models when some of the data is missing or incomplete, which is precisely the case in the MF-VAR where we are estimating with multiple different frequencies. It is important to understand the derivation of the EM algorithm to distinguish it from our proposed Variational Bayes-Expectation Maximization (Vb-EM) algorithm. EM algorithm or expectation maximization aims to find the maximum likelihood estimates in a model. The goal of EM-algorithm is to maximize the following, Dempster et al. (1977):

$$\widehat{\theta}_{ML} = \arg\min_{\theta} p(\mathbf{y}|\theta). \tag{2.1}$$

However the observed data $\mathbf{y}$ in eq. (2.1) is not entirely observed. Thus in most problems where implementing EM-algorithm benefits researcher is when they want to fill those missing observation with latent variable $\mathbf{z}$. Finding a bridge to connect the model to incomplete likelihood is the key to overcoming this difficulty. This is why EM-algorithm is an iterative algorithm that alternates between two steps: the E-step, which computes the expected values of the missing data given the current estimates of the global parameters, and after those missing observations in $\mathbf{y}$ is filled/imputed, then the M-step, which updates the estimates of the global parameters given the expected values of the missing data.

To understand the whole story of EM-algorithm, we first simply derive the evidence lower bound (shortly labelled as ELBO) to the log-incomplete-likelihood, denoted by $\log p(\mathbf{y}|\theta)$. It is worth noting that this log-incomplete-likelihood is not to be confused with the incomplete likelihood in other econometric literature:

$$\begin{aligned}
\log p(\mathbf{y}|\theta) &= \int q(\mathbf{z}) \log p(\mathbf{y}|\theta) d\mathbf{z}, \\
&= \int q(\mathbf{z}) \log \left(\frac{p(\mathbf{y}|\theta) p(\mathbf{z}|\mathbf{y};\theta)}{p(\mathbf{z}|\mathbf{y};\theta)} \right) d\mathbf{z}, \\
&= \int q(\mathbf{z}) \log \left(\frac{p(\mathbf{y},\mathbf{z}|\theta)}{p(\mathbf{z}|\mathbf{y};\theta)} \right) d\mathbf{z}, \\
&= \int q(\mathbf{z}) \log \left(\frac{p(\mathbf{y},\mathbf{z}|\theta) q(\mathbf{z})}{p(\mathbf{z}|\mathbf{y};\theta) q(\mathbf{z})} \right) d\mathbf{z}, \\
&= \int q(\mathbf{z}) \log \left(\frac{p(\mathbf{y},\mathbf{z}|\theta)}{q(\mathbf{z})} \right) d\mathbf{z} - \int q(\mathbf{z}) \log \left(\frac{p(\mathbf{z}|\mathbf{y};\theta)}{q(\mathbf{z})} \right) d\mathbf{z}, \\
&= \mathcal{L}(q(\mathbf{z}),\theta) + KL(q||p).
\end{aligned} \tag{2.2}$$

Let $q(\mathbf{z})$ is approximate density for latent parameters (at least for now).[1]

First term on the RHS $\mathcal{L}(q(\mathbf{z}),\theta)$ is evidence lower bound on log-incomplete-likelihood $\log p(\mathbf{y}|\theta)$. The second term on RHS $KL(q||p)$ is called Kullback-Leibler divergence (also known as relative entropy), measuring the distance between true hidden variables $p(\mathbf{z}|\mathbf{x};\theta)$ and approximate ones $q(\mathbf{z})$. Most of the time this measurement is intractable. Therefore EM-algorithm optimize evidence lower bound $\mathcal{L}(q(\mathbf{z}),\theta)$ on log-incomplete-likelihood by first apply Jensen's inequality. Since the log-incomplete-likelihood is concave function then we have:

$$\begin{aligned}
\log p(\mathbf{y}|\theta) &\geq \mathbb{E}_q \left[\log \left\{ \frac{p(\mathbf{z}|\theta) p(\mathbf{y}|\mathbf{z};\theta)}{q(\mathbf{z})} \right\} \right], \\
&\geq \mathbb{E}_q \left[\log p(\mathbf{z}|\theta) \right] + \mathbb{E}_q \left[p(\mathbf{y}|\mathbf{z};\theta) \right] - \mathbb{E}_q \left[q(\mathbf{z}) \right], \\
&\geq \mathcal{L}(q(\mathbf{z}),\theta) + KL(q||p).
\end{aligned} \tag{2.3}$$

Jensen's inequality is applied to demonstrate that maximizing ELBO $\mathcal{L}(q(\mathbf{z}),\theta)$ is equivalent to minimizing the intractable term $KL(q||p)$. Now let's assume that we can find

[1] The introduction of the notation $q(\mathbf{z})$ serves to enhance clarity in the subsequent section, where we demonstrate that this approximate distribution can be replaced with the exact posterior distribution using the Kalman filter and smoother recursion algorithm.

By representing the latent variables $\mathbf{z}$ with the distribution $q(\mathbf{z})$, we can effectively approximate the true posterior distribution. This approximation allows us to employ the Kalman filter and smoother recursion algorithm, which enables more accurate inference and estimation.

In the upcoming section, we will delve into the details of this approach and highlight its advantages in capturing the exact posterior distribution of the latent variables $\mathbf{z}$.

posterior of latent parameters conditional on global parameter at iteration $(t-1)$, $p(\mathbf{z}|\mathbf{y};\theta^{(t-1)})$ analytically.[2] Then simply substitute $q(\mathbf{z}) = p(\mathbf{z}|\mathbf{y};\theta^{(t-1)})$ and at the EM optimizing iteration (t), ELBO for EM algorithm becomes:

$$\begin{aligned}
\mathcal{L}(q(\mathbf{z}),\theta) &= \int q(\mathbf{z}) \log\left(\frac{p(\mathbf{y},\mathbf{z};\theta)}{q(\mathbf{z})}\right) d\mathbf{z},\\
&= \int q(\mathbf{z}) \log p(\mathbf{y},\mathbf{z};\theta) d\mathbf{z} - \int q(\mathbf{z}) \log q(\mathbf{z}) d\mathbf{z},\\
&= \int p(\mathbf{z}|\mathbf{y};\theta^{(t-1)}) \log p(\mathbf{y},\mathbf{z};\theta) d\mathbf{z} - \int p(\mathbf{z}|\mathbf{y};\theta^{(t-1)}) \log p(\mathbf{z}|\mathbf{y};\theta^{(t-1)}) d\mathbf{z},\\
&= Q(\theta^{(t-1)}|\theta^{(t-1)}) + H(q).
\end{aligned} \tag{2.4}$$

$H(q)$ is entropy[3] of $\mathbf{z}$ given observed evidence $\mathbf{y}$ and is implicitly function of the parameter of previous iteration $\theta^{(t-1)}$. It should be noted that it is constant with respect to global-parameter in current iteration $\theta^{(t)}$. Therefore it is irrelevant while optimizing for current $\theta^{(t)}$ when proceeding EM-algorithm.

EM proceeds by coordinate descent[4] from iteration (t) until the convergence is detected.

$$\begin{aligned}
&- \text{ E-step :}\quad \text{compute}\quad Q(\theta^{(t-1)}|\theta^{(t-1)}) = q(\mathbf{z})^{(t)} = p(\mathbf{z}|\mathbf{y};\theta^{(t-1)}),\\
&- \text{ M-step :}\quad \text{compute}\quad \theta^{(t)} = \arg\max_{\theta} Q(\theta^{(t-1)}|\theta^{(t-1)}) = \mathbb{E}_{\mathbf{z}|\mathbf{y};\theta^{(t-1)}}\left[\log p(\mathbf{y},\mathbf{z};\theta)\right].
\end{aligned} \tag{2.5}$$

where $\mathbb{E}_{\mathbf{z}|\mathbf{y};\theta^{(t-1)}}\left[\log p(\mathbf{y},\mathbf{z};\theta)\right]$ is expectation of *complete* log likelihood conditional on hidden variable given previous EM iteration of $\theta^{(t-1)}$. It is called complete in a sense

[2] For linear Gaussian state space model for example, this is the forward-backward Kalman filter and smoother.

[3] Entropy of a random variable is the average uncertainty inherent in the variable's possible outcomes. For instance, factorized Gaussian distribution entropy of variational family $H(q)$ is $\frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2}$, where σ^2 is variance.

[4] Coordinate descent can be viewed as parameter blocks in sampling-based method such as Gibbs sampling in Markov Chain Monte Carlo but instead of using sampling technique it is an optimization problem i.e. optimizing one block at a time while holding other block fixed.

that $\mathbf{y}, \mathbf{z}$ contain no missing observations since those missing data are already imputed with state/hidden variable from E-step. The mentioned expectation is crucial since it integrates out the state variables so that the M-step actually maximizes the log-incomplete-likelihood $\int_{\mathbf{z}} \log p(\mathbf{y}, \mathbf{z}; \theta) d\mathbf{z}$. As a result, maximum log likelihood $\log p(\mathbf{y}|\theta)$. Finally it is now obvious and one can notice that EM algorithm seeks to maximize evidence lower bound on log-incomplete-likelihood while variational inference maximizes ELBO on log marginal likelihood $p(\mathbf{y})$, which will be demonstrated in the next section. Thanks to a tractable mean-field variational family that assume each block is factorized, Vb is able to maximize ELBO on log marginal likelihood although $\log p(\mathbf{y})$ is intractable.

2.3 Variational Bayes with missing observations.

Similar to Vb in the previous chapter but now there is additional state/latent variables. The goal of Vb is to maximize evidence lower bound to the log marginal likelihood. From previous chapter we know that the log marginal likelihood can be bounded by the following equation, where now the additional hidden/latent variables $\mathbf{z}$ is added:

$$\log p(\mathbf{y}) \geq \int q(\mathbf{z}, \theta) \log \frac{p(\mathbf{y}, \mathbf{z}, \theta)}{q(\mathbf{z}, \theta)} d\mathbf{z} d\theta. \tag{2.6}$$

where $q(\mathbf{z}, \theta)$ is approximate distribution for state variables $\mathbf{z}$ and global parameters θ. The term on r.h.s is Evidence lower bound (ELBO) to the log marginal likelihood (or log evidence). Some also refer this as functional of the free distributions/free energy, $q(\mathbf{z}, q(\theta))$, where it is denoted:

$$\mathcal{L}(q(\mathbf{z}), q(\theta)) = \int q(\mathbf{z}, \theta) \log \frac{p(\mathbf{y}, \mathbf{z}, \theta)}{q(\mathbf{z}, \theta)} d\mathbf{z} d\theta. \tag{2.7}$$

We denote ELBO as $\mathcal{L}(q(\theta), q(\mathbf{z}))$, where it is a function of those two approximate distributions. The problem here is that the $\mathbf{y}$ is not entirely observed. As a result optimizing ELBO directly is intractable. To resolve the issue we iteratively maximize ELBO $\mathcal{L}$ above with respect to free distribution (approximate distribution) $q(\theta)$ and

$q(\mathbf{z})$ using variational calculus. To allow such algorithm to succeed one need to define the approximate distribution. Instead of estimate exact posterior distribution $p(\mathbf{z}, \theta|\mathbf{y})$, variational Bayes simplifies the problem by replacing with a more tractable, convenient factorized approximate distribution to $q(\mathbf{z}, \theta) = q(\theta)q(\mathbf{z})$. Such factorization implies that we assume the *global* and *latent/hidden* variables to be conditionally independent. This results in iterative algorithm directly analogous to typical EM-algorithm. With elementary calculus of variations to take functional derivatives of the evidence lower bound with respect to $q(\mathbf{z})$, and $q(\theta)$, each block while holding other fixed, the evidence lower bound above can be maximized by alternating between these two steps, see Bernardo et al. (2003):

$$\begin{aligned} \widehat{q}(\mathbf{z}^{t+1}) &\propto \exp\Big[\int \log p(\mathbf{z}, \mathbf{y}|\theta)\widehat{q}(\theta^t)d\theta\Big], \\ \widehat{q}(\theta^{(t+1)}) &\propto p(\theta)\exp\Big[\int \log p(\mathbf{z}, \mathbf{y}|\theta)\widehat{q}(\mathbf{z}^{(t+1)})d\mathbf{z}\Big]. \end{aligned} \tag{2.8}$$

where t denotes the number of Vb optimizing iterations. The first equation in eq. (2.8) derived from the partial derivative of ELBO w.r.t to latent variables distribution $\mathbf{z}$, i.e. $\frac{\partial \mathcal{L}}{\partial q(\mathbf{z})} = 0$. Similarly, the second equation in eq. (2.8) (optimal variational parameters for global parameters) is derived from partial derivative of ELBO w.r.t. global distribution and set to zero, $\frac{\partial \mathcal{L}}{\partial q(\theta)} = 0$.

People who are familiar with EM-algorithm Dempster et al. (1977), which already described in section 2.2, will notice that eq. (2.8) have something in common to such algorithm but rather than restricting the global parameters to point estimate (i.e. Dirac delta function) $q(\theta) = \delta(\theta - \theta^{\star})$, Vb algorithm assumes it to be a random variables parameterized by defined parameter density. It is worth noting that Vb algorithm for missing observations has been developed in machine learning literature, mostly for graphical models. For example Attias (1999) named it *EM-like algorithm with free-form optimization*[5], see eq (4)-(5) Attias (1999). Ghahramani & Beal (2000) further extended for in-complete data (data/evidence are not entirely observed), where they called *Variational Bayesian-Expectation Maximization* algorithm. The latter provides

[5] Free-form optimization is currently known as *coordinate descent* optimization.

more on theoretical proofs. Most of the previous work focus on approximating both state variables and global parameters whereas our work focus more on the latter and let the Kalman filter recursion algorithm estimate the state variables. This statement will become clear shortly after.

Alternating between those two equations in eq. (2.8) results in a tighter ELBO. To be more precise, at Vb optimizing iteration $t = 0$, Vb:E-step computes $\widehat{q}(\mathbf{z}^{(1)})$ with randomly initialized global parameters $\widehat{q}(\theta^{(0)})$. Vb:M-step computes $\widehat{q}(\theta^{(1)})$ given estimates produced from the Vb:E-step. These two steps are alternated until converged. For further details on how both approximate distribution for latent variables $q(\mathbf{z})$ and global parameters $q(\theta)$ are derived, interested reader are referred to Beal & Ghahramani (2001).

Next sub-section we provide that how our proposed algorithm is slightly modified and named variational Bayes-expectation maximization.

2.3.1 Variational Bayes-Expectation Maximization

As mentioned earlier, the Vb-EM algorithm has been previously developed for graphical models in the field of machine learning. However, in this study, we adapt and modify the Vb-EM algorithm specifically for the MF-VAR model. More specifically, we modify the structure of the approximate distribution, assuming it to be factorized as:

$$q(\mathbf{z}, \theta) = q(\theta) p(\mathbf{z}|\mathbf{y}, \theta). \tag{2.9}$$

This modification allows us to handle the complexities of the MF-VAR model effectively.

The factorized approximate distribution described above implies that we approximate only the global parameters (θ), while the state variables are computed from the exact posterior distribution conditioned on the global parameters and observed data. This exact posterior distribution is known as the Kalman filter and smoother recursion, which provides optimal estimates of the state/latent variables based on the current state of the global parameters. These global parameters in the MF-VAR model refer to the VAR coefficients and covariance, which are utilized in the state transition matrix and covariance in the state equation of the normal linear state-space model.

The use of exact posterior for state variables is motivated by the aim to maintain consistency of the state variables. Wang & Titterington (2004) demonstrate that the posterior of the state variables obtained from mean-field and variational Bayes approximation is mostly inconsistent, it is consistent under certain conditions, such as sufficiently small variances of the noise variables in the state equation. The consistency of the approximation also may improve as the sample size increases. However, since macroeconomic data typically has a moderate number of observations, the problem of inconsistency can arise. Another relevant work by Frazier et al. (2021) shows that in certain settings, the discrepancies in predictive performance can become significant over longer out-of-sample periods due to inaccuracies in the state variables. Therefore, to address these issues and ensure reliable results, we rely on the use of the exact posterior obtained through the Kalman filter and smoother algorithm.

Now replacing approximate distribution above to ELBO in eq. (2.7), the log marginal likelihood and ELBO becomes:

$$\log p(\mathbf{y}) \geq \int q(\theta)p(\mathbf{z}|\mathbf{y},\theta) log \frac{p(\mathbf{y},\mathbf{z},\theta)}{q(\theta)p(\mathbf{z}|\mathbf{y},\theta)} d\mathbf{z}d\theta \tag{2.10}$$

Maximizing the ELBO w.r.t the factorized approximate distribution $q(\theta)p(\mathbf{z}|\mathbf{y},\theta)$ results in $q(\theta)p(\mathbf{z}|\mathbf{y},\theta) = p(\mathbf{z},\theta|\mathbf{y})$ (if KL is 0 or else $q(\theta)p(\mathbf{z}|\mathbf{y},\theta) \approx p(\mathbf{z},\theta|\mathbf{y})$). With this approximate distribution, the ELBO can be maximized by alternating between the following equations.

$$\begin{aligned} &- \text{ Vb-E-step: } \; p(\mathbf{z}^{(t+1)}|\mathbf{y},\theta^{(t)}) \; \text{ via Kalman filter and Smoother recursion,} \\ &- \text{ Vb-M-step: } \; \widehat{q}(\theta^{(t+1)}) \propto p(\theta) \exp\left[\int \log p(\mathbf{z},\mathbf{y}|\theta^{(t)}) p(\mathbf{z}^{(t+1)}|\mathbf{y},\theta^{(t)}) d\mathbf{z}\right] \end{aligned} \tag{2.11}$$

Where the first equation above is obtained via Kalman filter recursion given the current global parameters $\theta^{(t)}$. For those who are familiar with EM-algorithm, it is noticeable that the Vb:E-step corresponds to the general E-step of EM-algorithm. However, the distinction between Vb-EM and the EM-algorithm lies in the second step/equation. The second equation represents the optimal global parameters conditional on current iteration of state latent variables. The derivation above is similar to Bernardo et al. (2003) eq.(6)-(7), except that in the Vb:E-step, we compute the exact conditional posterior of the state latent variables using the Kalman filter recursion, rather than using an approximate distribution. In our Vb-EM algorithm for the MF-VAR model, the approximate distribution is used only for the global parameters $q(\theta)$. Now it becomes evident why the proposed algorithm is named Vb-EM. We borrow the E-step from the EM-algorithm and modify the maximization steps accordingly. Instead of maximize the log likelihood $\log p(\mathbf{y}|\theta)$, we maximize the ELBO to log marginal likelihood. The mean-field approximate distribution can be straightforwardly employed where each block j of global parameters is factorized, i.e. $q(\theta_j) = \prod_j q(\theta_j)$. Surprisingly this is the only assumption required to implement the coordinate descent optimisation[6]. Thus the optimal global parameters can be expressed:

$$\widehat{q}(\theta_j^{(t+1)}) \propto p(\theta) \exp\left\{\mathbb{E}_{-\theta_j}\left[\int \log p(\mathbf{z}, \mathbf{y}|\theta) p(\mathbf{z}^{(t+1)}|\mathbf{y}, \theta^{(t)}) d\mathbf{z}\right]\right\} \tag{2.12}$$

where $\mathbb{E}_{-\theta_j}$ denotes expectation with respect to approximate distribution of the rest of optimal global parameters apart from block-j. Notice further that the expression above is similar to eq. (1.2) in previous chapter but now there is an integral of state variables. Such integral of state latent variable are cumbersome to analytically obtained but fortunately it can be expressed in expectation with respect to state latent variables:

$$\widehat{q}(\theta_j^{(t+1)}) \propto p(\theta) \exp\left\{\mathbb{E}_{-\theta_j}\left[\mathbb{E}_{\mathbf{z}}\left\{\log p(\mathbf{z}, \mathbf{y}|\theta_j)\right\}\right]\right\} \tag{2.13}$$

Expressing the integral as an expectation will aid in deriving the optimal parameters for global parameters $\widehat{q}(\theta)$[7].

[6] Coordinate descent optimization: it is an optimizing method to maximize function with respect to each variational parameter one at a time while holding other fixed. There is a very close relationship between conditional posterior from Gibbs sampling and this method, see Blei et al. (2017) for more details.

[7] Here the optimal variational parameters refers to the best parameters parameterizing the approximation distribution. For instance if approximation distribution is Gaussian then variational parameters are mean $\mathbf{m}$, and covariance $\mathbf{V}$, $\widehat{q}(\theta) \sim \mathcal{N}(\mathbf{m}, \mathbf{V})$.

2.3.2 Variational predictive distribution

For Vb-EM algorithm to be practically applicable for forecasting, the generating procedures should be conditioning on history of observed data. While multi-step forecasts are also possible, our primary focus is on real-time one-step-ahead forecast.

Let y_{T+1} be future value of interested variable. Given the observed data up to time T i.e. $y_{1:T}$, one-step ahead forecast is fully depicted by the conditional predictive distribution.

$$\begin{aligned} p(y_{T+1}|y_{1:T}) &= \int p(y_{T+1}, z_{1:T+1}, \theta|y_{1:T}) d\theta dz_{1:T+1}, \\ &= \int p(y_{T+1}|z_{T+1}, y_{1:T}, \theta) p(z_{T+1}|z_T) p(z_{1:T}, \theta|y_{1:T}) d\theta dz_{1:T+1}, \\ &= \int p(y_{T+1}|z_{T+1}, y_{1:T}, \theta) p(z_{T+1}|z_T) q(\theta) p(z_{1:T}|y_{1:T}, \theta) d\theta dz_{1:T+1}. \end{aligned}$$

The exact posterior $p(z_{1:T}, \theta|y_{1:T})$ is approximated by Vb-EM algorithm and will be replaced with $\approx q(\theta)p(z_{1:T}|y_{1:T}, \theta)$. The predictive distribution is then approximated as:

$$\widehat{q}(y_{T+1}|y_{1:T}) \approx \frac{1}{S}\sum_{s=1}^{S} p(y_{T+1}|z_{T+1}^{(s)}, y_{1:T}, \theta^{(s)}). \tag{2.14}$$

It should be noted we are able to evaluate predictive forecasts from MCMC and Vb-EM algorithm only, see for MCMC Ankargren & Jonéus (2020). Those forecast produced by EM algorithm is point estimates. Although linear state space model can simulate smoothed forecasts using method so-called *Simulation Smoothing*, see Durbin & Koopman (2012). Forecasts from such approach involves generating forecast conditional on variations of state and measurement covariance but the uncertainty of transition matrix (VAR-coefficients) are not considered.

2.4 Mixed Frequency VAR

Let N, T be the number of equation in VAR, and total number of observation, respectively. And p be the number of lag in VAR. The conventional VAR form with p lags predictors is written as Schorfheide & Song (2015), Ankargren & Yang (2019):

$$Y_t = B_1 Y_{t-1} + ... + B_p Y_{t-p} + \epsilon_t, \; \epsilon_t \sim \mathcal{N}(0, \Sigma^{-1}) \tag{2.15}$$

where Y_t is responsive vector with N dimension. $Y_{t-1}, .., Y_{t-p}$ are Np vector of covariates at time t. The vector Y_t can be decomposed into monthly and quarterly variables as $Y_t = (Y'_{m,t}, Y'_{q,t})'$ of $N_m \times 1$ being number of monthly frequency variables, and $N_q \times 1$ being number of quarterly frequency variables, and $N = N_m + N_q$. Notice that there are missing $Y_{q,t}$ in Y_t which has never been observed in monthly frequency. As a result estimating VAR in eq. (2.15) directly is not possible (at least not until those missing observations are filled). To deal with those missing observations, researchers assume that the missing $Y_{q,t}$ follows random walk process. Popular algorithm to impute those missing observations is linear Gaussian state space model where the recursion of Kalman filter and Smoother algorithm (KFS) are employed. KFS contains two main equations, which are measurement and state equations. To comprehend how KFS can be implemented in conjunction with VAR model, first denote $z_t = (Y'_t, Y'_{t-1}, ..., Y'_{t-p+1})$, and $z_{t-1} = (Y'_{t-1}, Y'_{t-2}, ..., Y'_{t-p})$, one can write VAR($p$) process in state equation form as followed:

$$\begin{aligned} z_t &= F(B) z_{t-1} + v_t, \; v_t \sim \mathcal{N}(0, \Omega(\Sigma^{-1})), \\ z_t &= \begin{pmatrix} B_1 & B_2 & \cdots & B_p \\ I_{N(p-1)} & & & 0_{N(p-1)\times N} \end{pmatrix} z_{t-1} + \begin{pmatrix} v_t \\ 0_{N(p-1)\times 1} \end{pmatrix}. \end{aligned} \tag{2.16}$$

where $F(B)$ is apparently a state transition, which partially contain VAR coefficients so that the state equation is equivalently formulated as VAR model eq. (2.15). Both $F(B)$ and $\Omega(\Sigma^{-1})$ are the corresponding companion form matrices to produce similar results to eq. (2.15). Equation (2.16) is state equation in state space model. The state variables z_t is then projected through the measurement equation which is given by:

$$Y_t = M_t \Lambda z_t. \tag{2.17}$$

where (again) $Y_t = \left(Y'_{m,t}, Y'_{q,t}\right)$, and M_t is a time-varying selection matrix selecting which variable in which at time t observation in Y_t is observed. The measurement equation above, with the state equation in eq. (2.16) can now be seen as a Normal linear state space model and text book methods exist involving the Kalman filter and smoother for its estimation can be straightforwardly implemented, see Durbin & Koopman (2012).

Finally one may notice that the reduced form VAR can also be written as:

$$Y = BX + E, \quad E_t \sim \mathcal{N}(0, \Sigma^{-1}) \tag{2.18}$$

where Y is $N \times (T-1)$ matrix of first N row of posterior mean from state variables z_t, for $t = 2, ..., T$, and X is $K \times (T-1)$ matrix of z_t, for $t = 1, 2, ..., T-1$. Readers might be confused on why Y, X have to be $T-1$ despite the fact that total number of observation is T. The reason behind this is quite simple, during the recursion of KFS involves computing the state-covariance between z_t, and z_{t-1}. Such state-covariance are required to approximate global parameters, in which our MF-VAR case is VAR coefficients, VAR-covariance, and the rest of hierarchical shrinkage parameters which will be defined shortly after. With the state variables in state equation being a random walk process, the state-covariance exists only a lag of itself. Since the number of lag in VAR is typically more than one $p > 1$. Therefore to make VAR(p) process corresponding with available covariance of state variables, (without loss of generality) one can use $t = 2, .., T$ in Y, and $t = 1, .., T-1$ for X. To make it more crystal, if we are dealing with a total number of observations 100, two equations in VAR, and 3 lags i.e. $T = 100, N = 2, p = 3$ and $m = Np$ then we have:

variables	Mathematic dimension notations	Dimensions
$\mathbf{z}$	$(m \times T)$	(6×100)
z_t	$(m \times 1)$	(6×1)
Y	$(N \times T - 1)$	(2×99)
X	$(m \times T - 1)$	(6×99)

Table 2.1: Dimension of state estimators, and Y, X in eq. (2.18).

Now it is obvious that doing so allow us to access the variance and covariance of smoothed state estimators from Vb:E-step which will be crucial to derive the optimal variational parameters in Vb:M-step. Next we define the prior for VAR parameters:

$$\begin{aligned} Y|B,\Sigma &\sim \mathcal{N}(BX, \Sigma^{-1}), \\ B|\Sigma,\lambda,\tau &\sim \mathcal{N}(0, \Sigma^{-1} \otimes (\lambda\tau)^{-1}), \\ \lambda|\vartheta &\sim \mathcal{G}(1/2, \vartheta), \\ \tau|\xi &\sim \mathcal{G}(1/2, \xi), \\ \vartheta &\sim \mathcal{G}(1/2, 1), \\ \xi &\sim \mathcal{G}(1/2, 1), \\ \Sigma &\sim \mathcal{W}(S_0, \nu_0). \end{aligned} \tag{2.19}$$

Similar to the previous chapter this is a conditionally conjugate prior for VAR coeffcient B and VAR covariance Σ (conditional on global and local shrinkage parameters $\lambda\tau$). It is noteworthy (again) that the probability density function of $p(B|\Sigma,\lambda,\tau) \sim \mathcal{N}(0, (\Sigma \otimes \lambda\tau)^{-1})$ takes the form:

$$\begin{aligned} p(B|\Sigma,\lambda,\tau) &= (2\pi)^{\frac{-NK}{2}} |\Sigma|^{\frac{K}{2}} |\lambda\tau|^{\frac{N}{2}} \exp\left(\mathrm{Tr}\left\{-\frac{1}{2}\Sigma B(\lambda\tau)B'\right\}\right), \\ &= (2\pi)^{\frac{-NK}{2}} |\Sigma|^{\frac{K}{2}} |\lambda\tau|^{\frac{N}{2}} \exp\left(\mathrm{Tr}\left\{-\frac{1}{2}(\lambda\tau)B'\Sigma B\right\}\right). \end{aligned}$$

2.5 Optimal variational parameters in Vb:M-step

As mentioned above that our VB:E-step is equivalent to Expectation step from EM-algorithm. Therefore we will proceed presenting the VB:M-step, where interested readers are referred to Durbin & Koopman (2012) for Kalman filter and smoother. The optimal variational parameters for VB-EM algorithm is similar to VB for typical VAR. The only difference that is while the optimal variational parameters are being derived, it is still in the expectation with respect to state variables, see eq. (2.13) for clarity. From previous chapter we know that the optimal variational parameters can be expressed as followed:

$$\begin{aligned}
\widehat{q}(B|\Sigma) &\sim \mathcal{N}_{N\times K}(\widehat{B}, \Sigma^{-1} \otimes \Phi^{-1}),\\
\widehat{q}(\Sigma) &\sim \mathcal{W}(\overline{S}, \overline{\nu}),\\
\widehat{q}(\lambda) &\sim \mathcal{G}(\overline{a}, \overline{b}),\\
\widehat{q}(\tau) &\sim \mathcal{G}(\overline{c}, \overline{d}),\\
\widehat{q}(\vartheta) &\sim \mathcal{G}(\overline{e}, \overline{f}),\\
\widehat{q}(\xi) &\sim \mathcal{G}(\overline{g}, \overline{h}).
\end{aligned} \tag{2.20}$$

where:

$$\begin{aligned}
\widehat{B} &= \mathbb{E}_{Y,X}\left[\Phi^{-1}(XY')\right],\\
\Phi &= \mathbb{E}_X\left[\widehat{\lambda}\widehat{\tau} + XX'\right],\\
\overline{S}^{-1} &= \mathbb{E}_Y\left[S_0 + YY' - \widehat{B}\Phi\widehat{B}'\right],\\
\overline{\nu} &= \nu_0 + T,\\
\overline{a} &= 1/2 + N/2,\\
\overline{b} &= \widehat{\vartheta} + 1/2\widehat{\tau}\mathrm{diag}(\widehat{B}'\widehat{\Sigma}\widehat{B} + N\Phi^{-1}),\\
\overline{c} &= 1/2 + NK/2,\\
\overline{d} &= \widehat{\xi} + 1/2\mathrm{Tr}\left\{\widehat{\lambda}\left(N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B}\right)\right\},\\
\overline{e} &= 1,\\
\overline{f} &= \widehat{\lambda} + 1,\\
\overline{g} &= 1,\\
\overline{h} &= \widehat{\tau} + 1.
\end{aligned} \tag{2.21}$$

where we denote $\widehat{q}(\bullet)$ as the optimal variational distribution of $\bullet$. The $\widehat{\bullet}$ denote the mean of parameters $\bullet$. For example if $X \sim \mathcal{G}(\bullet, \bullet')$ with $\bullet$ as shape, and $\bullet'$ as rate of Gamma distribution, then $\mathbb{E}[X] = \frac{\bullet}{\bullet'} = \widehat{X}$.

Notice that the optimal variational parameters are exactly as described in previous Chapter, with slightly difference in any optimal variational parameters that include state variables must be integrated out. Again such complex integral is intractable but the integral can be expressed in expectational form. To make it crystal, the inverse covariance of VAR coefficient is $\Phi = \mathbb{E}_X[\widehat{\lambda\tau} + XX']$. Taking the expectation with respect to X which is the state variables from Kalman filter and smoother we obtain, see Lemma 12.2 corollary 2.12.2.1:

$$\begin{aligned}\Phi &= \widehat{\lambda\tau} + \mathbb{E}_X[XX'],\\ &= \widehat{\lambda\tau} + \widehat{\mathbf{P}}_{T-1,T-1} + XX'.\end{aligned} \tag{2.22}$$

where if we denote $\widehat{P}_t$ as the covariance of state variables at time t, which obtained during the Kalman filter and smoother recursion. Then $\widehat{\mathbf{P}}_{T-1,T-1} = \sum_{t=2}^{T} \widehat{P}_{t-1}$. In similar fashion the optimal variational mean of VAR coefficient:

$$\begin{aligned}\overline{B} &= \mathbb{E}_{Y,X}[\Phi^{-1}XY'],\\ &= \Phi^{-1}\big[\widehat{\mathbf{P}}_{T-1,T} + XY'\big].\end{aligned} \tag{2.23}$$

where $\widehat{\mathbf{P}}_{T-1,T} = \sum_{t=2}^{T} \widehat{P}_{t-1,t}$ is the state-covariance between time t and $t-1$ for $t = 2, ..., T$.

Finally the optimal variational scale for Wishart distribution:

$$\begin{aligned}\overline{S}^{-1} &= \mathbb{E}_{Y,X}[S_0 + YY' - \widehat{B}\Phi\widehat{B}'],\\ &= [S_0 + (YY' + \widehat{\mathbf{P}}_{T,T}) - \widehat{B}\Phi\widehat{B}'].\end{aligned} \tag{2.24}$$

where $\widehat{\mathbf{P}}_{T,T} = \sum_{t=2}^{T} \widehat{P}_t$. The state and measurement equations as described in eqs. (2.16) and (2.17) are similar to the textbook from Durbin & Koopman (2012). As mentioned above that the state-covariance of $\widehat{\mathbf{P}}_{T-1,T}, \widehat{\mathbf{P}}_{T,T}$ can be obtained during the Kalman filter and smoother recursion, and the full derivation is explained in section (4.7) of Durbin & Koopman (2012). For convenience we also provided the derivation of $\widehat{\mathbf{P}}$ in Appendix C section 2.11. In the derivation section we set $\widehat{P}_{T-1,T} \equiv \widehat{P}'_{T,T-1} \equiv \mathbf{J}_T$ for notational clarity. It is important to note that if we ignore integral of state variable generated from Kalman filter and smoother, our proposed algorithm is equivalent to the method described in Gefang et al. (2020). However, in the following section, we demonstrate that this choice results in a ELBO to the log marginal likelihood that poorly converges.

2.6 Empirical applications

The goal of this work is to introduce an algorithm for approximating MF-VAR parameters by comparing Vb-EM and a more computational demanding MCMC methods, with a focus on measuring their performance in pseudo real-time out-of-sample forecasting, both point forecasts and density forecasts.[8] The study uses eleven macroeconomic series which is similar to the pioneered paper of MF-VAR model Schorfheide & Song (2015), see section 2.9 for more details. Computational time is also detailed in a sub-section. The method for evaluating predictive forecasts is also outlined in a sub-section.

2.6.1 Predictive density evaluation scores

MCMC and variational methods are Bayesian inference. In order to evaluate predictive performance, we provide scores for both point-forecast and predictive distribution accuracy. The point-forecasts is measured by Root Mean Square Error (RMSE) where it is formulated as:

$$\text{RMSE} = \sqrt{\frac{\sum_i \sum_t (y_{i,t} - \widehat{y}_{i,t})^2}{T - T_0 + 1}}. \tag{2.25}$$

$y_{i,t}$ is realizes, where subscription t denotes times, i is variable column index in vector $y_{i,t}$, i.e. GDPC1, FPIC1, GCEC1, UNRATE, AWHI, CPIAUCSL, INDPRO, PCEC96, FEDFUNDS, GS10, and SP500. T, T_0 is total number of forecast evaluation periods and number of observation in the first loop of forecast, respectively.

Second measure is to investigate how accurate density forecast is. Some also refer this to *Proper scoring rules.* It is designed for evaluating probabilistic forecasts. Scoring metric in this work focuses on the Continuous Ranked Probability Score (CRPS), see Gneiting & Raftery (2007), Gneiting et al. (2007), Gneiting & Ranjan (2011).

[8] It should be noted we are able to evaluate predictive forecasts from MCMC and Vb-EM algorithm only. Those forecast produced by EM algorithm is point estimates. Although linear state space model can simulate smoothed forecasts using method so-called *Simulation Smoothing*, see Durbin & Koopman (2012). Those forecast involves generating forecast conditional on variations of state and measurement covariance but the uncertainty of transition matrix is not included.

Suppose y is realizes, and f is density forecast. Denote F as cumulative density function (CDF) associated with the density f. One can write $F^{-1}(q)$ for quantile at level $q \in (0,1)$. Continuous ranked probability score can be equally defined in three different formulas:

$$\begin{aligned} \mathrm{CRPS}(f,y) &= \mathbb{E}_F|Y-y| - \frac{1}{2}\mathbb{E}_F|Y-Y'|, && (2.26)\\ &= \int_{\infty}^{\infty} \left(F(z) - \mathbb{I}\{y \leq z\}\right)^2 dz, && (2.27)\\ &= 2\int_0^1 \left(\mathbb{I}\{y < F^{-1}(q)\} - q\right)\left(F^{-1}(q) - y\right) dq. && (2.28) \end{aligned}$$

where Y, Y' are independent random variable with distribution function F. $\mathbb{I}(\cdot)$ is Dirac delta function. Equation (2.27) is cumulative ranked probability scores (CRPS) or Brier scores, and eq. (2.28) is quantile scores. First score to measure how accurate predictive forecast is called threshold weighted version of CRPS to which sometimes we refer as the threshold decomposition of CRPS, is introduced by Gneiting et al. (2007) which is equivalent eq. (2.27) with a small extension:

$$\mathrm{S}(f,y) = \int_{-\infty}^{\infty} \left(F(z) - \mathbb{I}\left\{y \leq z\right\}\right)^2 u(z)dz, \tag{2.29}$$

where u is Borel measure of positive weight function on real line. Gneiting & Raftery (2007) stated that this is augmented to encourage forecasters to be able to concentrate to specific area of predictive interests. Although we found that threshold decomposition CRPS is not much different from Brier scores but it is worth mentioning.

