Mixed frequency forecasting with Bayesian additive regression trees

Mona Sfaxi

Master's thesis, 30 ECTS
Stockholm, Sweden
Spring 2022

Department of Statistics
Stockholm University

Supervisor
Pär Stockhammar

No 2022:2

Abstract

A wide variety of extensions to Bayesian additive regression trees (BART) have been proposed during recent years. A few of the most current studies are devoted towards combining BART with vector autoregressive (VAR) models. This paper combines BART with a mixed frequency vector autoregressive (MF-BAVART) model similarly to Huber et al. (2020). The performance of the model is compared to other common mixed frequency Bayesian VAR (MF-BVAR) models using Swedish macroeconomic data, both for long forecast horizons and short ones, so called 'nowcasts'. The forecasting performance for longer horizons using a small model with three variables shows no improvement against the benchmark MF-BVAR model with a Minnesota prior and the MF-BAVART model is consistently less accurate in comparison. Additionally, the nowcasting results shows that the MF-BAVART model can compete with the other Bayesian models when it comes to the high frequency variables. A comparison before and during the pandemic indicates that accuracy improves for the low frequency variables when the nowcast is initiated at a later month in a quarter. Futhermore, the MF-BAVART model overall shows of similar results compared to the other Bayesian models but it generally does not perform better with any large margin but the algorithm comes at a much greater computational cost with increasing instability as the parameter space expands. The results also strongly indicates that GDP seems to benefit the most out of the mixed frequency setting, where the competing Bayesian models perform better but where the MF-BAVART model also makes some clear improvements and clearly surpasses the simple AR(1) model.

Keywords

Bayesian additive regression trees, VAR, Macroeconomic forecasting, nonparametric, Gibbs sampling, Metropolis Hastings, MCMC, State Space, Machine Learning

Acknowledgments

I first want to give my sincere thanks to my supervisor Pär Stockhammar for all your expert advice, which I tried to follow to the best of my ability. It's been a joy working with you. I want to thank you for your patience and also for taking this project on and the time you put into this, even though it was a bit risky and you probably received a bit of a headache sometimes.

I also want to give a huge thanks to Mattias Villani for always going out of his way for his students and for taking the time to discuss some models and concepts and for going through some ideas, I really appreciate it.

Finally, I'm also really grateful to Håkan Slättman for the tech support. I might not have finished in time without your help.

Table of contents

1 Introduction

1.1 Background

Vector autoregressive (VAR) models have been widely used among particularly researchers concerned with macroeconomic data and as forecasting tools in different government agencies ever since they were first introduced by Sims (1980)[1]. The research field is huge and policy makers often rely on forecasts to implement changes in the economy. It's also quite common to use Bayesian vector autoregressive (BVAR) models when the parameter space is quite rich with relatively few observations.

A common issue for macroeconomic data though is that they're often published with different frequencies. GDP is often published at a quarterly or yearly basis, while the inflation and unemployment rates are published at a monthly one. Mixed-frequency models have emerged during the last couple of years where the aim often is to produce so called 'Nowcasts'. This enables policy makers to receive forecasts on macroeconomic indicators for very short time spans, even for those that are normally produced at a quarterly basis. These models also allow users to include more observations instead of aggregating all higher frequency data into the lowest occurring frequency order (e.g. aggregating monthly data to quarterly) which particularly can be an issue with Swedish data since many series are quite short.

Two popular approaches used for forecasting data composed of different frequencies are mixed data sampling (MIDAS) regression and mixed frequency VAR (MF-VAR) models. The former was initially developed by Ghysels et al. (2002, 2005) and differs substantially from MF-VAR models since it does not assume an underlying autoregressive model. For instance, the approach in this paper using a MF-VAR is that all variables, even the quarterly ones revolve around a monthly frequency but that the monthly observations of the low frequency variables are latent (see section 3.4), while MIDAS does not impose this restriction. This has the affect that MF-VAR models has to be represented in a State space form while MIDAS does not. In comparing MIDAS regression with MF-VAR, Kuzin et al. (2011) observed that MIDAS seemed to produce more accurate nowcasts while the MF-VAR model worked better for longer time spans. They also mention the common issue while working with VAR models, which is that the parameter space rapidly expands. Formulating an MF-BVAR model can instead allow the user to shrink the parameter space with the use of an appropriate restrictive prior. Schorfheide and Song (2015), while using a Minnesota prior for their MF-BVAR, found that it yielded better forecasts on the macroeconomic indicators in their first dataset compared to the benchmark VAR that was used, which was just a standard VAR model with quarterly aggregated data (QF-VAR). But they also noted that this only seemed to hold for short run forecasts and didn't make a large impact in the long run. The authors also inferred that the forecast accuracy could be affected depending on which month in a given quarter the forecast horizon was initiated at.

At the same time, methods that are utilized more extensively in the traditional Machine learning literature have become more popular to use for time series forecasting. One common occurring issue in some Machine learning methods is that the models may sometimes be overly complex and difficult to interpret, especially in the neural net

[1] Christopher A. Sims was awarded the Nobel prize in economics 2012 for his contribution to econometric research.

framework. A quite popular approach are ensemble methods like Boosting and Bagging. The general practice in ensemble methods is to use a common machine learning method as for instance a regression (or classification) tree and then learn the model by training several different, often small trees and then average or add the results (taking majority votes for classification trees). By utilizing several weak learners, the variance or bias can often be greatly reduced and the model is also less prone to adapt or overfit to training data, thus making it more adaptable to new data. The objective is in most cases not to minimize the training error but to obtain a model that can achieve a good predictive performance and minimize the error[2] based on new unseen data. This also justifies the use of cross validation where the data is split into training and test sets in order to evaluate performance where the model utilizes the training data for learning the model but is blind towards the data in the test set (Lindholm et al., 2022, pp. 57).

A common issue with Bagging type methods, is that although the variance is reduced, the bias is pretty much unaffected. In a similar manner, Boosting mostly influences the potential bias, while using weak learners where the model parameters are updated through each iteration (Lindholm et al., 2022, pp. 135). Bayesian additive regression trees (BART), originally proposed by Chipman et al. (2010) being heavily influenced by their CART model in Chipman et al. (1998), belongs to the ensemble machine learning methods. But it allows the user to set a prior and easier evaluate model performance from a statistical point of view since it's a Bayesian approach. One of its attractive features is that it's nonparametric, so the user doesn't have to specify a certain model for the specific data. As comparison, in using linear regression, one might have several competing models with a different number of higher degree polynomials and possibly interaction terms. The nonlinear characteristic of BART allows for flexible adjustments to data without the user having to make that many decisions. But Lindholm et al. (2022, pp. 63) argue that increasing the complexity of a model (in this case as adding more variables and growing larger trees) often contributes to reducing bias but the variance generally also grows larger and too complex models tend to be more influenced by training data, which often leads to overfitting. Chipman et al. (2010) points to BART being robust thanks to its restrictive priors which often only leads to growing small trees. Their simulation study showed that it can still perform quite decently, even when the parameters in a model exceeds the number of observations by a large margin.

1.2 Research problem

Although quite recent, there have been a few studies done in combining BART with VAR models (BAVART), which have shown promising results. For instance in combining a mixed frequency BVAR (MF-BVAR) with a BART model (MF-BAVART), Huber et al. (2020), focused on the predictive performance of GDP from four large economies in Europe as response variables with the other covariates in monthly frequencies. They found that their model was particularly useful for time series data with huge outliers. Similarly to Schorfheide and Song (2015), their forecasts of the quarterly variable GDP were improved by making use of the monthly frequencies of the predictors in their sample. Huber and Rossini (2020), using a single frequency BAVART model, found that it was particularly useful for data that are highly nonlinear. Clark et al. (2021), while also using a BAVART model in a macroeconomic setting, maintained that using a nonlinear approach was convenient for heteroskedastic data and that forecasting accuracy could

[2]The error can be defined in several different ways but the general idea is that it's a quantifiable measure on how much the model predictions deviates compared to the true values.

be greatly improved by allowing for flexible models such as BART.

The literature concerning Nowcasting is often devoted to evaluating the predictive performance of variables of low frequencies, often GDP when using macroeconomic data. The performance of the high frequency variables are often not of main focus. This paper focuses on the merging of MF-VAR with BART in a similar manner as Huber et al. (2020) but the predictive performance of the monthly variables are also considered and several low frequency variables are also used in the same VAR model. The model's long- and short term forecasting accuracy is evaluated and compared against other traditional MF-BVAR models with Minnesota and steady state priors (see section 3.4.1). The data mainly consist of Swedish macroeconomic indicators, of which the unemployment rate, inflation and GDP are of most interest.

1.3 Outline

The outline of the study is as follows. In section 2, the data, which covers some of the most important indicators for the Swedish economy is presented and described. Section 3 covers the method used. An introduction to the general theory behind BART with a single outcome variable is first depicted followed by a very short description of VAR models, accompanied by a brief presentation of time series in State Space form before going into the details of the MF-BAVART model. Some ethical remarks is given in section 4. Section 5 is devoted to the results, with both long and short term forecasting comparisons between the MF-BAVART and more traditional MF-BVAR models. The nowcasting performance of all the Bayesian models are also compared to the performance of a basic AR(1) model. A discussion of the results is given in section 6, along with some concluding remarks and suggestions for further research. A list of some abbreviations used throughout the text can be found in Appendix C.

2 Data

The main indicators of interest are real Gross domestic product (GDP), Fixed consumer price index (CPIF inflation) and the unemployment rate. A few of the other variables are just used as predictors. Since the analysis is essentially based on a VAR-model, the lags of the main indicators are used to predict the outcomes as well as the lags of the other predictors. The original sample ranges from January 1996 to December 2021 and consists of data revolving the Swedish economy with no missing observations during that timespan. The final dataset ranges from January 1997 to December 2021, this is after performing some transformations on the original sample. Three sources were mainly used for retrieving the Swedish data, Statistics Sweden which are responsible for producing official statistics in Sweden, the Riksbank which is Sweden's central bank, and the national institute of economic research.

Table 1: Data Description

Variable	Freq.	Transformation	Source
Real Gross Domestic Product (GDP)	Quarterly	Annual %-change	i
Fixed Consumer Price Index (CPIF)	Monthly	Annual %-change	i
Unemployment	Monthly	Level	i
Government Bonds 5 years	Monthly	Level	ii
Government Bonds 10 years	Monthly	Level	ii
Treasury Bills 3 months	Monthly	Level	ii
Expected inflation in business sector	Quarterly	Level	iii
Total industry capacity utilization	Quarterly	Level	i
Production of total industry index (IPI)	Monthly	Annual %-change	v
Effective exchange rate index (KIX)	Monthly	Level	ii
Wage inflation public sector	Quarterly	Level	iv

i = statistics Sweden. ii = the Riksbank. iii = national institute of economic research.
iv = Swedish national mediation office. v = FRED.

Table 1 summarizes the dataset. It consists of a mix of monthly and quarterly series, some of which were already de-seasoned upon download from their respective sources. The seasonal patterns have been removed for the other series which could not be obtained without a seasonal pattern except for the interest rates (the 3 months treasury Bills and the 5 and 10 year Bonds) and the weighted currency index KIX and the expected inflation. The stl-function with its default setting, which is included in the basic stats pacage in r was used to remove the seasonal patterns. The stl-function uses Loes smoothing in order to decompose a series into different components (like trend and season) and makes it possible for the user to extract a de-seasoned series. The function only allows for additive models, but multiplicative models can easily be incorporated too by first using a log-transformation (Hyndman and Athanasopoulos, 2018).