In addition to uniform CRPS and threshold decomposition of CRPS, quantile weighted of continuous ranked probability score is measured.

$$QS_{\pi}(q,y) = (y-q)(\pi - \mathbb{I}\left\{y \leq q\right\}), \tag{2.30}$$

where q is quantile forecast, and π is selected quantile.

$$S(f,y) = \frac{1}{J-1}\sum_{j=1}^{J-1} v(\pi_j) QS_{\pi_j}(q,y). \tag{2.31}$$

where $\pi_j = j/J$. In this work we implemented 19 of $\pi \in .05, .1, ..., .95$. The quantile score can also be extended to quantile weighted scores emphasizing specific region of forecast density. Table 2.2 shows weighted implemented.

Emphasis	**Quantile Weight**
uniform	$v(\pi) = 1$
centre	$v(\pi) = \pi(1 - \pi)$
tails	$v(\pi) = (2\pi - 1)^2$
right tail	$v(\pi) = \pi^2$
left tail	$v(\pi) = (1 - \pi)^2$

Table 2.2: Quantile weights.

To begin with the convergence of proposed Vb-EM algorithm, as mention in previous section that the objective of any Vb method is to minimize the distance between approximate distribution and the true one. To minimize KL or maximize evidence lower bound, in other words. I first prove to readers that the proposed algorithm leads to a tighter evidence lower bound[9] relative to existing literature according to fig. 2.1. The lower bound to log marginal likelihood increases accordingly to the number of Vb-EM algorithm iterations. The literature method, however, has looser evidence lower bound.

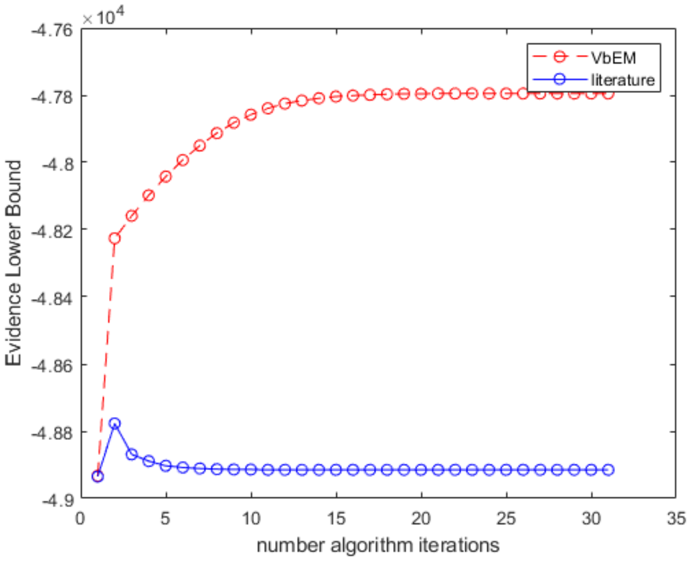


Figure 2.1: Evidence lower bound to log marginal likelihood, of Vb-EM algorithm (red-dashed-dot), and literature algorithm (blue-solid-dot), see eq. (2.35) for full derivations

[9] Derivation of an evidence lower bound to log marginal likelihood can be found in section 2.10.

2.6.2 Pseudo out-of-sample real-time forecasts

Forecast evaluating periods begins at 2000M1 through 2019M12. Assuming that each forecasts are proceed at the end of each month. We first illustrate nowcast results in tables 2.3 and 2.4. The first column, **M/Q** denotes which month within a quarter the nowcast is performed. For example **M 1/Q** is first month of a quarter, implying that the RMSE is evaluated in January, April, July and October. **(+0),(+1),(+2)** after **M/Q** is number indicating advantages of having additional already observed monthly observation within the quarter when the nowcasts are constructed. For instance, nowcast of January **M 1/Q (+0)** takes place at 31-January where forecaster get access to the first release of last December monthly observations. Thus no additional information from the forecast in first quarter at the end of January. **(+1)**, on the other hand, implies that forecaster has access first month within a first quarter.

We first present the nowcasts of quarterly variables in tables 2.3 and 2.4. This table evaluates the nowcast results each month within quarter. According to both RMSE and CRPS scores, all algorithms are indistinguishable.

The rest of monthly variable forecasts are presented in table 2.5. The table also includes RMSE, CRPS measurements during forecast evaluating periods plus relative RMSE, CRPS of proposed VBEM and other methods. Although we expect a higher forecast error, coming from approximation error by the nature of mean-field assumption from Vb-EM algorithm, the point-forecasts are almost identical across all macroeconomic series. This is surprisingly interesting, which may indicates that the correlation between global parameters in MF-VAR is low. As a result mean-field assumption does not affect the accuracy of posterior distribution. Predictive distribution is thus relatively close to MCMC. It is also important to note that although the relative of CRPS from VBEM to MCMC, particularly for fixed investment where MCMC performs better proportionately approximately 17% but CRPS value are already small. Consequently even if the relative number is large, the actual difference is insignificant.

Moving to quantile weighted of CRPS, the scores emphasizes five regions of forecast density. First is tails, where both-sided tails are primary focused. Second is uniform (no weights augmented), centre (using median/mean as weights), right and left tail. Both methods are highly accurate.

To visualize how close both algorithms produce predictive density, threshold weighted version of CRPS is plotted in figs. 2.2 and 2.3 for quarterly and monthly forecasts, respectively. The interpretation is simple, it visualizes threshold weighted version of CRPS which is again negatively-orientated. Thus the lower the line in figs. 2.2 and 2.3 indicates better forecasts. For example MCMC executes slighly better in centre region forecasting FPIC1, and GCEC1. VBEM, on the other hand, is more well founded in centre region in predicting GDPC1 for some unknown reasons. The integrated area under the curve of figs. 2.2 and 2.3 are presented in table 2.5. There is no evidence of any difference in forecasting monthly variables, indicating that the approximation errors from proposed algorithm is small. Finally we conclude that all three algorithms produce quite similar predictive distributions.

Measurement	**RMSE**								
Macro-series	**GDPC1**			**FPIC1**			**GCEC1**		
algorithm	*VBEM*	*EM*	*MCMC*	*VBEM*	*EM*	*MCMC*	*VBEM*	*EM*	*MCMC*
M 1/Q (+0)	0.023	0.023	0.025	0.039	0.039	0.038	0.031	0.031	0.034
M 2/Q (+1)	0.023	0.023	0.024	0.039	0.04	0.038	0.032	0.031	0.034
M 3/Q (+2)	0.022	0.022	0.023	0.039	0.039	0.038	0.032	0.032	0.034

Table 2.3: RMSE of nowcasting each month within quarter.

Measurement	**CRPS**								
Macro-series	**GDPC1**			**FPIC1**			**GCEC1**		
algorithm	*VBEM*	*EM*	*MCMC*	*VBEM*	*EM*	*MCMC*	*VBEM*	*EM*	*MCMC*
M 1/Q (+0)	0.007	-	0.009	0.017	-	0.016	0.012	-	0.012
M 2/Q (+1)	0.007	-	0.008	0.018	-	0.015	0.012	-	0.011
M 3/Q (+2)	0.007	-	0.008	0.021	-	0.016	0.013	-	0.011

Table 2.4: CRPS of nowcasting each month within quarter.

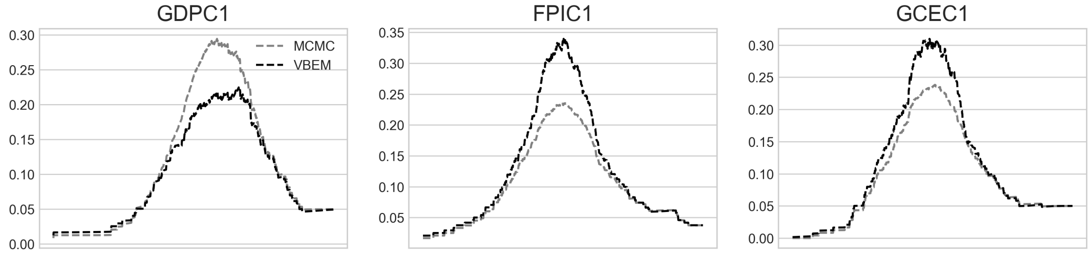


Figure 2.2: Threshold decomposition mean of CRPS for quarterly one-step-ahead forecast density for entire evaluation period, MCMC (dash-grey) and Vb-EM(dash-black). The area under the curve is presented as CRPS in table 2.5.

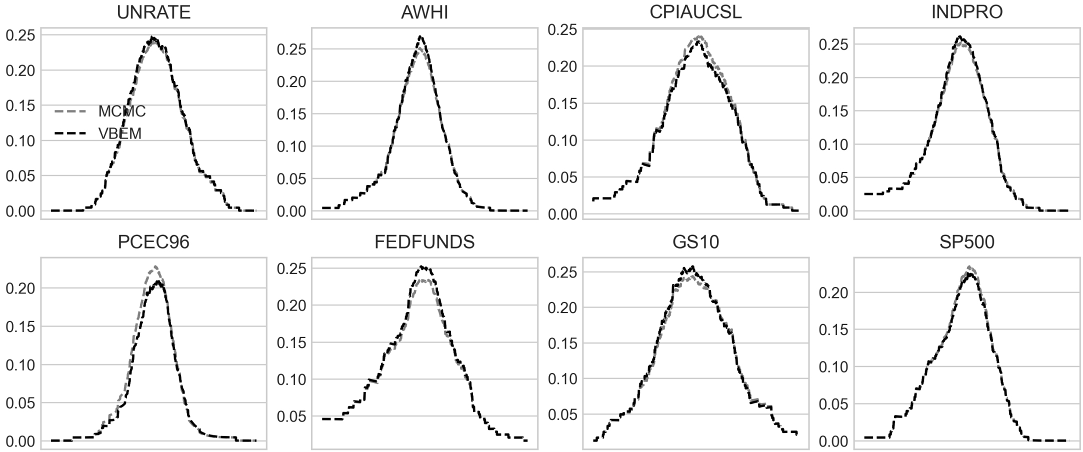


Figure 2.3: Threshold decomposition mean of CRPS for monthly variable forecast density, MCMC (dash-grey) and Vb-EM(dash-black).

Measure	RMSE			CRPS			RMSE		CRPS
Algorithms	*VBEM*	*EM*	*MCMC*	*VBEM*	*MCMC*	*EM*	*VBEM/MCMC*	*VBEM/EM*	*VBEM/MCMC*
GDPC1	0.0228	0.0226	0.0241	0.008	0.008	-	0.95	1.01	0.94
FPIC1	0.0389	0.0393	0.0379	0.019	0.016	-	1.03	0.99	1.17
GCEC1	0.0318	0.0314	0.0339	0.013	0.011	-	0.94	1.01	1.11
UNRATE	0.0003	0.0003	0.0003	0.000	0.000	-	1.03	0.97	1.03
AWHI	0.0261	0.0261	0.0261	0.004	0.004	-	1.00	1.00	1.01
CPIAUCSL	0.0031	0.0032	0.0033	0.002	0.002	-	0.94	0.98	0.96
INDPRO	0.0186	0.0188	0.0186	0.006	0.005	-	1.00	0.99	1.02
PCEC96	0.0037	0.0041	0.0038	0.002	0.002	-	0.96	0.90	0.91
FEDFUNDS	0.0014	0.0014	0.0013	0.001	0.001	-	1.01	1.00	1.04
GS10	0.0007	0.0007	0.0007	0.000	0.000	-	1.00	0.98	1.01
SP500	0.0435	0.0491	0.0446	0.024	0.024	-	0.98	0.89	0.98

Table 2.5: RMSE, and CRPS over forecasting periods. The last three columns are RMSE and CRPS of Vb-EM relative to MCMC and EM-algorithm.

Emphasis	**tails**	**uniform**	**centre**	**right**	**left**
GDPC1	0.99	0.93	0.91	0.98	0.89
FPIC1	1.27	1.16	1.12	1.12	1.26
GCEC1	1.19	1.10	1.07	1.06	1.21
UNRATE	1.04	1.03	1.03	1.03	1.04
AWHI	1.00	1.01	1.02	0.99	1.02
CPIAUCSL	0.97	0.95	0.95	0.95	0.96
INDPRO	1.02	1.02	1.01	1.00	1.03
PCEC96	0.86	0.91	0.93	0.92	0.89
FEDFUNDS	1.05	1.04	1.03	1.04	1.03
GS10	1.04	1.01	1.00	1.01	1.03
SP500	0.98	0.98	0.98	0.97	0.98

Table 2.6: Quantile weighted mean of CRPS of Vb-EM algorithm relative to MCMC (one-step-ahead) predictive forecast of eleven macroeconomic series.

2.7 Summary of different algorithms to estimate MF-VAR.

This section offers the differences between approximate algorithm for MF-VAR (apart from Gibbs sampling). First is proposed Vb-EM algorithm where the goal is to maximize evidence lower bound to the log marginal likelihood. Second is EM-algorithm from Brave et al. (2020) where E-step is similar to our algorithm. M-step, on the other hand, employ modified-OLS which is performing OLS estimates for VAR inside the expectation with respect to state estimators from E-step. Finally the literature's Vb algorithm for MF-VAR from Gefang et al. (2020) (labelled Vb-DEM hereafter). The summarization of all three algorithms are provided in tables 2.7 to 2.9.

It is important to note that the Vb-DEM algorithm performs per equation algorithm when approximating VAR parameters, see Carriero et al. (2019) for per equation algorithm and Gefang et al. (2020) for details of Vb-DEM approach. In table 2.8 we denote θ_i where subscription i is parameters of equation i in VAR. Since the per equation is implemented, we also make clear of the notation of each responsive variables in VAR i.e. $\widetilde{\mathbf{y}}_i$ is each responsive variable in each VAR equation. This is the VAR is written in structural-form thus the $\widetilde{\mathbf{y}}_i$ contains additional parameters arising from the lower-triangular matrix A. Finally the $\sigma^2 = \text{diag}(\sigma_1^2, ..., \sigma_N^2)$ is the diagonal of residual-variances from all equations.

There are two primary distinctions of our Vb-EM algorithm in comparison to the literature (apart from using equation). Firstly literature's Vb-DEM did not perform expectation with respect to estimates produced from E-step. Consequently the evidence lower bound never reach its local maximum. As plotted in fig. 2.1, it visualizes ELBO of proposed Vb-EM algorithm and literature Vb-DEM. Such empirical evidence suggests that Vb-EM algorithm is able to produce tighter lower bound to the log marginal likelihood as optimizing iteration proceeds. Vb-DEM, however, converges to some particular number which is obviously not optimal.

Variational Bayes-Expectation Maximization algorithm for MF-VAR

```
Inputs:
    observed evidence $\mathbf{y}$,
Initialize:
    - initialized variational parameters for any of $\theta = B, \Sigma, \lambda, \tau, \vartheta, \xi$
    - set $t = 0$
While not converged do:
  Vb:E-Step:
    - $p(\mathbf{z}^{(t+1)}|\mathbf{y}, \theta^{(t)})$ via Kalman Filter recursion.
  Vb:M-Step:
    $\widehat{q}(\theta_j^{(t+1)}) \propto p(\theta_j) \exp \left\{ \mathbb{E}_{-\theta_j} \left[ \mathbb{E}_{\mathbf{z}} \left\{ \log p(\mathbf{z}, \mathbf{y}|\theta_j) \right\} \right] \right\}.$
    set $t = t + 1$.
end while.
Outputs: optimal approximate distribution of global parameters $q(\theta)$ and state variables $\mathbf{z}$.
```

Table 2.7: Variational Bayes-Expectation Maximization algorithm for MF-VAR.

Literature algorithm for MF-VAR from Gefang et al. (2020)

```
Inputs:
  observed evidence $\mathbf{y}$,
Initialize:
  - initialized variational parameters for any of $\theta_i$ in VAR eq no.$i = 1, 2, ...N$,
      which are $B_i, \Sigma = A^{-1}\sigma^2 A^{-1'}, \lambda_i, \tau_i, \vartheta_i, \xi_i$
  - set $t = 0$
While not converged do:
    VB-D:E-Step:
        - $p(\mathbf{z}^{(t+1)}|\mathbf{y}, \theta^{(t)})$ via Kalman Filter recursion.
    VB-D:M-Step:
        for $i = 1 : N$
            $\widehat{q}(\theta_{i,j}^{(t+1)}) \propto p(\theta_{i,j}) \exp\left\{\mathbb{E}_{-\theta_{i,j}}\left[\log p(\mathbf{z}_i, \widetilde{\mathbf{y}}_i|\theta_{i,j})\right]\right\}.$
        end for $i$
    set $t = t + 1$.
end while.
Outputs: optimal approximate distribution of global parameters $q(\theta)$ and state variables $\mathbf{z}$.
```

Table 2.8: Literature variational Bayesian (Vb) algorithm for MF-VAR.

EM-algorithm for MF-VAR from Brave et al. (2020)

```
Inputs:
    observed evidence $\mathbf{y}$,
Initialize:
    - initialized global parameters $\theta = B, \Sigma$
    - set $t = 0$
While not converged do:
  E-Step:
    - $p(\mathbf{z}^{(t+1)}|\mathbf{y}, \theta^{(t)})$ via Kalman Filter recursion.
  M-Step:
    $\theta_{MLE}^{(t+1)} = \arg\min_\theta \mathbb{E}_\mathbf{z}\big[\log p(\mathbf{z}, \mathbf{y}|\theta^{(t)})\big]$ .
    set $t = t + 1$.
end while.
Outputs: Maximum likelihood of (point) global parameters $\theta_{MLE}$ and state variables $\mathbf{z}$.
```

Table 2.9: EM-algorithm for MF-VAR.

2.8 Conclusion

In this work we present an alternative algorithm to approximate MF-VAR model, namely Variational-Bayes Expectation Maximization (Vb-EM). The algorithm borrows the idea from EM-algorithm where the algorithm alternates between two steps until the convergence is detected. Vb:E-Step performs exact posterior distribution via the Kalman filter and smoother recursion. Vb:M-step where the global parameters in MF-VAR is approximated via the variational Bayes. This approach is novel in a sense that the approximation in second step contains complex integrals which is impossible to derive since the state estimators from Vb:E-step is multi-dimensional. Instead we express the integrals in expectation form, which is analytically available during the routine of Kalman filter and smoother recursions. We also derive the evidence lower bound for convergence monitoring, which we illustrate is tighter than existing method for approximating MF-VAR model. The legitimacy of proposed algorithm is evaluated via pseudo real-time out-of-sample forecasts of eleven US macroeconomic series. The empirical result suggests that there is no concrete evidence for Vb-EM algorithm to be outperformed by a more computational demanding MCMC (Gibbs sampling).

2.9 Appendix A: Data

No.	Macroeconomic Series	ALFRED acronyms	Date range	Transformation
1	Gross Domestic Product	GDPC1	1964M1 - 2019M12	log-level
2	Fixed Investment	FPIC1		log-level
3	Government Expenditures	GCEC1		log-level
4	Unemployment Rate	UNRATE		log-level/100
5	Hours Worked	AWHI		log-level
6	Consumer Price Index	CPIAUCSL		log-level
7	Industrial Production Index	INDPRO		log-level
8	Personal Consumption Expenditure	PCEC96		log-level
9	Federal Fund Rate	FEDFUNDS		log-level/100
10	Treasury Bond Yield	GS10		log-level/100
11	SP500	SP500		log-level

Table 2.10: Macroeconomic series: All observations are obtained through Federal Reserve Bank of St.Louis via fredapi (Python).

2.10 Appendix B: Derivation of evidence lower bound to the log marginal likelihood

The evidence lower bound to the log marginal likelihood is the lower bound to the $p(\mathbf{y})$, see eq. (2.7).

$$\mathcal{L}\left(q(\theta), q(\mathbf{z})\right) = \int q(\mathbf{z}, \theta) \log \frac{p(\mathbf{y}, \mathbf{z}, \theta)}{q(\mathbf{z}, \theta)} d\mathbf{z} d\theta.$$

Replace the approximate distribution $q(\theta, \mathbf{z})$ with $q(\theta)p(\mathbf{z}|\mathbf{y}, \theta)$, the evidence lower bound to the log marginal likelihood, we thus have:

$$\begin{aligned}
\mathcal{L}\left(q(\theta), p(\mathbf{z}|\mathbf{y}, \theta)\right) &= \int q(\theta)p(\mathbf{z}|\mathbf{y}, \theta) \log \frac{p(\mathbf{y}, \mathbf{z}, \theta)}{q(\theta)p(\mathbf{z}|\mathbf{y}, \theta)} d\theta d\mathbf{z}, \\
&= \mathbb{E}_{q(\theta), p(\mathbf{z}|\mathbf{y}, \theta)} \left[\log \frac{p(\mathbf{y}, \mathbf{z}, \theta)}{q(\theta)p(\mathbf{z}|\mathbf{y}, \theta)}\right], \\
&= \mathbb{E}_{q(\theta), p(\mathbf{z}|\mathbf{y}, \theta)} \left[\log p(\mathbf{z}|\mathbf{y}, \theta) + \log p(\mathbf{y}, \theta) - \log q(\theta) - \log p(\mathbf{z}|\mathbf{y}, \theta)\right], \\
&= \mathbb{E}_{q(\theta), p(\mathbf{z}|\mathbf{y}, \theta)} \left[\log p(\mathbf{y}|\theta) + \log p(\theta) - \log q(\theta)\right], \\
&= \int q(\theta) \log p(\mathbf{y}|\theta) d\theta + \int q(\theta) \log \frac{p(\theta)}{q(\theta)} d\theta, \\
&= \mathbb{E}_{q(\theta)} \left[\log p(\mathbf{y}|\theta)\right] - \mathrm{KL}(q(\theta)||p(\theta)).
\end{aligned} \tag{2.32}$$

where $\mathrm{KL}(q(\cdot)||p(\cdot))$ is the Kullback-Leibler divergence (relative entropy), measuring how one distribution $q(\cdot)$ is different from the second distribution $p(\cdot)$. Notice that the KL term measures the distance between the approximate distribution and the prior given to the parameters. To derive such term we first show the joint distribution of approximate distribution and joint prior distribution are formulated:

$$\begin{aligned}
q(\theta) &= q(B, \Sigma)q(\vartheta)q(\tau)q(\vartheta)q(\xi), \\
&= q(B|\Sigma)q(\Sigma)q(\lambda)q(\tau)q(\vartheta)q(\xi). \\
p(\theta) &= p(B|\Sigma, \lambda, \tau)p(\Sigma)p(\lambda|\vartheta)p(\tau|\xi)p(\vartheta)p(\xi).
\end{aligned} \tag{2.33}$$

Replacing two terms of eq. (2.33) into eq. (2.32), suppressing $\mathbb{E}_{q(\theta)}[\cdot]$ with $\mathbb{E}_q[\cdot]$, where it denotes the expectation of $[\cdot]$ with respect to all approximate distribution of global parameters $q(\theta)$ then we have:

$$\begin{aligned}
\mathrm{KL}(q(\theta)||p(\theta)) &= \mathbb{E}_q\left[\log\frac{q(\theta)}{p(\theta)}\right] = \mathbb{E}_q\left[\log q(\theta) - \log p(\theta)\right], \\
&= \mathbb{E}_q\big[\log q(B|\Sigma) + \log q(\Sigma) + \log q(\lambda) + \log q(\tau) + \log q(\vartheta) + \log q(\xi) \\
&\quad - \log p(B|\Sigma,\lambda,\tau) - \log p(\Sigma) - \log p(\lambda|\vartheta) - \log p(\tau|\xi) - \log p(\vartheta) - \log p(\xi)\big], \\
&= \mathbb{E}_q\left[\log\frac{q(B|\Sigma)}{p(B|\Sigma,\lambda,\tau)} + \log\frac{q(\Sigma)}{p(\Sigma)} + \log\frac{q(\lambda)}{p(\lambda|\vartheta)} + \log\frac{q(\tau)}{p(\tau|\xi)} + \log\frac{q(\vartheta)}{p(\vartheta)} + \log\frac{q(\xi)}{p(\xi)}\right], \\
&= \int_\tau d\tau q(\tau)\int_\lambda d\lambda q(\lambda)\int_\Sigma d\Sigma q(\Sigma)\int_B dB q(B|\Sigma)\log\frac{q(B|\Sigma)}{p(B|\Sigma,\lambda,\tau)} \\
&\quad + \int_\Sigma d\Sigma q(\Sigma)\log\frac{q(\Sigma)}{p(\Sigma)} + \int_\vartheta d\vartheta q(\vartheta)\int_\lambda d\lambda q(\lambda)\log\frac{q(\lambda)}{p(\lambda|\vartheta)} \\
&\quad + \int_\xi d\xi q(\xi)\int_\tau d\tau q(\tau)\log\frac{q(\tau)}{p(\tau|\xi)} + \int_\vartheta d\vartheta q(\vartheta)\log\frac{q(\vartheta)}{p(\vartheta)} + \int_\xi q(\xi)\log\frac{q(\xi)}{p(\xi)} \\
&= \mathrm{KL}(q(B|\Sigma)||p(B|\Sigma,\lambda,\tau)) + \mathrm{KL}(q(\Sigma)||p(\Sigma)) + \mathrm{KL}(q(\lambda)||p(\lambda|\vartheta)) \\
&\quad + \mathrm{KL}(q(\tau)||p(\tau|\xi)) + \mathrm{KL}(q(\vartheta)||p(\vartheta)) + \mathrm{KL}(q(\xi)||p(\xi)).
\end{aligned} \tag{2.34}$$

The prior of MF-VAR we are employing is conjugate prior for B and Σ conditional on shrinkage parameters $\lambda\tau$, such prior introduces conditional dependencies between parameters in some terms of KL above. To prove the point we demonstrate the general form of KL measurement between two multivariate normal distribution below. For example let $X_1 \sim \mathcal{N}_1(\mu_1, \Sigma_1)$, and $X_2 \sim \mathcal{N}_2(\mu_2, \Sigma_2)$, where $X_1, X_2 \in \mathbb{R}^k$, $\mu_1, \mu_2 \in \mathbb{R}^k$, and Σ_1, Σ_2 are multivariate normal distribution's covariance, then the $\mathrm{KL}(\mathcal{N}_1||\mathcal{N}_2)$ is then, see proof Duchi (2007):

$$\mathrm{KL}(\mathcal{N}_1||\mathcal{N}_2) = \frac{1}{2}\left\{\mathrm{Tr}\{\Sigma_2^{-1}\Sigma_1\} + (\mu_2 - \mu_1)'\Sigma_2^{-1}(\mu_2 - \mu_1) - k + \log\frac{|\Sigma_2|}{|\Sigma_1|}\right\}.$$

Such general formulation will not apply to $\mathrm{KL}(q(B|\Sigma)||p(B|\Sigma,\lambda,\tau))$,
since $q(B|\Sigma) \sim \mathcal{N}(\widehat{B}, \Sigma^{-1}\otimes V^{-1})$, and $p(B|\Sigma,\lambda,\tau) \sim \mathcal{N}(0, \Sigma^{-1}\otimes(\lambda\tau)^{-1})$. Both probability density function apparently conditionally depends on other parameters. Fortunately such derivation can be derived by changing integral into expectation with respect to the approximate distribution. To proof such statement, we shall begin by

deriving the first term in eq. (2.34). First we write such term as followed:

$$\begin{aligned}
\mathrm{KL}(q(B|\Sigma)||p(B|\Sigma,\lambda,\tau)) &= \mathbb{E}_q\left[\log q(B|\Sigma) - \log p(B|\Sigma,\lambda,\tau)\right], \\
&= \frac{K}{2}\log|\Sigma| + \frac{N}{2}\log|\Phi| - \frac{1}{2}\mathbb{E}_q\left[\mathrm{Tr}\left\{\Sigma(B-\widehat{B})\Phi(B-\widehat{B})'\right\}\right] \\
&\quad - \frac{K}{2}\log|\Sigma| - \frac{NK}{2}\log\tau - \frac{N}{2}\sum_{j=1}^{K}\log\lambda_j + \frac{1}{2}\mathbb{E}_q\left[\mathrm{Tr}\left\{\Sigma B(\lambda\tau)B'\right\}\right], \\
&= \frac{N}{2}\log|\Phi| - \frac{NK}{2}\mathbb{E}_q[\log\tau] - \frac{N}{2}\sum_{j=1}^{j=K}\mathbb{E}_q\left[\log\lambda_j\right] \\
&\quad + \frac{1}{2}\mathrm{Tr}\left\{(\widehat{\lambda}\widehat{\tau})\left[N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B}\right]\right\} - \frac{NK}{2}, \\
&= \frac{N}{2}\log|\Phi| - \frac{NK}{2}\left(1 + \psi(\overline{c}) - \log\overline{d}\right) - \frac{N}{2}\left(\psi(\overline{a}) - \log\overline{b}\right) \\
&\quad + \frac{1}{2}\mathrm{Tr}\left\{(\widehat{\lambda}\widehat{\tau})\left[N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B}\right]\right\}.
\end{aligned}$$

Next is KL of VAR covariance Σ. The simple formulation of KL is applicable, here is provided for convenience, see Bishop (2006).

$$\begin{aligned}
\mathrm{KL}(q(\Sigma)||p(\Sigma)) &= \frac{N+1}{2}\log|\overline{S}| + \frac{1}{2}N(N+1)\log(2) + \log\Gamma_N\left(\frac{\overline{\nu}}{2}\right) \\
&\quad - \left(\frac{\overline{\nu}-N-1}{2}\right)\psi_N\left(\frac{\overline{\nu}}{2}\right) + \frac{\overline{\nu}N}{2}.
\end{aligned}$$

Next is KL of local shrinkage parameters λ. Recall from optimal variational parameters that $\widehat{q}(\lambda) \sim \mathcal{G}(\overline{a},\overline{b})$, where $\overline{a} = 1/2 + N/2$, and $\overline{b} = \widehat{\vartheta} + 1/2\widehat{\tau}\mathrm{diag}(\widehat{B}'\widehat{\Sigma}\widehat{B} + N\Phi^{-1})$.

$$\begin{aligned}
\mathrm{KL}(q(\lambda)||p(\lambda|\vartheta)) &= \overline{a}\log\overline{b} - \log\Gamma(\overline{a}) + (\overline{a}-1)\mathbb{E}_q[\log\lambda] - \widehat{\overline{b}\lambda} \\
&\quad - \frac{1}{2}\mathbb{E}_q[\log\vartheta] + \log\Gamma(1/2) + \frac{1}{2}\mathbb{E}_q[\log\lambda] + \widehat{\vartheta}\widehat{\lambda}, \\
&= \overline{a}\log\overline{b} - \log\Gamma(\overline{a}) + (\overline{a}-1/2)\left[\psi(\overline{a}) - \log\overline{b}\right] - \widehat{\lambda}\left(\overline{b} - \widehat{\vartheta}\right) \\
&\quad - \frac{1}{2}\left(\psi(\overline{e}) - \log\overline{f}\right) + \log\Gamma(1/2).
\end{aligned}$$

Next is KL of global shrinkage parameters τ. Recall from optimal variational parameters that $\widehat{q}(\tau) \sim \mathcal{G}(\overline{c}, \overline{d})$, where $\overline{c} = 1/2{+}NK/2$, and $\overline{d} = \widehat{\xi}{+}1/2\mathrm{Tr}\left\{\widehat{\lambda}\left(N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B}\right)\right\}$.

$$\begin{aligned}
\mathrm{KL}(q(\tau)||p(\tau|\xi)) &= \overline{c}\log\overline{d} - \log\Gamma(\overline{c}) + (\overline{c}-1)\mathbb{E}_q[\log\tau] - \overline{d}\widehat{\tau} \\
&\qquad - \frac{1}{2}\mathbb{E}_q[\log\xi] + \log\Gamma(1/2) + \frac{1}{2}\mathbb{E}_q[\log\tau] + \widehat{\xi}\widehat{\tau}, \\
&= \overline{c}\log\overline{d} - \log\Gamma(\overline{c}) + (\overline{c}-1/2)\left(\psi(\overline{c}) - \log\overline{d}\right) - \widehat{\tau}\left(\overline{d} - \widehat{\xi}\right) \\
&\qquad - \frac{1}{2}\left(\psi(\overline{g}) - \log\overline{h}\right) + \log\Gamma(1/2).
\end{aligned}$$

Next is KL of hierarchical local, and global shrinkage parameters ϑ, ξ. Now (again) the typical formulation of distance KL between two distribution is readily applicable. Recall from optimal variational parameters that $\widehat{q}(\vartheta) \sim \mathcal{G}(\overline{e}, \overline{f})$, where $\overline{e}, \overline{g} = 1$, $\overline{f} = \widehat{\vartheta} + 1$, and $\overline{h} = \overline{\tau} + 1$.

$$\begin{aligned}
\mathrm{KL}(q(\vartheta)||p(\vartheta)) &= (\overline{e}-1)\psi(\overline{e}) - \log\Gamma(\overline{e}) + \log\Gamma(1/2) + 1/2\log\overline{f} + \overline{e}\left(\frac{1-\overline{f}}{\overline{f}}\right), \\
\mathrm{KL}(q(\xi)||p(\xi)) &= (\overline{g}-1)\psi(\overline{g}) - \log\Gamma(\overline{g}) + \log\Gamma(1/2) + 1/2\log\overline{h} + \overline{g}\left(\frac{1-\overline{h}}{\overline{h}}\right),
\end{aligned}$$

Finally the last term in eq. (2.32) is the term $\mathbb{E}_{q(\theta)}\left[\log p(\mathbf{y}|\theta)\right]$. As already described in section 2.2 that the EM-algorithm maximizes the $\log p(\mathbf{y}|\theta)$, the value is computed during the Kalman filter recursion, see sec (4.24) of Durbin & Koopman (2012). But here we provided for convenience.

$$\log p(\mathbf{y}|\theta) = -\frac{TN}{2}\log 2\pi - \frac{1}{2}\sum_{t=1}^{T}\left(\log|F_t| + v_t' F_t^{-1} v_t\right).$$

where v_t is the one-step-ahead forecast error. F_t is assumed to be nonsingular, and are typically called *updating step* of Kalman filter. Both are calculated routinely by the Kalman filter recursion. So the term $\log p(\mathbf{y}|\theta)$ is easily computed from the Kalman filter output, see eq-(7.2) of Durbin & Koopman (2012). Now taking the expectation with respect to the approximate distribution of global parameters $\mathbb{E}_{q(\theta)}[\bullet]$, the result remains unchanged i.e. $\mathbb{E}_{q(\theta)}\left[\log p(\mathbf{y}|\theta)\right] \equiv \log p(\mathbf{y}|\theta)$ since the we already use the mean of global parameters. In our next appendix C section 2.11 we briefly provide the Kalman filter smoother recursion, and how variance and covariance of smoothed state estimators.

Finally combine everything we have:

$$\begin{aligned}
\mathcal{L}\left(q(\theta), p(\mathbf{z}|\mathbf{y},\theta)\right) = -\frac{TN}{2}\log 2\pi - \frac{1}{2}\sum_{t=1}^{T}\left(\log|F_t| + v_t' F_t^{-1} v_t\right) - \frac{N}{2}\log|\Phi| + \frac{NK}{2}\left(1+\psi(\overline{c}) - \log\overline{d}\right) \\
&+ \frac{N}{2}\sum^{K}\left(\psi(\overline{a}) - \log\overline{b}\right) - \frac{1}{2}\mathrm{Tr}\left\{(\widehat{\lambda}\widehat{\tau})\left[N\Phi^{-1} + \widehat{B}'\widehat{\Sigma}\widehat{B}\right]\right\} \\
&- \left(\frac{N+1}{2}\right)\log|\overline{S}| - \frac{1}{2}N(N+1)\log(2) - \log\Gamma_N\left(\frac{\overline{\nu}}{2}\right) \\
&+ \left(\frac{\overline{\nu}-N-1}{2}\right)\psi_N\left(\frac{\overline{\nu}}{2}\right) - \frac{\overline{\nu}N}{2} \\
&+ \sum^{K}\Bigg(-\overline{a}\log\overline{b} + \log\Gamma(\overline{a}) - \sum^{K}(\overline{a}-1/2)\left[\psi(\overline{a}) - \log\overline{b}\right] + \widehat{\lambda}\left(\overline{b} - \widehat{\vartheta}\right) \\
&+ \frac{1}{2}\left(\psi(\overline{e}) - \log\overline{f}\right) - \log\Gamma(1/2)\Bigg) \\
&- \overline{c}\log\overline{d} + \log\Gamma(\overline{c}) - (\overline{c}-1/2)\left(\psi(\overline{c}) - \log\overline{d}\right) + \widehat{\tau}\left(\overline{d} - \widehat{\xi}\right) \\
&+ \frac{1}{2}\left(\psi(\overline{g}) - \log\overline{h}\right) - \log\Gamma(1/2) \\
&+ \sum^{K}\Bigg(-(\overline{e}-1)\psi(\overline{e}) + \log\Gamma(\overline{e}) - \log\Gamma(1/2) - 1/2\log\overline{f} - \overline{e}\left(\frac{1-\overline{f}}{\overline{f}}\right)\Bigg) \\
&- (\overline{g}-1)\psi(\overline{g}) + \log\Gamma(\overline{g}) - \log\Gamma(1/2) - 1/2\log\overline{h} - \overline{g}\left(\frac{1-\overline{h}}{\overline{h}}\right).
\end{aligned} \tag{2.35}$$

2.11 Appendix C: Kalman filter and smoother recursion, Variance and Covariance matrices of smoothed estimators

2.11.1 Kalman filter recursion

First write measurement and state equation from eqs. (2.16) and (2.17) in reduced form:

$$
\begin{aligned}
y_t &= G_t z_t, && (2.36)\\
z_t &= H z_{t-1} + c + v_t, \;\; v_t \sim \mathcal{N}(0, \Omega(\Sigma)), && (2.37)\\
z_0 &\sim \mathcal{N}(a_0, P_0). && (2.38)
\end{aligned}
$$

where $G_t = M_t \Lambda$ stacks all selection matrix and inter-temporal restriction. $H = F(B)$ is transition matrix. Denote α_t as filtered estimator[10] where $a_t = \mathbb{E}(z_t|y_t),\ P_t = \text{var}(z_t|y_t)$ are mean and covariance of filtered estimators conditional on observed data up to time t. Assuming mean and variance of filter estimators a_0, P_0 at time $t = 0$ are known, the initialized state and covariance of filtered estimators are given by:

$$
\begin{aligned}
a_1 &= H a_0 + c,\\
P_1 &= G_t P_0 G_t' + \Omega(\Sigma).
\end{aligned}
$$

Recursion of the following equations for time $t = 1, .., T$ leads to filtered state variables, see eq.(4.24) Durbin & Koopman (2012).

$$
\begin{aligned}
v_t &= y_t - G_t z_t, & F_t &= G_t P_t G_t',\\
a_{t+1} &= H a_t + c + K_t v_t, & K_t &= H P_t G_t' F_t^{-1}, \qquad (2.39)\\
P_{t+1} &= H P_t L_t' + \Omega(\Sigma). & L_t &= H - K_t G_t.
\end{aligned}
$$

[10] Notably, the state variables that fills missing observation in $\mathbf{y}$ are mean smoothed estimators not filtered estimators.