As indicated by the table, a few of the series have been transformed and consist of annual log-differences using the same period observation from the previous year (monthly in monthly data and quarterly in quarterly data). It's commonplace to either use this approach of differencing or to difference the data based on the most adjacent observation. This allows in the first instance for interpretations as year to year percentage changes and in the latter case as the percentage change in one observation from the previous one. The reason for using year to year changes is because some of the acquired variables are only available in that format, which deems it appropriate to use the same approach for all other variables. The reason for using logs is that there's otherwise an asymmetric issue with conventional percentage differences (Cole and Altman, 2017). Consider for instance a raise in price from 9 to 11. The percentage increase is given as $\frac{11-9}{9} \cdot 100 = 22.2\%$ while a decrease from 11 to 9 is given as $\frac{9-11}{11} \cdot 100 = -18.2\%$. Comparing this to the log difference $100 \cdot (ln(11) - ln(9)) = 20.1\%$ increase and $100 \cdot (ln(9) - ln(11)) = 20.1\%$ decrease, where the value from the latter method is symmetric.

A time series process y_t can be defined as stationary if $E(y_t) = \mu$ and if the covariance between two observations separated by the distance 'h' only depends on this distance Lütkepohl (2006, pp. 24)[3]. Apart from attaining data in growth rates by using (log) differences, doing so can also potentially transform a non-stationary series into a stationary

[3]It's possible to use a stricter definition of stationarity

one. Although the unemployment rate, exchange- and interest rates may exhibit some visible trends, they're considered as stationary in the long run. It's therefore common to treat them as such in research and neither seasonally adjust them, nor take their difference in Swedish applications, see for instance Lindholm et al. (2020), where the corresponding variables are in their respective levels. In Clark et al. (2021), the interest rates are also kept at their corresponding level rates, while the unemployment rate is transformed with a negative first difference. See Table 7 in Appendix A.1 for a summarizing table containing the Phillips Perron test for each untransformed variable, where the null hypotheisis is that the process contains a unit root, resulting in a non-stationary process. It clearly shows that some of the variables can't be considered as stationary, although the table strongly indicates that three out of four of the quarterly variables can be considered as level stationary while GDP seem to be stationary around a trend. Table 8 shows another picture, where the null hypothesis in the Phillips Perron test is rejected for each transformed variable.

A more detailed description and visualization of each variable is given in Appendix A.1.

3 Method

This section is devoted to the theoretical background of the different components comprising the MF-BAVART model as well as the formulation of the model itself. A short presentation of the hyperparameters is also given followed by a brief discussion about the use of different priors in the VAR models.

3.1 The BART model with a univariate response

Bayesian additive regression trees (BART) is a Bayesian application of ensemble methods. Parallels can be drawn to it and boosting since both methods typically rely on weak learners when learning the parameters. Cross-validation is often used in Boosting for choosing the number of trees and terminal nodes, while BART relies on priors. BART is built upon the use regression trees with binary splitting decisions. Each tree has one root, that may, or may not split into two new nodes. Generally, all nodes have the potential to split into new nodes and in turn form a deeper tree. Some covariates may be more important than others in deciding the splits. Each split of a node has a binary outcome. A tree may also have branches and internal, as well as terminal nodes. The terminal nodes are sometimes referred to as leaves. All (non-root) nodes have a branch connected to them (Tan and Roy, 2019). The basic univariate response model is usually defined as

$$y = f(\mathbf{x}) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2) \tag{3.1}$$

where $\mathbf{x} = (x_1, x_2, ..., x_k)'$ is a vector of covariates and $f(\mathbf{x})$ can be some non-linear function. In order to make predictions and inferences, the aim is to obtain

$$f(\mathbf{x}) = E(Y|\mathbf{x}) \tag{3.2}$$

Let $g_s(\mathbf{x}; \mathcal{T}, \mathcal{M})$ be a regression tree with parameters $\mathcal{T}$ and $\mathcal{M}$. Easing the notation of $g_s(\mathbf{x}; \mathcal{T}, \mathcal{M})$ to $g_s(\mathbf{x})$, the conditional mean of Y can be approximated by a sum of tree

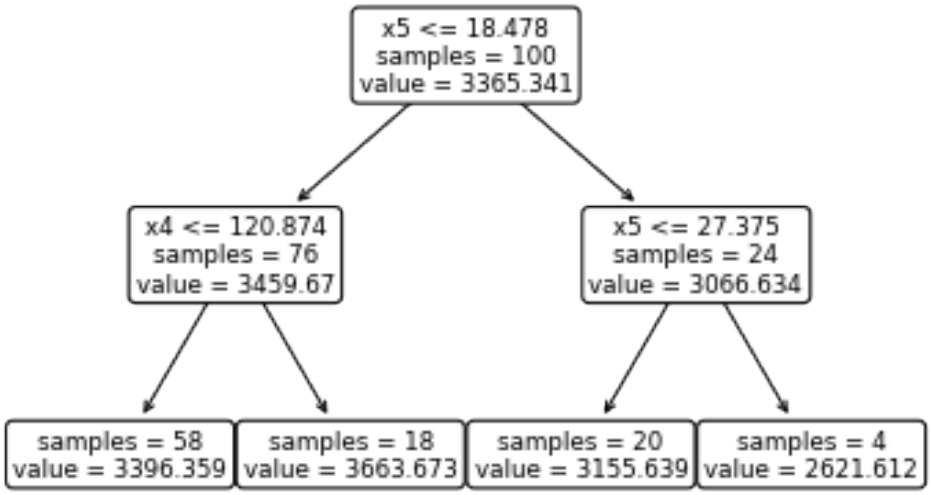


Figure 1: A Single Regression tree using 5 covariates with simulated data and a maximum depth of 2.

model

$$f(\mathbf{x}) \approx \sum_{s=1}^{S} g_s(\mathbf{x}) \tag{3.3}$$

(Chipman et al., 2010). Assuming that only one tree $\mathcal{T}$, with all it's branches and nodes, is used in the model, the model can be decomposed containing the parameter $\mathcal{T}$ and $\mathcal{M}$, where $\mathcal{M} = \{\mu_1, \mu_2, ..., \mu_B\}$. Each μ_b is a parameter value belonging to a terminal node in the tree.

An illustration of how a single tree may look like is given in Figure 1. A total of five covariates were used in a simulated dataset. All of them were continuous, but it would also be possible to use binary or discrete covariates. As seen by the tree structure, which has been pruned to a maximum depth of 2, only 2 of the covariates were used for splitting the nodes in this tree. The other covariates had the potential to be utilized too for deciding the splits, but they weren't in this case. As seen, the root node is split based on x_5. A value of $x_5 \leq 18.478$ results in a move to the left, otherwise, we move to the right branch. We can see that 76 of the 100 observations are in the left internal node and the next split is based on x_4. If $x_4 \leq 120.874$ we move to the left terminal node, otherwise we move to the right terminal node. We can also see that x_5 is used again for determining the split for the other internal node. This illustrates that while a binary covariate has a one-strike you're out rule in a tree, continuous covariates may be used several times in the same tree, depending on how many unique values they contain. Generalizing this to a total of S trees, each tree can be denoted as having the parameters $\mathcal{T}_s$ and $\mathcal{M}_s$, where $\{\mathcal{T}_1, ..., \mathcal{T}_S\}$ is a set of different tree structures (each potentially containing different number of nodes and depths) and $\mathcal{M}_s = \{\mu_{1s}, \mu_{2s}, ..., \mu_{Bs}\}$ refers to the obtained parameters' values in each of the terminal nodes in a given tree. Thus the approximated model becomes:

$$Y = \sum_{s=1}^{S} g_s(\mathbf{x}) + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2) \tag{3.4}$$

If there is more than one covariate $\mathbf{x}$, a tree can possibly also contain an interaction effect between the covariates. This occurs when the depth is larger than 1 (Chipman et al., 2010) and can for instance be seen in figure 1. The first split is based on x_5 while the second split in the left internal node is based on x_4 and so contains an interaction effect between x_4 and x_5.

All observations are assigned to a specific value of a terminal node based on the splitting criteria for that tree. This means that some observations may be assigned μ_{1s} while others may be assigned the value μ_{2s} or $\mu_{3s}, ...\mu_{Bs}$ and so on depending on how many terminal nodes are present in a tree. Each terminal node can also be seen as a predicted y-value and had there been several trees, then terminal nodes from these trees could be combined and added together to produce a prediction of y (Tan and Roy, 2019). An example of this in the single tree case is the leftmost terminal node where the 58 observations assigned to it are given the value 3396.359. The 18 observations adhering to the terminal node right next to it receives a value of 3663.673 and so on.

An alternative way to view BART is as a sum of ANOVA, where each tree is an ANOVA model (see Chipman et al., 2010).

3.1.1 The three prior components of BART

The priors serve as a regularization tool and can be denoted by

$$Pr((\mathcal{T}_1, \mathcal{M}_1), (\mathcal{T}_2, \mathcal{M}_2), ..., (\mathcal{T}_S, \mathcal{M}_S), \sigma^2)$$

Since σ^2 is assumed to be independent of the trees, the priors can be written as

$$Pr(\sigma^2) \prod_{s=1}^{S} P(\mathcal{T}_s, \mathcal{M}_s)$$

which could also be re-written as

$$Pr(\sigma^2) \prod_{s=1}^{S} Pr(\mathcal{M}_s|\mathcal{T}_s) Pr(\mathcal{T}_s) = Pr(\sigma^2) \prod_{s=1}^{S} \left[\prod_{b=1}^{B} Pr(\mu_{bs}|\mathcal{T}_s) \right] Pr(\mathcal{T}_s) \tag{3.5}$$

In summary, three types of priors are used; $Pr(\mathcal{T}_s), Pr(\mu_{bs}|\mathcal{T}_s)$ and $Pr(\sigma)$ since each tree is considered independent from other trees. The two latter priors are mainly chosen based on conjugacy to ease the computational burden. The prior for the tree structure $Pr(\mathcal{T}_j)$ consists of three parts. The first part is

$$\frac{\alpha}{(1+d)^\beta}, \qquad 0 < \alpha < 1, \ \beta \geq 0 \tag{3.6}$$

where α and β are hyperparameters and d represents the depth of the tree. This controls how likely a node is to split, and it's what serves as a regularization tool on the tree structure. Higher values of α all things equal will make a node splitting more probable. All the while higher values of β all things equal will have the opposite effect and reduce the amount of terminal nodes in a tree.

The second two parts of the prior for the tree structure $\mathcal{T}_s$ consist of choosing which covariate to split on and then choosing which value of said covariate to split upon. The covariate to split on is chosen uniformly. This means that all covariates have an equal

probability of being chosen. The value to split on, given the covariate that was selected, is also decided uniformly. This is the default settings in Chipman et al. (2010) where they argue that it yields robust results across different settings even though it's possible to specify more informed splitting rules for the covariates.

The prior $Pr(\mu_{bs}|\mathcal{T}_s) \sim N(\mu_\mu, \sigma_\mu^2)$ and the prior adopted for σ^2 is often based on a conjugate distribution Chipman et al. (2010) as well as Huber et al. (2020) used an inverse-χ^2 distribution where $\sigma^2 \sim \nu\lambda/\chi^2_\nu$. As shown these two priors require four hyperparameters of which a more thorough discussion is given in section 3.4.2.

3.1.2 The posterior distribution

The posterior distribution denoted by

$$Pr((\mathcal{T}_1, \mathcal{M}_1)), ..., (\mathcal{T}_S, \mathcal{M}_S), \sigma^2|y) \propto \\ Pr(y|(\mathcal{T}_1, \mathcal{M}_1), ..., (\mathcal{T}_S, \mathcal{M}_S), \sigma^2) Pr((\mathcal{T}_1, \mathcal{M}_1), ..., (\mathcal{T}_S, \mathcal{M}_S), \sigma^2) \quad (3.7)$$

with observed y is obtained through the use of a Metropolis Hastings algorithm inside a Gibbs sampler. Although two well known procedures, the drawback is that it can be quite slow, especially as the number of conditional draws in the Gibbs sampler increases. Even though packages for BART are usually written in high level languages with C++ or Java extensions, it's still quite slow.

Each draw from the posterior is obtained by first drawing tree structure $\mathcal{T}_s$ and terminal node parameters $\mathcal{M}_s$ conditionally on σ, the outcome variable as well as the *set* of all other tree structures and node parameters, excluding the one with the s'th index, denoted as $s^{\complement}$ where $s \neq s^{\complement}$, i.e

$$Pr(\mathcal{T}_j, \mathcal{M}_s|\mathcal{T}_{s^\complement}, \mathcal{M}_{s^\complement}, \sigma, y) \quad (3.8)$$

3.1.3 A closer look inside the Gibbs Sampler

Let the s'th residual be denoted as

$$R_s = y - \sum_{s^\complement} g(\mathbf{x}; \mathcal{T}_{s^\complement}, \mathcal{M}_{s^\complement}) \quad (3.9)$$

Then drawing the posterior for the s'th tree structure and terminal node parameters is equivalent to drawing $\mathcal{T}_s, \mathcal{M}_s$ conditional on R_s, and σ, i.e.

$$Pr(\mathcal{T}_j, \mathcal{M}_s|R_s, \sigma)$$

Since the prior for $\mathcal{M}_s$ is conjugate, it can easily be integrated out, thus leaving the draws of $\mathcal{T}_s$ unbothered by $\mathcal{M}_s$ and doing so gives us

$$Pr(\mathcal{T}_s|R_s, \sigma) \quad (3.10)$$

The posterior for the Tree structure in (3.10) is intractable but if can be approximated by the Gibbs sampler. As previously mentioned, it consist of three parts.

The Gibbs sampler is first initiated by setting all the S root nodes $\mu^{(\mathbf{0})} = \frac{\bar{y}}{S}$. This is followed by a calculation of $R_s^{(1)}$ in the first iteration. It's a simple calculation for R_1,

in the first iteration. For some intuition on a general Gibbs sampler, see Appendix B.1. The same basic principle applies as illustrated there, at the end there will be a total of S leave one out residual (R_s) values in the i'th iteration. Following this is a Metropolis Hastings algorithm; A proposal of a new tree structure for the s'th tree is given as

$$\mathcal{T}_s^* \sim Pr(\mathcal{T}_s^* | \mathcal{T}_s^{(i-1)}) \tag{3.11}$$

where

$$Pr(\mathcal{T}_s^* | \mathcal{T}_s^{(i-1)}) = \pi_1 \cdot \frac{1}{Bs} \cdot \frac{1}{\mathcal{L}} \cdot \frac{1}{\kappa} \tag{3.12}$$

Here π_1 represents the probability of growing the tree, Bs is the total number of terminal nodes in the previous given tree, $\mathcal{L}$ represents the remaining variables that are left who didn't already strike out and κ are the remaining distinct values in a variable.

Similarly $\Pr(\mathcal{T}_s^{(i-1)} | \mathcal{T}_s^*)$ is given as

$$Pr(\mathcal{T}_s^{(i-1)} | \mathcal{T}_s^*) = \pi_2 \cdot \frac{1}{\breve{n}} \tag{3.13}$$

π_2 denotes the pruning probability so this equation refers to the pruning part of the algorithm, which is the opposite of the growth step. $\breve{n}$ denotes the total number of internal nodes that gave rise to only two terminal nodes (Tan and Roy, 2019).

Apart from the growth and pruning step, there's also a swap and a change step, which Chipman et al. (2010) argue are important for the performance of the algorithm. The swap step picks an internal node at random and replaces the old criteria for splitting the node with a new random one. The change step changes the splitting decision between two internal nodes that have a branch connected to them. They are used in a similar manner as the growth and pruning proposals given by (3.12) and (3.13). See Chipman et al. (2010) or Kapelner and Bleich (2016, pp. 36) for more details.

After drawing the proposal $\mathcal{T}_s^*$ the ratio

$$r = \frac{q(\mathcal{T}_s^* | \mathcal{T}_s^{(i-1)})}{q(\mathcal{T}_s^{(i-1)} | \mathcal{T}_s^*)} \frac{Pr(R_s | X, \mathcal{T}_s^*, \mathcal{M}_s)}{Pr(R_s | X, \mathcal{T}_s^{(i-1)}, \mathcal{M}_s)} \frac{Pr(\mathcal{T}_s^*)}{Pr(\mathcal{T}_s^{(i-1)})} \tag{3.14}$$

is calculated. The proposal $\mathcal{T}_s^*$ is accepted with a probability of $min(1, r)$. The new tree $\mathcal{T}_s^{(i)}$ is then updated to the proposed value T_s^* if this was accepted, otherwise $\mathcal{T}_s^{(i)} = \mathcal{T}_s^{(i-1)}$. This constitutes the Metropolis Hastings step inside the Gibbs sampler and the proposal and it's potential acceptance as a new tree structure could arguably be viewed as the main learning part of the algorithm. After a few hundred or thousand iterations, the algorithm will start to learn which variables are of more importance for splitting the nodes. Even if the variable chosen to split upon is decided randomly, covariates that are less important for predicting the outcome will less likely be chosen and accepted.

Finally, each terminal node μ_{bs} in a tree $\mathcal{T}_s$ has a normal posterior given as

$$Pr(\mu_{bs} | \mathcal{T}_s, R_s, \sigma) \propto exp \left\{ \frac{\left(\mu_{bs} - \frac{\sigma_\mu^2 \sum_l R_{bsl} + \sigma_\mu^2 \mu_\mu}{n_{bs} \sigma_\mu^2 + \sigma^2} \right)^2}{2 \frac{\sigma^2 \sigma_\mu^2}{n_{bs} \sigma_\mu^2 + \sigma^2}} \right\} \tag{3.15}$$

where R_{bsl} is the set of observations going to the terminal node in the s'th tree and n_{bs} is the total number of observations going to said node.

Since σ^2 is considered independent, its posterior is a scaled inverse-χ^2_ν given by

$$Pr(\sigma|\mathcal{T}_1, \mathcal{M}_1, ..., \mathcal{T}_S, \mathcal{M}_S, y) \sim Inv - \chi^2\Big(\nu + T, \quad \frac{\nu\lambda + \sum_{t=1}^{T}(Y_t - \sum_{s=1}^{S} g(x_t, \mathcal{T}_s\mathcal{M}_s))^2}{\nu + T}\Big) \tag{3.16}$$

Where T denotes the total sample size. Therefore, the posterior for the tree structures and terminal nodes in (3.8) needs to be drawn first.

A summary of the algorithm is given below

Gibbs sampling for BART

Input:
Data; **X** and **y**
hyperparameters: $\alpha, \beta, \mu_\mu, \sigma_\mu, \lambda, \nu, \pi_1, \pi_2, \pi_3, \pi_4$
number of Trees: S
number of iterations: N + burn-in

Initialize starting values for all the S root nodes $\mu_1^{(0)}, \mu_2^{(0)}, ..., \mu_S^{(0)}$

For i = 1,2,...N+burn-in
For s = 1,2,...,S
Calculate $R_s^{(i)} = y - \sum_{s\complement} g(\mathbf{x}; \mathcal{T}_{s\complement}, \mathcal{M}_{s\complement})$
Draw Tree structure proposal $\mathcal{T}_s^* \sim Pr(\mathcal{T}_s^*|\mathcal{T}_s^{(i-1)})$
Calculate the ratio:

$$r = \frac{q(\mathcal{T}_s^*|\mathcal{T}_s^{(i-1)})}{q(\mathcal{T}_s^{(i-1)}|\mathcal{T}_s^*)} \frac{Pr(R_s|X, \mathcal{T}_s^*, \mathcal{M}_s)}{Pr(R_s|X, \mathcal{T}_s^{(i-1)}, \mathcal{M}_s)} \frac{Pr(\mathcal{T}_s^*)}{Pr(\mathcal{T}_s^{(i-1)})}$$

Set $\mathcal{T}_s^{(i)} = \mathcal{T}_s^*$ with probability min(1,r),
otherwise use $\mathcal{T}_s^{(i)} = \mathcal{T}_s^{(i-1)}$
Draw $Pr(\mu_{bs}|\mathcal{T}_s^{(i)}, R_s^{(i)}, \sigma)$ as in (3.15)

Draw $Pr(\sigma|\mathcal{T}_1^i, \mathcal{M}_1^i, ..., \mathcal{T}_S^i, \mathcal{M}_S^i, y)$ as in (3.16)

Remove the burn-in draws

Output: N posterior draws $Pr((\mathcal{T}_1, \mathcal{M}_1)), ..., (\mathcal{T}_S, \mathcal{M}_S), \sigma^2|y)$

For more rigorous details and derivations see Kapelner and Bleich (2016) or Tan and Roy (2019).

3.2 VAR models

A general VAR(p) model is assumed to be stationary and consists of 1, 2,..., p lags. The simplest model, a VAR(1) with a total of K response variables can be denoted as

$$\mathbf{y}_t = \mathbf{v} + \mathbf{A}_1\mathbf{y}_{t-1} + \mathbf{u}_t \tag{3.17}$$

where $\mathbf{y}_t$ is a K-dimensional vector of response variables. $\mathbf{v}$ is a K-dimensional vector of intercepts. $\mathbf{A}_1$ represents a symmetric (K x K) coefficient matrix for time point t-1 (and in the general case $\mathbf{A}_p$ represents the coefficient matrix for the p'th lag of $\mathbf{y}_t$). Naturally the $\mathbf{y}_{t-1}$ vector consist of the values $y_{1,t-1}, y_{2,t-1}, ..., y_{K,t-1}$. In summary, each response variable depends on its own first lag, as well as the first lag of all the other response variables. Writing out the matrices, this model would look like

$$\mathbf{y}_t = \begin{bmatrix} y_{1t} \\ y_{2t} \\ \vdots \\ y_{Kt} \end{bmatrix}, \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_K \end{bmatrix},$$

$$\mathbf{A}_1 = \begin{bmatrix} \gamma_{11,1} & \gamma_{12,1} & \cdots & \gamma_{1K,1} \\ \gamma_{21,1} & \gamma_{22,1} & \cdots & \gamma_{2K,1} \\ \vdots & & \ddots & \vdots \\ \gamma_{K1,1} & \gamma_{K2,1} & \cdots & \gamma_{KK,1} \end{bmatrix}, \mathbf{y}_{t-1} = \begin{bmatrix} y_{1,t-1} \\ y_{2,t-1} \\ \vdots \\ y_{K,t-1} \end{bmatrix}, \mathbf{u}_t = \begin{bmatrix} u_{1,t-1} \\ u_{2,t-1} \\ \vdots \\ u_{K,t-1} \end{bmatrix}$$

Generalizing this to a VAR(p) model with K response variables, we have

$$\mathbf{y}_t = \mathbf{v} + \mathbf{A}_1\mathbf{y}_{t-1} + \mathbf{A}_2\mathbf{y}_{t-2} + , ..., + \mathbf{A}_p\mathbf{y}_{t-p} + \mathbf{u}_t \tag{3.18}$$

where $\mathbf{u}_t$ is iid. white noise with mean vector equal to $\mathbf{0}$ and covariance matrix $\mathbf{\Sigma}$. It's also possible to write the stationary VAR(p) model as a moving average (MA) process but maximum likelihood methods would be required in order to estimate the coefficients. A VAR(p) process as given by (3.18) can also be written in companion form (Lütkepohl, 2006, pp. 15), i.e. in simplified terms, represented as a VAR(1) process such as

$$\mathbf{Y}_t = \breve{\mathbf{v}} + \mathbf{A}\mathbf{Y}_{t-1} + \mathbf{U}_t \tag{3.19}$$

where the dimensions of each matrix is $\mathbf{Y}_t = Kpx1$, $\breve{\mathbf{v}} = Kpx1$, $\mathbf{A} = KpxKp$, $\mathbf{U}_t = Kpx1$. Explicitly writing out these matrices gives us

$$\mathbf{Y_t} := \begin{bmatrix} \mathbf{y}_t \\ \mathbf{y}_{t-1} \\ \vdots \\ \mathbf{y}_{t-p+1} \end{bmatrix}, \breve{\mathbf{v}} := \begin{bmatrix} \mathbf{v} \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

$$\mathbf{A} := \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 & ... & \mathbf{A}_{p-1} & \mathbf{A}_p \\ \mathbf{I}_K & 0 & ... & 0 & 0 \\ \vdots & & \ddots & \vdots & \vdots \\ 0 & 0 & ... & \mathbf{I}_K & 0 \end{bmatrix}, \mathbf{U} := \begin{bmatrix} \mathbf{u}_t \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

The companion form of the process is often useful when formulating models. Even though the VAR(p) model has the coefficient matrices $\mathbf{A}_1, ..., \mathbf{A}_p$ it's usually also relevant to observe the impulse response functions (IRF) which can give a measure of how a shock in one variable effects another one along with the duration of the shock.