2.11.2 Kalman Smoother recursion

Kalman smoother is different from Kalman filter recursion in the sense that smoother-latent parameters are generated conditional on whole observed observations rather than the observation up to time t from Kalman filter, i.e. $\mathbb{E}(z_t|y_1, ..., y_T) = \widehat{z}_t$ and $\widehat{P}_t = \text{Var}(z_t|y_1, ..., y_T)$. The state smoother equations are iteratively formulated backward for time $t = T+1, ..., 1$ as follows, see eq.(4.44) Durbin & Koopman (2012):

$$\begin{aligned} r_t &= G_t' F_t^{-1} v_t + L_t' r_{t+1}, \\ \widehat{z}_t &= a_t + P_t r_t, \\ \widehat{z}_0 &= a_0 + P_0 H' r_1, \end{aligned} \qquad \begin{aligned} \widehat{P}_t &= P_t - P_t N_t P_t', \\ N_t &= G_t' F_t^{-1} G_t + L_t' N_{t+1} L_t, \end{aligned} \tag{2.40}$$

where L_t, F_t, v_t, P_t are computed during the recursion of Kalman filter eq. (2.39). Initialize $r_{T+1} = 0$ and $N_{t+1} = 0$. The smoothing estimates can be obtained straightforwardly for $t = T, ..., 1$.

2.11.3 Variance and covariance of smoothed state variables

The full derivation of covariance of smoothed estimators can be found in Durbin & Koopman (2012) sec 4.7. They stated that the original derivation were developed by Koopman (1993), and JONG & Mackinnon (1988). Here we provide the final formulation. $\widehat{P}_t = \text{Var}(z_t|y_1, .., y_T)$ is already computed during smoothing recursion, see eq. (2.40). For $J_t = \text{Cov}(z_t, z_{t+1}|y_1, .., y_T)$. This covariance of smoothed estimators are similar in modified OLS for global parameters in EM-algorithm, see Brave et al. (2020) and proposed Vb-EM algorithm for MF-VAR is formulated as:

$$\begin{aligned} J_t &= P_t \mathbf{L}_t (I_m - N_{t+1} P_{t+1}), \\ \mathbf{L}_t &= H(I_m - K_{t+1} G_{t+1}) I_m \end{aligned} \tag{2.41}$$

where I_m is $m \times m$ dimensional identity matrix. These variance and covariance of smoothed estimators $\widehat{P}_t, \widehat{P}_{t,t-1}, J_t$ from Kalman filter and smoother algorithm will be a key feature to approximate optimal variational parameters for global-parameters in

Vb:M-step, see eqs. (2.22) to (2.24) for clarity i.e. $\mathbf{J}_T = \sum_t^T J_t$.

2.12 Appendix D: Lemmas

Lemma 2.12.1 *Suppose that $X \sim \mathcal{N}_{N,K}(M, \Delta \otimes \Omega)$, and let A be $N \times N$ matrix then the expectation of X follows immediately, see Gupta & Nagar (2018) page 60.*

$$\mathbb{E}[X'AX] = Tr\{\Delta A'\}\Omega + M'AM.$$

Additionally let $\widetilde{A}$ be $K \times K$ matrix then the expectation of X follows:

$$\mathbb{E}[X\widetilde{A}X'] = Tr\{\Omega\widetilde{A}'\}\Delta + M\widetilde{A}M'.$$

Lemma 2.12.2 *Let $\mathbf{x}$ be $D \times T$ matrix with $\mathbf{x}_t$ being D dimensional vector at time t, with mean $\mathbf{m}_t$, and covariance $\mathbf{V}_t$, i.e. $\mathbf{x}_t \sim \mathcal{N}_D(\mathbf{m}_t, \mathbf{V}_t)$. Then $\mathbb{E}_{\mathbf{x}}[\mathbf{x}_t\mathbf{x}_t'] = \mathbf{V}_t + \mathbf{m}_t\mathbf{m}'_t$. Thus it follows immediately that:*

$$\begin{aligned}\mathbb{E}_{\mathbf{x}}\left[\mathbf{x}\mathbf{x}'\right] &= \sum_{t=1}^{T}\left(\mathbf{V}_t + \mathbf{m}_t\mathbf{m}_t'\right), \\ &= \sum_{t=1}^{T}\left(\mathbf{V}_t\right) + \mathbf{m}\mathbf{m}'.\end{aligned}$$

And a consequence of Lemma 12.2 is the statement in the next corollary.

Corollary 2.12.2.1 *Again, let $\mathbf{x}$ be $D \times T$ random matrix with $\mathbf{x}_T$ being $D \times T - 1$ matrix of $\mathbf{x}$ at time $t = 2, ..., T$, and $\mathbf{x}_{T-1}$ being $D \times T - 1$ matrix of $\mathbf{x}$ at time $t = 1, ..., T - 1$. Then it follows immediately that:*

$$\mathbb{E}\left[\mathbf{x}_T\mathbf{x}_{T-1}'\right] = \sum_{t=2}^{T}\left(\mathbf{V}_{t,t-1} + \mathbf{m}_t\mathbf{m}_{t-1}'\right).$$

where $\mathbf{V}_{t,t-1}$ is covariance between $\mathbf{x}_t, \mathbf{x}_{t-1}$. Additionally it also follows that:

$$\mathbb{E}\left[\mathbf{x}_{T-1}\mathbf{x}_{T-1}'\right] = \sum_{t=2}^{T}\left(\mathbf{V}_{t-1,t-1} + \mathbf{m}_{t-1}\mathbf{m}_{t-1}'\right).$$

Chapter 3

Forecasting macroeconomic variables with Gaussian process VAR.

3.1 Introduction

Gaussian process VAR attracts significant attention from economists recently. Particularly in the context of the COVID-19 pandemic, which has led the macroeconomic variables' volatilities to rise or fall to unprecedented levels. As a result, conventional VAR models may not be suitable, and their parameters can be unreliable Lenza & Primiceri (2020). The development of structural VAR models, such as Cholesky-transformed VARs, which allow for estimation equation by equation, has enabled economists to explore more advanced regression models in VARs Clark et al. (2022), Huber et al. (2020), Hauzenberger et al. (2021).

Gaussian process VAR, a non-parametric model, has shown better ability to handle outliers that emerged during the pandemic compared to traditional Bayesian VAR models with stochastic volatility Huber et al. (2020), Clark et al. (2022). In this chapter, I therefore investigate two additional algorithms base on GP-VAR. Firstly GP-VAR using Deep neural network (DNN) serving as mean function for $\mathcal{GP}$ prior (labelled: GP-DNN-VAR). Although this algorithm is popular among machine learning researchers but as far as the knowledge goes, it has never been explored in economic field yet, see for instance Fortuin et al. (2019).[1] The majority of GP-VAR in economic applications place more emphasis on learning the GP kernel parameters than on the mean function in $\mathcal{GP}$ prior by simply put zero mean GP prior on top of non-parametric function, see for examples Clark et al. (2022), Hauzenberger et al. (2021). Employing the mean function in $\mathcal{GP}$ prior is also known as *centering Gaussian process*. The benefit of having mean function parameterized by DNN rather than assuming it to be zero, is especially pronounced when the observed data is away from zeros. We investigate whether this might improve the out-of-sample forecast performance of key macroeconomic variables during periods of high volatility. Hence manipulating those mean function is worth exploring.

Secondly, Heteroscedastic-GP-VAR (HGP-VAR). HGP-VAR differs from GP-VAR in that GP-VAR models typically assume that the likelihood residual are normally distributed with constant covariance across all observations. In HGP-VAR applications, however, has input/predictors-dependent noise (heteroscedastic covariance). This model is designed to model residuals adequately in the presence of heteroscedasticity. From macroeconomic point of view, it is often important to allow for error covariance to change over time. Most common approach is employing stochastic volatility into Bayesian VAR model (BVAR-SV), which assumes log volatilities to follow a random walk. HGP-VAR model, on the other hand, assumes the error covariance to be in non-parametric functional form. In addition to these models, GP-VAR model is also provided for relative comparisons.

The main contribution of this chapter is to assess whether the forecasting performance of these GP-VAR models can be improved as opposed to a traditional workhorse Bayesian VAR with stochastic volatility model. To investigate this idea, we evaluate point and predictive densities using metrics such as root mean square error (RMSE), cumulative ranked probabilistic scores (CRPS), and quantile scores (QS).

The roadmap for this chapter can be categorized as followed: We begin with section 3.2, by first briefly introducing Gaussian process model with economic data, types

[1] Note that it is not the same model where the specific kernel function mimic multi-layer random neural networks, for examples Lee et al. (2017), Matthews et al. (2018).

of kernels and how kernels affect the posterior distribution, compositional kernels (combining multiple kernels) and derivation of analytical expression of log marginal likelihood. Section 3.3 shows how VAR is formulated both for parametric model, Bayesian VAR with SV, and all types of GP-VARs (non-parametric). Section 3.4 shows how GP-VARs can be estimated by simply perform regression one equation at a time, and how each Gaussian process parameters are learned. Section 3.5 presents full in-sample estimations. These include interpreting the meaning of kernel hyper-parameters from macroeconomic point of views. We are comparing the trade-off between the conditional mean of parametric and non-parametric models. To shed some light on how different volatilities of heteroscedastic models i.e. BVAR-SV (a random walk SV), and HGP-VAR (non-parametric functional form Heteroscedastic variance) are, we visualize volatilities results resulted from both models and HGP-VAR model. Next is section 3.6, where the forecast performances are evaluated. Finally section 3.7 draws the conclusions and section 3.8 reports appendices.

3.2 Gaussian process regression

Gaussian Processes (GPs) Rasmussen (2003) are a straightforward approach to generalise the concept of a multivariate normal distribution. A multivariate normal distribution defines random variables that are vectors, whereas a GP describes random variables that are real-valued functions defined over some input domain or covariates. If the input domain is real numbers, then the random variable described by a GP can be considered as a vector of an infinite extent and infinite resolution that is indexed by a real number. GPs are popular tool for regression model, identifying an unknown real-value function from noisy function observations at some covariates locations.

Let y as targets/outputs (observed data) that we wish to model. We do not know which distribution targets are distributed but in Gaussian process regression, we assume the output y of a function f at covariate $\mathbf{x}$ can be written as:

$$\begin{aligned} y &= f(\mathbf{x}) + \epsilon, \\ f(\mathbf{x}) &\sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \end{aligned} \tag{3.1}$$

with $f(\mathbf{x})$ is a real-value function at input $\mathbf{x}$. A GP represents a distribution denoted as $\mathcal{GP}(\bullet)$ over real-valued functions $f : \mathbb{R}^M \to \mathbb{R}$ that map the vector in $\mathbb{R}^M$ to some feature space $\mathbb{R}$. The $\mu(\mathbf{x})$ is $\mathcal{GP}$ prior mean, $\mu(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$, and $k(\mathbf{x}, \mathbf{x}')$ is kernel (covariance) value evaluated at covariate/input $\mathbf{x}$ i.e. $k(\mathbf{x}, \mathbf{x}') = \mathbb{E}\left[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{x}') - \mu(\mathbf{x}'))\right] = \mathrm{cov}\left[f(\mathbf{x}), f(\mathbf{x}')\right]$.

In particular if the GP is evaluated at any finite subset of $\mathbf{x} = (x_1, ..., x_T)'$, the notation becomes $T-$dimensional multivariate normal random variables of $\mathbf{f}$:

$$\mathbf{f} \sim \mathcal{N}(\mu_{\mathbf{f}}, \mathbf{K}_{\mathbf{ff}}) \tag{3.2}$$

where $\mu_{\mathbf{f}}$, and $\mathbf{K}_{\mathbf{ff}}$ are T-dimensional mean vector and $T \times T$ covariance matrix evaluated at covariates $(x_1, ..., x_T)$, i.e. $\mu_{\mathbf{f}}(t) = \mu(x_t)$, and $\mathbf{K}_{\mathbf{ff}}(t, \tau) = k(x_t, x_\tau)$, respectively. The brackets $\mathbf{K}_{\mathbf{ff}}(t, \tau)$ indicates the number of row t and column τ index of covariance $\mathbf{K}_{\mathbf{ff}}$, in a manner analogous to Matlab matrix notations. From most literature this notation is unfortunately hidden for notational clarity, which may confuse readers who is new to the topic. It is noteworthy that if zero mean $\mathcal{GP}$ prior is assumed i.e. $\mu_{\mathbf{f}} = \mathbf{0}_T$ the parameters $\mathbf{f}$ relies solely on the kernel department.

The error ϵ is scalar i.i.d random variables that represent observation noise. The distribution of error term is generally assumed to be $\epsilon \sim \mathcal{N}(0, I_T\sigma^2)$, where σ^2 is error variance. The GP regression is fully specified if error variance σ^2, and kernel hyperparameters θ are known.

There are numerous methods for estimating GP-regression, Rasmussen (2003), Bishop (2006), Williams & Rasmussen (2006) For faster computation plus the availability of analytical expression of log-marginal likelihood, throughout the chapter, I will focus more on maximizing log marginal likelihood with respect to all parameters σ^2, θ.

First let $\mathbf{y} = (y_1, .., y_T)'$, the joint probability model is $p(\mathbf{y}, \mathbf{f}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f})$ where

$p(\mathbf{f})$ is $\mathcal{GP}$ prior, and $p(\mathbf{y}|\mathbf{f})$ is the likelihood. With Bayes Theorem, the posterior mean and covariance of latent function at input x, and observed data y is given by, see Rasmussen (2003):

$$\begin{aligned} p(f(x)|y) &\sim \mathcal{N}(\mu(x), k_y(x, x')), \\ \mu(x) &= \mathbf{K}_{x\mathbf{f}}(\sigma^2 I + \mathbf{K}_{\mathbf{ff}})^{-1} y, \\ k_y(x, x') &= k(x, x') - \mathbf{K}_{x\mathbf{f}}(\sigma^2 I + \mathbf{K}_{\mathbf{ff}})^{-1} \mathbf{K}_{\mathbf{f}x'}. \end{aligned} \tag{3.3}$$

where $\mathbf{K}_{\mathbf{ff}}$ is the $T \times T$ covariance function evaluated between every observations of inputs, $\mathbf{K}_{x\mathbf{f}}$ is T-dimensional row vector of kernel function values between x and all inputs, notably, $\mathbf{K}_{\mathbf{f}x} = \mathbf{K}'_{x\mathbf{f}}$. The Gaussian process regression is completely specified if all of these hyper-parameters (σ^2, θ) are known. Therefore the posterior GP requisites estimating of likelihood variance σ^2 and kernel hyper-parameters (θ), which can be obtain by maximizing a log marginal likelihood. Fortunately such objective function is available in closed form. This will be noted later but now lets introduce to the kernel department.

Next sub-section provides some examples of kernels, resulting in different kernel hyper-parameters. Moreover illustrating on how different kernel can be influential to in-sample predictions and how to manipulate (combine) them.

3.2.1 Kernels

Kernel functions are sometimes also referred to as ”generalised dot products” since they compute the dot product of two vectors x_t and x_τ in some (possibly extremely high dimensional) feature space.

Recall that in order to configure our distribution i.e. $\mathbf{f} \sim \mathcal{GP}(\mu_{\mathbf{f}}, \mathbf{K}_{\mathbf{ff}})$, we must define $\mu_{\mathbf{f}}$ and $\mathbf{K}_{\mathbf{ff}}$. In Gaussian processes, it is commonly assumed that $\mu_{\mathbf{f}} = \mathbf{0}_T$, which simplifies the conditioning equations. We can always assume such a distribution, even if the actual $\mu_{\mathbf{f}} \neq \mathbf{0}_T$, and add mean function back to posterior function values during the prediction step. This method is also known as data *centering.* Some literature configure mean function in $\mathcal{GP}$ prior to be parameterized by deep neural networks (Matthews et al. (2017), Fortuin et al. (2019)). For now $\mathcal{GP}$ prior with zero mean function are assumed.

There are multiple kernels to choose from. For a very simple example, the radial basis function, also known as the Quadratic Exponential kernel is in the form:

$$\mathbf{K_{ff}}(t,\tau) = k(x_t, x_\tau) = \gamma^2 \exp\left(-\frac{\| x_t - x_\tau \|^2}{2l^2}\right). \tag{3.4}$$

where (again) $\mathbf{K_{ff}}(t,\tau)$ is the covariance of $\mathcal{GP}$ prior at row number t, and column number τ. The symbol $\| x_t - x_\tau \|^2$ is recognized as the squared Euclidean distance (L2-norm) between time t, and τ of covariates. $l, \gamma^2 \in \mathbb{R}^+$ are kernel hyper-parameters, typically called kernel variance γ^2, and length parameters l. Each kernel contains a set of parameters that define the exact shape of the covariance function. These are frequently referred to as *hyper-parameters* because they can be thought of as describing a distribution over function parameters rather than being parameters that directly specify a function. For example, the length parameter l of kernel above controls the band of around the conditional mean produced by non-parametric function. As the length parameter is increased, the banding becomes more pronounced, as points further apart become more correlated. In the following discussion I will illustrate only the different kernel, resulting in the posterior only. For textbook treatment of how kernel hyper-parameters affect the shape of prior function, Duvenaud (2014), Görtler et al. (2019).

To begin with fig. 3.1 illustrates Gaussian process regression (GPR) with US gross domestic product data, regressed with its own single lag. Here those figures show the posterior prediction mean of GPR model with four different kernels, namely Radial basis function kernel, Linear, Matern52 and Matern32 kernel, respectively. The grey bar represents 95% confident interval. It is noteworthy that the posterior is obtained by maximizing closed-form log marginal likelihood with respect to kernel hyper-parameters, and likelihood variance. It is quite obvious that the Matern52/Matern32 kernel outperforms the rest especially when the US GDP is in the COVID-19 pandemic periods. Linear kernel, on the other hand, inadequately model US GDP. Because GPR assumes zero mean function in $\mathcal{GP}$ prior, the posterior mean and variance are entirely dependent to the kernel hyper-parameters. As a result, selecting a proper kernel is important to the GPR model. Next sub-section briefly introduce how can we manipulate kernels by either via the addition or multiplication multiple kernels.

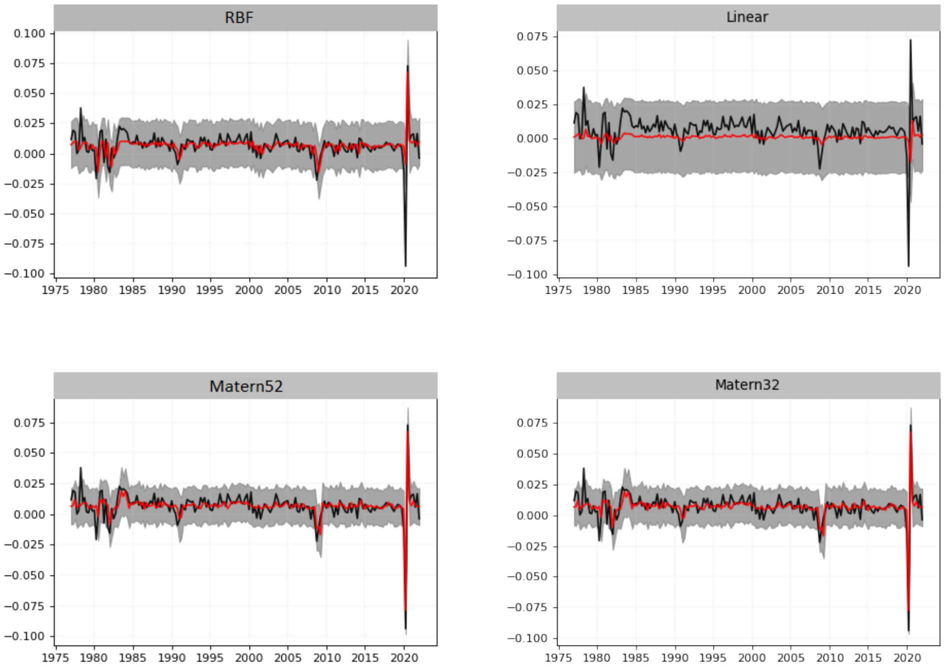


Figure 3.1: Gaussian process regression (GPR) with US gross domestic product, regressed with its own single lag. Each figure illustrate the posterior of GPR with four different kernels. Radial basis function (also known as Squared Exponential kernel), Linear, Matern52 and Matern32 kernel, respectively. Red-solid line is posterior prediction mean, Grey-band represents 95% confident interval, and black-solid line illustrates actual observed US GDP.

3.2.2 Compositional kernels.

As discussed above, kernel function determines the efficacy of Gaussian processes. This property enables specialists to incorporate domain knowledge into the process and gives Gaussian processes the flexibility to identify trends in observed data. By selecting an appropriate bandwidth for the RBF kernel, for instance, we regulate the smoothness of the resulting function. The benefit of using kernel as covariance in $\mathcal{GP}$ prior is the ability to combine together *(compositional kernels)* Duvenaud et al. (2013), consequently a more specialised kernel. The most frequent kernel compositions would consist of addition and multiplication MacKay et al. (2003). Let's consider the following two kernels, Periodic kernel $k_{\text{periodic}}(\cdot, \cdot')$, and Linear kernel $k_{\text{linear}}(\cdot, \cdot')$:

$$k_{\text{periodic}}(\cdot, \cdot') = \gamma^2 \exp\left(-\frac{2\sin^2(\pi| \cdot - \cdot' |/\rho)}{l^2}\right),$$
$$k_{\text{linear}}(\cdot, \cdot') = \gamma_b^2 + \gamma_a^2(\cdot - c)(\cdot' - c).$$

The multiplication would simply be $k_{\text{combined}}(\cdot, \cdot') = k_{\text{periodic}}(\cdot, \cdot') \odot k_{\text{linear}}(\cdot, \cdot')$, where $\odot$ is element-wise product.

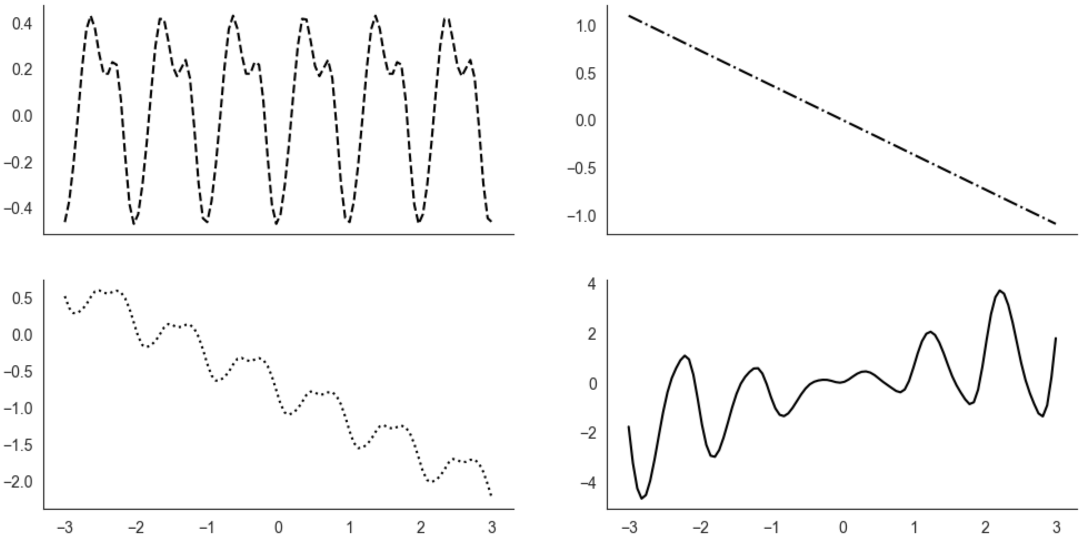


Figure 3.2: Samples generated from $\mathcal{GP}$ prior distribution using different kernels. Top figures from left to right, is Periodic and Linear kernel, respectively. Bottom figures are the combination of both kernels, that is, kernel addition (left) and kernel multiplication (right).

Figure 3.2 illustrates the samples generated from $\mathcal{GP}$ prior distribution using two different kernels, and compositional kernels. As the prior distribution does not yet contain any additional information, the Periodic kernel's samples are shown as a wave pattern (top left), suitable to potentially model seasonal observations in economic data. Linear kernel, however, has negative slope (top right). Bottom left and right figures demonstrate how the impact of the kernel combination and how it may preserve the quality features of the individual kernels. Notice that after adding a periodic and a linear kernel (bottom left), the trend of a linear kernel is integrated into the combined kernel, resulting in a periodic function that follow linear trend. When the identical kernels are multiplied together, the outcome is a periodic function with an amplitude that grows linearly away from the linear kernel parameter c. The advantage of combining kernels will be discussed empirically in next section.

3.2.3 Marginal likelihood of Gaussian process regression

As discuss in the previous sub-section that the Gaussian process regression is completely specified by its mean function and covariance function. Let θ stacks all parameters in kernel department which will be estimated. The log marginal likelihood is in the form:

$$p(\mathbf{y}) = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f})d\mathbf{f}. \tag{3.5}$$

The marginal likelihood from eq. (3.5) above refers to the marginalization over the function values $\mathbf{f}$.[2] From sampling-based perspective such as MCMC, these parameters can be estimated by setting prior to kernel hyper-parameters θ, then Bayes' rule is readily applied. Most of the time researcher use MCMC method such as Gibbs sampling because the normalizing constant (log marginal likelihood or log evidence) is hard to obtain or sometime impossible. In Gaussian process regression, however, the marginal likelihood is available in closed form, Rasmussen (2003), Bishop (2006):

$$\log p(\mathbf{y}) = \log \left[\mathcal{N}(\mathbf{y}|\mathbf{0}_T, \sigma^2 I_T + \mathbf{K_{ff}})\right] \tag{3.6}$$

where $\mathcal{N}(\mathbf{y}|\mathbf{0}_T, V)$ is multivariate normal distribution with zero mean and V covariance of random variables $\mathbf{y}$. The benefit of maximizing a closed-form log marginal likelihood w.r.t interested parameters are it is significantly faster than sampling-based such as MCMC.

[2] It is important to note that there are no integrals over the likelihood variance σ^2 and kernel hyper-parameters θ, as these are treated as hyper-parameters.

3.3 VAR models

3.3.1 Bayesian VAR with SV

Our benchmark model is BVAR with stochastic volatility (BVAR-SV). Quite often, economist found significant improvements in forecast performance from having heteroscedastic variance in BVAR model, see for examples, Li & Koopman (2021), Marcellino et al. (2016), Carriero et al. (2015), Kastner & Frühwirth-Schnatter (2014). Therefore, we select BVAR-SV model as our benchmark, which is in the form:

$$\mathbf{y}_t = \beta \mathbf{x}_t + \mathbf{Q}\mathbf{y}_t + \epsilon_t, \quad \text{where } \epsilon_t \sim \mathcal{N}(0, \boldsymbol{\Sigma}_t). \tag{3.7}$$

where $\mathbf{y}_t = (y_{1t}, ..., y_{Nt})'$ is $N \times 1$ vector of macroeconomic responsive variables, $\mathbf{x}_t$ is K-dimensional vector of predictor, $K = Np$ where p is number of lag in VAR. β represents $N \times K$ VAR coefficients, mapping $\mathbf{x}_t$ linearly to the endogenous variables. The $N \times N$ lower-triangular matrix $\mathbf{Q}$ with zeros on its diagonal, whose role is to define the contemporaneous relationship across the elements in $\mathbf{y}_t$. From eq. (3.7), the VAR covariance is $\boldsymbol{\Sigma}_t = \text{diag}(e^{h_{1t}}, ..., e^{h_{Nt}})$ which stack the volatility diagonally at each point in time. The log-volatility is assumed to follow a *stationary* AR(1) process:

$$\begin{aligned}
h_t &= \mu + \phi(h_{t-1} - \mu) + \nu\eta_t, \\
h_0|\mu, \phi, \nu &\sim \mathcal{N}(\mu, \nu^2/(1-\phi^2)), \\
\mu &\sim \mathcal{N}(a_\mu, b_\mu), \\
\phi &\in (-1, 1) \sim \mathcal{U}(-1, 1), \\
\nu &\sim \mathcal{IG}(a_\nu, b_\nu).
\end{aligned} \tag{3.8}$$

where $\mathcal{N}(a_\mu, b_\mu)$ denotes normal distribution with mean a_μ and variance b_μ. The μ, ϕ, ν parameters are *level* of log-variance, the *persistent* of log-variance, and the *volatility* of log-variance, respectively. They are assumed to be unknown and will be estimated.[3]

[3] Without the stationary control i.e. assume that $\phi = 1$ sometime leads to an explosive forecasts especially at longer horizon, see for example Cogley & Sargent (2005). These results are consistent with literature, see for examples, Loaiza-Maya et al. (2021), Chan & Yu (2022), Gefang et al. (2022).

The random walk is stationary in a sense that the prior for persistent of log-variance ϕ is assumed to be continuous uniform distribution with the lower bound and upper bound between $-1, 1$ i.e. $\phi \sim \mathcal{U}(-1, 1)$. As a result, $|\phi| < 1$ control the random walk to be stationary across observations. The representation of SV model above is also known as *centered parameterization* SV, Kastner & Frühwirth-Schnatter (2014).

VAR coefficients in eq. (3.7) are $\beta, \mathbf{Q}$. To avoid the over-fitting, the Horseshoe prior distribution Carvalho et al. (2010) is specified.

$$\begin{aligned} \beta_j|\lambda_j^2, \tau^2 &\sim \mathcal{N}(0, \lambda_j^2, \tau^2), \\ \lambda_j^2|\psi_j &\sim \mathcal{IG}(\frac{1}{2}, \frac{1}{\psi_j}), \\ \tau^2|\xi &\sim \mathcal{IG}(\frac{1}{2}, \frac{1}{\xi}), \\ \psi_1, ..., \psi_K, \xi &\sim \mathcal{IG}(\frac{1}{2}, 1). \end{aligned} \tag{3.9}$$

where $z \sim \mathcal{IG}(\alpha, \beta)$ denotes inverse Gamma distribution with probability density function $p(z|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} z^{-\alpha-1} \exp\left(-\frac{\beta}{z}\right)$. The λ_j^2 is local shrinkage parameters controlling the tightness of VAR coefficient individually. The global shrinkage parameters τ^2, on the other hand, controls the overall tightness of coefficients.

From this point of view it is straightforward to obtain the conditional posterior distribution. Readers are referred to Makalic & Schmidt (2015) for brevity.

After estimating all necessary parameters, the VAR covariance can be obtained by simply computing the product of $\mathbf{H}_t = \mathbf{A}^{-1}\Sigma_t\mathbf{A}^{-1'}$, where $\mathbf{A} = \mathbf{Q} - \mathbf{I}_N$, and $\mathbf{I}_N$ is $N \times N$ identity matrix.

3.3.2 Gaussian process-VARs

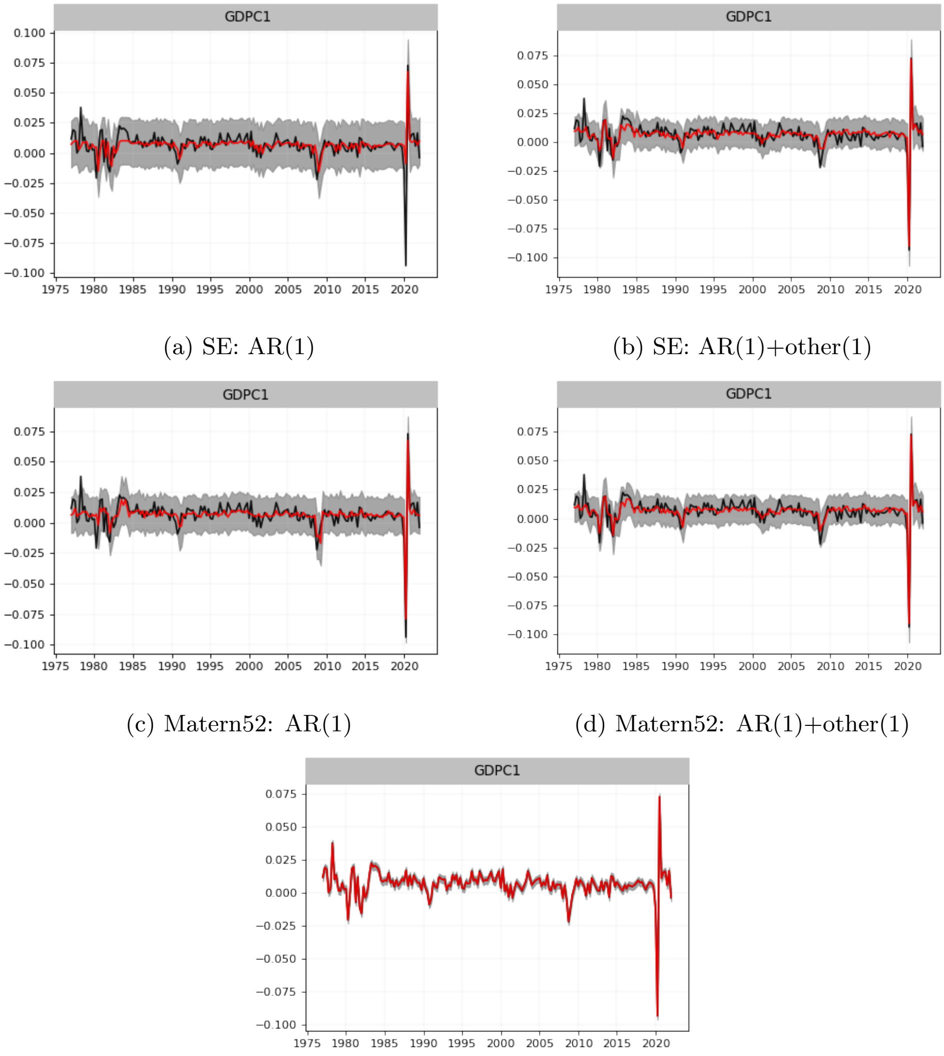
This sub-section introduce how Gaussian process regression can be extended to a multivariate case (GP-VAR) where each equation in GP-VAR can be estimated one equation in a time. GP-VAR has been successfully found to be a successful model for forecasting macroeconomic variables particularly during periods of economic turbulence. For example Huber et al. (2020) employ GP-VAR in Mixed-frequency VAR and found that

To avoid such problem μ, ϕ are treated as unknown parameters.

forecast performance is improved during the Covid-19 pandemic for European GDP growth.

While a recent paper Clark et al. (2022) demonstrates that simple GP-regression accurately predicts US inflation compared to the unbeatable unobserved component stochastic volatility (UC-SV) model, I investigated GP-VAR model using multiple US-macroeconomic variables, and found it to be accurate in terms of in-sample predictions. This leads to the hypothesis that GP-VARs might also potentially perform well in terms of out-of-sample prediction. To convince readers, I plot fig. 3.3, which illustrate the posterior GP regression of US GDP data. Each figure in fig. 3.3 has different GP-regression model configurations, that is, different kernel and covariates (or inputs). In the following discussion, black-solid line is GPRs conditional mean, red-solid is actual observed US GDP, and grey-band represents 95% confident interval.

To begin with first figure, fig. 3.3a, the predictor is simply a lag of GDP observations, notably the squared exponential kernel is employed (labelled: AR(1), squared exponential). Secondly fig. 3.3b, shows similar model with a lag of additional 7 other important macroeconomic and financial variables (labelled: AR(1)+other(1), squared exponential). The third and fourth figs. 3.3c and 3.3d show AR(1), and AR(1)+other(1) with Matern52 kernel, respectively. Finally, the most crucial one, fig. 3.3e illustrates model with AR(1)+other(1) setting with two composite kernels (combination), namely squared exponential and Matern52. Obviously with squared exponential kernel and Matern52 alone, adding more important predictors/inputs to the GP-regression model barely makes any difference to the posterior GP-regression, and thus cannot model US GDP data adequately. With appropriate predictors and some kernel manipulations, GP-regression is able to capture US GDP almost perfectly. The 95% confident interval is narrower, indicating how confident the GP-regression model is. Those additional 7 macroeconomic/financial variables (and possibly more variables) may carry some predictive information which can potentially improve the out-of-sample predictive performance especially in a long forecast horizon. Therefore, investigating the use of GP-VARs instead of simple GP regression and conducting iterative forecasts is worthwhile.