3.2.1 Stochastic volatility

Stochastic volatility is well used for modeling assets in the financial market. Different types of GARCH models have traditionally been used for VAR-modeling as a way to deal with serially correlated errors. The stochastic volatility model allows the variance

to vary and can be seen as an alternative to GARCH models. It uses a state space approach to model the log squared latent states as an AR(1) process (Kastner, 2016). The reader is referred to Kastner and Frühwirth-Schnatter (2014) or Kastner (2016) for details since the topic will not be treated in depth here but one of the alternative specified MF-BAVART models in section 5 utilizes the approach.

3.2.2 The curse of dimensionality

It's easy to see from (3.18) that a VAR(p) model with K equations can easily become inflated with a large number of parameters to estimate, even when p is chosen rather small. The dataset used here contains 11 variables. Using them all, with each variable being an outcome variable and with $p = 5$ lags would give us $11 \cdot (11 \cdot 5 + 1) = 616$ parameters to estimate, 56 in each equation. The non-lagged dataset itself consists of 300 observations (100 observations of the low frequency variables). Using a Bayesian approach allows for creating flexible models with the opportunity to impose regularization priors on the parameters. Flexible models would otherwise be difficult to implement in a VAR setting. A further discussion of common regularization priors is given in section 3.4.1. And for an extensive textbook treatment on multivariate time series and VAR models see Lütkepohl (2006) or Hamilton (1994).

3.3 Time series using a State Space representation

Mixed frequency (B)VAR models contain data of different frequencies. The objective of using these models is often to be able to produce more accurate forecasts for low-frequency indicators, often in the nearest time span, resulting in so called "Nowcasts". One could view the low frequency data as a missing data problem. To be able to estimate the missing data points, the mixed frequency VAR model is given in state space form. The Kalman filter, which has many different types of extensions and was originally derived by Kalman (1960), can then be used to impute the missing values. State space models and the Kalman filter have traditionally been well used in fields such as physics and engineering although the theory lends itself well to statistics too. It can differ substantially from the traditional ARIMA methods usually applied in the econometrics field. A brief introduction to State Space models is given below. For an introductory treatment through a statistical perspective the reader is referred to Commandeur and Koopman (2007) and for a more rigorous treatment of the topic see Durbin and Koopman (2012).

Consider the basic multivariate time series model in State space form as

$$\begin{aligned} \mathbf{y}_t &= \mathbf{V}_t\theta_t + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \mathbf{\Sigma}_{\eta_t}) \\ \theta_{t+1} &= \mathbf{W}_t\theta_t + \mathbf{M}\varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \mathbf{\Sigma}_{\varepsilon_t}), \quad t = 0, 1, ... \end{aligned} \tag{3.20}$$

This is a linear system where the first equation represents what's observed in the system while the second one, denoted as the state equation represents the unobserved. Estimation of the state vector θ_t is achieved recursively. The $\mathbf{y}_t$ vector can be viewed as the output and its dimension is Kx1 and independent of the dimension of the state vector. $\mathbf{\Sigma}_{\eta_t}$ and $\mathbf{\Sigma}_{\varepsilon_t}$ denotes the covariance matrices of the errors. Assuming that there's no trend, cyclical or seasonal component the whole model reduces to a multivariate local level model, Which means that it's a time series only containing a stochastic level. Its

counterpart in linear regression would be if the intercept was set to vary at each time point. The basic multivariate linear model in (3.20) as treated by Durbin and Koopman (2012, pp. 76) gives the same results regardless of approaching it from a frequentist point of view or a Bayesian one. The standard practice using ARIMA methods is to remove present disturbances such as trends and seasonality from the data in order to make it stationary. But they can instead be incorporated in state space models since it assumes that the underlying system is stochastic. Therefore the system matrices $\mathbf{V}_t, \mathbf{W}_t, \mathbf{M}_t, \mathbf{\Sigma}_{\eta_t}, \mathbf{\Sigma}_{\varepsilon_t}$ can be constructed somewhat differently depending on the application, for instance whether a seasonal component is included in the model or not. And as seen from the general implementation, the matrices can also vary with time, Although many times $\mathbf{M} = \mathbf{I}$ with the other matrices being time-invariant. And since $\mathbf{y}_t$ and θ_t can be of different dimensions the $\mathbf{V}$ matrix serves as a bridge between them. As an example of the connection between state space- and VAR models, borrowing from the system defined in Lütkepohl (2006, pp. 611) where the mean is assumed to be zero ($\check{\mathbf{v}} = \mathbf{0}$), results in the following VAR(p) process in companion form, illustrated again for convenience

$$\mathbf{Y}_t = \mathbf{A}\mathbf{Y}_{t-1} + \mathbf{U}_t$$

By first giving a sightly different representation of the system in (3.20) to ease the notation, with $t = 1, 2, ...$ instead of $t = 0, 1, ...$

$$\begin{aligned} \mathbf{y}_t &= \mathbf{V}_t\theta_t + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \mathbf{\Sigma}_{\eta_t}) \\ \theta_t &= \mathbf{W}_{t-1}\theta_{t-1} + \varepsilon_{t-1}, \quad \varepsilon_t \sim \mathcal{N}(0, \mathbf{\Sigma}_{\varepsilon_t}) \end{aligned} \tag{3.21}$$

and by assuming that $\mathbf{M}_t = \mathbf{I}$[4], the correspondence between the above state space model and the simple VAR(p) process can be seen as

$$\begin{aligned} \mathbf{y}_t &:= [\, \mathbf{I}_K \vdots 0 \vdots \cdots \vdots 0] \, \mathbf{Y}_t \\ \mathbf{V}_t &:= [\, \mathbf{I}_K \vdots 0 \vdots \cdots \vdots 0] \\ \eta_t &:= \mathbf{0} \\ \theta_t &:= \mathbf{Y}_t \\ \mathbf{W}_t &:= \mathbf{A}_t \\ \varepsilon_{t-1} &:= \mathbf{U}_t \end{aligned} \tag{3.22}$$

There are generally similarities between the linear state space model and linear regression, although, the state vector is assumed to vary over time and is updated for each new time point. An analogous counterpart in linear regression would be if the intercept and coefficient vector were set to vary at point in time. The Kalman filter for a linear model can be derived using the properties of the mean and covariance matrix in multivariate normal regression theory (see Meinhold and Singpurwalla, 1983, or Durbin and Koopman, 2012). The normality assumption, as well as the linear one can be relaxed if needed. Schorfheide and Song (2015) used a linear state space model for their mixed frequency BVAR but BART is essentially a nonlinear application and would require a deviation from the linearity assumption, while still presuming normality in its basic form. The complete derivation of the Kalman filter is outside the scope of this paper, but for the interested reader see for instance Durbin and Koopman (2012, pp. 82). Appendix B.2 gives a short description and display of the Kalman filter recursions to provide some intuition of the model.

[4]It's also possible to include an additional vector of covariates in the observed equation in (3.20) or (3.21).

3.4 Mixed frequency BAVART

To formulate the MF-BAVART model using a state space representation, assume that the data can be classified into two groups, one of low frequency kind and one of high frequency kind, or formally as $\mathbf{y}_t = [\ \mathbf{y}'_{q,t}, \mathbf{y}'_{m,t}]\ '$ where $\boldsymbol{y}_{q,t}$ refers to the low frequency monthly observations and $\boldsymbol{y}_{m,t}$ refers to the high frequency monthly ones. For the low frequency variables the observed vector in the observed equation can be viewed as the quarterly observations such as

$$\mathbf{y}_{Q,t} = \frac{1}{9}(\mathbf{y}_{q,t} + 2\mathbf{y}_{q,t-1} + 3\mathbf{y}_{q,t-2} + 2\mathbf{y}_{q,t-3} + \mathbf{y}_{q,t-4}) \tag{3.23}$$

where t in this case is the point in time for each quarter, i.e. {March, June, Sep, Dec}. This is also known as the triangular weighting scheme in the MF-VAR literature. The scheme is often applied to data consisting of growth rates and it's also the method used in Mariano and Murasawa (2003) for their MF-VAR model and in Huber et al. (2020). Another common method is the intra-quarterly averaging scheme [5] (see Schorfheide and Song, 2015), often used for data in log form. Although Ankargren and Yang (2019) maintain that the two approaches produce quite similar results. Similarly, the state vector can in this instance be seen as the monthly observations of the low frequency variables that are of actual interest to us. But they're latent and can't be observed directly. On the other hand, the high frequency variables are seen for each point in time.

Not much needs to be changed conceptually or notation wise from the BART model with the univariate outcome variable in section 3.1. The same general principle applies in that a Metropolis Hastings algorithm is employed inside a Gibbs sampler to draw all the tree structures. The difference lies mainly in the prior and hyperparameter choices and that there are now a total of K outcome variables, where each $\mathbf{y}_k$ represents a VAR equation. By closely following Huber et al. (2020), this results in the State space model

$$\begin{aligned} \mathbf{y}_{Q,t} &= \frac{1}{9}(\mathbf{y}_{q,t} + 2\mathbf{y}_{q,t-1} + 3\mathbf{y}_{q,t-2} + 2\mathbf{y}_{q,t-3} + \mathbf{y}_{q,t-4}) \\ \mathbf{y}_t &= F(\mathbf{X}_t) + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \boldsymbol{\Sigma}) \end{aligned} \tag{3.24}$$

Applying the general representation as in (3.20), The observed equation is the triangular weighting scheme as given by (3.23), while the second equation is the state equation. As seen, both the high and low frequency variables are baked into the state equation. This increases the burden of the algorithm, but there exists alternatives to formulating the state equation in the MF-BVAR literature so that the high frequency variables are separated from the low frequency ones (see for instance Schorfheide and Song (2015) for the alternative specification). Furthermore, $F(\mathbf{X_t})$ is some nonlinear function and $\mathbf{X}_t = [\ \mathbf{y}'_{t-1}, \mathbf{y}'_{t-2}, ..., \mathbf{y}'_{t-p}]\ '$. Each equation also has the exact same set of covariates included. Forward filtering backward sampling (FFBS) was used in Huber et al. (2020) to simulate the latent states conditionally on the BART parameters, but it can't be applied directly since the algorithm requires a linear model. See Appendix B.3 for a short explanation of FFBS and how it can be implemented in a nonlinear setting.

There are now a total of k = 1, 2,..K different equations, each equation approximates $\mathbf{y}_t$ by a sum of tree model. Each tree $\mathcal{T}_{k,s}$ and its corresponding set of terminal nodes $\mathcal{M}_{k,s}$ are still considered as independent. But the equations are dependent of each other because of the covariance matrix $\boldsymbol{\Sigma}$. The principle for drawing the trees, terminal node

[5] The intra-quarterly averaging scheme would be given as $\mathbf{y}_{Q,t} = \frac{1}{3}(\mathbf{y}_{q,t} + \mathbf{y}_{q,t-1} + \mathbf{y}_{q,t-2})$

parameters, and variances σ_k^2 maintain as in section 3.1.3 for each of the K different equations. The covariance matrix $\mathbf{\Sigma}$ can be decomposed as

$$\mathbf{\Sigma} = \mathbf{CHC}' \tag{3.25}$$

where the diagonal elements of $\mathbf{H}$ are the equation specific variances σ_k^2 and $\mathbf{C}$ is a lower triangular matrix, with a diagonal of ones. Each row of $\mathbf{C}$, can be formulated as a vector $\mathbf{c_k}$ with dimension (k-1) by only including the lower, non-diagonal elements of the k'th row from the matrix $\mathbf{C}$. Each element in $\mathbf{c_k}$ is penalized by a horseshoe prior such as

$$\begin{aligned} Pr(c_{k,j}|\tau_{k,j}, \psi) &\sim \mathcal{N}(0, \tau_{k,j}\psi) \\ \tau_{k,j} &\sim Cauchy(0,1), \quad \tau \geq 0 \\ \psi &\sim Cauchy(0,1), \quad \psi \geq 0 \end{aligned} \tag{3.26}$$

Let $f_k(\boldsymbol{X}_t)$ be some nonlinear function for the k'th equation, then in order to approximate each $\mathbf{y}_{k,t}$, the state equation for each VAR equation can be formulated such as

$$\mathbf{y}_{k,t} = \sum_{s=1}^{S} g_{k,s}(\mathbf{X}_t) + \mathbf{c}'_k \boldsymbol{\mathcal{Z}}_{k,t} + \zeta_{k,t}, \qquad \zeta_{k,t} \sim \mathcal{N}(0, \sigma_k^2) \tag{3.27}$$

by following Huber et al. (2020) who used an implementation similarly to Carriero et al. (2019) to enable estimation of each VAR-equation sequentially, where $\boldsymbol{\mathcal{Z}}_k$ is a vector containing structural shocks from the previous equations, i.e. $\boldsymbol{\mathcal{Z}}_{k,t} = [\zeta_{1,t}, ..., \zeta_{k-1,t}]'$ and $\mathbf{c}_k$ is defined as before. If k=1 then naturally $\mathbf{c}'_k \boldsymbol{\mathcal{Z}}_{k,t}$ will be equal to zero and (3.27) is reduced similarly to the univariate BART response (3.4). Specifying the model in this manner allows for sequential estimation of each VAR-equation where the posterior of each $\mathbf{c}_k$ also is normal and where the mean and variance include the shocks $\boldsymbol{\mathcal{Z}}_k$ (see Huber et al., 2020 for more details).

Finally, by letting $\mathbf{\Xi}$ denote all the parameters used in BART as the trees, terminal nodes, error covariance matrix as well as all the latent states in each equation the one step ahead posterior predictive distribution can be given as

$$Pr(\boldsymbol{y}_{T+1}|\boldsymbol{X}_{T+1}) = \int Pr(\boldsymbol{y}_{T+1}|\mathbf{\Xi}, Y) Pr(\mathbf{\Xi}|\boldsymbol{Y}) d\mathbf{\Xi} \tag{3.28}$$

where $Pr(\mathbf{\Xi}|\boldsymbol{Y})$ is simply the posterior distribution given by the MF-BAVART model and

$$Pr(\boldsymbol{y}_{T+1}|\mathbf{\Xi}, \boldsymbol{Y}) \sim \mathcal{N}(F(\boldsymbol{X}_{T+1}), \mathbf{\Sigma}) \tag{3.29}$$

Obtaining a forecast value h steps ahead can be done through simulation with an iterative scheme, each time by updating the last observation with the $T + h - 1$ forecast value.

A summary of the MF-BAVART algorithm can be seen below.

The MF-BAVART algorithm

Input:
Data: $\boldsymbol{X}$ and $\boldsymbol{Y}$
Latent states and Parameters: $\boldsymbol{y}_t, \mathcal{T}, \mathcal{M}, \Sigma$
Hyperparameters: $S, \alpha, \beta, \mu_{k\mu}, \sigma^2_{k\mu}, \lambda_k, \nu_k, \pi_1, \pi_2, \pi_3, \pi_4,$
Number of equations: K
Number of Iterations: N + Burn-in

i. Construct weight-matrix used for quarterly aggregates (triangular or intra-quarterly average).
ii. Impute missing low frequency values with appropriate method to initiate a first draw of the matrix in (B.8)
iii. Initialize BART parameters by

For k = 1,..., K
Initialize starting values for all S root nodes $\boldsymbol{\mu}^{(0)}_{k,1}, \boldsymbol{\mu}^{(0)}_{k,2}, ..., \boldsymbol{\mu}^{(0)}_{kS}$.
For s = 1,2,...,S
Draw the BART parameters for each equation as in (3.1.3) using a single iteration.
Store equation specific tree structures, node parameters and variances.
iv. Initiate horseshoe prior

For i = 1,..., N + Burn-in
Compute The Moore-Penrose inverse of $\boldsymbol{X}$
For k = 1,..., K
For s = 1, 2,..., S
Calculate $R^{(i)}_{k,s} = \boldsymbol{y_k} - \sum^{\complement}_{k,s} g_k(\mathbf{X}; \mathcal{T}^{\complement}_{k,s}, \mathcal{M}^{\complement}_{k,s}) - \boldsymbol{Z}_k \boldsymbol{c}_k$
Draw Tree structure proposal ${\mathcal{T}^*}_{k,s} \sim Pr({\mathcal{T}^*}_{k,s} | \mathcal{T}^{(i-1)}_{k,s})$
Calculate the ratio

$$r = \frac{q(\mathcal{T}^*_{k,s}|\mathcal{T}^{(i-1)}_{k,s})}{q(\mathcal{T}^{(i-1)}_{k,s}|\mathcal{T}^*_{k,s})} \frac{Pr(R_{k,s}|X,\mathcal{T}^*_{k,s},\mathcal{Z}_k,\mathcal{M}_{k,s})}{Pr(R_{k,s}|X,\mathcal{T}^{(i-1)}_{k,s},\mathcal{Z}_k,\mathcal{M}_{k,s})} \frac{Pr(\mathcal{T}^*_{k,s})}{Pr(\mathcal{T}^{(i-1)}_{k,s})}$$

Set $\mathcal{T}^{(i)}_{k,s} = \mathcal{T}^*_{k,s}$ with probability min(1,r),
Otherwise use $\mathcal{T}^{(i)}_{s} = \mathcal{T}^{(i-1)}_{s}$
Draw $Pr(\mu_{k,bs}|\mathcal{T}^{(i)}_{k,s}, R^{(i)}_{k,s}, \sigma_k)$ as in (3.15)
Draw $Pr(\sigma_k|\mathcal{T}^{i}_{k,1}, \mathcal{M}^{i}_{k,1}, ..., \mathcal{T}^{i}_{k,S}, \mathcal{M}^{i}_{k,S}, y_k)$ as in (3.16)

Compute $Proj(\boldsymbol{A'F})$[a]
Update horseshoe prior
Draw the latent states $\boldsymbol{y}_t$ conditional on all BART parameters.
Draw h steps ahead posterior predictive distribution:
$Pr(\boldsymbol{y}_{T+1}|\boldsymbol{X}_{T+1})$ as given by (3.28).
Aggregate all low frequency variables.

Output: N posterior draws of $\boldsymbol{y_1}, \boldsymbol{y_2}, ..., \boldsymbol{y_T}$

[a] See Appendix B.3

3.4.1 Priors

The MF-VAR models in Schorfheide and Song (2015) were constructed using a Minnesota type prior (with a normal inverse Wishart distribution) while Huber et al. (2020) used a horseshoe prior for their implementation of MF-BAVART. The Minnesota prior as originally proposed by Litterman (1986), which is a widely used prior in the BVAR literature, assumes though that the error covariance matrix $\mathbf{\Sigma}$ is known. An Inverse Wishart prior can be used when the covariance matrix is unknown (Dieppe et al., 2016). The prior serves as a shrinkage tool on parameters, which is often needed when the parameter space is large. One downside though is that all parameters are shrunk equally towards zero using Minnesota style priors. Another prior which has been used extensively in Swedish macroeconomic forecasting is the steady-state prior proposed by Villani (2009) which allows the user to incorporate prior knowledge into the unconditional mean of the system. Louzis (2019) presented a hierarchical steady-state prior, which allows the prior to have fatter tails. This is meaningful since economists usually claim that several indicators are composed of certain values in the long run, like for instance the long run inflation rate in Sweden which has a target of 2% by the Riksbank. Although Ankargren (2019) mention that these priors, not including the horsehoe prior, are often used for relatively small BVAR models. The advantage of the horseshoe prior is that it only shrinks the less important parameter values towards zero, while those with large coefficients remain (Chan, 2021). Carvalho et al. (2010) argue that one of the attractive feautures of the horseshoe prior, apart from relieving the user to choose among different hyperparameters, is that it can perform well when the data is noisy and could be a useful default option. Although a further discussion of these priors are out of the scope of this paper, two of the comparison models in section 5.2 consist of the two steady-state priors while the main comparison model utilizes the Minnesota prior.

3.4.2 Hyperparameters

The choice of hyperparameters are mainly influenced by previous findings. Chipman et al. (2010) found that their default values produced solid results across a number of different settings. It also relived the user from having to find suitable values themselves, for instance by expert opinion or cross-validation. Even if they argue that cross-validation often gave the best outcomes.

The choice of the number of trees S was 200 in Chipman et al. (2010) and 250 in Huber and Rossini (2020) and in Huber et al. (2020). A value of 250 was also chosen here after running the algorithm using the whole training data to evaluate the fit of the model (see section 5.1). It should be noted that a very large number of trees makes the computational burden much greater, and the marginal precision gains smaller.

The default for α and β in $\frac{\alpha}{(1+d)^{\beta}}$ was 0.95 and 2 respectively. This setting promotes smaller trees, most likely not exceeding a depth of 3 for most trees. Chipman et al. (2010) argue that using smaller tree structures can also serve as a way to indicate which covariates are of more importance for predicting the outcome variable, since these will be used more extensively for splitting the tree-nodes in each iteration. Although it was not utilized here, counting how many times a covariate appears in a splitting decision could in that way indicate its importance compared to the other covariates.

π_1 and π_2, the growth and pruning parameters respectively both have default values of 0.25. The 'change' probability π_3 is usually given a value of 0.40 while the 'swap' probability π_4 is often chosen to be 0.10.

The prior for the diagonal elements in $\mathbf{H}$, i.e. the variance parameters σ_s^2 from each equation are given as inverse-χ^2

$$\sigma_k^2 \sim \nu_k \lambda_k / \chi^2_{\nu_k} \tag{3.30}$$

Where the hyperparameters ν_k and λ_k can be chosen by cross validation or by using the approach in Chipman et al. (2010) where the degrees of freedom ν is set in favor between 3 and 10 and affects the shape of the distribution. λ can be selected sp that an estimated value of σ_k given by $\hat{\sigma_k}$ is such that $Pr(\sigma_k < \hat{\sigma_k}) = \upsilon$ [6] where υ is a quantile value, usually chosen to be between 0.75 and 0.99. In the replication files from Huber et al. (2020), the values for ν and υ were set to 5 and 0.99 respectively for the real data, while they used the values $T_q/2, 0.75$ for their simulation study. Chipman et al. (2010) argue against using values of ν less than 3 since that will make the model more prone to overfit. Their default recommendation is to use $(\nu = 3, \upsilon = 0.90)$, using higher values of υ, all things equal, will form a tighter prior on σ_k^2. An alternative would also be to use different values for (ν, υ) depending on which outcome variable estimation is performed on. This would likely produce more accurate results since all variables are different, but choosing among different values would quickly become cumbersome in a VAR-setting, especially as the number of variables increase.