(a) SE: AR(1)

(b) SE: AR(1)+other(1)

(c) Matern52: AR(1)

(d) Matern52: AR(1)+other(1)

(e) SE+Matern52: AR(1)+other(1)

Figure 3.3: Gaussian process regression (GPR) of US GDP observation with five different settings.

Note: Figure 3.3a uses only lagged values of US GDP as explanatory variables with Squared Exponential kernel (labelled SE:AR(1)). Figure 3.3b uses one own lag and a single lag of other seven important macroeconomic and financial variables (labelled SE: AR(1)+other(1)), (Also Squared Exponential kernel). Figure 3.3c is AR(1) with Matern52 kernel (labelled: Matern52: AR(1)). Figure 3.3d is AR(1)+other(1), with Matern52 kernel (labelled: Matern52: AR(1)+other(1)). Finally and most importantly one, fig. 3.3e (labelled: SE+Matern52: AR(1)+other(1)) but two kernels are combined, namely Squared exponential and Matern52.

GP-VAR begins by first denote $\mathbf{X}_t = (\mathbf{x}_t, ..., \mathbf{x}_t)'$ with $\mathbf{x}_t$ as K-dimensional vector of covariates at time t. $\mathbf{y}_t = (y_{1t}, ..., y_{Nt})'$ as an $N \times 1$ macroeconomic responsive variables which is similar to previous sub-section. The GP-VAR can be expressed in the form:

$$\mathbf{y}_t = F(\mathbf{X}_t) + A^{-1}\epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0_N, \Sigma). \tag{3.10}$$

where $F(\mathbf{X}_t) = (f_1(\mathbf{x}_t), ..., f_N(\mathbf{x}_t))'$, with $\mathcal{GP}$ prior on top of each function in GP-VAR, i.e. $f_1(\mathbf{x}_t), ..., f_N(\mathbf{x}_t)) \sim \mathcal{GP}(\mu_1(\mathbf{x}_t), k_1(\mathbf{x}_t, \mathbf{x}_\tau)), ..., \mathcal{GP}(\mu_N(\mathbf{x}_t), k_N(\mathbf{x}_t, \mathbf{x}_\tau))$. $\epsilon_t = (\epsilon_{1t}, ..., \epsilon_{Nt})'$, and $\Sigma = diag(\sigma_1^2, ..., \sigma_N^2)$ stacking residual variances from each equation diagonally. The reason to have an additional term A^{-1} in reduced-form GP-VAR is to capture the contemporaneous relations between the element in $\mathbf{y}_t$, and those lower elements in squared matrix A^{-1} will be used to compute GP-VAR covariance. The idea of expression above is borrowed from a paper Carriero et al. (2022) but instead of mapping VAR explanatory variables to responsive variables linearly, the responsive variables are assumed to be in the form of unknown functions $F(\cdot)$ which is potentially non-linear functions and input-dependent. Formulating such fashion reduces tremendous computational cost as described in chapter 1.

Centering Gaussian process with feed-forward Deep neural networks

So far we have discussed the $f(\cdot) \sim \mathcal{GP}(\mu(\cdot), k(\cdot, \cdot'))$ prior with zero mean function $\mu(\cdot) = 0$, implying that the Gaussian process relies solely on the kernel. This has been the primary focus on the economic literature related to the GP-VAR, see for examples Clark et al. (2022), Hauzenberger et al. (2021). As discussed in the section 3.2.1, manipulating the mean function of $\mathcal{GP}$ prior might help us in solving the target more efficiently. To begin with a very simple linear mean function, when evaluated at input x_t at time t, it is in the form:

$$\mu(x_t) = Wx_t + b. \tag{3.11}$$

where W is weight/coefficients/effect size mapping input x_t linearly to the target, and b is referred to *bias*. Notice that it is exactly similar to the simple linear regression with OLS weights and constant term. The benefit of doing so help the model to solve the target more effectively especially when the target is away from zero. Since the $\mu(\cdot)$ is in the $\mathcal{GP}$ prior it can be estimated before the kernel hyper-parameters, or even after.

Moreover it can be extended to nonlinear parameterization such as neural network and deep neural network. In this sub-section I focus more on the latter, demonstrating how simple feed-forward deep-neural network (DNN) can be served as a mean function. Similarly the mean function of $\mathcal{GP}$ prior when evaluating at input point x_t can then be written as:

$$\mu(x_t) = W_3 h(W_2 h(W_1 h(W_0 x_t + b_0) + b_1) + b_2) + b_3. \tag{3.12}$$

where W_i, b_i is weight matrices and biases of DNN layer i, respectively. $h(\cdot)$ represents an activation function, sigmoid or ReLu for instances.[4] The example above shows only three layers in DNN. However, from a practical standpoint, the number of layers and weight size can be adjusted as *deep* as required.

The visualization of $\mathcal{GP}$ prior mean function of both examples are shown in figs. 3.4 and 3.5. It is obvious to see that if no activation employed, and only a single layer is used, the mean function collapses to the simple OLS estimators which is similar to eq. (3.11).

Each equations in GP-VAR will have individual mean function parameterized by DNN, with the goal to potentially center GP model according to each characteristic of responsive variable in GP-VAR.

[4] Activation functions are required to prevent linearity in the broadest sense. Without them, data would only go through the nodes and layers of a network with linear functions. In this chapter I employ a Sigmoid activation function, which is, $h(z) = \frac{1}{1+e^{-z}}$.

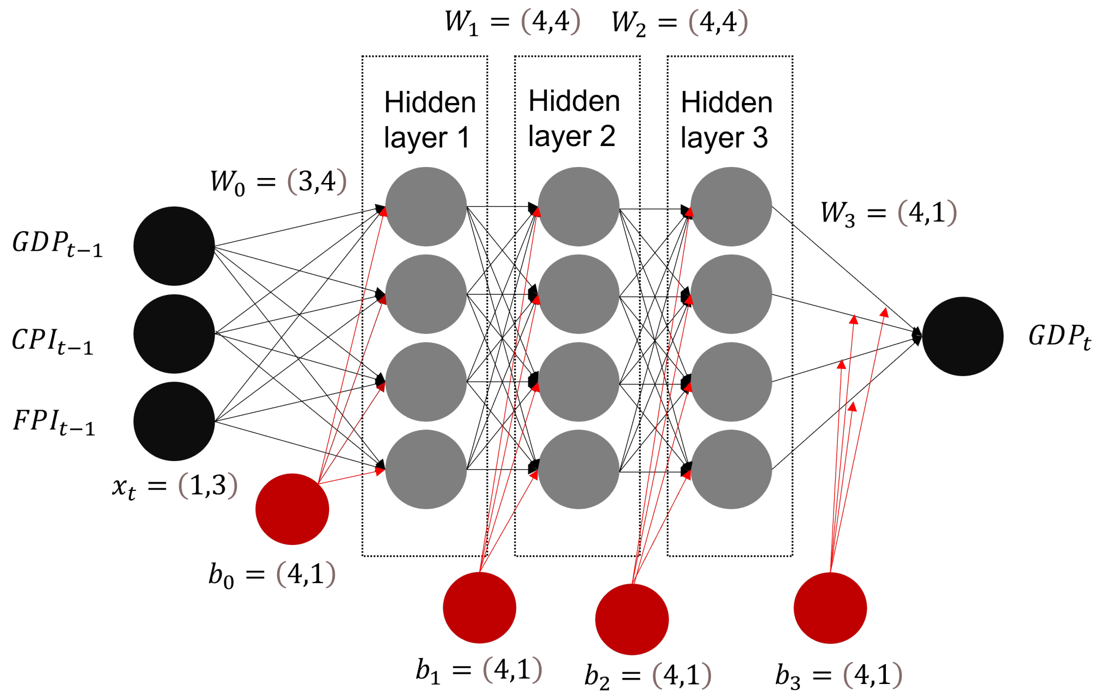


Figure 3.4: Visualization of feed-forward Deep neural networks, where three hidden layers are configured, each has 4 nodes.

Note: The $\mathcal{GP}$ prior is $f(x_t) \sim \mathcal{GP}(\mu(x_t), k(x_t, x_\tau))$, where $\mu(x_t) = W_3 h(W_2 h(W_1 h(W_0 x_t + b_0) + b_1) + b_2) + b_3$ with $x_t = (\text{GDP}_{t-1}, \text{CPI}_{t-1}, \text{FPI}_{t-1})$ denote the input/explanatory variables. GDP_t represent the observed target. Each arrow from left to right is (W), red circle is biases (b). The rectangle square indicates the activation function before feeding the results from layer to a next layer. The $(\cdot,\cdot)$ is number of row and column of Weight and biases matrices.

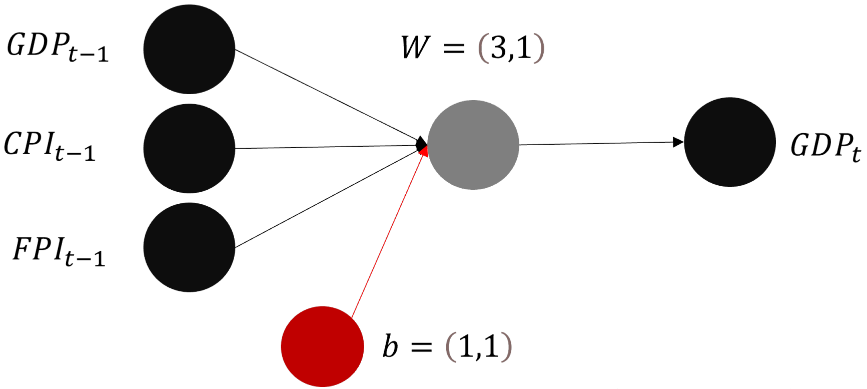


Figure 3.5: Visualization of linear mean function for $\mathcal{GP}$ prior (OLS estimates).

Note: The $\mathcal{GP}$ prior is $f(x_t) \sim \mathcal{GP}(\mu(x_t), k(x_t, x_\tau))$, where $\mu(x_t) = W x_t + b$. With $\text{GDP}_{t-1}, \text{CPI}_{t-1}, \text{FPI}_{t-1}$, denote explanatory variables. GDP_t is an observed target.

Heteroscedastic Gaussian process-VAR

The previous sub-section provides the details of how GP-VAR and it is noticeable that a GP-VAR covariance is constant over time (Homoscedastic). In economic modelling, it is often important to allow the error covariance to vary over time. Most common approach is to assume the error terms' log volatility to follow a random walk i.e. adding stochastic volatility (SV). Several studies suggest that incorporating SV improves forecasting performance (Li & Koopman (2021), Marcellino et al. (2016), Carriero et al. (2015), Kastner & Frühwirth-Schnatter (2014).) In this section, the constant error variance in GP-VAR assumption is relaxed by allowing the error distribution to be time-variant and input/predictor-dependent, that is in a non-parametric functional form. The Heteroscedastic-GP-VAR can be written as:

$$\begin{aligned} \mathbf{y}_t &= F_1(\mathbf{X}_t) + A^{-1}\epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0_N, \Sigma_t), \\ \Sigma_t &= diag\left(\left[e^{F_2(\mathbf{X}_t)}\right]^2\right). \end{aligned} \tag{3.13}$$

where $F_1(\mathbf{X}_t) = (f_{11}(\mathbf{x}_t), ..., f_{1N}(\mathbf{x}_t))'$, and $F_2(\mathbf{X}_t) = (f_{21}(\mathbf{x}_t), ..., f_{2N}(\mathbf{x}_t))'$. The additional subscription in $f_{ij}(\cdot)$ is added, where j subscription represents the number of equation in VAR. The i subscription, on the other hand, refers to number of GP-latent functions, where $i = 1, 2$ for conditional mean, and conditional variance functions, respectively. Both functions satisfy $\forall f_{1j}(\mathbf{x}), f_{2j}(\mathbf{x}) \in \mathbb{R}$, for $j = 1, .., N$. To ensure that the conditional variance produced by $F_2(\cdot)$ is always positive, the exponential transformation ($e^{(\cdot)}$) is employed. The role of inverse square matrix A is similar to (homoscedastic) GP-VAR case, that is for capturing contemporaneous relations between each responsive variables, see eq. (3.10).

The error covariance in HGP-VAR, denoted as Σ_t, is now time-variant across observations. It is important to note that the error variance is expressed in a non-parametric functional form within the framework of a Gaussian process. This formulation differs from the popular stochastic volatility (SV) model, which assumes the log variance follows a random walk AR(1) process, where the current volatility depends on its own lagged value. In contrast, the HGP-VAR model adopts a non-linear functional expression for the error variance, which is also dependent on covariates/inputs. By changing the variables and the number of lags in HGP-VAR, different patterns of heteroscedastic variance can be captured. From a pseudo out-of-sample forecast perspective, allowing the volatility to be input-dependent in HGP-VAR may offer advantages over a random walk SV model. The underlying logic is straightforward: in the case of a random walk SV, there is no observed target for forecasting, meaning the predictive distributions rely solely on previously estimated parameters. In contrast, HGP-VAR can leverage the latest observed predictors since its inputs are lagged responsive variables. This hypothesis will be explored further in Section section 3.6.

Next, the $\mathcal{GP}$ prior is specified for each unknown function in the HGP-VAR, similar to the (homoscedastic) GP-VAR case.

$$
\begin{aligned}
f_{11}(\cdot), ..., f_{1N}(\cdot) &\sim \mathcal{GP}(\mu(\cdot), k(\cdot, \cdot')),\\
f_{21}(\cdot), ..., f_{2N}(\cdot) &\sim \mathcal{GP}(\mu(\cdot), k(\cdot, \cdot')).
\end{aligned}
$$

For HGP-VAR model, the mean function in $\mathcal{GP}$ prior is assumed to be zero i.e. $\mu(\cdot) = 0$.

Selection of GP-VARs kernels

Although it is worth mentioning that there are automatic algorithm to do so, Duvenaud et al. (2013), Duvenaud (2014), Hwang et al. (2016), Steinruecken et al. (2019) but kernels for GP-VAR and GP-DNN-VAR are selected manually by ones that has the highest values of closed form log marginal likelihood since it is analytically available for those two models. For HGP-VAR, however, the log marginal likelihood is not analytically available, it is thereforeselected by the one that has the highest ELBO instead. After bunches of trails the table below summarizes kernel for GP-VARs models.

GP-VARs variables	GP-VAR	GP-DNN-VAR	HGP-VAR
GDPC1	Matern12+ RBF + White + Linear	Matern12+ RBF + White + Linear	RBF + Linear
FPI	Matern12 + RBF×Linear + White	Matern12 + RBF×Linear + White	RBF + Linear
GCEC1	Matern12 + RBF + Linear	Matern12 + RBF + Linear	RBF + Matern52
INDPRO	Matern12 + RBF + White + Linear	Matern12 + RBF + White + Linear	RBF + Linear
UNRATE	Matern12 + RBF + Linear	Matern12 + RBF + Linear	RBF + Linear
ICSA	Matern12 + RBF + White + Linear	Matern12 + RBF + White + Linear	RBF + Linear
CPIAUCSL	Matern12 + RBF×Linear + White	Matern12 + RBF×Linear + White	RBF + Linear
FEDFUNDS	Matern12 + RBF + Linear	Matern12 + RBF + Linear	RBF + Linear

Table 3.1: Kernel selection for each equation in GP-VARs models.

Kernel types	Kernel hyper-parameters
Matern12	$k(\cdot,\cdot') = \gamma^2 \exp\left(\| \cdot - \cdot' \|\right)$
Matern52	$k(\cdot,\cdot') = \gamma^2 \left(1 + \sqrt{5} \| \cdot - \cdot' \| + \frac{5}{3} \| \cdot - \cdot' \|^2\right) \exp\left(\| \cdot - \cdot' \|\right)$
Radial Basis Function (RBF)	$k(\cdot,\cdot') = \gamma^2 \exp\left(-\frac{(\cdot - \cdot')^2}{2l^2}\right)$
White	$k(\cdot_n,\cdot_m) = \delta(n,m)\gamma^2$
Linear	$k(\cdot,\cdot') = \gamma^2 \cdot \cdot'$

Table 3.2: Kernels hyper-parameters.

where $\| \cdot - \cdot' \|$ is Euclidean distance between $\cdot$ and $\cdot'$. For White kernel, $\cdot_n, \cdot_m$ denotes input at time n and m, respectively. $\delta(n,m)$ represents Kronecker delta i.e. $\delta(n,m) = 0$ if $n \neq m$ and $\delta(n,m) = 1$ if $n = m$.

3.4 Estimation and predictive density of GP-VARs models

Our GP-VAR models are estimated one equation at a time, utilizing a per equation algorithm as recently proposed by Carriero et al. (2022). Each sub-section in this section explains the estimation procedure for each equation in all types of GP-VAR models, as well as how predictive densities are computed (for $h > 1$, simulations are required).

3.4.1 Learning each equation in GP-VAR

As mention above that GP-VARs will be performed by one equation at a time. Therefore in the following sub-sections, the notation will be in a regression form similar to ones already described in section 3.2. Recall that we have data sets of T observations $\{x_t, y_t\}_{t=1}^{T}$, where y_t, x_t is typically called output and input at time t, respectively. Let $\mathbf{x} = \{x_t\}_t^T, \mathbf{y} = \{y_t\}_t^T$, the output is assumed to be a noisy realization of GP latent function $\mathbf{f}$, recall that the likelihood of Gaussian process regression can be written as:

$$y_t|f, x_t \sim \mathcal{N}(f(x_t), \sigma^2),$$
$$\mathbf{y} \sim \mathcal{N}(\mathbf{f}, I_T\sigma^2).$$

Equation above implies that the output observation is assumed to be normally distributed with a constant variance σ^2. The mean of the likelihood is input-dependent and given a $\mathcal{GP}$ prior over latent function i.e. $f(x) \sim \mathcal{GP}(\mu(x), k(x, x'))$, where $\mu(x), k(x, x')$ are mean and kernel function evaluated at input x, respectively. Similar to previous section $\mathbf{f} \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$. Denote $\mathbf{y} = (y_1, .., y_T)'$, the joint probability model is $p(\mathbf{y}, \mathbf{f}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f})$ where $p(\mathbf{f})$ is GP prior, and $p(\mathbf{y}|\mathbf{f})$ is the likelihood. With Bayes Theorem, the posterior mean and covariance of latent function at input x, and output y is easily obtained, see in section 3.2, and Rasmussen (2003).

Predictive density of GP-VAR

Once all hyperparameters and likelihood variance from each equation in GP-VAR are estimated, one can obtain the forecast density conditional on new input points i.e. $p(y_{T+1}|\mathbf{y}, \mathbf{x}^\star)$. To achieve so, we first derive the joint distribution between already estimated model $p(\mathbf{y}_{T+1})$, where $\mathbf{y}_{T+1}$ denotes vector $(y_1, ..., y_{T+1})'$.

$$p(\mathbf{y}_{T+1}) = \mathcal{N}(\mathbf{0}, \mathbf{C}_{T+1}), \tag{3.14}$$

where:

$$\mathbf{C}_{T+1} = \begin{pmatrix} \mathbf{K_{ff}} & \mathbf{K_{ff^\star}} \\ \mathbf{K_{f^\star f}} & \mathbf{K_{f^\star f^\star}} \end{pmatrix}$$

Another way of writing the forecast density at new input points is:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^\star \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K_{ff}} & \mathbf{K_{ff^\star}} \\ \mathbf{K_{f^\star f}} & \mathbf{K_{f^\star f^\star}} \end{bmatrix}\right)$$

where $\mathbf{K_{ff}} = k(\mathbf{x}, \mathbf{x}')$ is kernel values evaluated at every regression input points, $\mathbf{K_{ff^\star}} = k(\mathbf{x}, \mathbf{x}^\star)$ is kernel value evaluated between every input points and new/predictive input points, and lastly $\mathbf{K_{f^\star f^\star}} = k(\mathbf{x}^\star, \mathbf{x}^{\star'})$ is kernel values evaluated between predictive input points.

Since the GP is chosen to be a prior on top of unknown function, the forecast density is also Gaussian and tractable, where it leads to normal distribution that can be fully described by the mean and covariance as followed: $\mathbf{f}^\star | \mathbf{x}, \mathbf{y}, \mathbf{x}^\star \sim \mathcal{N}(\overline{\mathbf{f}}^\star, \text{cov}(\mathbf{f}^\star))$, where:

$$\begin{aligned} \overline{\mathbf{f}}^\star &= \mathbf{K_{f^\star f}} \mathbf{K}^{-1} \mathbf{y}, \\ \text{cov}(\mathbf{f}^\star) &= \mathbf{K_{f^\star f^\star}} - \mathbf{K_{f^\star f}} \mathbf{K}^{-1} \mathbf{K_{ff^\star}}. \end{aligned} \tag{3.15}$$

Notice from posterior predictive density above is in the regression context. To measure the uncertainty from GP-VAR context one can recover GP-VAR covariance from $A^{-1} \Sigma A^{-1'}$ where $\Sigma = diag(\sigma_1^2, ..., \sigma_N^2)$. The GP-VAR forecast density can be summarized in table 3.3 below.

3.4.2 Learning each equation in GP-DNN-VAR.

Given that some macroeconomic observations are away from zero (after stationary transformation). Assuming zero mean in $\mathcal{GP}$ prior may or may not be suitable i.e. $f(x) \sim \mathcal{GP}(m(x), k(x, x'))$, where $m(x) = 0$. Manipulating the mean function in $\mathcal{GP}$ prior might help us in solving the target more efficiently. Even if the OLS weights in the mean function can improve the accuracy of the GP-models. However, the example

```
Predictive density for GP-VAR
one step ahead predictive densities
for $s = 1 : S$
    for $j = 1 : N$
        $p(\mathbf{f}_j^\star|\mathbf{x}_{T+1}) \sim \mathcal{N}(\overline{\mathbf{f}}_j^\star|cov(\mathbf{f}_j^\star))$, eq. (3.15).
        -compute $A^{(s)}$,
    end for $j = 1 : N$
end for $s = 1 : S$

compute GP-VAR covariance:
  $\mathbf{H}^{(s)} = A_{(s)}^{-1}\Sigma A_{(s)}^{-1'}$, where $\Sigma = diag(\sigma_1^2, ..., \sigma_N^2)$
set $\mathbf{F}_{T+1} = (\mathbf{f}_1^\star, ..., \mathbf{f}_N^\star)'$
GP-VAR covariance: $\mathbf{F}_{T+1} = \mathbf{F}_{T+1} + \mathbf{H}^{\frac{1}{2}}\varepsilon$, where $\varepsilon \sim \mathcal{N}(0,1)$

Two-and-more-steps-ahead predictive densities with Monte Carlo Estimates
for $h = 2 : 12$
    for $s = 1 : S$
        for $j = 1 : N$
            $\int p(\mathbf{f}_j^\star|\mathbf{x}_{T+h-1}^{(s)}, .., \mathbf{x}_{T+h-1}^{(s)})d\mathbf{x}_{T+h-1}, .., \mathbf{x}_{T+h-1}^{(s)}$.
            $\approx \sum_{s=1}^{S} p(\mathbf{f}_j^\star|\mathbf{x}_{T+h-1}^{(s)}, .., \mathbf{x}_{T+h-1}^{(s)})$.
    set $\mathbf{F}_{T+h} = (\mathbf{f}_1^\star, ..., \mathbf{f}_N^\star)'$
    GP-VAR forecasts: $\mathbf{F}_{T+h} = \mathbf{F}_{T+h} + \mathbf{H}^{\frac{1}{2}}\varepsilon$, where $\varepsilon \sim \mathcal{N}(0,1)$.
        end for $j = 1 : N$
    end for $s = 1 : S$
end for $h = 2 : 12$
```

Table 3.3: Algorithm to simulate predictive density for GP-VAR.
Note: S is total number of samples, N is total number of equations in VAR, $A_{(s)}^{-1}$ is inverse of lower-triangular A at sample number (s).

below demonstrates how the feed-forward deep neural network with three hidden layers can be employed:

$$m(x) = W_3 h(W_2 h(W_1 x + b_1) + b_2) + b_3. \tag{3.16}$$

where W_i is weight matrices of layer i, the b_i is biases, and $h(\cdot)$ is activation function in neural network, sigmoid or ReLu for instances. For notational clarity I suppress all weight matrices and biases into vector ϕ i.e. $\phi = (W_i, b_i)$ for $i = 1, 2, 3$. This kind of manipulating $\mathcal{GP}$ prior has been explored in machine learning field, see for example Fortuin et al. (2019).

Approximating these weights and biases are uncomplicated because of two following reasons. First it is still in GPR expression, meaning that the log marginal likelihood is still available in closed form. Secondly the deep neural network is augmented in GP prior i.e. $f(\cdot) \sim \mathcal{GP}(m(\cdot), k(\cdot,\cdot))$, implying that it can be approximated simultaneously with hyper-parameters, before or even after kernel hyper-parameters are known. Through multiple tests, I have found that the order of approximation does not significantly affect the results, but approximating them separately is computationally more efficient. The GPD-R algorithm is summarized in table 3.4.

```
Algorithm for Gaussian process regression with Deep neural network mean function
Inputs:
    observed evidence, and inputs $\mathbf{y}, \mathbf{x}$
While not converged do:
  Kernel hyperparameters and likelihood variance:
    - maximize $\log p(\mathbf{y})$ w.r.t kernel hyperparameters and likelihood variance, $\theta, \sigma^2$, see eq. (3.6)
end while.
Deep mean function:
    - maximize a closed form of $\log p(\mathbf{y})$ w.r.t mean function of $\mathcal{GP}$ prior $m(\mathbf{x})$                , see eq. (3.6)
Outputs: Optimized kernel hyper-parameters $\theta$, likelihood variance $\sigma^2$, and non-parametric functions $\mathbf{f}$.
```

Table 3.4: Algorithm for Gaussian process regression with deep neural network mean function.

It is worth noting that with the mean function in $\mathcal{GP}$ prior is not zero, the log marginal likelihood now becomes:

$$\log p(\mathbf{y}) = -\frac{1}{2}(\mathbf{y} - m(\mathbf{x}))'(\mathbf{K} + \sigma^2 I)^{-1}(\mathbf{y} - m(\mathbf{x})) - \frac{1}{2}\log|\mathbf{K} + \sigma^2 I| - \frac{T}{2}\log 2\pi. \quad (3.17)$$

where $m(\mathbf{x})$ is deep neural network mean function evaluated at input $\mathbf{x}$. Additionally the predictive distribution at new input (predictors) $\mathbf{x}^\star$ is $\mathbf{f}^\star|\mathbf{x}, \mathbf{y}, \mathbf{x}^\star \sim \mathcal{N}(\overline{\mathbf{f}}^\star, \mathrm{cov}(\mathbf{f}^\star))$, which is:

$$\overline{\mathbf{f}}^\star = m(\mathbf{x}^\star) + \mathbf{K}_{\mathbf{f}^\star\mathbf{f}} \left[\mathbf{K}\right]^{-1} \left(\mathbf{y} - m(\mathbf{x})\right). \quad (3.18)$$

The covariance of predictive distribution is still similar to those in eq. (3.15).

Predictive density of GP-DNN-VAR

The predictive of GP-DNN-VAR is identical in the case of GP-VAR except the predictive mean in each equation can be expressed as:

$$
\begin{aligned}
p(\mathbf{f}^{\star}|\bullet) &\sim \mathcal{N}(\overline{\mathbf{f}}^{\star}|\mathrm{cov}(\mathbf{f}^{\star})),\\
\overline{\mathbf{f}}^{\star} &= m(\mathbf{x}^{\star}) + \mathbf{K}_{\mathbf{f}^{\star}\mathbf{f}}\mathbf{K}^{-1}(y - m(\mathbf{x})),\\
\mathrm{cov}(\mathbf{f}^{\star}) &= \mathbf{K}_{\mathbf{f}^{\star}\mathbf{f}^{\star}} - \mathbf{K}_{\mathbf{f}^{\star}\mathbf{f}}\mathbf{K}^{-1}\mathbf{K}_{\mathbf{f}\mathbf{f}^{\star}}.
\end{aligned}
$$

Therefore predictive densities for GP-DNN-VAR can be obtained by simulating similarly to GP-VAR case except there is additional $m(\mathbf{x}^{\star}), m(\mathbf{x})$ in $\overline{\mathbf{f}}^{\star}$, see table 3.3.

3.4.3 Learning each equation in HGP-VAR.

Recall from eq. (3.13) that there are two latent-GP functions in HGP-VAR model, an additional GP function results in the log marginal likelihood is no longer available in a closed form. Thus an approximation approach is required. However, there are multiple available options. For sampling-based method such as MCMC, see Goldberg et al. (1997), variational inference Lázaro-Gredilla & Titsias (2011), Laplace approximation Jylänki et al. (2011) and expectation propagation Hernández-Lobato et al. (2014). Despite that, the most recent approximating method enables us to approximate the model by utilizing the *sparse Gaussian process.* This is by far the best choice to learn such model in terms of robustness and computational efficiency, Saul et al. (2016). Heteroscedastic Gaussian Process (HGP) contains two GP latent functions where first is parameterizing the mean of the likelihood and second is for the time-varying variance. In each equation in HGP-VAR, the likelihood can be expressed as follow:

$$
\begin{aligned}
f_1(x_t) &\sim \mathcal{GP}(0, k_1(x_t, x_\tau)),\\
f_2(x_t) &\sim \mathcal{GP}(0, k_2(x_t, x_\tau)),\\
y_t|f_1, f_2, x_t &\sim \mathcal{N}(f_1(x_t), \left[e^{f_2(x_t)}\right]^2).
\end{aligned}
\tag{3.19}
$$

where $\forall f_1(\cdot), f_2(\cdot) \in \mathbb{R}$. The second GP latent function $f_2(\cdot)$ implies that it is input-dependent. The time-varying standard deviation for likelihood approximated by second GP latent function is guaranteed to be positive by exponential transformation $e^{f_2(\cdot)}$.

Before delving into how Heteroscedastic Gaussian process regression learns the two GP-latent functions, it is necessary to understand the concept of sparse Gaussian process regression (SGPR). As mentioned in Equation (8), learning Gaussian process regression involves inverting the $T \times T$ kernel matrix $\mathbf{K_{ff}}$, resulting in a computational complexity of $\mathcal{O}(T^3)$. This becomes computationally challenging when the number of observations, T, is large. To address this issue, sparse Gaussian process regression introduces the concept of *inducing* variables.

The key idea behind SGPR is to use a smaller set of inducing inputs, denoted as m-dimensional vectors, where $m \ll T$, to represent the entire set of T observed data points. This is based on the rationale that not all observed inputs are equally informative in mapping to the responsive variable in regression. The formulation of SGPR is as follows:

$$\begin{aligned} f_1(x_t) &\sim \mathcal{GP}(0, k(x_t, x_\tau)), \\ y_t | f_1, x_t &\sim \mathcal{N}(f_1(x_t), \sigma^2). \end{aligned}$$

Notice that it is exactly similar to GPR but the way it learns parameter is different. I will now describe how a single GP-latent function can be approximated from SGPR, then later on in this section, we will adding another GP-latent functions to parameterize the time-varying residual standard deviation. For now denote the inducing-inputs $\mathbf{z} = \{z_t\}_{t=1}^m$ and specify zero mean $\mathcal{GP}$ prior for the inducing inputs then we have:

$$p(\mathbf{u}) = \mathcal{N}(\mathbf{u}|0, \mathbf{K_{uu}}) \tag{3.20}$$

where $\mathbf{u} = (u(z_1), .., u(z_m))' \in \mathbb{R}^m$ and $\mathbf{K_{uu}} = k(\mathbf{z}, \mathbf{z}')$ is $m \times m$ covariance of inducing function. The joint distribution of $p(\mathbf{y}, \mathbf{f})$ with auxiliary inducing-variables $\mathbf{u}$ now becomes, see Titsias (2009), Hensman et al. (2013), Saul et al. (2016):

$$p(\mathbf{y}, \mathbf{f}, \mathbf{u}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u}). \tag{3.21}$$

The joint distribution of second term $p(\mathbf{f}, \mathbf{u})$ in eq. (3.21) is in the form:

$$\begin{pmatrix} \mathbf{f} \\ \mathbf{u} \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{\mathbf{f}} \\ \mu_{\mathbf{u}}, \end{bmatrix} \begin{bmatrix} \mathbf{K_{ff}} & \mathbf{K_{fu}} \\ \mathbf{K_{uf}} & \mathbf{K_{uu}} \end{bmatrix} \right) \tag{3.22}$$

where $\mu_{\mathbf{f}}, \mu_{\mathbf{u}}$ denote the marginal means of function $\mathbf{f}$, and $\mathbf{u}$, respectively. $\mathbf{K_{ff}}$, $\mathbf{K_{fu}}$, $\mathbf{K_{uf}}$, $\mathbf{K_{uu}}$ are (cross) covariance matrices. To be more precise, $\mathbf{K_{ff}} = k(\mathbf{x}, \mathbf{x}')$ is $T \times T$ covariance evaluated at every inputs. $\mathbf{K_{fu}} = \mathbf{K_{uf}}'$, with $k(\mathbf{x}, \mathbf{z})$, and finally $\mathbf{K_{uu}} = k(\mathbf{z}, \mathbf{z}')$. The conditional distribution which is also Gaussian can be written as:

$$\begin{aligned} p(\mathbf{f}, \mathbf{u}) &= p(\mathbf{f}|\mathbf{u})p(\mathbf{u}), \\ p(\mathbf{f}|\mathbf{u}) &\sim \mathcal{N}(\mu_{\mathbf{f}} + \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}(\mathbf{u} - \mu_{\mathbf{u}}), \mathbf{K_{ff}} - \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\mathbf{K_{uf}}). \end{aligned} \tag{3.23}$$

Sparse Gaussian process aims to approximate inducing inputs $\mathbf{u}$. Suppose zero mean $\mathcal{GP}$ prior is employed in $\mathbf{f}$ i.e. $\mu_{\mathbf{f}} = 0$ (see eq. (3.23)) then the distribution of latent function for likelihood conditional on the inducing variables can be written in similar fashion:

$$\begin{aligned} p(\mathbf{f}|\mathbf{u}) &= \mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{S}), \\ \mathbf{m} &= \mathbf{\Phi}\mathbf{u}, \\ \mathbf{S} &= \mathbf{K_{ff}} - \mathbf{\Phi}\mathbf{K_{uu}}\mathbf{\Phi}'. \end{aligned} \tag{3.24}$$

where $\mathbf{\Phi} \equiv \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}$. According to eq. (3.24), the conditional prior of likelihood latent function on inducing variables $p(\mathbf{f}|\mathbf{u})$ are fully specified if and only if the inducing variables are known i.e. $\mathbf{z} = (z_1, .., z_m)'$. Since exact posterior of two latent functions are intractable $p(\mathbf{f}, \mathbf{u})$. The more convenient and tractable of joint variational distribution $q_\theta(\mathbf{f}, \mathbf{u})$ is introduced, which assume to factorize as:

$$q_\phi(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u})q_\phi(\mathbf{u}). \tag{3.25}$$

It is now obvious that SGPR never intends to estimate $\mathbf{u}$ function but instead approximate it with normal distribution which can be written as:

$$q_\phi(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{b}, \mathbf{W}). \tag{3.26}$$

where subscription $\phi = \{\mathbf{b}, \mathbf{W}\}$, which are variational parameters (mean and covariance) of normal distribution parameterizing inducing inputs. With these variational parameters in hand, one can obtain the marginal variational distribution over $\mathbf{f}$ by integrating out inducing variables $\mathbf{u}$. Fortunately Hensman et al. (2013), Matthews et al. (2017) showed that this has analytical expression as follow, see also eq. (3.24):

$$\begin{aligned} q_\phi(\mathbf{f}) &= \int q_\phi(\mathbf{f}, \mathbf{u}) d\mathbf{u} = \mathcal{N}(\mathbf{f}|\mu, \mathbf{\Sigma}), \\ \mu &= \mathbf{\Phi}\mathbf{b}, \\ \mathbf{\Sigma} &= \mathbf{K_{ff}} - \mathbf{\Phi}(\mathbf{K_{uu}} - \mathbf{W})\mathbf{\Phi}'. \end{aligned} \tag{3.27}$$

Next paragraph demonstrates how required parameters in sparse GPR model can be obtained by maximizing evidence lower bound to the marginal likelihood w.r.t. $\phi = \{\mathbf{b}, \mathbf{W}\}$.

Evidence lower bound to the log marginal likelihood This subsection briefly introduces how to derive the evidence lower bound to the log marginal likelihood for sparse Gaussian process. The objective is to minimize the Kullback-Leibler (KL) divergence between already defined approximate distribution $q_\phi(\mathbf{f}, \mathbf{u})$ to the exact posterior distribution $p(\mathbf{f}, \mathbf{u}|\mathbf{y})$ by first write the KL:

$$\begin{aligned} KL\big(q_\phi(\mathbf{f}, \mathbf{u})||p(\mathbf{f}, \mathbf{u}|\mathbf{y})\big) &= \mathbb{E}_{q_\phi(\mathbf{f}, \mathbf{u})}\Big[\log \frac{q_\phi(\mathbf{f}, \mathbf{u})}{p(\mathbf{f}, \mathbf{u}|\mathbf{y})}\Big], \\ &= \log p(\mathbf{y}) + \mathbb{E}_{q_\phi(\mathbf{f}, \mathbf{u})}\Big[\log \frac{q_\phi(\mathbf{f}, \mathbf{u})}{p(\mathbf{f}, \mathbf{u}|\mathbf{y})}\Big], \\ &= \log p(\mathbf{y}) - \mathcal{L}(\phi). \end{aligned} \tag{3.28}$$

with

$$\mathcal{L}(\phi) \equiv \mathbb{E}_{q_\phi(\mathbf{f}, \mathbf{u})}\Big[\log \frac{p(\mathbf{f}, \mathbf{u}, \mathbf{y})}{q_\phi(\mathbf{f}, \mathbf{u})}\Big]. \tag{3.29}$$

Titsias (2009), Hensman et al. (2013), Leibfried et al. (2020) provides how the final formula for this evidence lower bound is derived. Here provide for convenience:

$$\mathcal{L}(\phi) = \log \big[\mathcal{N}(\mathbf{y}|0, \sigma^2 I + \mathbf{K_{fu}K_{uu}^{-1}K_{ff}})\big] - \frac{1}{2\sigma^2} Tr\big[\mathbf{K_{ff}} - \mathbf{K_{fu}K_{uu}^{-1}K_{ff}}\big]. \tag{3.30}$$

Finally the necessary parameters can be approximated by maximizing ELBO above with respect to variational parameters of inducing variables $\phi = \{\mathbf{b}, \mathbf{W}\}$, and the rest of parameters i.e. residual variance, and kernel hyper-parameters (σ^2, θ). If ELBO reaches its maximum if and only if the approximate posterior distribution equals the true posterior distribution everywhere! i.e. $q_\phi(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}, \mathbf{u})$. The KL becomes zero, and as a result the ELBO recovers to the log marginal likelihood exactly.

Now we go back to our HGP-R, instead of only one GP-latent function as in SGPR case, there are now two latent functions that needs to be approximated i.e. $f_1(\cdot), f_2(\cdot)$, eq. (3.19). Adding one or even more should be straightforward as long as those GP-functions are independent and each function has its own inducing points i.e. $\{\mathbf{u}_i\}_{i=1}^2$. Now denote $\mathbf{f}_1 = f_1(\mathbf{x}), \mathbf{f}_2 = f_2(\mathbf{x})$, $\mathbf{u}_1 = \big(u_1(z_{1,1}), ..., u_1(z_{1,m})\big)'$, $\mathbf{u}_2 = \big(u_2(z_{2,1}), ..., u_2(z_{2,m})\big)'$, where $z_{j,m}$ is m inducing-inputs of latent function j. The joint distribution of both latent functions factorizes in the form:

$$p(\mathbf{f}_1, \mathbf{f}_2|\mathbf{u}_1, \mathbf{u}_2) = p(\mathbf{f}_1|\mathbf{u}_1)p(\mathbf{f}_2|\mathbf{u}_2). \tag{3.31}$$

The log marginal likelihood is:

$$\begin{aligned}\log p(\mathbf{y}) = \log \int & p(\mathbf{y}|\mathbf{f}_1, \mathbf{f}_2)p(\mathbf{f}_1, \mathbf{f}_2|\mathbf{u}_1, \mathbf{u}_2),\\ & \times p(\mathbf{u}_1)p(\mathbf{u}_2)d\mathbf{f}_1 d\mathbf{f}_2 d\mathbf{u}_1 d\mathbf{u}_2.\end{aligned}$$

Similar to sparse GPR, the HGP model is fully specified by first defining the approximate distribution for both inducing distributions i.e. $\mathbf{u}_1, \mathbf{u}_2$ which can be written as:

$$p(\mathbf{f}_1, \mathbf{f}_2, \mathbf{u}_1, \mathbf{u}_2|\mathbf{y}) \approx q_\phi(\mathbf{f}_1, \mathbf{f}_2, \mathbf{u}_1, \mathbf{u}_2). \tag{3.32}$$

where,

$$q_{\phi}(\mathbf{f}_1, \mathbf{f}_2, \mathbf{u}_1, \mathbf{u}_2) = p(\mathbf{f}_1|\mathbf{u}_1)p(\mathbf{f}_2|\mathbf{u}_2)q_{\phi_1}(\mathbf{u}_1)q_{\phi_2}(\mathbf{u}_2). \tag{3.33}$$

Notice again that eq. (3.33) above is exactly the same to eq. (3.25) except that there are now two independent latent functions and two inducing inputs which are approximated by normal distributions $q_{\phi_1}(\mathbf{u}_1), q_{\phi_2}(\mathbf{u}_2)$ parameterized by its own variational parameters $\phi = (\phi_1, \phi_2)$. With all these ingredients we are now ready to obtain those parameters by maximizing the evidence lower bound to the log marginal likelihood:

$$\begin{aligned}\log p(\mathbf{y}) &= \log \int p(\mathbf{y}|\mathbf{f}_1, \mathbf{f}_2)p(\mathbf{f}_1|\mathbf{u}_1)p(\mathbf{f}_2|\mathbf{u}_2)p(\mathbf{u}_1)p(\mathbf{u}_2)d\mathbf{f}_1 d\mathbf{f}_2 d\mathbf{u}_1 d\mathbf{u}_2,\\ &\geq \int q(\mathbf{f}_1)q(\mathbf{f}_2)\log p(\mathbf{y}|\mathbf{f}_1, \mathbf{f}_2)d\mathbf{f}_1 d\mathbf{f}_2 - \Big[KL\big(q(\mathbf{f}_1)||p(\mathbf{u}_1)\big) + KL\big(q(\mathbf{u}_1)||p(\mathbf{u}_2)\big)\Big].\end{aligned} \tag{3.34}$$

where $q(\mathbf{f}_j) = \int p(\mathbf{f}_j|\mathbf{u}_j)q(\mathbf{u}_j)d\mathbf{u}_j$ for $j = 1, 2$, which again similar to eq. (3.27). In fact one can extend to more than two latent functions by replicating the identical expression with the assumption that all j latent functions i.e. $\mathbf{f}_1, ..., \mathbf{f}_j$ are priori independent. Thus posterior is variationally factorized. $KL\big(q(\cdot)||p(\cdot)\big)$ denotes the KL divergence between $q(\cdot)$ and $p(\cdot)$. The variational posterior for inducing distribution are $q_{\phi_1}(\mathbf{u}_1) = \mathcal{N}(\mathbf{u}_1|\mathbf{b}_1, \mathbf{W}_1)$, and $q_{\phi_2}(\mathbf{u}_2) = \mathcal{N}(\mathbf{u}_2|\mathbf{b}_2, \mathbf{W}_2)$, respectively. Finally the posterior of each latent function conditional on inducing distributions are:

$$\begin{aligned}q_{\phi_1}(\mathbf{f}_1) = p(\mathbf{f}_1|\mathbf{u}_1) &= \mathcal{N}(\mathbf{f}_1|\mu_1, \mathbf{\Sigma}_1),\\ \mu_1 &= \mathbf{\Phi}_1\mathbf{b}_1,\\ \mathbf{\Sigma}_1 &= \mathbf{K}_{\mathbf{fu}_1} - \mathbf{\Phi}_1(\mathbf{K}_{\mathbf{u}_1\mathbf{u}_1} - \mathbf{W}_1)\mathbf{\Phi}_1',\end{aligned} \tag{3.35}$$

where $\mathbf{\Phi}_1 \equiv \mathbf{K}_{\mathbf{fu}_1}\mathbf{K}_{\mathbf{u}_1\mathbf{u}_1}^{-1}$. The similar fashion applies for $q_{\phi_2}(\mathbf{f}_2)$. This also implies that two latent functions of heteroscedastic Gaussian process regression are estimated by finding variational parameters (mean and variance) of inducing distribution i.e. $\phi = (\phi_1, \phi_2) = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{W}_1, \mathbf{W}_2\}$ that actually maximize evidence lower bound in eq. (3.34). The two KL terms in eq. (3.34) are available in closed form since it is KL distance between two normal distributions. Finally the term $\int q(\mathbf{f}_1)q(\mathbf{f}_2)\log(\mathbf{y}|\mathbf{f}_1,\mathbf{f}_2)d\mathbf{f}_1 d\mathbf{f}_2$, which Lázaro-Gredilla & Titsias (2011), and Saul et al. (2016) already derived a very final simple form:

$$\begin{aligned}\int q(\mathbf{f}_1)q(\mathbf{f}_2)\log p(\mathbf{y}|\mathbf{f}_1,\mathbf{f}_2)d\mathbf{f}_1 d\mathbf{f}_2 &= \int \mathcal{N}(\mathbf{f}_1|\mu_1,\mathbf{\Sigma}_1)\mathcal{N}(\mathbf{f}_2|\mu_2,\mathbf{\Sigma}_2)\log\mathcal{N}\big(\mathbf{y}|\mathbf{f}_1,\exp(\mathbf{f}_2)\big)d\mathbf{f}_1 d\mathbf{f}_2,\\ &= \log\mathcal{N}\left(\mathbf{y}|\mu_1,\exp\left(\mu_2-\frac{\mathbf{\Sigma}_2}{2}\right)\right)-\frac{\mathbf{\Sigma}_2}{4}-\frac{\mathbf{\Sigma}_1\exp(-\mu_2+\frac{\mathbf{\Sigma}_2}{2})}{2}.\end{aligned} \tag{3.36}$$

Predictive density of HGP-VAR.

The forecast density for HGP-VAR begins with the approximate distribution for each equation in VAR, from eq. (3.33).

$$q_\phi(\mathbf{f}_1,\mathbf{f}_2,\mathbf{u}_1,\mathbf{u}_2) = p(\mathbf{f}_1|\mathbf{u}_1)p(\mathbf{f}_2|\mathbf{u}_2)q_{\phi_1}(\mathbf{u}_1)q_{\phi_2}(\mathbf{u}_2).$$

With the assumption of two independent latent functions $\mathbf{f}_1,\mathbf{f}_2$, the posterior after the variational lower bound is maximized becomes:

$$\begin{aligned}p(\mathbf{f}^\star|\mathbf{y},\mathbf{x},\mathbf{x}^\star) &= \int p(\mathbf{f}^\star|\mathbf{x},\mathbf{f})p(\mathbf{f}|\mathbf{u}_1)p(\mathbf{u}_1|\mathbf{y})d\mathbf{f}d\mathbf{u}_1,\\ &\approx \int p(\mathbf{f}^\star|\mathbf{u}_1)q(\mathbf{u}_1)d\mathbf{u}_1,\\ &= q(\mathbf{f}^\star),\\ q_{\phi_1}(\mathbf{f}_1^\star) &\sim \mathcal{N}(\mathbf{K}_{\mathbf{f}_1^\star\mathbf{u}_1}\mathbf{K}_{\mathbf{u}_1\mathbf{u}_1}^{-1}\mathbf{b}_1,\mathbf{K}_{\mathbf{f}_1^\star\mathbf{u}_1}-\mathbf{K}_{\mathbf{f}_1^\star\mathbf{u}_1}\mathbf{K}_{\mathbf{u}_1\mathbf{u}_1}^{-1}\left(\mathbf{K}_{\mathbf{u}_1\mathbf{u}_1}^{-1}-\mathbf{W}_1\right)\mathbf{K}_{\mathbf{u}_1\mathbf{u}_1}^{-1}\mathbf{K}_{\mathbf{f}_1^\star\mathbf{u}_1}),\\ q_{\phi_2}(\mathbf{f}_2^\star) &\sim \mathcal{N}(\mathbf{K}_{\mathbf{f}_2^\star\mathbf{u}_2}\mathbf{K}_{\mathbf{u}_2\mathbf{u}_2}^{-1}\mathbf{b}_2,\mathbf{K}_{\mathbf{f}_2^\star\mathbf{u}_2}-\mathbf{K}_{\mathbf{f}_2^\star\mathbf{u}_2}\mathbf{K}_{\mathbf{u}_2\mathbf{u}_2}^{-1}\left(\mathbf{K}_{\mathbf{u}_2\mathbf{u}_2}^{-1}-\mathbf{W}_2\right)\mathbf{K}_{\mathbf{u}_2\mathbf{u}_2}^{-1}\mathbf{K}_{\mathbf{f}_2^\star\mathbf{u}_2}).\end{aligned} \tag{3.37}$$

where $\mathbf{b}_j,\mathbf{W}_j$ are mean and covariance parameterizing approximate distribution of inducing variables for j latent function, i.e. $q_{\phi_1}(\mathbf{u}_1)=\mathcal{N}(\mathbf{u}_1|\mathbf{b}_1,\mathbf{W}_1)$, and $q_{\phi_2}(\mathbf{u}_2)=\mathcal{N}(\mathbf{u}_2|\mathbf{b}_2,\mathbf{W}_2)$. For one-step-ahead forecast, notice that after inducing variables $\mathbf{u}_1,\mathbf{u}_2$, variational parameters ϕ_1,ϕ_2 of two GP-latent functions, and kernel hyper-parameters are approximated. Predictive density is readily to be computed. For two or more-step ahead it requires Monte Carlo Approximation which the algorithm can be summarized in table 3.5.

Predictive density for HGP-VAR

one step ahead predictive densities

```
for $s = 1 : S$
    for $j = 1 : N$
        -$q_{\phi_1}(\mathbf{f}_1^\star)$, eq. (3.37),
        -$e^{q_{\phi_2}(\mathbf{f}_2^\star)}$, eq. (3.37),
        -obtain $A^{(s)}$,
    end for $j = 1 : N$
end for $s = 1 : S$
```

compute HGP-VAR covariance:

```
    $\mathbf{H}_{T+1}^{(s)} = A_{(s)}^{-1}\Sigma_{T+1}^{(s)}A_{(s)}^{-1\prime}$, where $\Sigma_{T+1}^{(s)} = diag\left(\left[e_{(s)}^{\mathbf{f}_{2jT+1}^\star}\right]^2, ..., \left[e_{(s)}^{\mathbf{f}_{2NT+1}^\star}\right]^2\right)$
 set $\mathbf{F}_{T+1} = (\mathbf{f}_1^\star, ..., \mathbf{f}_N^\star)'$
HGP-VAR covariance: $\mathbf{F}_{T+1} = \mathbf{F}_{T+1} + \mathbf{H}_{T+1}^{\frac{1}{2}}\varepsilon$, where $\varepsilon \sim \mathcal{N}(0,1)$
```

Two-and-more-steps-ahead predictive densities with Monte Carlo Estimates

```
for $h = 2 : 12$
    for $s = 1 : S$
        for $j = 1 : N$
            $\int q_{\phi_1}(\mathbf{f}_1^\star|\mathbf{x}_{T+h-1}^{(s)}, .., \mathbf{x}_{T+h-1}^{(s)})d\mathbf{x}_{T+h-1}, .., \mathbf{x}_{T+h-1}^{(s)}.$
            $\int e^{q_{\phi_2}(\mathbf{f}_2^\star|\mathbf{x}_{T+h-1}^{(s)},..,\mathbf{x}_{T+h-1}^{(s)})d\mathbf{x}_{T+h-1},..,\mathbf{x}_{T+h-1}^{(s)}}.$
            $\approx \sum_{s=1}^S q_{\phi_1}(\mathbf{f}_1^\star|\mathbf{x}_{T+h-1}^{(s)}, .., \mathbf{x}_{T+h-1}^{(s)}).$
            $\approx \sum_{s=1}^S e^{q_{\phi_2}(\mathbf{f}_2^\star|\mathbf{x}_{T+h-1}^{(s)},..,\mathbf{x}_{T+h-1}^{(s)})}.$
    set $\mathbf{F}_{T+h} = (\mathbf{f}_1^\star, ..., \mathbf{f}_N^\star)'$
    set $\mathbf{H}_{T+h} = A_{(s)}^{-1}\Sigma_{T+h}^{(s)}A_{(s)}^{-1\prime}$
    HGP-VAR forecasts: $\mathbf{F}_{T+h} = \mathbf{F}_{T+h} + \mathbf{H}_{T+h}^{\frac{1}{2}}\varepsilon$, where $\varepsilon \sim \mathcal{N}(0,1)$.
        end for $j = 1 : N$
    end for $s = 1 : S$
end for $h = 2 : 12$
```

Table 3.5: Algorithm to simulate predictive density for HGP-VAR.
Note: S is total number of samples, N is total number of equations in VAR, $A^{(s)}, A_{(s)}^{-1}$ is squared matrix A and inverse of lower-triangular A at sample number (s), respectively. $\left[e_{(s)}^{\mathbf{f}_{2jT+1}^\star}\right]^2$ is exponential predictive variance of second latent-GP function of equation j in VAR at time $T+1$

3.4.4 Model summarizations

The goal of this sub-section is to provide an overall picture of included models, estimation method and parameters. As for GP-VARs models, we suppressed all hyper-parameters in kernel department into θ. The summary of models can be seen from the table 3.6 below.

Model Acronyms	Model descriptions	Estimation method	Parameters
BVAR-SV	Bayesian VAR with stochastic volatility	MCMC (Gibbs sampling)	$\beta, \mathbf{A}, h_{t=0}^{T}, \mu, \phi, \sigma_h^2.$
GP-VAR	Gaussian process VAR	maximize closed form log marginal likelihood	$\theta, \Sigma, \mathbf{A}$
GP-DNN-VAR	Gaussian process VAR with Deep-neural networks as mean function	maximize closed form log marginal likelihood	$\theta, \Sigma, W, b, \mathbf{A}$
HGP-VAR	Heteroscedastic Gaussian process VAR	maximize ELBO	$(\phi_1 = \mathbf{b}_1, \mathbf{W}_1, \phi_2 = \mathbf{b}_2, \mathbf{W}_2), \theta, \Sigma_t, \mathbf{A}$

Table 3.6: Acronyms of models, model descriptions, approximation method and parameters.

Optimization

The optimization of the evidence lower bound in HGP models and the estimation of feed-forward neural networks can be computationally challenging. To address this issue, stochastic gradient descent (SGD) is commonly used to speed up the computation of ELBO and train deep neural networks. For GP-DNN-VAR, we use the Adam algorithm Kingma & Welling (2014), while for HGP-VAR, we utilize the Adaptive subgradient (AdaGrad) Duchi et al. (2011).

3.5 Full Sample Estimation

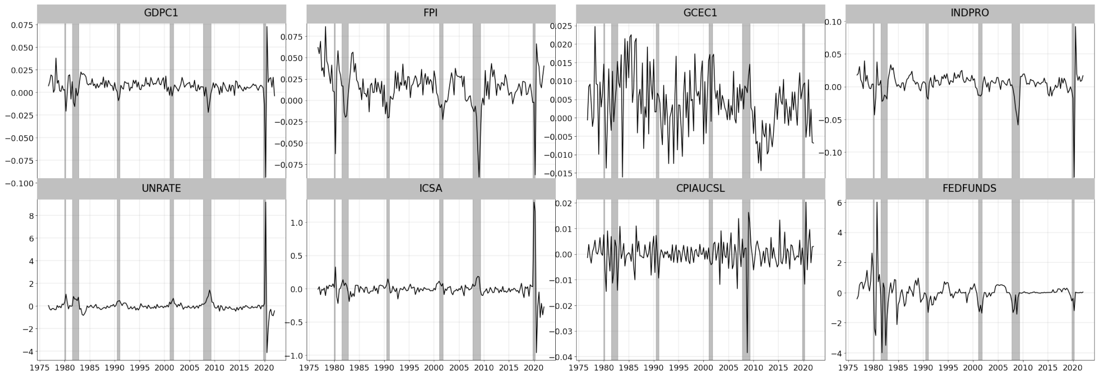


Figure 3.6: Data sets after transformation according to McCracken & Ng (2020). Black-solid line indicates observed data and vertical grey-band represents the US recession periods.

3.5.1 Capture macroeconomic correlations with Radial Basis Function kernel.

One of many benefits of having Gaussian process kernel as covariance in feature space is that we are able to explore the meaning behind the hyper-parameters in kernels. Obviously different kernel hyper-parameters leads to different meaning. In this subsection I present the meaning behind hyper-parameters of Radial basis function (RBF) kernel, where it is able to capture correlation between input and output in Gaussian process regression. RBF is in the form:

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\parallel \mathbf{x} - \mathbf{x}' \parallel^2}{2l^2}\right),$$

Here l is often called (in machine learning field) lengthscale. Recall that in Gaussian process regression (with zero mean $\mathcal{GP}$ prior) we have $\mathbf{f} \sim \mathcal{N}(0, \mathbf{K})$, the correlation between $f(\mathbf{x})$ and $f(\mathbf{x}')$ is exactly $k(\mathbf{x}, \mathbf{x}')$. Therefore with a Gaussian RBF kernel, any two points have a positive correlation, but it quickly approaches zero as the distance between them increases. For example when $\mathbf{x}$ and $\mathbf{x}'$ are l apart, the correlation is $\exp\left(-\frac{l^2}{2l^2}\right) = \exp(-\frac{1}{2}) \approx 0.61$ when l is one. We can exploit an advantage of this by running Gaussian process regression using RBF kernel. To potentially see the dynamic of such correlation especially during the recessions, one can run GPR in a loop by increasing one observation at a time. Here I plot the mentioned correlation results from GPR with US GDP as endogenous variable, and 8 lags of GDP itself and other essential macroeconomic variables, such as inflation, unemployment rate, industrial production, and federal fund rates. The results are visualized in figs. 3.7 to 3.11.

Like mentioning above, that the correlation presented in those figures are approximation, implying that even the computation suggests that there is zero correlation, does not mean it is exactly zero. Since RBF kernel recognizes zero correlation so quickly as the distance between input-output is increased. Therefore the correlation represents for relative purposes over time.

Let's start with fig. 3.7, which aims to visualize how the correlation between US GDP and its lagged values changes with different lag lengths over time. It can be observed that US GDP lagged by one period does not show any significant correlation with itself until the beginning of the Covid-19 pandemic, where it spikes to nearly one towards the end of the pandemic. On the other hand, the second and fourth lagged values appear to be uncorrelated. Interestingly, the sixth lagged value shows a high correlation of approximately 80 percent before the US sub-prime crisis. However, this correlation drops dramatically to almost zero afterwards, and then increases to around 35 percent two years before the start of the pandemic.

Next, we examine the correlations between lagged values of US inflation (lag one to eight) and US GDP, as shown in fig. 3.8. Prior to the Sub-prime crisis, there is a strong correlation between lag one, two, and three of the inflation rate and US GDP. However, these correlations abruptly decrease to almost zero immediately after the Sub-prime crisis, with the exception of the second lagged inflation rate, which remains around 70 percent. Interestingly, during the middle of the Covid pandemic, these correlations experience a drastic decrease and approach zero. Notably, lag five and six of the US inflation rate show no correlation with US GDP, which is quite unprecedented.

Another interesting correlation is between Federal fund rate and US GDP, fig. 3.9. Surprisingly the pattern captured by lengthscale l from RBF kernel indicates that the correlation of Federal fund rate and US GDP almost always drops during the crisis (Covid-19 pandemic included). For example, lag five of federal fund rate (labelled as FEDFUNDS-lag-5 in fig. 3.9), where the correlation spike to roughly 90 percent then dropped to the almost zero right before the Covid-pandemic. The similar patterns can be seen to lag one, two, six, seven and eight, accordingly.

Astoundingly, GCEC1 or government spending expenditures are almost zero-correlated to US GDP, see fig. 3.11.

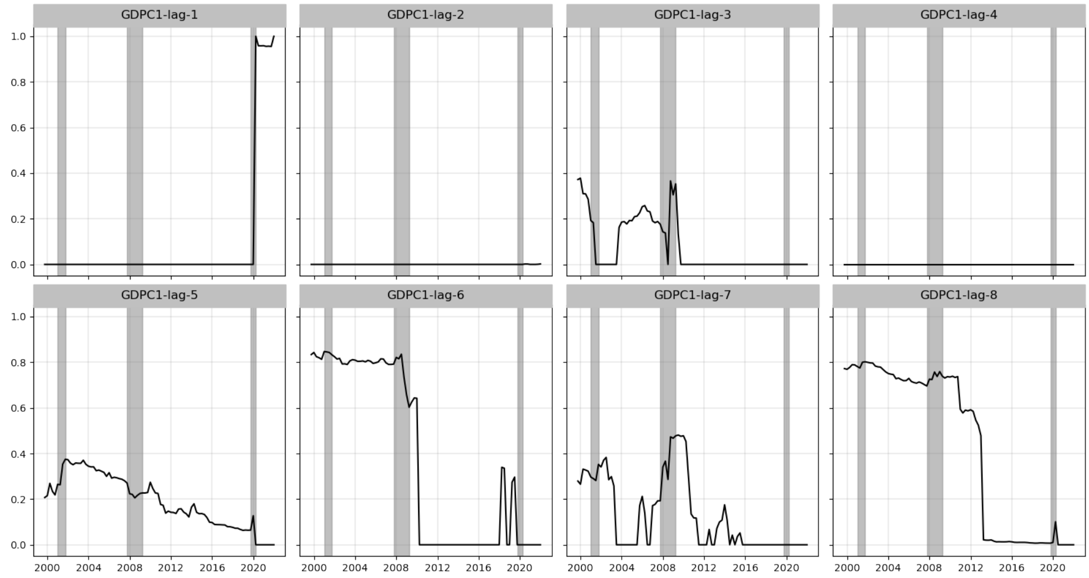


Figure 3.7: Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x})+\epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US GDP. The lengthscale l is transformed into correlation from $\exp\left(-\frac{l^2}{2l^2}\right)$ where it is visualized in black-solid line. The grey vertical bar refers to US recessions.

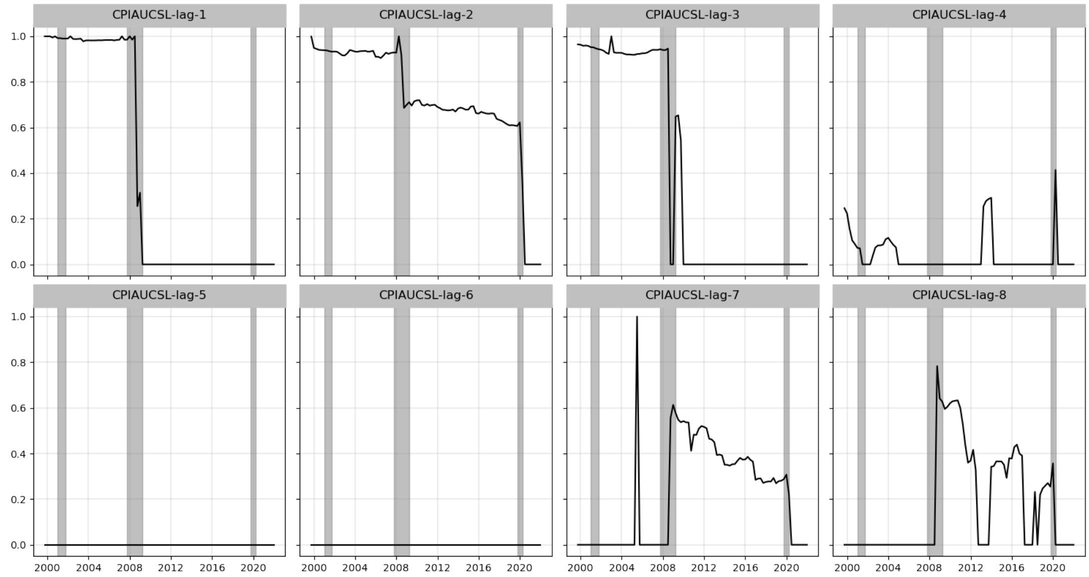


Figure 3.8: Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US inflation rate. The lengthscale l is transformed into correlation from $\exp\left(-\frac{l^2}{2l^2}\right)$ where it is visualized in black-solid line. The grey vertical bar refers to US crisis.

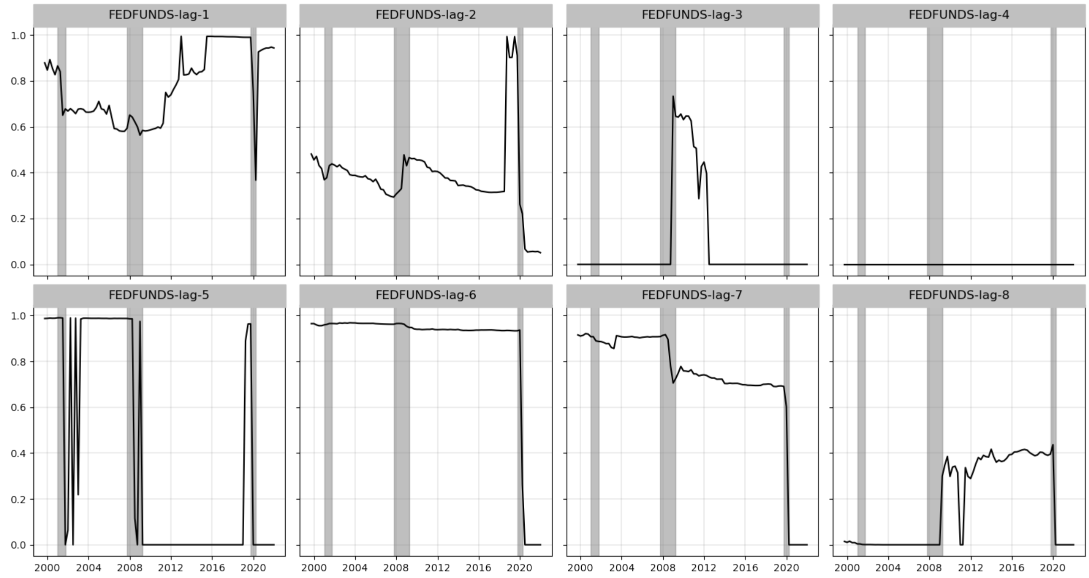


Figure 3.9: Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged Federal fund rate. The lengthscale l is transformed into correlation from $\exp\left(-\frac{l^2}{2l^2}\right)$ where it is visualized in black-solid line. The grey vertical bar refers to US recessions.

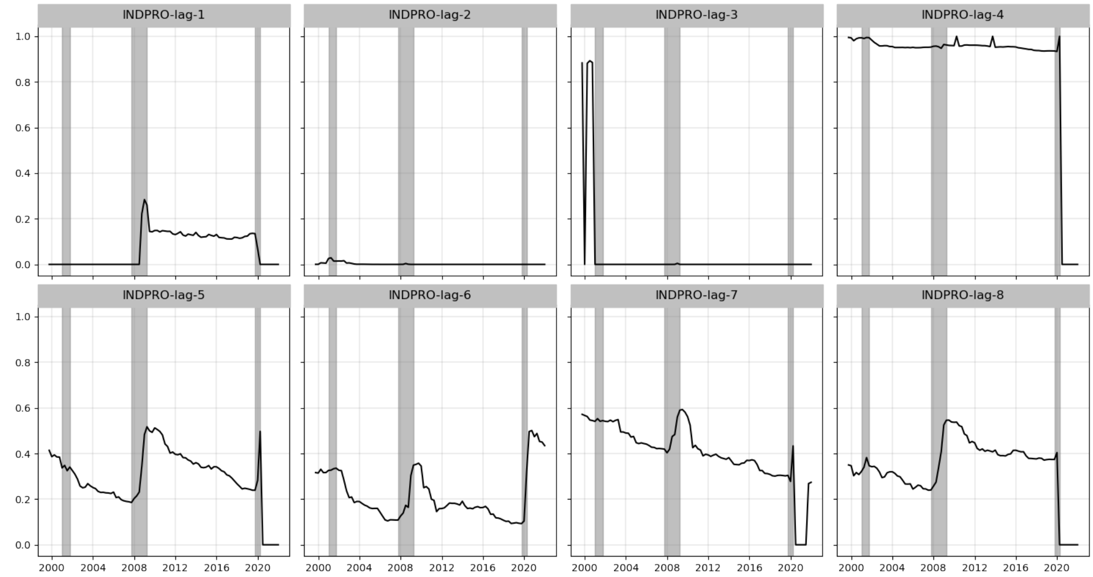


Figure 3.10: Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US industrial production. The lengthscale l is transformed into correlation from $\exp\left(-\frac{l^2}{2l^2}\right)$ where it is visualized in black-solid line. The grey vertical bar refers to US recessions.

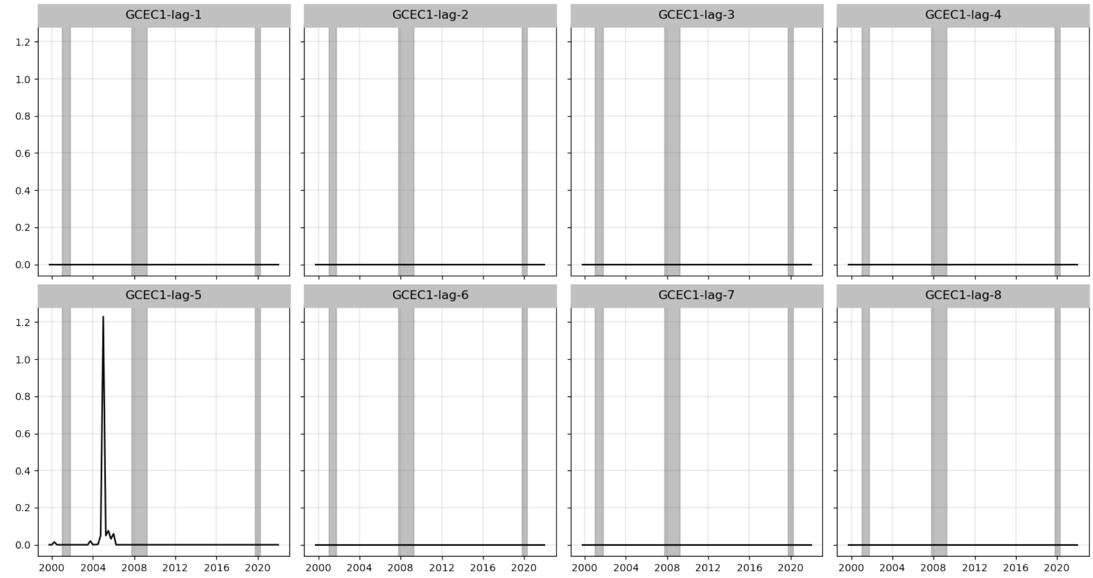


Figure 3.11: Correlation computed from lengthscale l in RBF kernel of Gaussian process regression i.e. $\mathbf{y} = f(\mathbf{x}) + \epsilon$, where $\mathbf{y}$ is US GDP, $\mathbf{x}$ is lagged US government spending expenditures. The lengthscale l is transformed into correlation from $\exp\left(-\frac{l^2}{2l^2}\right)$ where it is visualized in black-solid line. The grey vertical bar refers to US recessions.

3.5.2 Conditional mean

This sub-section, the posterior mean of prediction mean from both parametric, and non-parametric VARs conditional on the homoskedastic/heteroscedastic residual variance and lower triangular A, are investigated and illustrated in figs. 3.12 to 3.15,.,. Will be shortly called *conditional mean* hereafter. In the following discussion the red-solid lines indicate the conditional mean of each model, namely GP-VAR, GP-DNN-VAR, HGP-VAR, and BVAR-SV, respectively. The black-solid lines represent the actual observed US GDP (fig. 3.12), unemployment rate (fig. 3.13), inflation (fig. 3.14) and industrial production (fig. 3.15). The vertical grey fills are US recessions according to National Bureau of Economic Research (NBER).

We can start by examining the US GDP depicted in fig. 3.12. The conditional means of GP-VAR, GP-DNN-VAR, and HGP-VAR models exhibit remarkable similarity. However, the BVAR-SV model stands out as noticeably distinct. The dominance of stochastic volatility in parametric models' conditional mean is a widely recognized phenomenon in econometrics. This issue has been extensively discussed in the literature, as exemplified by the work of Korobilis (2021) and other related studies.

Looking at the conditional mean of US unemployment rate in fig. 3.13, we observe that HGP-VAR stands out among the non-parametric VAR models, particularly during periods of high turbulence such as the Covid-19 pandemic. The conditional mean represented by the red solid line in the plot exhibits distinct characteristics, capturing both the spikes and downturns in the US unemployment rate. In contrast, BVAR-SV appears to overlook the left tail of the distribution. Following the spike in the first quarter of 2020, the BVAR-SV conditional mean fails to decline in accordance with the actual observed UNRATE variable.

Next is US inflation fig. 3.14, it is obvious that the posterior conditional mean of GP-DNN-VAR performs poorest in terms of accuracy relative to the rest of models.

GP-VAR and HGP-VAR are almost identical. For BVAR-SV, again, it ignores the decline of inflation during the US sub-prime crisis (2008). Such similar pattern also happens in US industrial production, see fig. 3.15.

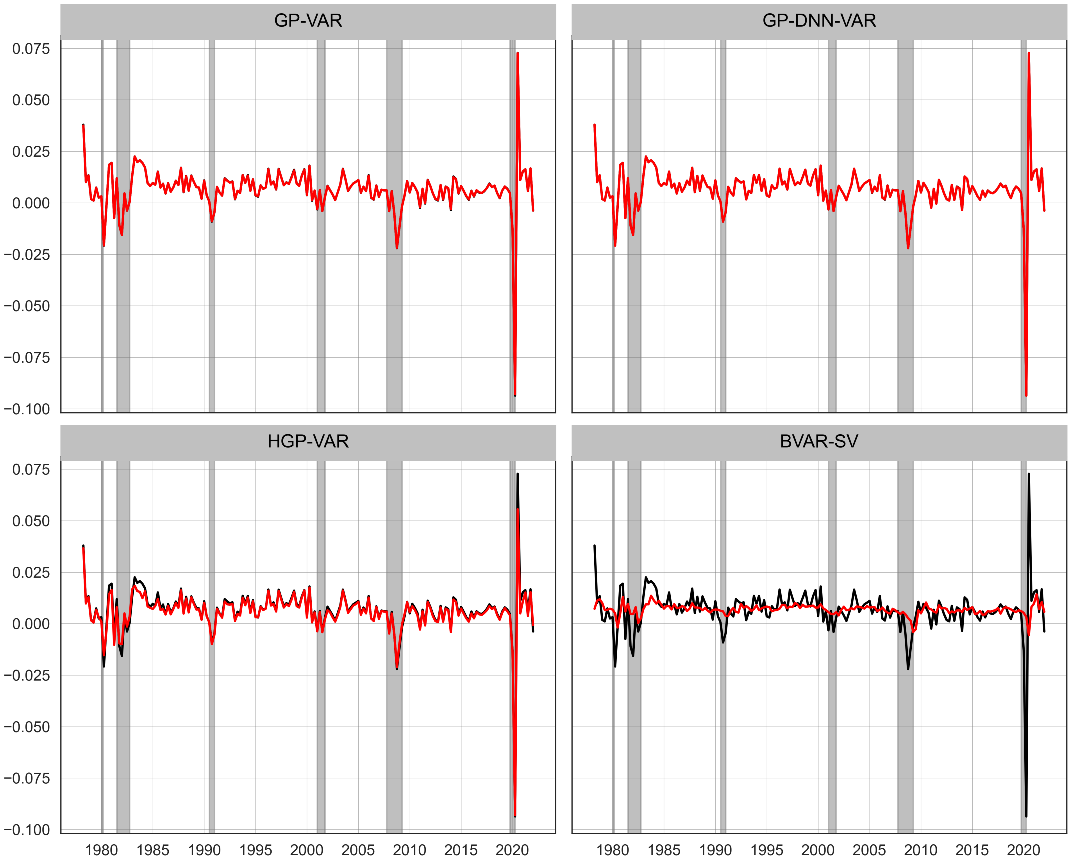


Figure 3.12: **US GDP** conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. The 'conditional' refers to the conditional posterior mean of non-parametric functions $F(\mathbf{X})$ conditional to the other parameters, namely, lower triangular A, and residual variances in each VAR equation.