The choice of hyperparameters for μ_μ and σ_μ is also done through standard settings as given by Chipman et al. (2010). Each node in a tree is independent and its prior is given as

$$Pr(\mu_{k,bs} | \mathcal{T}_{k,s}) \sim \mathcal{N}(\mu_{k,\mu}, \sigma^2_{k,\mu}) \tag{3.31}$$

Since each equation approximates $\mathbf{y}_k$ with a sum of tree model, the expected value of $\mathbf{y}_k$ given the covariates is also normal denoted as

$$\mathcal{N}(S\mu_{k,\mu}, S\sigma^2_{k,\mu}) \tag{3.32}$$

By re-scaling each $\mathbf{y}_k$, the center can be set to 0. The prior in (3.31) can then be reformulated as

$$Pr(\mu_{k,bs} | \mathcal{T}_{k,s}) \sim \mathcal{N}(0, \sigma^2_{k,\mu}) \tag{3.33}$$

Following Huber et al. the hyperparameter $\sigma_{k,\mu}$ is chosen as

$$\frac{0.5(max(\mathbf{y}_k) - min(\mathbf{y}_k))}{\mathcal{Q}\sqrt{S}} \tag{3.34}$$

where the choice of $\mathcal{Q}$ is set to a default of 2. A higher value of $\mathcal{Q}$ gives a more tight prior on each terminal node. Increasing the number of trees will have a similar effect, thus giving each individual node less weight in the sum.

3.4.3 Residual diagnostics

The univariate outcome BART model assumes normality in the error terms and this assumption is extended to the multivariate VAR case too as seen in (3.24). The standard

[6] $\hat{\sigma_k}$ is often given as a naive estimate as the standard deviation of the outcome variable $\mathbf{y_{\bullet k}}$ or as the standard deviation retained from OLS.

residual diagnostics tests for time series data using ARIMA models can be used, where the residual of observation t is denoted as the difference between the actual value and the point estimate as $e_{t,k} = y_{t,k} - \hat{y}_{t,k}$. Two such tests are the commonly used Jarque-Bera- and the Ljung-Box test.

The Jarque-Bera tests whether the normality assumption of the data is valid or not. It uses the test-statistic $\frac{T}{6}Sk^2 + \frac{1}{4}(\mathcal{K}-3)^2$, where $Sk = \frac{\frac{1}{T}\sum_{t=1}^{T}(y-\bar{y})^3}{\frac{1}{T}(\sum_{t=1}^{T}(y-\bar{y})^2)^{3/2}}$ denotes the skewness and $\mathcal{K} = \frac{\frac{1}{T}\sum_{t=1}^{T}(y-\bar{y})^4}{\frac{1}{T}(\sum_{t=1}^{T}(y-\bar{y})^2)^2}$ denotes the kurtosis. The test statistic asymptotically follows a χ_2^2-distribution and consist of the hypotheses

H0: The error terms are normally distributed.
HA: The error terms are not normally distributed.

(MathWorks, 2022). As seen from the test statistic, having a few huge outliers can affect the test result though and make it more probable to reject H0.

The Ljung-Box test on the other hand, also known as a Portmanteau test, tests for serial correlation in the data. It's composed of the test statistic $T(T+2)\sum_{j=1}^{p}\frac{\hat{\rho}_j^2}{T-j}$, where $\hat{\rho}_j$ denotes the sample autocorrelation for the j'th lag, which is also asymptotically χ_p^2-distributed and uses the hypotheses

H0: There is no autocorrelation in the error term.
HA: There's autocorrelation in the error term.

In this case, it's preferable to not reject either null hypothesis (Hyndman and Athanasopoulos, 2018).

Additionally to these tests another common test is the ARCH-LM test, proposed by Engle (1982), which is specifically used to test for autoregressive conditional heteroskedasticity. It's performed through an OLS on the squared residuals of lag p where

H0: the squared error terms ε_t^2 follow a white noise process.
HA: ε_t^2 are serially correlated.

3.4.4 Evaluation of the model and forecasting accuracy

The root mean squared error (RMSE) is generally defined as

$$\sqrt{\frac{1}{n}\sum_{t}^{n}(y_t - \hat{y}_t)^2} \tag{3.35}$$

Where n is the total number of observations used for comparison. It's applied in two different ways here to evaluate accuracy. The first approach is to start at a certain date and perform out-of sample forecasts with forecast horizon h using the posterior predictive distribution and then calculate the mean squared error for all those observations compared to their actual values. After this is done, the training data is incremented by

a few more observations and the procedure is repeated until it's no longer possible to compare the full set of forecast values to the actual values. This can give an indication of how much the predictive performance is affected by adding more training data.

The other approach, which is commonly used in macroeconomic forecasting and will henceforth be denoted as the root mean squared forecast error (RMSFE) is to start at a specific date in the sample and use all the observations up to this date as training data and then perform an out of sample forecast with a horizon size of h. The results from each forecast observation is then stored. After this, one or a few more observations are included in the training data and an out of sample forecast is performed again and the values from each forecast horizon is stored once more. This procedure is repeated, each time including more training data until the end of the sample has been reached. The RMSFE is then calculated for each forecast horizon, for instance the RMSFE is calculated for all out of sample forecasts belonging to horizon 1, another RMSFE is calculated for all forecasts belonging to horizon 2,..., up to horizon h. What's left is a vector of RMSFE values of length h. Following this procedure for multiple models allows for comparisons of accuracy between them. This also implies that the number of values used to calculate the RMSFE values for the first horizon is larger than for the last horizon.

A common test used for comparing forecast accuracy between two models is the Dieabold-Mariano test where the null hypothesis states that there's no difference in accuracy between two models and the alternative can be two sided, less or greater for one of the models. The test uses the forecasting residuals from each model. The test is robust and can be used for instance even in cases with non-normality or simultaneous correlation between forecast errors (see Diebold and Mariano (1995) for more details).

3.4.5 Further requirements

A requirement for implementing the algorithm is that the data start off in the same period. Using the approach by Schorfheide and Song (2015) and Ankargren and Yang (2019), it's possible though to have the series ending at different time points, which is often the case in reality since it's probable to observe additional monthly observations after the last published quarterly data. But for purposes of this paper, all observations end at the same time point, December 2021. Any discussion of issues with possible delays of publication from the various time series are not treated here. Furthermore, since the triangular weighting scheme in the first equation of (3.24) assumes five lags for each covariate, the choice of lags was straightforward. The initialization of the algorithm is done through the first five observations in the data. These observations are then disregarded.

4 Ethical considerations

The code used here for the MF-BAVART algorithm was made publicly available in R by the authors in Huber et al. (2020) and extracted from Michael Pfarrofer's github page, one of the coauthors. And the Bayesian comparison models were implemented using the 'mfbvar' package in R by Ankargren and Yang (2019). But a considerable amount still had to be coded from scratch. Code and replication files can be made available by

request.

5 Results

This section is divided into two parts. The first part consist of a more extensive comparison between two different small models, using only the three main indicators of most interest GDP, Unemployment and CPIF (inflation). Some model diagnostics will also be performed on the simple model, these mainly involve residual diagnostics in order to check if the model fulfills the requirements, with normally distributed errors and no serial correlation. The second part of this section extends the parameter space by including additional covariates. The focus on this part is set towards evaluating Nowcasting accuracy in a mixed frequency setting and to assess whether the MF-BAVART model can compete with other contemporary mixed frequency BVAR models in macroeconomic forecasting. The results from a simple AR(1) model is also included for an overall comparison between the Bayesian mixed frequency models and a univariate simple one.

The abbreviation 'MF' standing for mixed frequency will henceforth be dropped in the text since all the models will be of the mixed frequency kind except the AR(1) model which will be denoted the same. The MF-BAVART will be denoted as BART and the MF-BVAR, simply as BVAR.

5.1 A small model comparison

The basic approach used in Schorfheide and Song (2015) was to use a Minnesota prior for their BVAR. With the Minnesota prior traditionally being a well exploited prior in macroeconomic forecasting, the performance of the simple BART model with three variables will therefore be compared against that model.

The BART model is first estimated using the whole dataset reaching from January 1997 - December 2021. Since five lags are used for the triangular weighting scheme, the first five observations (i.e. from January-May) are lost. The model is evaluated using the tests described in section 3.4.3. The root mean squared error (RMSE) was calculated for the whole dataset. This was first done several times, using different number of MCMC-draws in order to obtain an approximation of how many iterations may be suited for the small dataset. Some variations were also used on the different hyperparameters in order to discover how sensitive the model performance is to changes in the hyperparameters, giving an indication of which setting would be more appropriate to use. The number of MCMC draws did not have that much of an impact on the RMSE-values, which is promising since the algorithm is quite costly computational wise. The number of draws will therefore be kept at 7000 for the three-variable BART model, of which half is disregarded as burn-ins. Although the results for the RMSE won't be much affected by the sparse iteration number, the variance is likely to be larger as a result.

The residual diagnostics test for the final chosen model gives a mixed result and can be seen in Table 2. The Jarque-Bera test on the residuals for two out of the three VAR equations rejects the null hypothesis of normality. A possible reason for that may be the few large outliers that are present in each variable. QQ-plots on the residuals of the three variables can be seen in Figure 10 in Appendix A.2 which seem to indicate

normality for all three variables except for a few outliers at the end, most prominent in GDP.

Table 2: Residual Diagnostics

	Unemployment		CPIF		GDP	
	χ^2	p-value	χ^2	p-Value	χ^2	p-value
Jarque Bera	9.6039	0.0082	2.1489	0.3415	123.4800	0.0000
Ljung-Box (lag 1)	0.0002	0.9885	3.0101	0.0828	3.6633	0.0556
Ljung-Box (lag 4)	0.5285	0.9707	3.2365	0.5191	12.1140	0.0165
Ljung-Box (lag 12)	12.7720	0.3858	61.4080	0.0000	15.9270	0.1946
ARCH LM (lag 1)	11.6790	0.0006	1.0904	0.2964	2.0824	0.1490
ARCH LM (lag 4)	14.7020	0.0054	3.8486	0.4269	18.1560	0.0012
ARCH LM (lag 12)	19.4640	0.0779	15.0000	0.2414	21.8590	0.0391

A more serious issue is the Ljung-Box tests, testing for autocorrelation in the residuals displayed in the same table. The test is not rejected for the first lag for any of the variables, suggesting that there's no yearly autocorrelation. The test is rejected for GDP on the forth lag and for the 12'th lag of CPIF. Figure 2 displays the autocorrelation function plotted against 20 lags. The graphs illustrate that there seem to be autocorrelation in the 12th lag in the residuals of CPIF and in the 4th lag for GDP, confirming the findings in Table 2 by using the Ljung-Box test. An AR(5) model was also estimated for each variable, since the BART model implemented here uses five lags because of the triangular weighting scheme. The results on the residuals were similar, but the serial correlation on the 4th lag of GDP was not present, but it indicated that there is serial correlation in the 12th lag for CPIF. As previously mentioned in section 2, the default setting in the stl-function in r was employed to remove seasonal patterns in the data. Due to the above results of the residual diagnostic tests for CPIF, some alternative settings were applied too with the stl-function, which uses different time-spans to estimate the seasonal pattern. The AR(5) was then re-estimated, resulting in a similar outcome. Therefore, another method which is the commonly used X12-ARIMA[7], for removing seasonal patterns, was also applied to the variable. The default method in r was implemented, as well as some alternative ones to remove the seasonal patterns, with all of them still leading to similar result in the Ljung-Box test. Hyndman and Athanasopoulos (2018) mention that autocorrelated error terms may imply that more information (as additional covariates for instance) needs to be added to a model.

The results for the ARCH-LM differs somewhat from the Ljung-Box test. The test is rejected for the variable GDP on the second and 12'th lag and for the Unemployment rate for both lag one and four, and reversely not rejected for any of the lags when it comes to CPIF. Although this shouldn't affect the point estimates in said BART equations, this can give an indication that the standard errors produced may be on the larger side and that the estimates are not efficient. A possibility to amend this could be to use an alternative GARCH or Stochastic Volatility (SV) model.

[7]X12-ARIMA, was developed by the US Census Bureau and is commonly used by practitioners to remove seasonality in time series data. With its implementation in r, using the x12-package, the series can be decomposed into different components (trend, season, cycle, trading day and irregular), the function can handle additive and multiplicative models. See Kowarik et al. (2014) for a description of the method used in their r package.

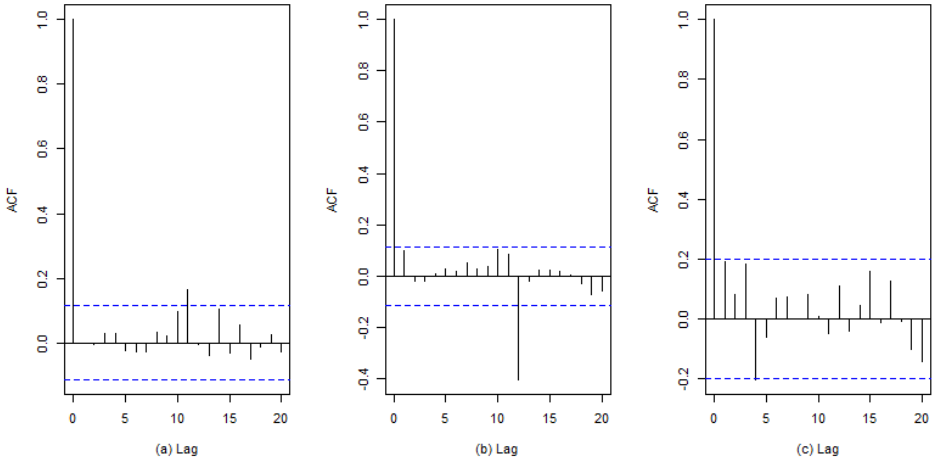
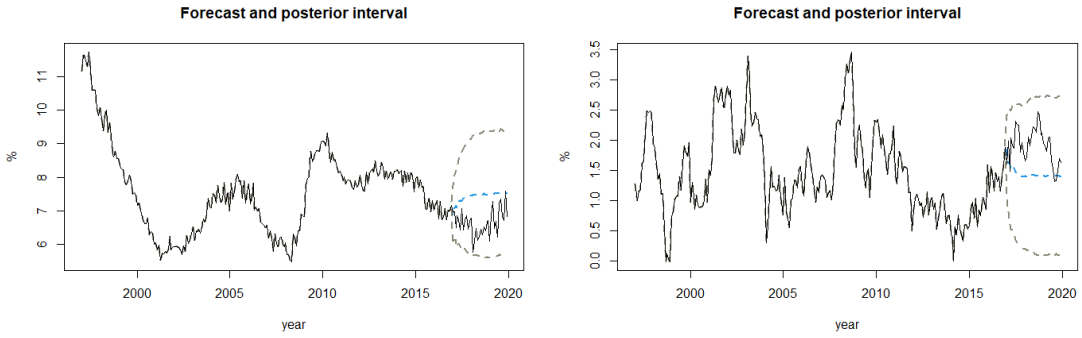


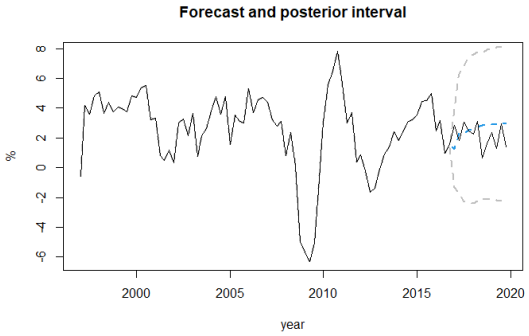
Figure 2: ACF-plots obtained from the residuals of BART using all observations as training data with VAR-equations (a) Unemployment, (b) CPIF, (c) GDP

The mean of each equation's residuals along with their 95% credible intervals can be found in Appendix A.2 in Table 9. As presented, the mean is approximately zero for the residuals for all three variables. Residual means differing from zero can be a sign of bias (Hyndman and Athanasopoulos, 2018). The setting is not ideal based on these tests, but it's still possible to continue since the focus is on forecasting.

So far the whole data has been used as training to learn the model. But the interest is to obtain good predictions on future observations. The data has therefore been split into a training set and a forecast set with a horizon of 36 months. The first training set reaches from January 1997 to December 2010 and the first forecast starts at January 2011 and ends at December 2013. The median of the posterior predictive distribution for each month is extracted for each forecast month. The RMSE is then calculated by comparing the predicted values to the observed ones and the quarterly variable is aggregated to make comparisons possible to the real observations. The training data is then increased with an additional quarter, i.e. the start of the forecast horizon is then April 2011, and the end is at March 2014 and the RMSE is recalculated. This procedure of adding a quarter of observations to the training data is repeated 33 times for the results shown in Figure 3 - 5. These values are thus obtained from the full three year forecasts. The final three year forecast starts at January 2019 and thus ends in December 2021. Because of the extreme observations due to the pandemic, the predictions that include the pandemic months, reaching from March 2020 to the end of 2021, are blind towards the extreme shifts that occurred in that time span. A visual representation of the final 36 months forecast can be seen in Figure 11 for the interested reader but it's not meaningful to compare these values to the actual ones since the model hasn't had time to adjust to the extreme values. A closer visual look is set towards the forecast values between January 2017 to December 2019.



(a) Unemployment **(b)** CPIF



(c) GDP

Figure 3: 36 month forecast values of Unemployment rate, CPIF and GDP using BART with 95 % posterior bands (in lighter dashed lines) plotted along with the actual values. Monthly forecast values for GDP have been aggregated to quarterly for comparison.

Starting with the monthly variables (a and b) in Figure 3, it's clear that the long term predictions using BART misses the quite wide fluctuations in the actual values. Even though the mean estimates seems to get closer to the real values by the end of the forecasts, for both the Unemployment rate and CPIF. The predictions on GDP seen in 3c are closer to the observed values. The actual values for each variable are within their 95% posterior CI bands respectively. The graphs also illustrate that the long term forecasts generally follow quite smooth, level lines, which is consistent with how the posterior predictive distribution given by 3.28 in BART uses the last observation to predict a new one. The wider posterior intervals reflect the increasing uncertainty with each new prediction.

The RMSE for each 3 year forecast can be seen in Figure 4. Note that the RMSE for GDP is only calculated by comparing the aggregated observations that occur in an actual quarter. The monthly observations that fall outside a quarter are disregarded for this comparison and the result could be slightly different had the algorithm been initiated with a different starting month in a quarter (the first month in a quarter was always used here). The first long term forecast is set in January 2011 and the last one starts in January 2019. Starting from the top, the RMSE values from the predictions for the Unemployment rate start off at around 1 and gain a small accuracy at the start by including more data, but it later moves up again, and the pandemic seem to enforce this pattern indicated by the vertical line in the graph that includes all 3 year forecasts

after March 2017. The value for CPIF, also starting off with an RMSE value of 1, has a similar pattern but the line is more stable, even though the pandemic effect may have a very small impact too. The largest impact can be seen in GDP, which starts off with a relatively high RMSE of approximately 3.5 for the first long term forecast, but quickly drops to low levels circling around 1.2 as the model receives more training data. But the RMSE increases dramatically around the vertical line in the graph, mirroring the pandemic affect on GDP.

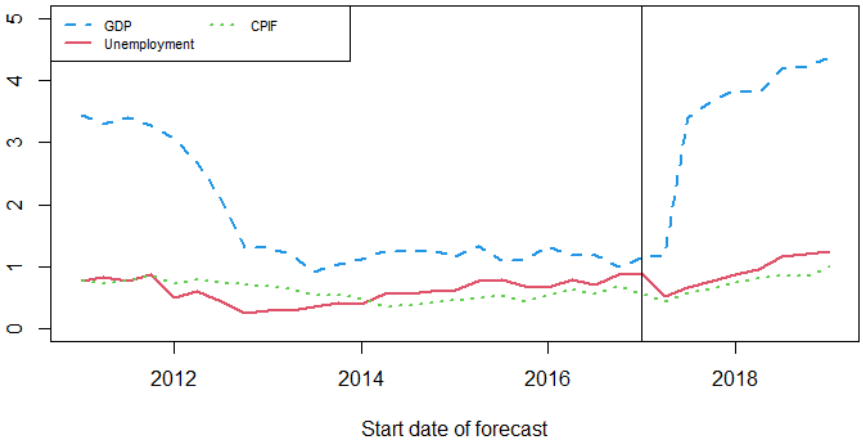


Figure 4: RMSE values obtained from 3 year ahead forecasts of each VAR-equation (Unemployment, CPIF and GDP) with the first set of values starting at January 2011. Observations to the right of the vertical line include predictions for observations occurring in the second quarter of 2020

Furthermore, the BART model had a tendency to either underestimate, or overestimate most of the observations in each set of forecasts. Although the RMSE values present the error of the model, they don't reveal in which direction these errors tend to take. To explore the path of potential bias, the mean difference between the forecast values and the actual values corresponding to each 36 month period was calculated. This means that a positive difference indicates an overestimation by the forecasts and a negative difference points towards underestimation. The results can be seen in Figure 5.

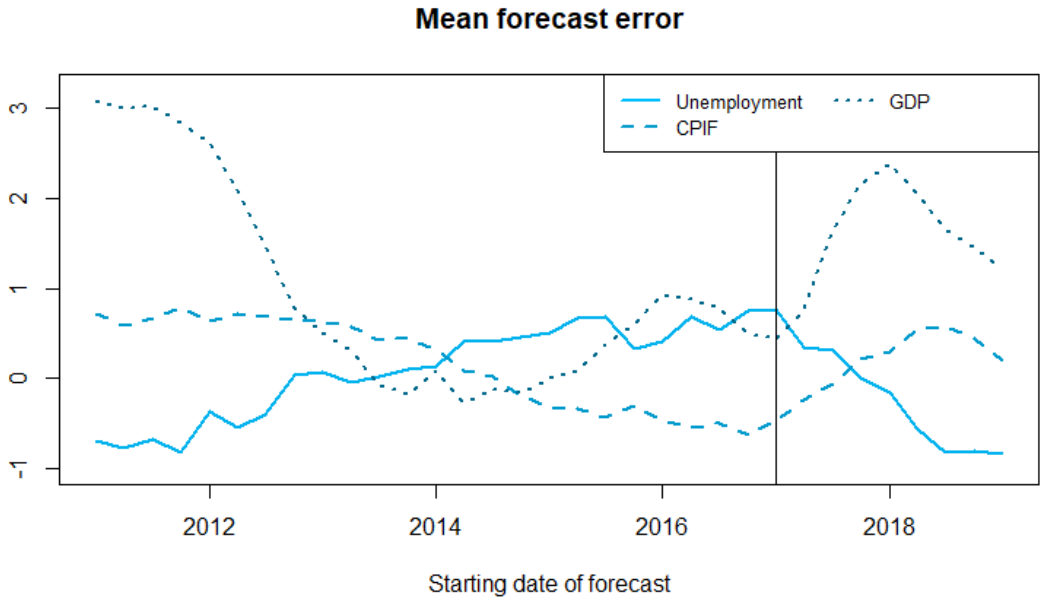


Figure 5: Mean error in each 3 year forecast for Unemployment, CPIF and GDP. Observations to the right of the vertical line include predictions for observations occurring in the second quarter of 2020.