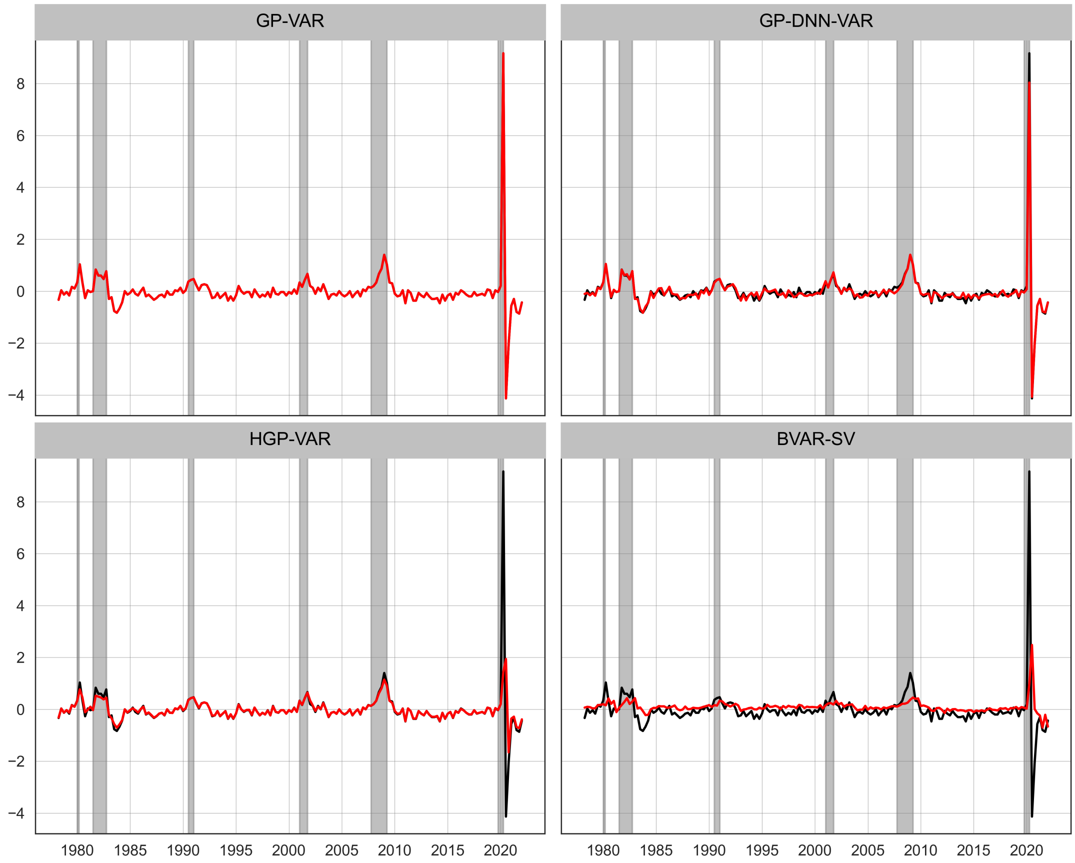


Figure 3.13: **US unemployment rate** conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. The 'conditional' refers to the conditional posterior mean of non-parametric functions $F(\mathbf{X})$ conditional to the other parameters, namely, lower triangular A, and residual variances in each VAR equation.

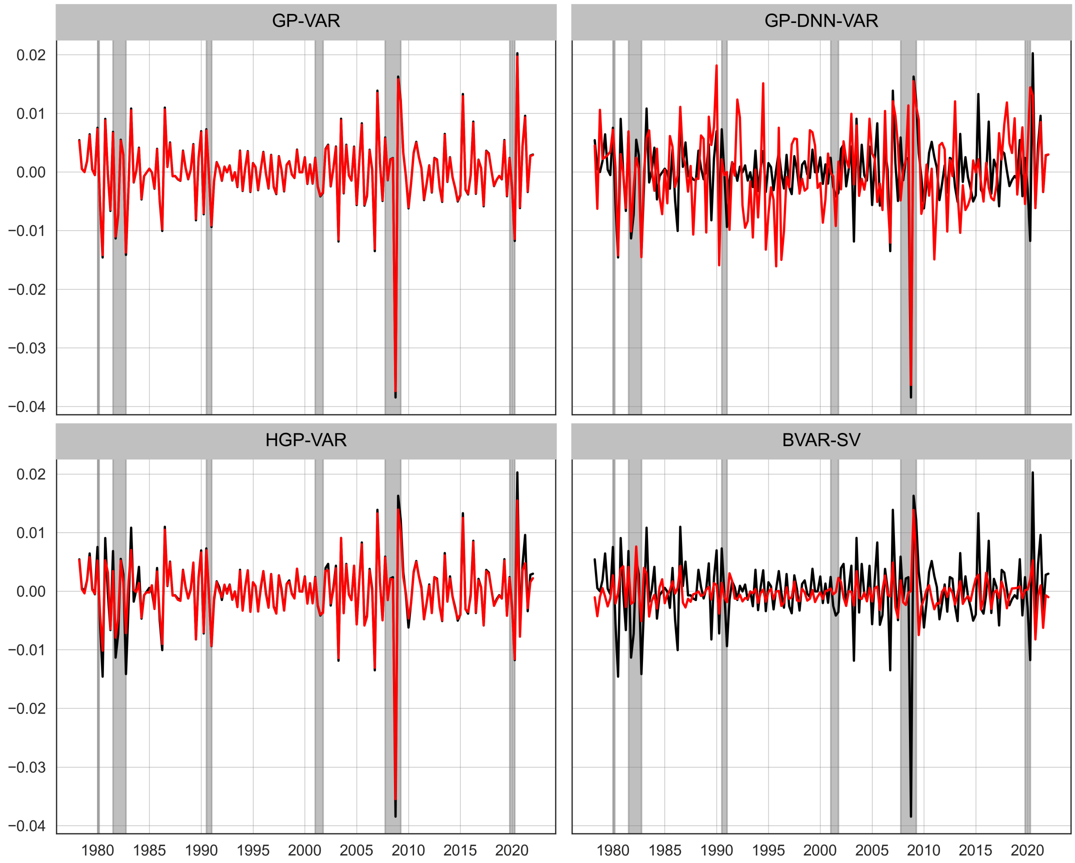


Figure 3.14: **US inflation** conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. The 'conditional' refers to the conditional posterior mean of non-parametric functions $F(\mathbf{X})$ conditional to the other parameters, namely, lower triangular A, and residual variances in each VAR equation.

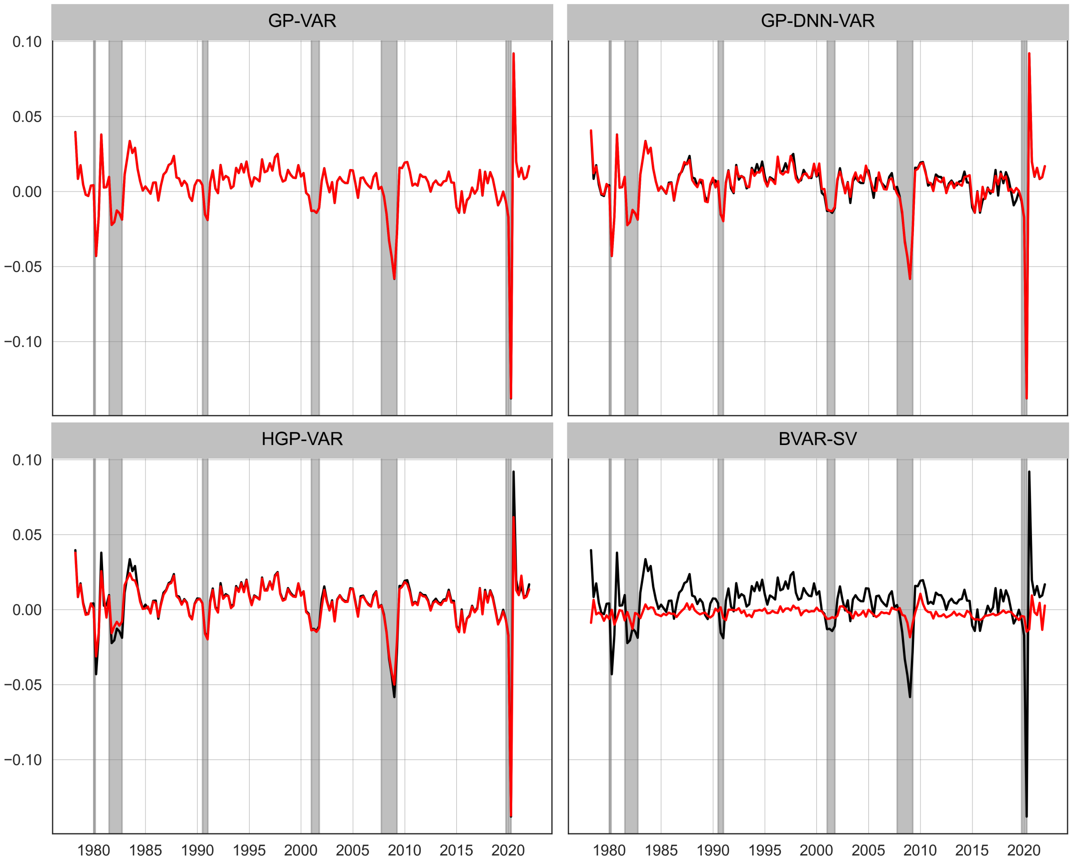


Figure 3.15: **US industrial production** conditional posterior mean of $F(\mathbf{X})$ from GP-VARs models. The 'conditional' refers to the conditional posterior mean of non-parametric functions $F(\mathbf{X})$ conditional to the other parameters, namely, lower triangular A, and residual variances in each VAR equation.

3.5.3 Volatilities

From all models included in the chapter, only two models are able to relax the homoscedastic variance assumption, namely BVAR-SV, and HGP-VAR. In this subsection, we compare heteroscedastic variances of those two different models. We begin by, first visualize the volatility from both models by not recover full time-varying covariance from VAR i.e. $\Sigma_t = e^{h_t/2}$ for BVAR-SV, see eq. (3.7), and $\Sigma_t = e^{\mathbf{F}_2(\mathbf{X_t})}$ for HGP-VAR eq. (3.13), which are presented in fig. 3.16 (simply denote *volatilities* hereafter). Then fig. 3.17 shows the VAR-volatility after recover the full TVP-Covariance, that is $diag(\mathbf{H_t})^{1/2}$ from eqs. (3.7) and (3.13) (denote VAR-volatility hereafter). In the following discussion, the vertical grey bar indicates the US crisis periods. Black and red-solid-lines refer to volatility produced by HGP-VAR, and BVAR-SV, respectively. To see which macroeconomic variable have the highest volatility over time, those figures share similar y-axis value (y-ticks).

In fig. 3.16, the volatilities of most variables in HGP-VAR are generally higher compared to those in BVAR-SV, except for the US unemployment rate and Federal funds rate. Notably, the spike in volatility of the unemployment rate variable in BVAR-SV is approximately three times higher than that in HGP-VAR. The significant difference in volatilities between the two models can be attributed to their distinct formulations. In BVAR-SV, the volatilities are conditioned on the VAR coefficients, resulting in higher volatilities when the conditional mean from the VAR model fails to adequately capture the dynamics of the dependent variables (indicated by larger squared residuals). On the other hand, HGP-VAR represents volatility as a non-parametric function over time, rather than assuming a random walk process as in BVAR-SV. Consequently, the two models exhibit different volatility patterns, as depicted in fig. 3.16. Despite these disparities, both models are effective in capturing the spikes in volatilities during periods of US economic crises.

Moving to the VAR-volatilities shown in fig. 3.17, it is evident that the US unemployment rate and Federal funds rate variables exhibit the most notable VAR-volatilities. Following the transition from volatilities to VAR-volatilities, both BVAR-SV and HGP-VAR exhibit higher residual variances, particularly during the periods of Covid-19 surges. Prior to the recovery, the volatilities of the US unemployment rate in BVAR-SV and HGP-VAR were approximately 3.0 and 1.0, respectively (refer to fig. 3.16 for reference). However, these figures significantly rose to around 20.0 during the recovery period before experiencing a sudden drop. On the other hand, the Federal funds rate volatility in BVAR-SV did not exhibit any spike during this period, whereas HGP-VAR saw a spike of approximately 10.0. For the remaining variables, the VAR-volatilities are almost indistinguishable from the volatilities.

Due to the expression of volatility, VAR-volatility generated from HGP-VAR model, which is input/predictors-dependent, the volatility relies also on VAR predictors (we use exact similar predictors for both non-parametric functions i.e. $F_1(\mathbf{X}), F_2(\mathbf{X})$, see eq. (3.13)). This means that different covariates in HGP-VAR results in different shapes of volatilities/VAR-volatilities. Different covariates here refer to both differences in numbers of variables and lags in HGP-VAR. To prove the point and illustrate that configure numbers of lag in HGP-VAR is extremely influential to the shape and value of VAR-volatility over time. the visualization of VAR-volatilities of eight US macroeconomic variables are plotted in fig. 3.18. The figure presents VAR-volatilities with three different number of lags, one lag (dash-dotted-line), six lags (dashed-line) and nine lags (solid line). Those sub-figures share x-axis but not y-axis for a clearer understanding and make it simple to compare.

Over time, it is evident that the HGP-VAR model with nine lags exhibits higher VAR-volatilities for all macroeconomic variables. The spikes in VAR-volatility correspond to periods of US crises, and models with additional lags in HGP-VAR show prolonged periods of elevated VAR-volatilities compared to models with fewer lags. For instance, consider the VAR-volatility of US GDP during the US subprime crisis between 2007 and 2009 (sub-figure top left of fig. 3.18). The VAR-volatility of US GDP in the one-lag HGP-VAR model spiked up to 0.15 and rapidly dropped back to the steady state. In contrast, the VAR-volatility of US GDP in the six-lag HGP-VAR model gradually increased to around 0.50, then decreased to the steady state (similar to the one-lag HGP-VAR but taking more time). The year before the six-lag HGP-VAR model reached its steady volatility is approximately 2010. A similar consistent pattern can be observed for the nine-lag HGP-VAR model, where it took even longer for the model to adjust back. This behavior is attributed to the fact that the conditional VAR-volatilities are dependent on explanatory variables and follow a non-parametric functional form. Thus, outliers from lagged responsive variables contribute to the volatility at each point in time.

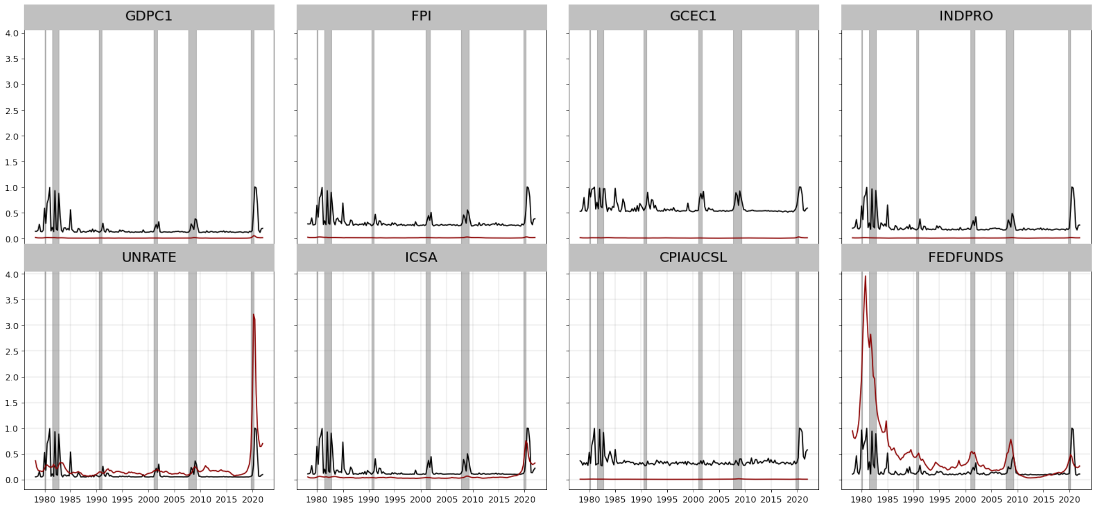


Figure 3.16: Volatility of eight variables BVAR-SV (red-solid) and HGP-VAR (black-solid).

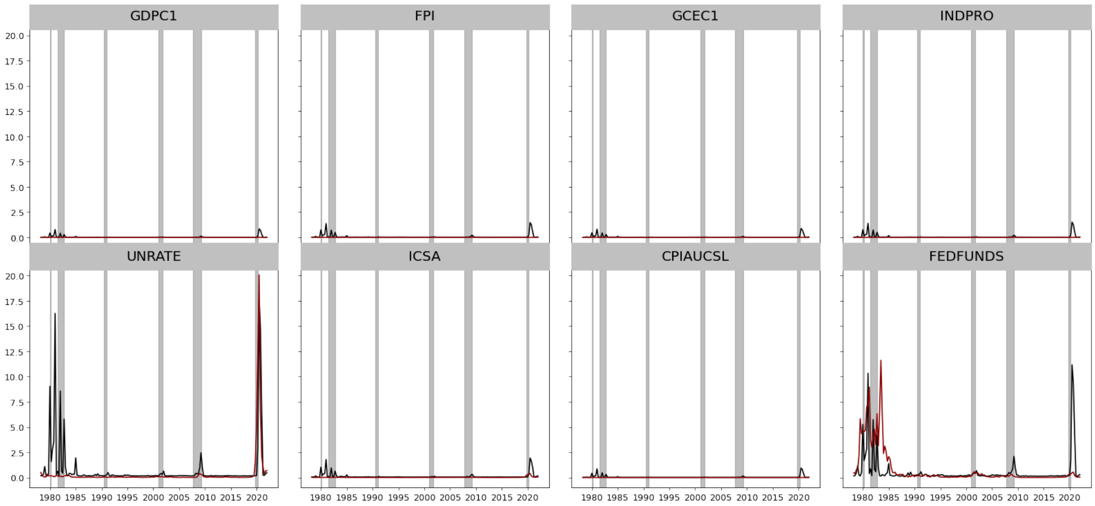


Figure 3.17: VAR-Volatilities of eight variables BVAR-SV (red-solid) and HGP-VAR (black-solid).

While the HGP-VAR models may not accurately capture the true macroeconomic volatilities, they demonstrate superior performance in terms of out-of-sample forecasting compared to BVAR-SV and other models, as will be discussed in more detail in Section section 3.6.

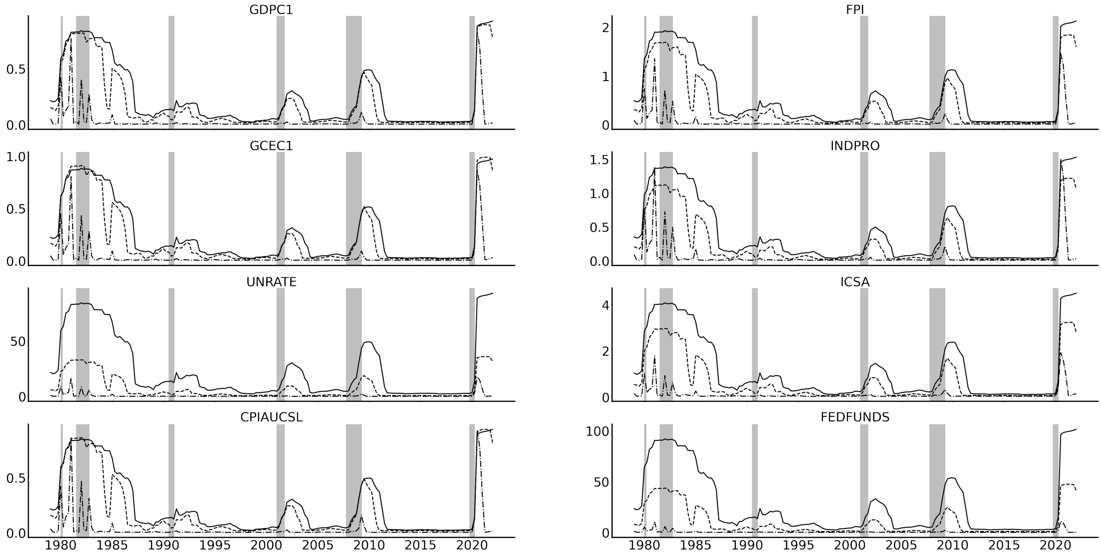


Figure 3.18: Conditional volatilities of HGP-VAR model with different number of lag of explanatory variables. Dash-dotted-line (one lag), dash-line (six lags), and solid-line (nine lags). Grey-vertical-fills (US recession according to NBER).

3.6 Forecasting Results

Evaluations of forecasting performance are evaluated for 2002-Q2 through 2019-Q1 (labelled: non-pandemic) and 2020-Q1 through 2022-Q1 (labelled: pandemic), separately. Root mean square error (RMSE), Cumulative ranked probabilistic scores (CRPS), and Quantile scores (QS) are the measuring matrices. Those results are presented in a relative number to the benchmark model (BVAR-SV). Since all measuring scores are negatively-orientated, any score less than one indicates a more accurate forecast than BVAR-SV. We evaluated the 12-steps-ahead pseudo out-of-sample iterated forecasts for 2002Q2 through 2019-Q1, and 4-steps-ahead forecasts during the Covid-19 pandemic periods which begins at the first quarter of 2020 (2020-Q1) and ends at first quarter of 2022-Q1. We show the forecast results of crucial US macroeconomic variables, which are gross domestic product (GDPC1), industrial production (INDPRO), unemployment rate (UNRATE) and inflation rate (CPIAUCSL). Notably, GP-DNN-VAR forecast scores will sometimes have $(\cdot)$ next to the score itself, which is the standard deviation of 20 RMSE/CRPS/QS with exact model configurations.[5] The reasoning for this is that GP-DNN-VAR model are approximated by stochastic gradient descent, which is a gradient-based-optimizers. As a result those learned parameters can be slightly different each time the model is trained, see sections 3.4.2 and 3.4.3 for more details on how parameters in each model are learned.

3.6.1 During 2002Q2 through 2019Q1

To begin with forecast results during 2002Q2 - 2019Q1, table 3.7 shows the RMSE scores of $h = 1, 2, ..., 12$ of all non-parametric models relative to the benchmark. The models include GP-VAR, HGP-VAR, and GP-DNN-VAR. The bold number denotes the lowest relative scores at each forecast horizon.

According to the results presented in table 3.7, it is evident that the GP-VAR

[5] This is due to the number of deep neural network i.e. number of notes and weights and bias are large, resulting in slightly different in each training. If none is shown, it indicates that those deviations are very low that it will be omitted for clarity.

model outperforms the other models in terms of point forecasting for three out of the four variables of interest. For example, if we look at the fourth column of table 3.7, we can see that the relative RMSE of the GP-VAR model is the lowest across all forecast horizons. A similar pattern is observed in the point forecasts of industrial production and unemployment rate. It is important to note that while some of the RMSE values for GP-VAR are only slightly lower than the other models, there are cases where the difference is quite substantial. For instance, when forecasting the first horizon of industrial production, the GP-VAR model outperforms the HGP-VAR model by approximately double. This trend is also observed in the predictions of the unemployment rate, where the HGP-VAR model performs worse than the GP-VAR model.

When it comes to predicting the US inflation rate, it is noteworthy that the HGP-VAR model generally outperforms both the GP-VAR and GP-DNN-VAR models. In terms of point forecasting, both the GP-VAR and HGP-VAR models exhibit better performance compared to the BVAR-SV model, particularly for the first three-steps-ahead forecasts. However, it is important to highlight that the majority of the RMSE values from the GP-DNN-VAR model are worse than those of the benchmark BVAR-SV model.

The advantage of incorporating heteroscedastic variance in Gaussian process VAR models becomes particularly pronounced when evaluating the accuracy of forecast densities, as shown in columns 6 to 8 of table 3.7. In this regard, the HGP-VAR model stands out as it performs exceptionally well compared to the other models in forecasting GDPC1 and UNRATE. Specifically, for GDPC1, HGP-VAR achieves the lowest relative Continuous Ranked Probability Score (CRPS) compared to the other models for forecast horizons $h = 2, ..., 12$, with the exception of $h = 1$. GP-VAR, on the other hand, slightly outperforms HGP-VAR in this regard.

Moving to forecasting US inflation rate, it is evident that BVAR-SV model performs quite well in comparison to GP-VAR and GP-DNN-VAR. Despite that HGP-VAR performs either equally to the benchmark or always better for all forecasting horizons. Perhaps this is an another solid empirical result suggesting that heteroscedastic variance is an essential assumption in order to forecast inflation well.[6]

Overall the final verdicts regarding to forecasting four focused US macroeconomic variables during the non-pandemic periods. Firstly GP-VAR is skilled in terms of point forecasts especially the very first forecasting horizon. When consider forecast densities, however, the rest of models are dominated by HGP-VAR, forecasting US inflation rate in particular. Secondly GP-DNN-VAR (centered-GP-VAR with Deep neural networks) performs worse to a very simple GP-VAR model. Unfortunately we are unable to know which predictors that actually cause the point forecast to be over or under-estimated since it is a deep neural network. This is often considered to be the weakness of all deep generative models where we are not able to see what happen inside each hidden-state in the model. That is why it is called *Deep-*.

[6] From literature point of view, the unbeatable model to perform best is unobserved component stochastic volatility (UC-SV) which also happens to have heteroscedastic variance and time-varying trend.

		RMSE			CRPS		
		HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR
GDPC1	h = 1	0.55	**0.38**	0.78	0.37	**0.35**	0.78
	h = 2	0.80	**0.77**	1.21	**0.65**	0.70	1.16
	h = 3	0.98	**0.82**	1.46	**0.67**	0.69	1.40
	h = 4	1.01	**0.86**	1.45	**0.67**	**0.67**	1.27
	h = 5	1.16	**1.05**	1.72	**0.71**	0.73	1.26
	h = 6	1.10	**0.99**	1.56	**0.68**	0.69	1.13
	h = 7	1.15	**1.09**	1.64	**0.66**	0.67	1.08
	h = 8	1.30	**0.97**	1.28	**0.61**	0.66	0.92
	h = 9	1.24	**1.12**	1.38	**0.62**	0.65	0.87
	h = 10	1.18	**1.06**	1.40	**0.59**	0.65	0.90
	h = 11	1.16	**1.06**	1.22	**0.58**	0.61	0.78
	h = 12	1.18	**1.05**	1.24	**0.57**	0.59	0.76
INDPRO	h = 1	0.96	**0.43**	0.63	0.36	**0.31**	0.54
	h = 2	1.03	1.03	1.32	**1.00**	1.09	1.53
	h = 3	1.05	**0.73**	0.95	**0.69**	0.75	1.08
	h = 4	1.11	**1.03**	1.41	**0.97**	1.04	1.54
	h = 5	1.06	**1.05**	1.47	**0.95**	1.07	1.51
	h = 6	1.11	**1.02**	1.31	**0.91**	1.02	1.33
	h = 7	1.08	1.08	1.30	**0.91**	1.04	1.28
	h = 8	1.11	**1.08**	1.26	**0.88**	1.05	1.18
	h = 9	1.08	1.08	1.13	**0.86**	1.01	1.00
	h = 10	**1.06**	1.08	1.08	**0.85**	1.01	0.96
	h = 11	1.06	**1.05**	1.08	**0.82**	0.93	0.95
	h = 12	1.06	**1.05**	1.06	**0.81**	0.94	0.91
UNRATE	h = 1	0.51	**0.25**	0.34 (.015)	0.24	**0.19**	0.26 (.012)
	h = 2	1.05	**0.50**	0.68 (.008)	0.44	**0.41**	0.65 (.009)
	h = 3	0.75	**0.57**	0.70 (.008)	**0.46**	0.47	0.61 (.010)
	h = 4	1.08	**0.91**	1.24 (.016)	**0.54**	0.61	0.86 (.018)
	h = 5	1.08	**0.96**	1.22 (.019)	**0.54**	0.61	0.76 (.017)
	h = 6	1.05	**0.97**	1.38 (.017)	**0.51**	0.58	0.79 (.014)
	h = 7	1.07	**0.97**	1.35 (.012)	**0.50**	0.56	0.74 (.014)
	h = 8	1.06	**1.00**	1.21 (.015)	**0.47**	0.54	0.61 (.007)
	h = 9	1.06	**1.01**	1.02 (.018)	**0.45**	0.51	0.50 (.010)
	h = 10	1.06	**1.05**	1.15 (.027)	**0.44**	0.52	0.53 (.015)
	h = 11	1.05	1.04	1.00 (.013)	**0.42**	0.49	0.45 (.007)
	h = 12	1.05	1.03	1.02 (.027)	**0.40**	0.48	0.45 (.017)
CPIAUCSL	h = 1	**0.44**	0.89	1.23	0.87	**0.85**	1.36
	h = 2	**0.75**	0.99	1.41	**1.00**	1.12	1.61
	h = 3	1.02	**0.98**	1.36	**1.00**	1.13	1.57
	h = 4	**0.65**	0.99	1.10	**0.99**	1.12	1.28
	h = 5	**0.72**	1.03	1.31	**1.00**	1.19	1.49
	h = 6	**0.98**	1.03	1.33	**0.99**	1.19	1.38
	h = 7	**0.90**	1.03	1.13	**0.97**	1.17	1.22
	h = 8	**1.00**	1.04	1.22	**0.96**	1.16	1.32
	h = 9	**1.00**	1.04	1.23	**0.95**	1.21	1.27
	h = 10	**1.01**	1.04	1.14	**0.94**	1.21	1.20
	h = 11	**1.00**	1.03	1.11	**0.93**	1.17	1.20
	h = 12	**1.00**	1.02	1.06	**0.93**	1.14	1.10

Table 3.7: RMSE of GP-VAR, HGP-VAR, GP-DNN-VAR models relative to BVAR-SV for `h=1,...,h=12`, one to twelve-steps-ahead forecasts during non-pandemic periods.

Note: *The number in bracket refers to the standard deviation of 20 RMSE from exact similar model configurations. For example* $RMSE_{mean} = \sum_{i=1}^{20} \frac{RMSE_1,...,RMSE_{20}}{20}$, $RMSE_{std} = \sqrt{\sum_{i=1}^{20} \frac{\left(RMSE_i - RMSE^2_{mean}\right)}{20}}$ *where* $RMSE_{mean}, RMSE_{std}$ *is RMSE mean and standard deviation, respectively.*

		TAILS $\nu(\pi) = (2\pi - 1)^2$			UNIFORM $\nu(\pi) = 1$			CENTRE $\nu(\pi) = \pi(1 - \pi)$			RIGHT $\nu(\pi) = \pi^2$			LEFT $\nu(\pi) = (1 - \pi)^2$		
		HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR
GDPC1	h = 1	**0.42**	**0.40**	**0.86**	**0.37**	**0.35**	**0.77**	**0.35**	**0.33**	0.74	0.36	**0.35**	0.76	0.38	0.36	0.80
	h = 2	**0.69**	**0.78**	**1.24**	**0.65**	**0.70**	**1.16**	**0.65**	0.68	1.16	0.83	**0.80**	1.62	**0.56**	0.66	0.93
	h = 3	**0.65**	**0.70**	**1.31**	**0.68**	**0.69**	**1.42**	**0.71**	0.71	1.53	0.56	**0.50**	1.20	**0.87**	0.99	1.78
	h = 4	**0.66**	**0.69**	**1.24**	**0.68**	**0.67**	**1.28**	**0.71**	**0.69**	1.39	0.81	**0.74**	1.74	**0.61**	0.66	1.06
	h = 5	**0.66**	**0.70**	**1.11**	**0.72**	**0.74**	**1.29**	**0.81**	0.81	1.54	0.76	**0.70**	1.44	**0.74**	0.82	1.28
	h = 6	**0.63**	**0.67**	**0.97**	**0.69**	**0.71**	**1.17**	**0.78**	0.79	1.42	0.63	**0.58**	1.09	**0.82**	0.91	1.39
	h = 7	**0.62**	**0.62**	**0.97**	**0.68**	**0.68**	**1.12**	**0.79**	0.79	1.37	0.69	**0.62**	1.20	**0.74**	0.81	1.19
	h = 8	**0.59**	**0.63**	**0.84**	**0.62**	**0.67**	**0.95**	**0.72**	0.77	1.17	0.56	**0.54**	0.90	**0.80**	0.94	1.18
	h = 9	**0.59**	0.63	0.82	**0.64**	0.67	0.90	**0.77**	0.80	1.11	0.65	**0.61**	0.97	**0.73**	0.83	0.99
	h = 10	**0.58**	0.64	0.86	**0.61**	0.67	0.94	**0.74**	0.79	1.19	0.58	**0.56**	0.92	**0.77**	0.92	1.19
	h = 11	**0.58**	0.62	0.79	**0.59**	0.61	0.80	**0.75**	0.76	1.06	0.61	**0.53**	0.87	**0.75**	0.89	1.02
	h = 12	**0.57**	0.62	0.77	**0.57**	0.60	0.78	**0.74**	0.76	1.03	0.59	**0.52**	0.85	**0.75**	0.90	0.99
INDPRO	h = 1	0.35	**0.30**	0.48	0.36	**0.31**	0.54	0.36	**0.31**	0.56	0.45	**0.38**	0.69	0.31	**0.26**	0.43
	h = 2	**1.01**	1.17	1.53	**1.00**	1.09	1.54	**1.02**	1.06	1.58	1.05	**1.00**	1.72	**0.99**	1.18	1.43
	h = 3	**0.76**	0.88	1.14	**0.69**	0.75	1.08	**0.67**	0.70	1.08	0.61	**0.57**	1.06	**0.80**	0.97	1.15
	h = 4	**1.00**	1.14	1.52	**0.98**	1.04	1.55	**1.00**	1.02	1.65	1.00	**0.94**	1.74	**1.00**	1.17	1.49
	h = 5	**0.99**	1.20	1.51	**0.95**	1.07	1.53	**0.99**	1.06	1.66	**0.99**	1.03	1.60	**0.99**	1.18	1.60
	h = 6	**0.95**	1.14	1.25	**0.93**	1.03	1.36	**0.98**	1.04	1.54	**0.93**	0.97	1.49	**1.00**	1.18	1.36
	h = 7	**0.96**	1.17	1.23	**0.92**	1.05	1.31	**0.98**	1.07	1.48	**0.97**	1.04	1.45	**0.97**	1.17	1.32
	h = 8	**0.94**	1.18	1.21	**0.90**	1.06	1.21	**0.97**	1.11	1.37	**0.93**	1.04	1.26	**0.98**	1.22	1.34
	h = 9	**0.93**	1.16	1.03	**0.88**	1.03	1.03	**0.96**	1.10	1.15	**0.94**	1.06	1.14	**0.96**	1.17	1.08
	h = 10	**0.93**	1.17	1.02	**0.87**	1.03	0.99	**0.96**	1.09	1.14	**0.92**	1.06	1.12	**0.97**	1.18	1.07
	h = 11	**0.90**	1.11	1.03	**0.84**	0.95	0.99	**0.96**	1.03	1.17	**0.90**	0.94	1.05	**0.97**	1.17	1.16
	h = 12	**0.90**	1.12	1.04	**0.84**	0.96	0.95	**0.96**	1.06	1.12	**0.89**	0.90	1.01	**0.98**	1.25	1.15
UNRATE	h = 1	0.25	**0.19**	0.25	0.24	**0.19**	0.26	0.23	**0.18**	0.25	0.22	**0.17**	0.23	0.26	**0.20**	0.28
	h = 2	0.51	0.51	0.76	0.45	**0.41**	0.65	0.44	**0.39**	0.65	0.41	**0.38**	0.55	0.55	**0.54**	0.92
	h = 3	**0.49**	0.53	0.62	0.47	0.47	0.62	0.51	**0.49**	0.69	0.71	**0.68**	0.84	**0.36**	0.38	0.54
	h = 4	**0.56**	0.68	0.89	**0.56**	0.62	0.89	**0.66**	0.71	1.06	**0.62**	0.65	0.84	**0.60**	0.75	1.18
	h = 5	**0.56**	0.70	0.78	**0.56**	0.64	0.80	**0.70**	0.75	1.02	**0.69**	0.71	0.88	**0.58**	0.75	0.96
	h = 6	**0.55**	0.69	0.82	**0.54**	0.62	0.83	**0.70**	0.76	1.12	0.73	0.73	0.97	**0.54**	0.73	1.00
	h = 7	**0.55**	0.68	0.80	**0.53**	0.59	0.79	**0.69**	0.73	1.06	0.68	**0.66**	0.93	**0.57**	0.77	0.96
	h = 8	**0.53**	0.66	0.65	**0.50**	0.58	0.65	**0.69**	0.75	0.93	0.74	0.74	0.94	**0.49**	0.68	0.67
	h = 9	**0.53**	0.64	0.54	**0.49**	0.55	0.55	**0.69**	0.74	0.81	0.70	**0.69**	0.75	**0.53**	0.70	0.62
	h = 10	**0.52**	0.68	0.58	**0.48**	0.56	0.58	**0.69**	0.77	0.87	**0.69**	0.71	0.80	**0.52**	0.75	0.67
	h = 11	**0.52**	0.66	0.52	**0.46**	0.54	0.50	**0.69**	0.76	0.77	**0.69**	0.71	0.72	**0.52**	0.72	0.59
	h = 12	**0.51**	0.66	0.55	**0.45**	0.53	0.50	**0.68**	0.77	0.78	**0.69**	0.73	0.75	**0.51**	0.71	0.59
CPIAUCSL	h = 1	0.84	0.84	1.35	0.86	**0.85**	1.36	0.86	**0.85**	1.34	**0.80**	0.81	1.31	0.91	**0.87**	1.38
	h = 2	**1.01**	1.28	1.71	**1.00**	1.12	1.61	**1.00**	1.04	1.58	**1.00**	1.18	1.61	**1.00**	1.08	1.64
	h = 3	**1.03**	1.32	1.62	**1.00**	1.13	1.57	**0.99**	1.05	1.57	**1.02**	1.19	1.59	**1.00**	1.11	1.59
	h = 4	**1.01**	1.29	1.30	**0.99**	1.12	1.28	**0.99**	1.06	1.30	**1.01**	1.18	1.33	**0.99**	1.11	1.27
	h = 5	**1.03**	1.39	1.55	**1.00**	1.19	1.50	**1.00**	1.13	1.54	**1.02**	1.23	1.52	**1.00**	1.22	1.56
	h = 6	**1.03**	1.40	1.41	**0.99**	1.19	1.40	**1.01**	1.13	1.45	**1.03**	1.25	1.39	**1.00**	1.21	1.48
	h = 7	**1.00**	1.36	1.30	**0.98**	1.17	1.23	**1.00**	1.13	1.31	**1.02**	1.23	1.33	**0.99**	1.21	1.28
	h = 8	**1.00**	1.39	1.38	**0.97**	1.17	1.33	**1.00**	1.13	1.41	**1.01**	1.26	1.36	**0.99**	1.19	1.44
	h = 9	**1.00**	1.45	1.35	**0.95**	1.21	1.29	**1.00**	1.19	1.39	**1.01**	1.31	1.36	**0.99**	1.27	1.39
	h = 10	**1.00**	1.47	1.32	**0.95**	1.22	1.22	**1.00**	1.19	1.30	**1.01**	1.31	1.27	**0.99**	1.27	1.34
	h = 11	**1.00**	1.45	1.34	**0.94**	1.18	1.22	**1.00**	1.16	1.30	**1.02**	1.31	1.24	**0.98**	1.23	1.39
	h = 12	**1.00**	1.40	1.24	**0.93**	1.15	1.12	**1.00**	1.14	1.19	**1.02**	1.26	1.16	**0.99**	1.22	1.26

Table 3.8: Quantile Scores during non-pandemic periods (selected variables).

3.6.2 During pandemic 2020Q1 through 2022Q1

Recently, multiple literatures have suggested that Vector Autoregressive Regression in a non-parametric form is able to handle outliers exceptionally well in contrast to parametric model, Huber et al. (2020), Clark et al. (2023, 2022) . In this sub-section we provide some empirical results regarding of how accurate Gaussian process VAR forecasts essential US macroeconomic variables relative to a BVAR-SV model. Additionally to investigate if GP-DNN-VAR or relaxing the homoscedastic to Heteroscedastic-GP-VAR (HGP-VAR) actually improves the out-of-sample forecasts both in point and density forecasts when macroeconomic variables are experiencing significant fluctuations or erratic behavior?

The comparison between non-parametric models and the benchmark is in a similar fashion to the previous sub-section, the point and forecast densities are evaluated with RMSE, CRPS and Quantile scores, respectively. We first report the point forecasts during the high turbulence of macroeconomic variables in table 3.9. The numbers in each column report the forecast results of macroeconomic variables. The bold number refers to the lowest relative RMSE score to BVAR-SV model. To aid the interpretation, see one-step-ahead prediction of US unemployment rate for example, the relative RMSE of HGP-VAR and GP-VAR to the benchmark is $0.89, 0.92$, implying that the forecasts produced by both model is 11 and 8 percent better than BVAR-SV, respectively. From here if we look at the $h = 2, 3$ and 4, it is obvious that both homoscedastic and heteroscedastic GP-VAR models are preferably skilled at longer forecasting horizon exercises. Such consistent pattern can be seen for other variables as well. For example at $h = 4$, relative RMSE of HGP-VAR and GP-VAR are 0.72 in contrast to $h = 1$ which are $0.96, 0.95$. Such improvement is up to about 20 percent.

In contrast to GP-VAR, the forecasting results of GP-DNN-VAR are less satisfactory for all variables and time horizons. It should also be highlighted that the performance of GP-DNN-VAR is characterized by high variability and inconsistency.

To summarize, we find that Homoscedastic GP-VAR outperforms Heteroscedastic-

GP-VAR in terms of point forecasts.

In terms of forecast densities, evaluated with CRPS. I found all four US variables with at almost every forecasting horizons, HGP-VAR dominates the benchmark and the rest of the models. Starting with gross domestic product, HGP-VAR outperforms BVAR-SV at $h = 1, 2, 4$ (23, 22, 26 percent respectively) but small margin at $h = 3$ (only 1 percent). With homoscedastic-GP-VAR, there were massive increase at $h = 3$ (comparatively worse than a benchmark upto 25 percent). Despite that GP-VAR do quite well at $h = 2$ and $h = 4$. GP-DNN-VAR do worse relative to all models, at $h = 3$ in particular. Also there is a massive spike in relative-CRPS (4.26 according to table 3.9). For industrial production, there is a significant difference between BVAR-SV and HGP-VAR (30 percent) at $h = 1$, and small margin (11 percent) at $h = 4$. Surprisingly GP-DNN-VAR forecasts industrial production well at $h = 3$. Among non-parametric models, it is the only model that actually beats the benchmark in terms of forecast density. Lastly, in the case of unemployment rate and inflation rate, HGP-VAR stands out as the top-performing non-parametric model. Although this evaluation focuses on the overall accuracy of the forecast densities during the Covid-19 periods, a more detailed assessment using Quantile scores to specifically examine accuracy of different regions of the predictive distribution will be discussed in the following section. This allows for a more nuanced evaluation of the model's performance across the entire distribution of possible outcomes.

Table 3.10 presents the relative QS of various non-parametric models compared to a benchmark for four-steps-ahead forecasting exercises.

In the case of US GDP forecasts (row 4-6) HGP-VAR consistently outperforms the benchmark BVAR-SV as well as GP-VAR and GP-DNN-VAR models in terms of relative QS in both tails regions, center region (for $h = 1, 2, 4$), right region (for $h = 1, 2$), and left region (for all h). The gains in relative QS can be up to 39% over BVAR-SV, 41% over GP-VAR, and 58% over GP-DNN-VAR in both tails regions. This indicates that incorporating heteroscedasticity in the model helps improve the predictive density, especially during turbulent periods.

In the case of industrial production, HGP-VAR is the best-performing non-parametric model in most cases, except for the $h = 3$ forecast horizon where GP-DNN-VAR performs better. At $h = 1$, HGP-VAR outperforms all models, especially BVAR-SV, in the both-tails (51%), centre (19%), right (28%) , and left regions (36%) of the predictive distribution. It totally beats constant variance GP-VAR for across all horizons and all regions. These results are particularly important during turbulent periods since they show how well the model adjusts the predictive densities toward extreme values.

When considering the unemployment rate variable, there is a notable increase in the Quantile scores (QS) for all types of GP-VAR models at the forecast horizon of $h = 3$, with the benchmark model performing the best. However, it is important to highlight that among all the GP-VAR models, HGP-VAR remains the top performer. For forecast horizons of $h = 1, 2, 4$, HGP-VAR outperforms the other models significantly. This suggests that HGP-VAR has a stronger ability to capture the uncertainty and variability in the unemployment rate forecasts compared to the other non-parametric models.

When examining the forecasting performance for US inflation rate, it appears that GP-VAR demonstrates greater skill in predicting inflation compared to linear models, particularly in the tails region (31%), the right region (35%), and the left region (-2%) at the forecast horizon of $h = 1$. However, when compared to HGP-VAR, GP-VAR still struggles in the left region of the predictive density, where HGP-VAR outperforms it for $h = 2, 3, 4$. In terms of other regions and for all forecast horizons, GP-VAR is not significantly different from time-varying-variance-GP-VAR. At $h = 3, 4$, HGP-VAR shows a notably better performance than GP-VAR and GP-DNN-VAR, although not to the extent of surpassing the benchmark model. Overall, the findings suggest that BVAR-SV remains a reliable model for accurately forecasting US inflation, particularly in the left region of the predictive distributions.

For $h = 1, 2, 3$ GP-VAR forecasts the best for GDP tails forecast (both left and right tails, labelled: QS (tails)). It can be also seen that HGP-VAR forecast densities are as superior as GP-VAR for all $h = 1, 2, 3, 4$ in contrast to BVAR-SV.

For non-weighted QS (QS (uniform)), all three non-parametric models are exactly to QS-tails. For center region of predictive density, both GP-VAR and HGP-VAR outperforms the rest.

Moving to the right region of density (QS right), we found that allowing the variance in Gaussian process VAR is able to deliver better forecast densities. For example, QS-right of HGP-VAR at three-steps-ahead prediction is 0.2 which is doubly lower than GP-VAR at 0.4, as well as, approximately 80 percent relative better than BVAR-SV. If looking at other QS for the rest of variables, it is apparent that either HGP-VAR performs better than the rest of models i.e. BVAR-SV, GP-DNN-VAR, GP-VAR or only slightly beaten. Such identical patterns can be observed for almost all variables.

In addition to mentioned results, I also provide the forecast performance over the number of lags and variables in GP-VARs in section 3.8.2. The forecast performances are evaluated during the pandemic periods.

Next section visualizes how predictive densities of each model look like during the pandemic time.

		RMSE			CRPS		
		HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR
GDPC1	h = 1	0.96	**0.95**	1.06	**0.77**	1.04	1.24
	h = 2	**0.86**	**0.86**	0.92	**0.78**	0.83	0.91
	h = 3	**1.42**	1.53	4.89	**0.99**	1.25	4.26
	h = 4	**0.72**	**0.72**	2.07	**0.74**	0.86	2.12
INDPRO	h = 1	0.94	**0.92**	1.51	**0.70**	0.97	1.81
	h = 2	0.99	0.99	**0.91**	**0.99**	1.05	1.12
	h = 3	1.56	1.56	**1.40**	1.15	1.11	**0.98**
	h = 4	**0.93**	0.94	5.88	0.89	**0.84**	4.79
UNRATE	h = 1	0.97	**0.96**	1.19	**0.85**	1.01	1.40
	h = 2	**0.86**	0.87	1.30	**0.83**	1.02	1.69
	h = 3	**1.30**	1.40	4.76	**1.41**	2.04	6.74
	h = 4	**0.81**	0.88	4.11	**0.81**	1.35	4.35
CPIAUCSL	h = 1	0.92	**0.89**	1.95	**0.87**	0.88	2.23
	h = 2	1.02	**1.01**	1.92	**1.05**	1.09	2.27
	h = 3	**1.05**	**1.05**	3.01	**1.06**	1.13	3.58
	h = 4	**0.88**	0.91	1.74	**0.86**	0.92	1.73

Table 3.9: RMSE of GP-VAR, HGP-VAR, GP-DNN-VAR models relative to BVAR-SV for `h=1,...,h=4` during pandemic periods (2020Q1 through 2022Q1).

3.7 Conclusion

The use of non-parametric VAR models, specifically GP-VARs, has gained significant attention in the field of economics due to their superior out-of-sample forecasting performance, particularly in turbulent macroeconomic data, compared to the traditional BVAR-SV models. In this study, we have examined more advanced GP-VAR mod-

		TAILS $\nu(\pi) = (2\pi - 1)^2$			UNIFORM $\nu(\pi) = 1$			CENTRE $\nu(\pi) = \pi(1 - \pi)$			RIGHT $\nu(\pi) = \pi^2$			LEFT $\nu(\pi) = (1 - \pi)^2$		
		HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR	HGP-VAR	GP-VAR	GP-DNN-VAR
GDPC1	h = 1	**0.61**	1.02	1.19	**0.80**	1.04	1.24	**0.87**	1.05	1.27	**0.73**	1.01	1.30	**0.78**	1.05	1.18
	h = 2	**0.85**	0.93	1.03	**0.77**	0.82	0.93	**0.74**	0.78	0.90	**0.86**	0.89	1.09	**0.68**	0.76	0.75
	h = 3	**0.74**	0.95	4.23	**1.04**	1.17	4.55	**1.15**	1.25	4.67	0.99	**0.98**	5.12	**0.98**	1.22	4.03
	h = 4	**0.71**	0.80	2.49	**0.74**	0.77	2.10	**0.75**	0.76	2.01	1.13	**0.80**	3.03	**0.69**	0.72	1.54
INDPRO	h = 1	**0.51**	0.91	1.77	**0.72**	0.97	1.81	**0.81**	0.99	1.83	**0.72**	0.95	2.35	**0.64**	0.96	1.42
	h = 2	**1.01**	1.13	1.03	**0.98**	1.09	1.15	**0.97**	1.07	1.19	**1.00**	1.04	1.07	**0.98**	1.19	1.21
	h = 3	1.04	1.78	**0.79**	1.22	1.78	**0.93**	1.28	1.78	**0.98**	0.75	1.17	**0.65**	1.81	2.66	**1.26**
	h = 4	**0.91**	1.59	5.39	**0.90**	1.32	5.06	**0.90**	1.24	4.96	**0.91**	1.49	9.94	**0.90**	1.31	2.56
UNRATE	h = 1	**0.74**	0.93	1.36	**0.87**	1.01	1.41	**0.92**	1.03	1.42	**0.91**	1.06	1.16	**0.74**	0.88	1.73
	h = 2	**0.87**	0.89	1.89	**0.80**	0.84	1.64	**0.78**	0.83	1.56	**0.76**	0.92	1.65	0.86	**0.80**	1.72
	h = 3	**1.47**	1.84	7.71	**1.43**	1.68	6.90	**1.41**	1.62	6.65	**1.24**	1.67	6.95	**1.63**	1.74	7.17
	h = 4	**0.83**	1.32	5.67	**0.82**	1.06	4.48	**0.81**	0.99	4.18	**0.77**	1.13	2.64	**0.89**	1.05	7.94
CPIAUCSL	h = 1	0.84	**0.69**	2.14	0.85	**0.82**	2.18	**0.86**	**0.86**	2.19	0.73	**0.65**	1.96	1.04	**1.02**	2.50
	h = 2	1.03	**0.97**	2.50	1.01	**0.97**	2.17	**1.01**	0.97	2.07	1.03	**0.90**	2.31	**1.00**	1.07	2.10
	h = 3	**1.03**	1.15	4.44	**1.04**	1.05	3.49	1.04	**1.02**	3.26	1.09	**0.99**	3.78	**0.99**	1.14	3.51
	h = 4	**0.85**	1.09	2.04	**0.85**	0.96	1.63	**0.85**	0.93	1.52	**0.83**	0.85	1.20	**0.86**	1.10	2.16

Table 3.10: Quantile scores of GP-VAR, HGP-VAR, GP-DNN-VAR models relative to BVAR-SV for `h=1,...,h=4` during pandemic periods (2020Q1 through 2022Q1).

els, namely GP-DNN-VAR and HGP-VAR. The former utilizes a feed-forward neural network to parameterize the mean in GP prior function, while the latter introduces a dynamic covariance structure to the VAR model using another latent-GP function. In addition, HGP-VAR has a unique feature that utilizes inputs/covariates to perform out-of-sample forecasts, where the latest observed data contributes to the forecasts.

Our findings show that during non-pandemic periods, HGP-VARs outperform BVAR-SV models by a significant margin, both in terms of point and density forecasts. Moreover, the HGP-VAR model consistently outperforms other models in specific regions of the predictive distributions. During the Covid-19 pandemic, the forecasting gains are less pronounced than non-pandemic periods, but the HGP-VAR model remains the best-performing model in both point and density forecasts.

For future research, there are several ways to manipulate the HGP-VAR model. One possible approach is to alter the covariate of the second GP-function that parameterizes the HGP-VAR covariance. The behavior of the HGP-VAR covariance over time can be significantly influenced by the covariate/inputs of the second GP-function, which shapes the posterior predictive density band and the duration of its response to large changes in covariates. Therefore, changing the covariate for the second GP-function has the potential to result in further forecast gains.

3.8 Appendix

3.8.1 Data

Fred-Acronyms	**Title**	**Units before transformation**	**Transform-code**
GDPC1	Real Gross Domestic Product	Billions of Chained 2012 Dollars	5
FPI	Fixed Private Investment	Billions of Dollars	5
GCEC1	Real Government Consumption Expenditures and Gross Investment	Billions of Chained 2012 Dollars	5
INDPRO	Industrial Production: Total Index	Index 2017=100	5
UNRATE	Unemployment Rate	Percent	2
ICSA	Initial Claims	Number	5
CPIAUCSL	Consumer Price Index for All Urban Consumers: All Items in U.S. City Average	Index 1982-1984=100	6
FEDFUNDS	Federal Funds Effective Rate	percent	2

Table 3.11: US macroeconomic variables: Source: alfred module, a package from python. The transformation code is according to McCracken & Ng (2020).

3.8.2 Additional Results

Forecast performance over number of lags and variables in GP-VAR.

The goal of this section is to shed light to the question, ”Does increasing the number of lags and variables in Gaussian process VAR improve forecast performance under exceptionally high turbulence of four focused US macroeconomic variables?” Based on the forecast performance in earlier sections, see section 3.6.2, GP-DNN-VAR appears to perform the poorest among the non-parametric models, and in comparison to the benchmark (BVAR-SV). As a result, I only test GP-VAR and HGP-VAR models here.

In the following discussion, the figures have two lines, red and black-solid line where each visualizes RMSE, CRPS, plotted against the number of steadily increasing lags and variables in non-parametric-VARs. It is worth noting that for black-solid line which refers to consistently increasing number of variables in VAR all have 6 lags.

To begin with the GP-VAR model where the RMSE results are plotted in fig. 3.19 for $h = 1, 2$, and fig. 3.20 for $h = 3, 4$. Those four figures empirically suggest that increasing number of variables in GP-VAR does not improve the point forecast of all four US essential macroeconomic variables. It actually is worse in most cases. It is also obvious that expanding the number of lags in GP-VAR almost has no effects in terms of point forecast performance.

The accuracy of forecast density, on the other hand, are plotted in fig. 3.21, and fig. 3.22 for $h = 1, 2, 3, 4$. For $h = 1$ forecasting of US GDP, industrial production and unemployment rate, increasing the number of variables makes the model worse. Despite that we found that the CRPS of US inflation at forecasting horizon $h = 1$ is lower as the number of variables are added in GP-VAR. CRPS keeps decreasing until GP-VAR has 48 variables then CRPS bounce back again. It can also be seen that those improvement seems to be insignificant.

Overall in terms of point forecasts, we found that the best number of variables and lags in GP-VAR is 8 variables with 6 lags for one-step, three-steps, and four-steps-ahead forecast of US GDP, industrial production, unemployment rate and inflation. For $h = 2$, it is slightly different for inflation, where the best number of variables is still 8 but the number of lag is 28. The rest of US variables performances remains unchanged.

For predictive density which is evaluated with CRPS, we found that the best number of variables in forecasting GDPC1, INDPRO and UNRATE is 8 variables with (again) six lags. For US inflation, however, adding more lags in GP-VAR actually improves the accuracy of forecast density. For $h = 1$ best number of lag to forecast US inflation is 18, whereas $h = 2, 3, 4$ is 58.

Moving to HGP-VAR, the RMSE and CRPS with different model configurations between number of lags and variables in VAR are visualized in figs. 3.23 and 3.24 ,figs. 3.25 and 3.26. For one-step-ahead forecast exercise $h = 1$, HGP-VAR performs slightly better in terms of point forecasts as the number of variables in VAR increases, fig. 3.25. Such results can be seen for all four variables. Although the shape of the line in HGP-VAR looks upward and downward and seems to be varied with number of lags and variables in VARs but the y-axis number empirically proves that the difference between point-forecast performances over numbers of lags and variables in HGP-VARs is unrecognised.

When considering forecast density accuracy over lags and variables configurations in HGP-VAR at $h = 1$, it is obvious that CRPS of GDPC1 both lines has positive slope thus making it is worsen (roughly -10%) as those lags and variables are increased. The similar consequences also happen with INDPRO and UNRATE variables. US inflation, however, CRPS is massively improved (up to approximately 38%). The CRPS is at its minimum at 18 variables with 6 lags. For $h = 2, 3, 4$, there is not enough conclusive results suggesting the difference of forecast density performance over the model configurations in HGP-VAR model.

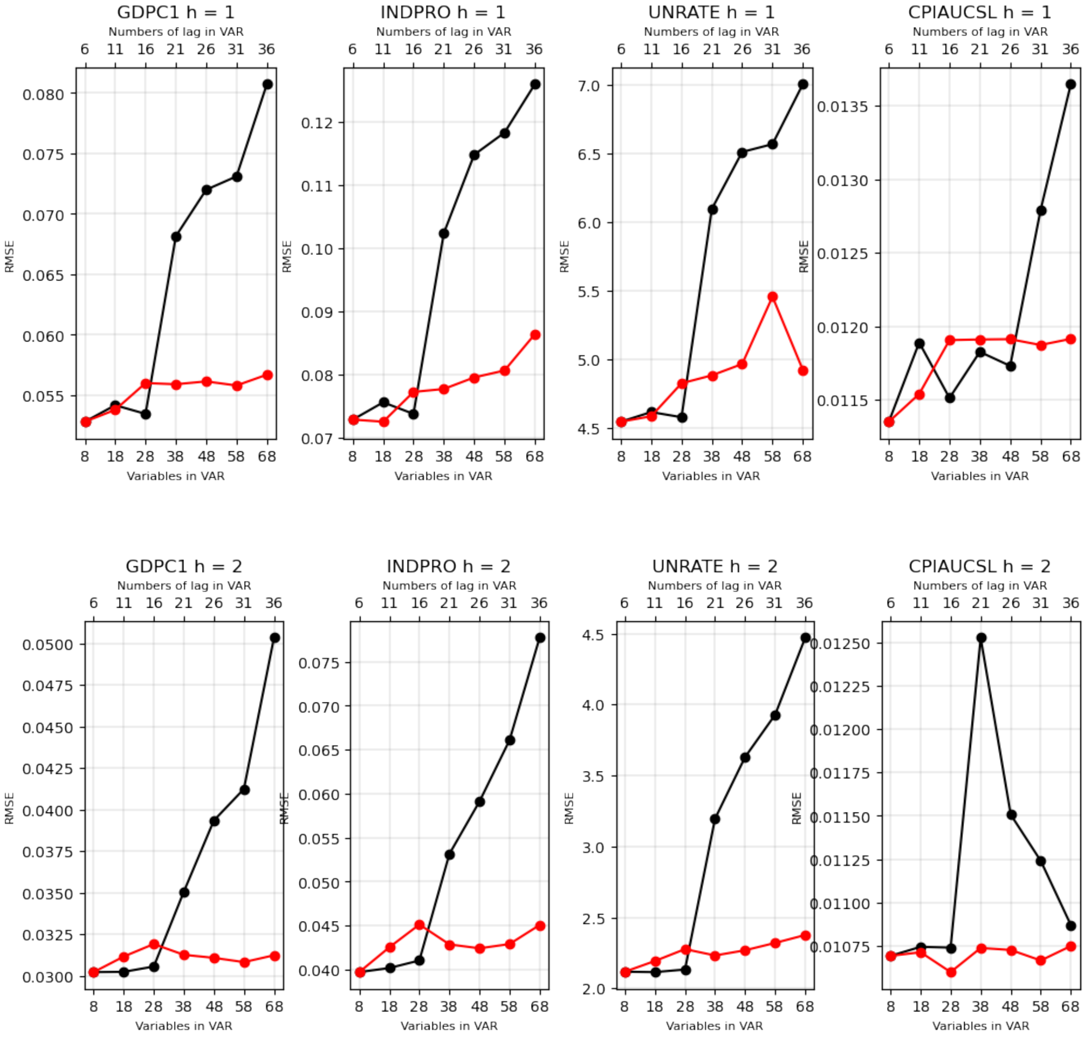


Figure 3.19: RMSE of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 1, h = 2$ forecasting horizons, with consistently increasing number of lags and number of variables in GP-VAR. Red-line denotes RMSE of GP-VAR with increasing 5 GP-VAR lags at a time until 36 lags. Whereas black-line refers to RMSE of GP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily add 10 variables at a time until it reaches 68 variables GP-VAR.

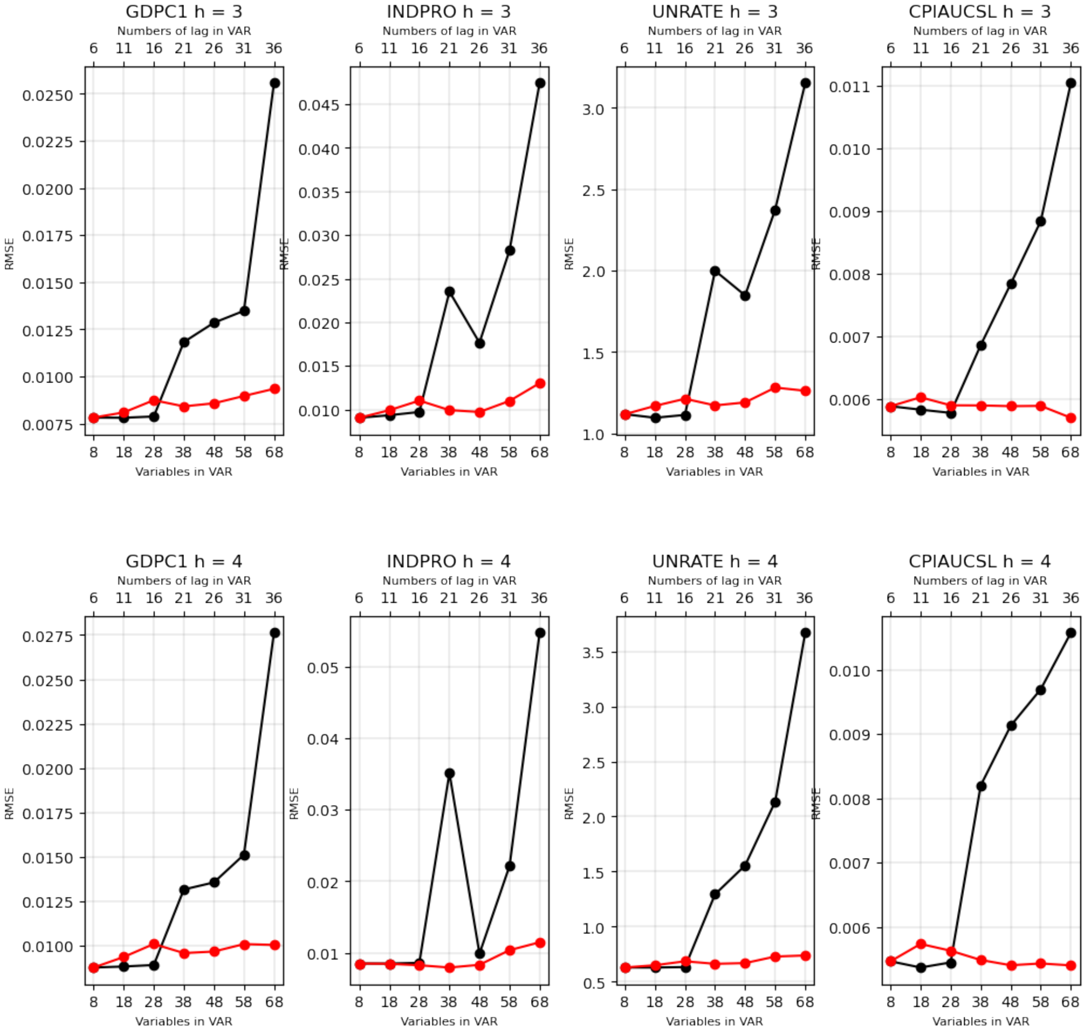


Figure 3.20: RMSE of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 3, h = 4$ forecasting horizons, with consistently increasing number of lags and number of variables in GP-VAR. Red-line denotes RMSE of GP-VAR with increasing number of lags 5 at a time until 36 lags. Whereas black-line refers to RMSE of GP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily increase 10 variables at a time until it reaches 68 variables GP-VAR.

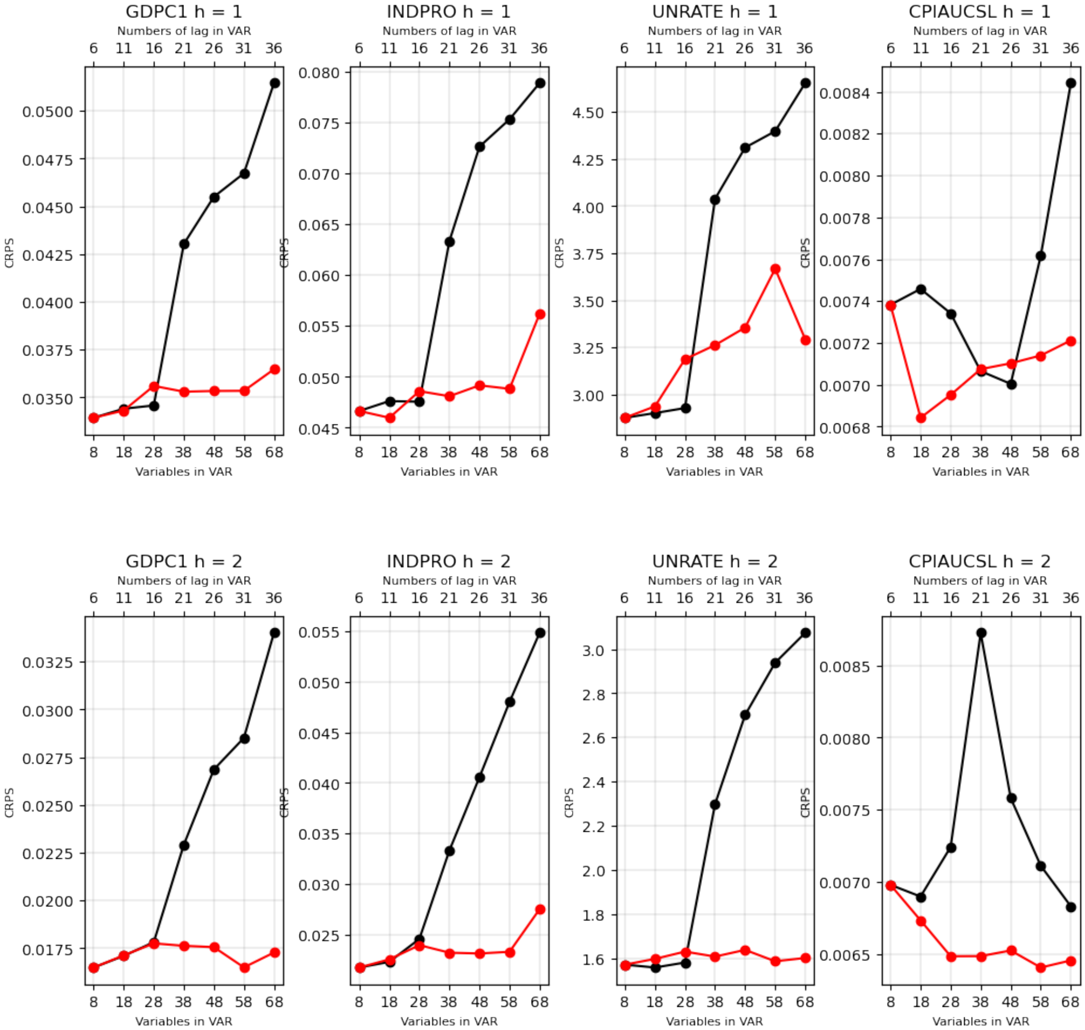


Figure 3.21: CRPS of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 1, h = 2$ forecasting horizons, with consistently increasing number of lags and number of variables in GP-VAR. Red-line denotes CRPS of GP-VAR with increasing number of lags 5 at a time until 36 lags. Whereas black-line refers to CRPS of GP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily increase 10 variables at a time until it reaches 68 variables GP-VAR.

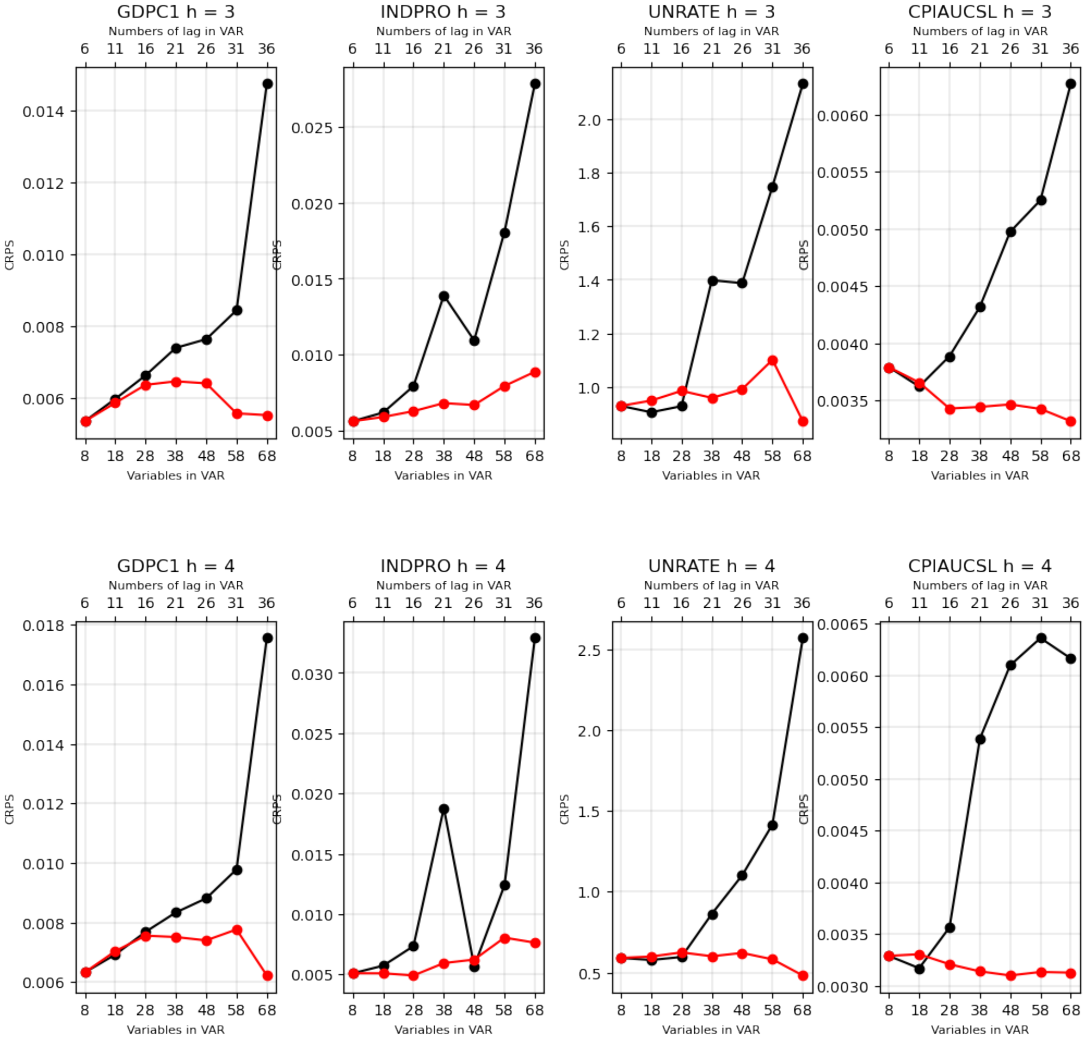


Figure 3.22: CRPS of four focused macroeconomic variables during the pandemic, performed by GP-VAR model at $h = 3, h = 4$ forecasting horizons, with consistently increasing number of lags and number of variables in GP-VAR. Red-line denotes CRPS of GP-VAR with increasing number of lags 5 at a time until 36 lags. Whereas black-line refers to CRPS of GP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily increase 10 variables at each trial until it reaches 68 variables GP-VAR.

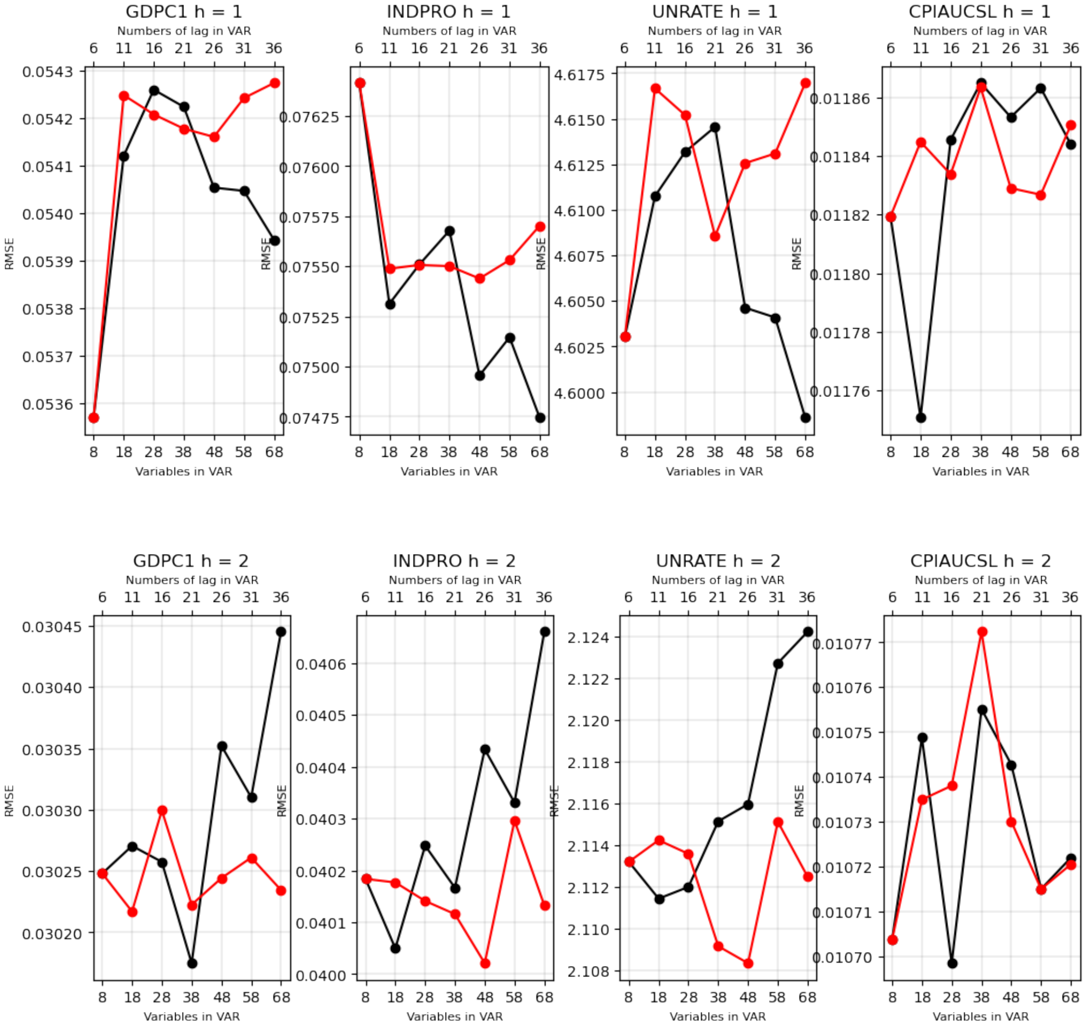


Figure 3.23: RMSE of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 1, h = 2$ forecasting horizons, with consistently increasing number of lags and number of variables in HGP-VAR. Red-line denotes RMSE of HGP-VAR with increasing number of lags 5 at a time until 36 lags. Whereas black-line refers to RMSE of HGP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily increase 10 variables at each trial until it reaches 68 variables HGP-VAR.

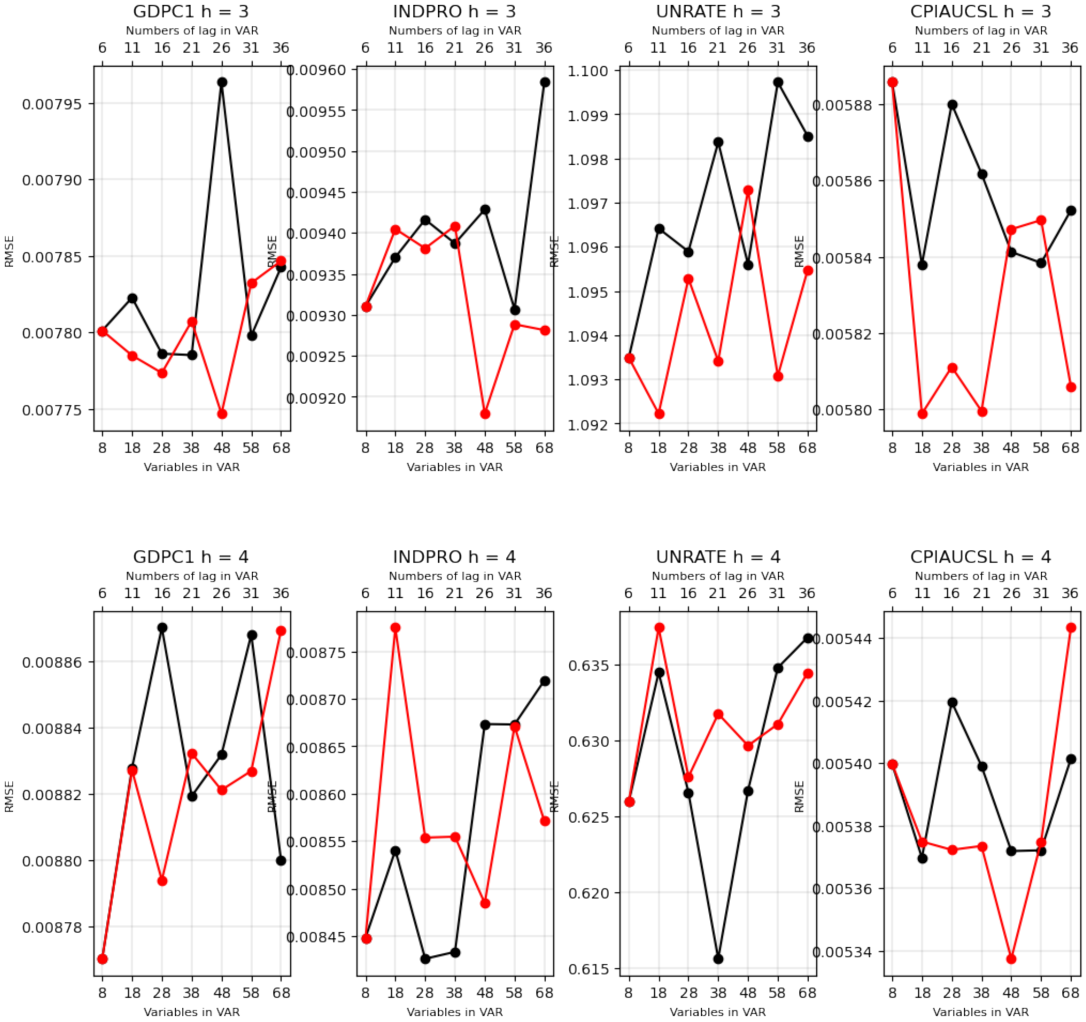


Figure 3.24: RMSE of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 3, h = 4$ forecasting horizons, with consistently increasing number of lags and number of variables in HGP-VAR. Red-line denotes RMSE of HGP-VAR with increasing number of lags 5 at a time until 36 lags. Whereas black-line refers to RMSE of HGP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily increase 10 variables at a time until it reaches 68 variables HGP-VAR.

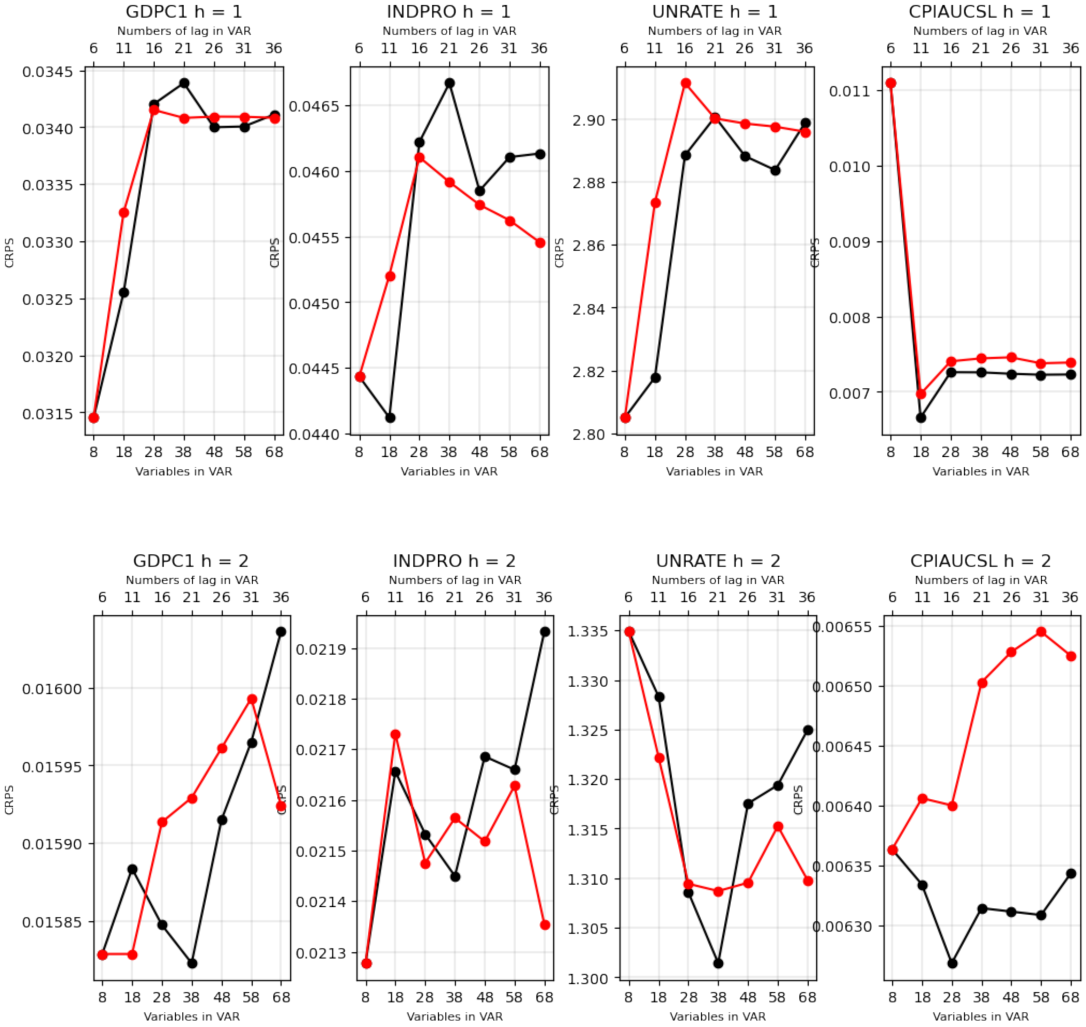


Figure 3.25: CRPS of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 1, h = 2$ forecasting horizons, with consistently increasing number of lags and number of variables in HGP-VAR. Red-line denotes CRPS of HGP-VAR with increasing number of lags 5 at a time until 36 lags. Whereas black-line refers to CRPS of HGP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily increase 10 variables at each trial until it reaches 68 variables HGP-VAR.

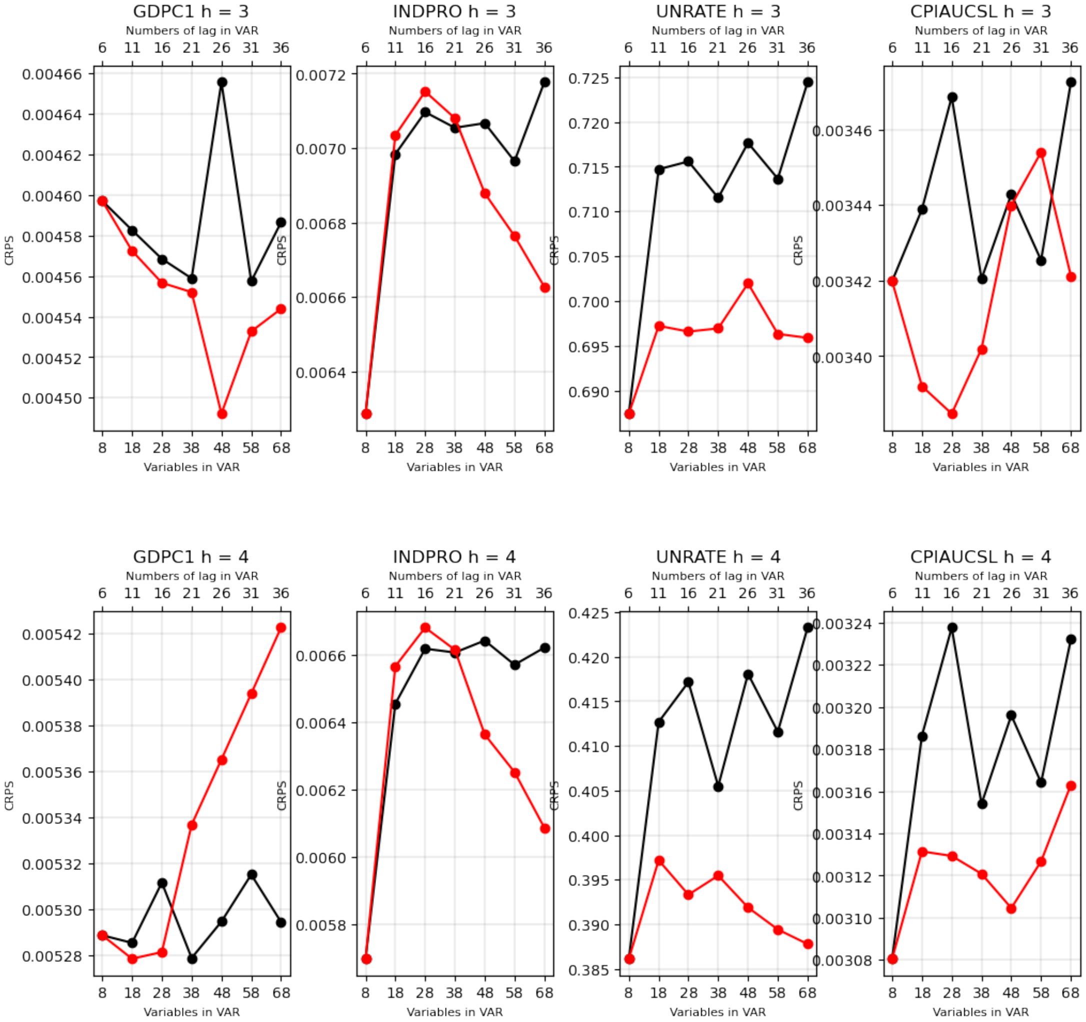


Figure 3.26: CRPS of four focused macroeconomic variables during the pandemic, performed by HGP-VAR model at $h = 3, h = 4$ forecasting horizons, with consistently increasing number of lags and number of variables in HGP-VAR. Red-line denotes CRPS of HGP-VAR with increasing number of lags 5 at a time until 36 lags. Whereas black-line refers to CRPS of HGP-VAR with increasing number of variables (all use 6 lags). Beginning at 8 variables, and steadily increase 10 variables at each trial until it reaches 68 variables HGP-VAR.

Forecast densities

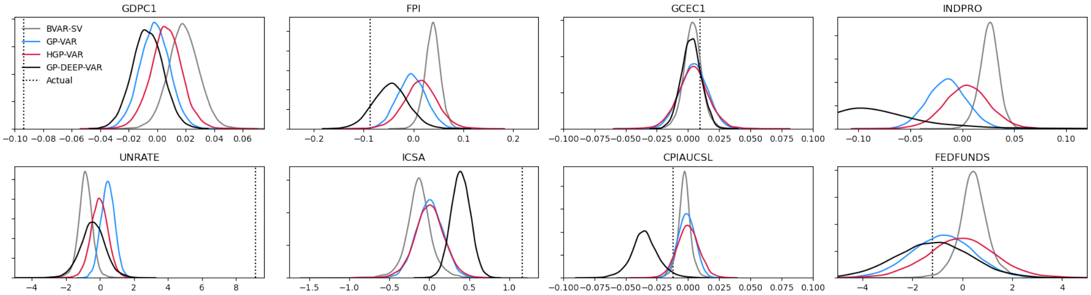


Figure 3.27: Forecast density of one-step-ahead forecast of 2020Q2.

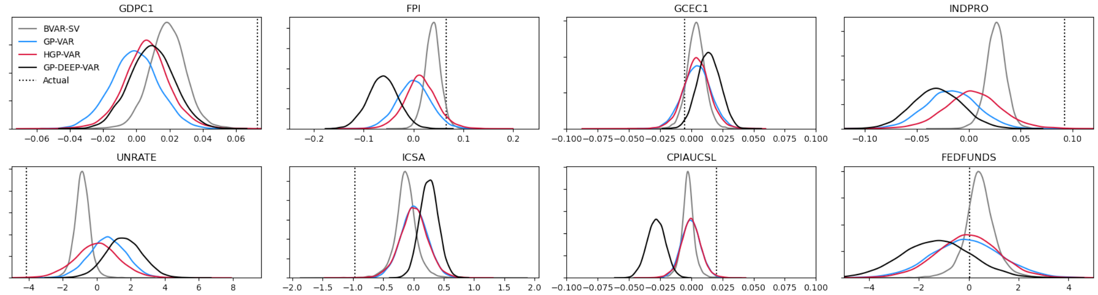


Figure 3.28: Forecast density (one-step-ahead) of 2020Q3.

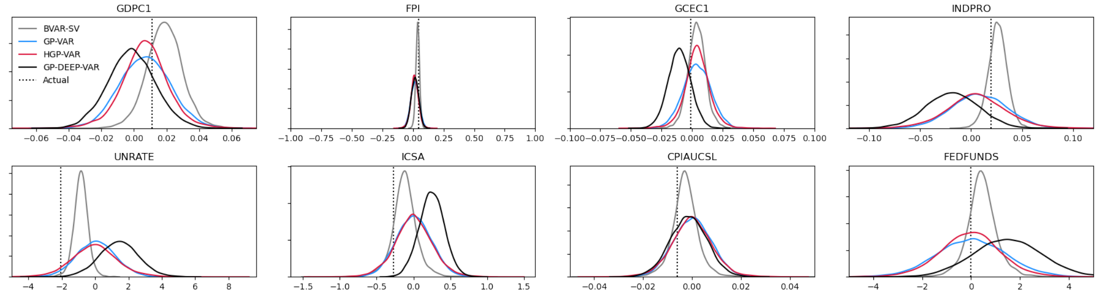


Figure 3.29: Forecast density (one-step-ahead) of 2020Q4.

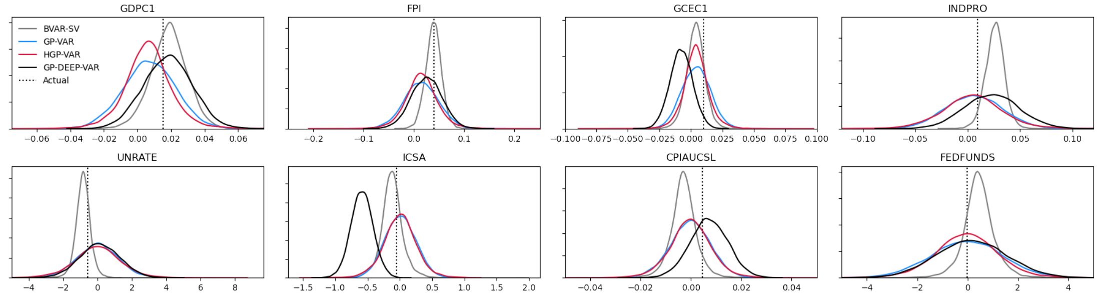


Figure 3.30: Forecast density (one-step-ahead) of 2021Q1.

Chapter 4

Conclusions

This thesis contributes to the literature on forecasting US core macroeconomic variables by deriving the variational Bayes for reduced-form VARs and Mixed Frequency VARs, discussing the advantages of several advancements in non-parametric VAR models. Firstly, the thesis develops a variational Bayes (Vb) approach for system-wide VARs with Horseshoe shrinkage priors. This methodology improves the estimation and inference of reduced-form VAR models by incorporating Bayesian techniques and efficient variable selection while enjoy the very cheap computational costs. Secondly, the thesis extends the Vb approach to Mixed-Frequency VARs (MF-VARs), which are essential tools for economists to perform real-time out-of-sample forecasts. This contribution enhances the accuracy and timeliness of macroeconomic forecasts by effectively combining high-frequency and low-frequency data. Again the estimation is carried through the variational Bayes. Finally, the thesis explores non-parametric VAR models, specifically Gaussian process VARs (GP-VARs). This approach departs from several assumptions commonly made in GP-VAR literature. Firstly, it relaxes the assumption of homoscedastic variance by incorporating an additional non-parametric function, allowing for time-varying error covariance. Secondly, it no longer assumes zero mean function for the GP prior, providing greater flexibility in capturing complex patterns in the data.

In Chapter 1 of the thesis, a contribution is made to the econometric literature by developing a variational Bayes (Vb) approach for system-wide VARs, specifically reduced-form VARs, with a Horseshoe shrinkage prior. Previous literature on variational Bayes typically employed the VB approach on a single equation at a time, known as the Cholesky-transformed VAR. However, it was found that this approach led to inadequate estimation of the VAR covariance and inconsistent results. This chapter addresses this limitation by proposing a novel Vb approach that improves the estimation of the VAR covariance. Additionally, the chapter explores the issue of the prior specification for the lower triangular elements in the square matrix A, which represents the contemporaneous relations between the responsive variables in the VAR. By using a Horseshoe shrinkage prior, the proposed Vb approach addresses the challenge of selecting appropriate prior distributions for the VAR covariance. Overall, the contribution of Chapter 1 lies in developing an improved Vb approach for estimating system-wide VARs, enhancing the accuracy and consistency of the estimated VAR covariance.

Chapter 2 of the thesis presents the derivation of the variational Bayes method for the Mixed-Frequency VAR (MF-VAR) model. This method, called ”variational Bayes-expectation maximization,” utilizes a two-step iterative process involving Vb-E-step and Vb-M-step to maximize the evidence lower bound. The Vb-E-step updates the state variables, while the Vb-M-step updates the variational parameters. This process is performed in each iteration of the Coordinate descent optimization algorithm. The thesis provides a proof that the evidence lower bound is tight, ensuring that the variational parameters and state variables generated from the model converge accurately. The model itself is a Gaussian linear state-space model.

In Chapter 3 of the thesis, the focus is on exploring the advantages of non-parametric Gaussian process VARs (GP-VARs) compared to traditional approaches in economic literature. The GP-VARs proposed in the thesis relax multiple assumptions commonly made in economic models. Firstly, the mean function of the Gaussian process prior for non-parametric functions is manipulated by parameterizing it with a deep neural network. This allows for more flexibility and captures complex patterns in the data, while almost of economic literature assume it to be zeros. Secondly, the thesis introduces a second non-parametric function to model the heteroscedastic covariance in GP-VARs. This means that the VAR covariance becomes time-varying, which is important from an economic modeling perspective. In contrast to the common assumption of residual variances being a random walk, the heteroscedastic GP-VARs allow for a non-parametric and input/predictor-dependent volatility. This introduces nonlinearity and the ability to manipulate VAR predictors to potentially improve out-of-sample forecasts. The empirical application of the GP-VARs to US macroeconomic variables in the thesis demonstrates that the GP-VARs with time-varying volatility outperform other included models. This highlights the advantage of incorporating non-parametric modeling and time-varying volatility in forecasting macroeconomic variables.

Bibliography

Ankargren, S. & Jonéus, P. (2020), 'Simulation smoothing for nowcasting with large mixed-frequency vars', *Econometrics and Statistics* .

Ankargren, S. & Yang, Y. (2019), 'Mixed-frequency bayesian var models in r: the mfbvar package'.

Arias, J. E., Rubio-Ramirez, J. F. & Shin, M. (2022), 'Macroeconomic forecasting and variable ordering in multivariate stochastic volatility models', *Journal of Econometrics* .

Attias, H. (1999), 'A variational baysian framework for graphical models', *Advances in neural information processing systems* **12**.

Bańbura, M., Giannone, D. & Reichlin, L. (2010), 'Large bayesian vector auto regressions', *Journal of applied Econometrics* **25**(1), 71–92.

Beal, M. J. & Ghahramani, Z. (2001), 'The variational kalman smoother', *Gatsby Computational Neuroscience Unit, University College London, Tech. Rep. GCNU TR* **3**, 2001.

Bernardo, J., Bayarri, M., Berger, J., Dawid, A., Heckerman, D., Smith, A., West, M. et al. (2003), 'The variational bayesian em algorithm for incomplete data: with application to scoring graphical model structures', *Bayesian statistics* **7**(453-464), 210.

Bishop, C. M. (2006), *Pattern recognition and machine learning*, springer.

Blei, D. M., Kucukelbir, A. & McAuliffe, J. D. (2017), 'Variational inference: A review for statisticians', *Journal of the American statistical Association* **112**(518), 859–877.

Bognanni, M. (2022), ‘Comment on “large bayesian vector autoregressions with stochastic volatility and non-conjugate priors”’, *Journal of Econometrics* **227**(2), 498–505.

Brave, S. A., Butters, R. A. & Justiniano, A. (2016), ‘Forecasting economic activity with mixed frequency bayesian vars’.

Brave, S. A., Butters, R. A. & Kelley, D. (2020), ‘A practitioner’s guide and matlab toolbox for mixed frequency state space models’, *Available at SSRN 3532455* .

Carriero, A., Chan, J., Clark, T. E. & Marcellino, M. (2021), ‘Corrigendum to: Large bayesian vector autoregressions with stochastic volatility and non-conjugate priors’.

Carriero, A., Chan, J., Clark, T. E. & 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. & Marcellino, M. (2015), ‘Realtime nowcasting with a bayesian mixed frequency model with stochastic volatility’, *Journal of the Royal Statistical Society. Series A,(Statistics in Society)* **178**(4), 837.

Carriero, A., Clark, T. E. & Marcellino, M. (2019), ‘Large bayesian vector autoregressions with stochastic volatility and non-conjugate priors’, *Journal of Econometrics* **212**(1), 137–154.

Carriero, A., Clark, T. E. & Marcellino, M. G. (2020), ‘Nowcasting tail risks to economic activity with many indicators’.

Carriero, A., Kapetanios, G. & Marcellino, M. (2009), ‘Forecasting exchange rates with a large bayesian var’, *International Journal of Forecasting* **25**(2), 400–417.

Carriero, A., Kapetanios, G. & Marcellino, M. (2012), ‘Forecasting government bond yields with large bayesian vector autoregressions’, *Journal of Banking & Finance* **36**(7), 2026–2047.

Carvalho, C. M., Polson, N. G. & Scott, J. G. (2010), ‘The horseshoe estimator for sparse signals’, *Biometrika* **97**(2), 465–480.

Chan, J. C., Koop, G. & Yu, X. (2021), 'Large order-invariant bayesian vars with stochastic volatility', *arXiv preprint arXiv:2111.07225* .

Chan, J. C. & Yu, X. (2022), 'Fast and accurate variational inference for large bayesian vars with stochastic volatility', *arXiv preprint arXiv:2206.08438* .

Clark, T. E., Huber, F., Koop, G. & Marcellino, M. (2022), 'Forecasting us inflation using bayesian nonparametric models', *arXiv preprint arXiv:2202.13793* .

Clark, T. E., Huber, F., Koop, G., Marcellino, M. & Pfarrhofer, M. (2023), 'Tail forecasting with multivariate bayesian additive regression trees', *International Economic Review* **64**(3), 979–1022.

Cogley, T. & Sargent, T. J. (2005), 'Drifts and volatilities: monetary policies and outcomes in the post wwii us', *Review of Economic dynamics* **8**(2), 262–302.

Cross, J. L., Hou, C. & Poon, A. (2019), 'Macroeconomic forecasting with large bayesian vars: Global-local priors and the illusion of sparsity', *International Journal of Forecasting* .

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977), 'Maximum likelihood from incomplete data via the em algorithm', *Journal of the Royal Statistical Society: Series B (Methodological)* **39**(1), 1–22.

Diebold, F. X. & Mariano, R. S. (1995), 'Comparing predictive accuracy', *Journal of Business and Economic Statistics* **13**(3), 253–263.

Dieppe, A., Legrand, R. & Van Roye, B. (2016), 'The bear toolbox'.

Duchi, J. (2007), 'Derivations for linear algebra and optimization', *Berkeley, California* **3**(1), 2325–5870.

Duchi, J., Hazan, E. & Singer, Y. (2011), 'Adaptive subgradient methods for online learning and stochastic optimization.', *Journal of machine learning research* **12**(7).

Durbin, J. & Koopman, S. J. (2012), *Time series analysis by state space methods*, Oxford university press.

Duvenaud, D. (2014), Automatic model construction with Gaussian processes, PhD thesis, University of Cambridge.

Duvenaud, D., Lloyd, J., Grosse, R., Tenenbaum, J. & Zoubin, G. (2013), Structure discovery in nonparametric regression through compositional kernel search, *in* 'International Conference on Machine Learning', PMLR, pp. 1166–1174.

Fortuin, V., Strathmann, H. & Rätsch, G. (2019), 'Meta-learning mean functions for gaussian processes', *arXiv preprint arXiv:1901.08098* .

Fox, C. W. & Roberts, S. J. (2012), 'A tutorial on variational bayesian inference', *Artificial intelligence review* **38**, 85–95.

Frazier, D. T., Loaiza-Maya, R. & Martin, G. M. (2021), 'A note on the accuracy of variational bayes in state space models: Inference and prediction', *arXiv preprint arXiv:2106.12262* .

Gefang, D., Koop, G. & Poon, A. (2020), 'Computationally efficient inference in large bayesian mixed frequency vars', *Economics Letters* p. 109120.

Gefang, D., Koop, G. & Poon, A. (2022), 'Forecasting using variational bayesian inference in large vector autoregressions with hierarchical shrinkage', *International Journal of Forecasting* .

Gefang, D., Koop, G. & Poon, A. (2023), 'Forecasting using variational bayesian inference in large vector autoregressions with hierarchical shrinkage', *International Journal of Forecasting* **39**(1), 346–363.

Ghahramani, Z. & Beal, M. (2000), 'Propagation algorithms for variational bayesian learning', *Advances in neural information processing systems* **13**.

Ghysels, E. & Ozkan, N. (2015), 'Real-time forecasting of the us federal government budget: A simple mixed frequency data regression approach', *International Journal of Forecasting* **31**(4), 1009–1020.

Giannone, D., Lenza, M. & Primiceri, G. E. (2015), 'Prior selection for vector autoregressions', *Review of Economics and Statistics* **97**(2), 436–451.

Gneiting, T., Balabdaoui, F. & Raftery, A. E. (2007), 'Probabilistic forecasts, calibration and sharpness', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **69**(2), 243–268.

Gneiting, T. & Raftery, A. E. (2007), 'Strictly proper scoring rules, prediction, and estimation', *Journal of the American Statistical Association* **102**(477), 359–378.

Gneiting, T. & Ranjan, R. (2011), 'Comparing density forecasts using threshold- and quantile-weighted scoring rules', *Journal of Business & Economic Statistics* **29**(3), 411–422.

Goldberg, P., Williams, C. & Bishop, C. (1997), 'Regression with input-dependent noise: A gaussian process treatment', *Advances in neural information processing systems* **10**.

Gupta, A. K. & Nagar, D. K. (2018), *Matrix variate distributions*, Chapman and Hall/CRC.

Görtler, J., Kehlbeck, R. & Deussen, O. (2019), 'A visual exploration of gaussian processes', *Distill* . https://distill.pub/2019/visual-exploration-gaussian-processes.

Hauzenberger, N., Huber, F., Marcellino, M. & Petz, N. (2021), 'Gaussian process vector autoregressions and macroeconomic uncertainty', *arXiv preprint arXiv:2112.01995* .

Hensman, J., Fusi, N. & Lawrence, N. D. (2013), 'Gaussian processes for big data', *arXiv preprint arXiv:1309.6835* .

Hernández-Lobato, D., Sharmanska, V., Kersting, K., Lampert, C. H. & Quadrianto, N. (2014), 'Mind the nuisance: Gaussian process classification using privileged noise', *Advances in Neural Information Processing Systems* **27**.

Huber, F., Koop, G., Onorante, L., Pfarrhofer, M. & Schreiner, J. (2020), 'Nowcasting in a pandemic using non-parametric mixed frequency vars', *arXiv preprint arXiv:2008.12706* .

Hwang, Y., Tong, A. & Choi, J. (2016), Automatic construction of nonparametric relational regression models for multiple time series, *in* 'International Conference on Machine Learning', PMLR, pp. 3030–3039.

JONG, P. D. & Mackinnon, M. J. (1988), 'Covariances for smoothed estimates in state space models', *Biometrika* **75**(3), 601–602.

Jylänki, P., Vanhatalo, J. & Vehtari, A. (2011), 'Robust gaussian process regression with a student-t likelihood.', *Journal of Machine Learning Research* **12**(11).

Kastner, G. & Frühwirth-Schnatter, S. (2014), 'Ancillarity-sufficiency interweaving strategy (asis) for boosting mcmc estimation of stochastic volatility models', *Computational Statistics & Data Analysis* **76**, 408–423.

Kingma, D. P. & Welling, M. (2014), Stochastic gradient vb and the variational auto-encoder, *in* 'Second International Conference on Learning Representations, ICLR', Vol. 19.

Koop, G. & Korobilis, D. (2013), 'Large time-varying parameter vars', *Journal of Econometrics* **177**(2), 185–198.

Koop, G. M. (2003), *Bayesian econometrics*, John Wiley & Sons Inc.

Koop, G. M. (2013), 'Forecasting with medium and large bayesian vars', *Journal of Applied Econometrics* **28**(2), 177–203.

Koop, G., McIntyre, S. & Mitchell, J. (2020), 'Uk regional nowcasting using a mixed frequency vector auto-regressive model with entropic tilting', *Journal of the Royal Statistical Society: Series A (Statistics in Society)* **183**(1), 91–119.

Koop, G., McIntyre, S., Mitchell, J. & Poon, A. (2020), 'Regional output growth in the united kingdom: More timely and higher frequency estimates from 1970', *Journal of Applied Econometrics* **35**(2), 176–197.

Koop, G., McIntyre, S., Mitchell, J., Poon, A. et al. (2020), 'Reconciled estimates of monthly gdp in the us'.

Koopman, S. J. (1993), 'Disturbance smoother for state space models', *Biometrika* **80**(1), 117–126.

Korobilis, D. (2021), 'High-dimensional macroeconomic forecasting using message passing algorithms', *Journal of Business & Economic Statistics* **39**(2), 493–504.

Lázaro-Gredilla, M. & Titsias, M. K. (2011), Variational heteroscedastic gaussian process regression, *in* 'ICML'.

Lee, J., Bahri, Y., Novak, R., Schoenholz, S. S., Pennington, J. & Sohl-Dickstein, J. (2017), 'Deep neural networks as gaussian processes', *arXiv preprint arXiv:1711.00165* .

Leibfried, F., Dutordoir, V., John, S. T. & Durrande, N. (2020), 'A tutorial on sparse gaussian processes and variational inference', *CoRR* **abs/2012.13962**.
URL: *https://arxiv.org/abs/2012.13962*

Lenza, M. & Primiceri, G. E. (2020), How to estimate a var after march 2020, Technical report, National Bureau of Economic Research.

Li, M. & Koopman, S. J. (2021), 'Unobserved components with stochastic volatility: Simulation-based estimation and signal extraction', *Journal of Applied Econometrics* **36**(5), 614–627.

Loaiza-Maya, R., Smith, M. S., Nott, D. J. & Danaher, P. J. (2021), 'Fast and accurate variational inference for models with many latent variables', *Journal of Econometrics* .

MacKay, D. J., Mac Kay, D. J. et al. (2003), *Information theory, inference and learning algorithms*, Cambridge university press.

Makalic, E. & Schmidt, D. F. (2015), 'A simple sampler for the horseshoe estimator', *IEEE Signal Processing Letters* **23**(1), 179–182.

Marcellino, M., Porqueddu, M. & Venditti, F. (2016), 'Short-term gdp forecasting with a mixed-frequency dynamic factor model with stochastic volatility', *Journal of Business & Economic Statistics* **34**(1), 118–127.

Matthews, A. G. d. G., Rowland, M., Hron, J., Turner, R. E. & Ghahramani, Z. (2018), 'Gaussian process behaviour in wide deep neural networks', *arXiv preprint arXiv:1804.11271* .

Matthews, A. G. d. G., Van Der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z. & Hensman, J. (2017), 'Gpflow: A gaussian process library using tensorflow.', *J. Mach. Learn. Res.* **18**(40), 1–6.

McCracken, M. & Ng, S. (2020), Fred-qd: A quarterly database for macroeconomic research, Technical report, National Bureau of Economic Research.

Ormerod, J. T. & Wand, M. P. (2010), 'Explaining variational approximations', *The American Statistician* **64**(2), 140–153.

Rasmussen, C. E. (2003), Gaussian processes in machine learning, *in* 'Summer School on Machine Learning', Springer, pp. 63–71.

Saul, A. D., Hensman, J., Vehtari, A. & Lawrence, N. D. (2016), Chained gaussian processes, *in* 'Artificial Intelligence and Statistics', PMLR, pp. 1431–1440.

Schorfheide, F. & Song, D. (2015), 'Real-time forecasting with a mixed-frequency var', *Journal of Business & Economic Statistics* **33**(3), 366–380.

Steinruecken, C., Smith, E., Janz, D., Lloyd, J. & Ghahramani, Z. (2019), The automatic statistician, *in* 'Automated Machine Learning', Springer, Cham, pp. 161–173.

Titsias, M. (2009), Variational learning of inducing variables in sparse gaussian processes, *in* 'Artificial intelligence and statistics', PMLR, pp. 567–574.

Wang, B. & Titterington, D. (2004), 'Lack of consistency of mean field and variational bayes approximations for state space models', *Neural Processing Letters* **20**, 151–170.

Williams, C. K. & Rasmussen, C. E. (2006), *Gaussian processes for machine learning*, Vol. 2, MIT press Cambridge, MA.

Wohlrabe, K. (2009), Forecasting with mixed-frequency time series models, PhD thesis, lmu.