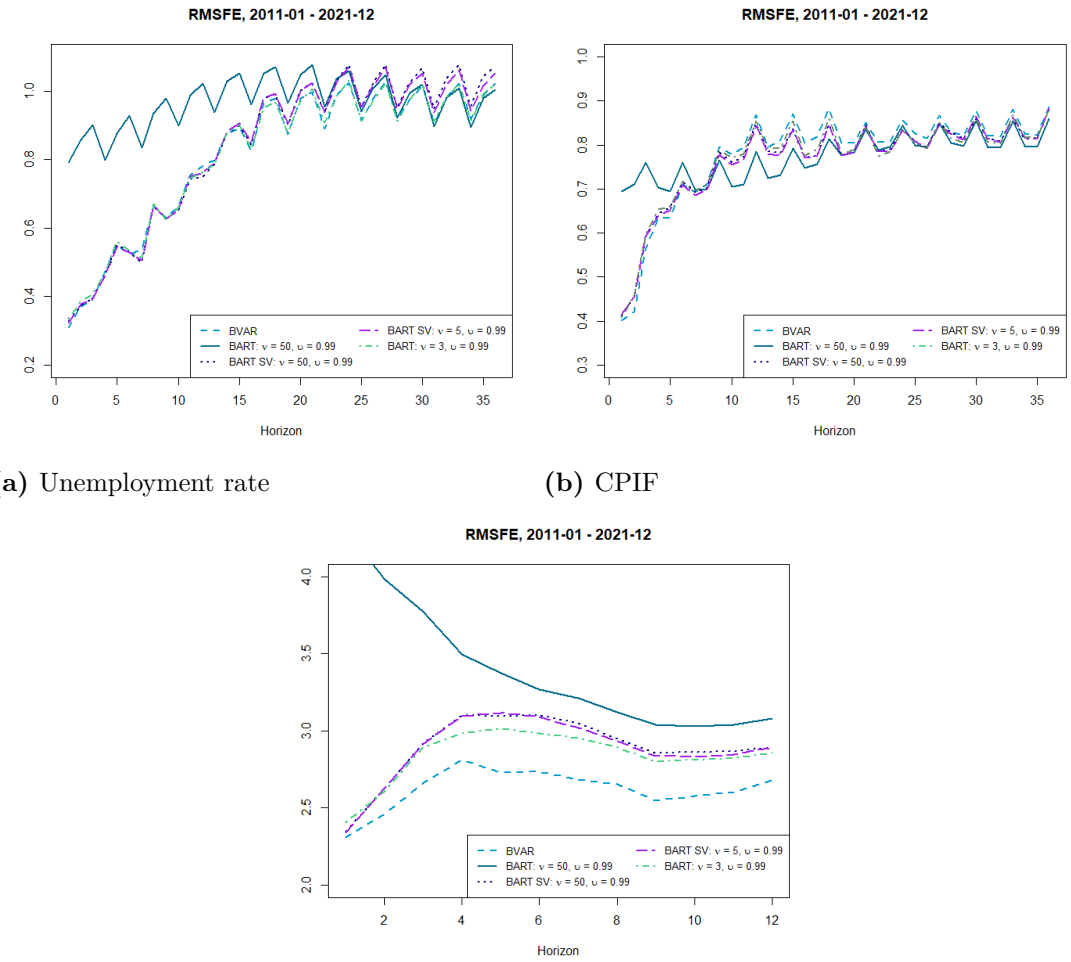
The graph shows that GDP clearly on average overestimates the true values when fewer observations are used in the training data. But the first couple of three year forecast periods are still rather close to the financial crisis in 2008, which may potentially have an affect too. The mean forecast error goes down dramatically for GDP, but the tendency is often positive, and it rises dramatically again when forecast values that include the pandemic period are also included. The paths for the Unemployment rate and CPIF are much more stable, where they seem to fluctuate around 1 and -1. Ideally, values as close to zero as possible would be preferred as it could give an indication of a small bias. The graph also indicates that the model is specified correctly from an economic perspective mirroring Okun's law, where GDP and the unemployment rate in theory move in opposite directions, which is also reflected in the graph.

Next is an illustration of the RMSFE values, as given in section 3.4.4 with a total of 33 full forecast horizons up to December 2019. The last added training data is the third quarter of 2021 and thus the three first forecast horizons consist of 44 values, horizon 4-6 consist of 43 values, horizon 7-9 consist of 42 values and so on since the training data is added on a quarterly basis. Figure 6 shows the outcome for each VAR equation using BART with a few different hyperparameter settings with two of them including Stochastic volatility versus a BVAR model with a Minnesota style prior with tightness parameter = 0.2 (see Ankargren and Yang (2019, p. 5)). It's clear that the RMSFE on the high frequency variables produce almost indistinguishable results for the two different models except for BART with $\nu = 50, \upsilon = 0.99$. The low frequency variable tells another story. It shows that the basic BVAR model outperforms BART for every single horizon. Again, it's clear that BART with $\nu = 50, \upsilon = 0.99$ performs very poorly and completely misses the mark for short horizons. The difference between all the models seem to decrease for longer horizons.

Another interesting pattern for GDP is that the RMSFE values start off at lower levels

for small horizons and reach a peak at around horizon 4 (which is after a year since it's quarterly data) and later steadily drops down to lower levels. The pattern may be due to some of the earlier horizons including more observations occurring in the pandemic period. The zigzag behavior on the unemployment rate and CPIF could possibly be explained by the fact that each forecast always starts off at the first month of a quarter and the training data is always increased by the quarter and not by the month.

One flaw of these forecasts and calculated errors is that the true potential of using mixed frequencies is not utilized. A more nuanced view of the forecast accuracy can be achieved by increasing the training data on a monthly basis for each forecast horizon, instead of increasing it on quarterly one. In doing so for the low frequency variables, a comparison can be made between using different starting values in a given quarter to illustrate if the accuracy improves.



(a) Unemployment rate

(b) CPIF

(c) GDP

Figure 6: RMSFE values for Unemployment (a), CPIF (b), GDP(c) using a horizon of 36 months. The lighter hyphened lines are values obtained from a BVAR model using a Minnesota style prior and the other ones are from BART with different hyperparameter settings.

5.2 Nowcasting with BART

This section is devoted to Nowcasting since it's often the main objective when using mixed frequency models. The number of variables are also increased from three to eleven and four of them are in low frequencies as seen from Table 1. The maximum forecast horizon is set to three months and the first Nowcast is initiated at January 2019 and the last is initiated at December 2021. The same benchmark comparison model, a BVAR with a Minnesota style prior, with the same hyperparameter setting as in section 5.1 is also used here since that model outperformed BART in the previous section. Additionally, a simple AR(1) model and two other BVAR models using steady state priors are also included as comparisons. The listed models are given in Table 3 (with a slight abuse of notation for BART since it's the elements in each $\mathbf{c}_k$ in (3.25) that are imposed by the horseshoe prior) .

Table 3: Models used for Nowcasting

Model	Description	Prior on error covariance matrix
BART	Bayesian Additive regression trees	Horseshoe
BVAR	Bayesian VAR with Minnesota prior	Inverse Wishart
SS	Bayesian VAR with Steady State prior	Inverse Wishart
SSNG	Bayesian VAR with Steady State hierarchical Prior	Inverse Wishart
AR(1)	Autoregressive of order 1	-

The Steady State prior, as mentioned in section 3.4.1, is commonly implemented by the Swedish Riksbank but it requires the user to set a prior for the unconditional mean (see Louzis, 2019 or Ankargren et al., 2020). This can be done by a number of measures, one approach is to use the 95% prior probability interval. But an approach with cross-validation was used here by computing the RMSFE and comparing the forecast values from January 2019-December 2021 to the actual ones with several different specifications before settling for the intervals given in Table 10 in Appendix A.2 which gave the lowest RMSFE for the variable GDP. It should be noted that the values produced by the grid-search were very similar across different settings.

A comparison before as well as during the pandemic likewise to Huber et al. (2020) is also made to discover if BART can compete with other contemporary BVAR models during a time of high uncertainty. The BART model used here consists of 250 trees with $\nu = 3$ and $\upsilon = 0.99$, also denoted as an 'aggressive' setting in Chipman et al. (2010). Theses settings are chosen based on the long term forecast results in section 5.1 which seemed to produce overall slightly better results compared to the other BART settings while still being kept simple. The number of MCMC iterations is increased to 20 000, of which 10 0000 are discarded as burn-ins. The same number of MCMC draws and burnins is used for the other Bayesian models.

Table 4 illustrate the RMSFE results for BART and the other competing models for the high frequency variables with a forecast horizon of three months. Overall, it's clear that the forecast accuracy appears to decrease as the horizon length increases which is to be expected. Although, this is not always the case as seen from the table, it's more pronounced during the pandemic period. Comparing BART with the other Bayesian models gives no clear indication of which is to be preferred, although the comparison models seemingly performs better for the interest rates and the KIX index. Overall, it's

a mixed outcome between the two models but they also produce very similar results. It's also clear that the AR(1) model performs quite well and sometimes even beats all the other models (see the results on the KIX index during the pandemic for instance).

Switching the focus to the low frequency variables, we can see that the Nowcast results in Table 5 differs a bit from the one made on the high frequency variables. Instead of displaying the horizon number, the number of the month in a quarter the Nowcast was initiated at is displayed. As a concrete example, Month 1 for the first quarter means that the Nowcast was initiated at January, with a three month horizon. The forecast value for March is extracted by using the triangular weighting scheme from (3.23) to compute the quarterly forecast value. Likewise, for Month 2, the Nowcast is initiated in February and the forecast for March is just two months ahead instead of three this time. Proceeding in this way, allows the user to utilize more of the monthly observations of the high frequency variables for the forecasts. Intuitively, this should give a good opportunity to improve the accuracy as more observations are available in a given quarter. As seen by the table for BART, the Pre-Pandemic forecasts does not seem to make much improvements, if any at all. In contrast to the results for BART, the other Bayesian models give more accurate results for GDP proceeding forward towards month 3. But likewise to BART, the accuracy does not always increase as the nowcast is made at a later month of a quarter during the pre-pandemic period.

The results for the Pandemic period clearly shows that the accuracy overall improves when the Nowcast is initiated at the third month of a quarter. It's a sign that the accuracy generally benefits, during a period characterized by higher uncertainty, by adding more observations to all the different Bayesian models. It's also clear that while the measure of GDP in the first month is more accurate using BART, the other Bayesian models perform much better as we move along in a quarter with SSNG being the best one. The result for the second variable, Capacity utilization is quite similar across all the different models including the AR(1) even if the RMSFE for BART is the lowest compared to all the other comparisons. The biggest surprise may be the result for the other two remaining variables; the expected inflation and the wage inflation where all the Bayesian models perform similarly but the AR(1) model outperforms them all. Table 11 and 12 shows the results for the Diebold Mariano test between BART and the benchmark BVAR model for each equation where the alternative hypothesis is that the BVAR model is less accurate than BART. The results overall reflects those given in Table 4 and 5, that the forecasts produced by BART generally don't produce more accurate results than the BVAR model even if there are a few exceptions.

It should be noted though that the BART model sometimes had issues with convergence and that the algorithm had to be restarted sometimes. [8]

[8]The BART model will fail to converge if the covariance matrix $\mathbf{\Sigma}$ is not positive definite. This will cause a failure in the Cholesky decomposition used in the Kalman filter since it requires a positive definite symmetric matrix. The issue may be due to combining certain variables with others but the findings here suggests that the BAVART model is particularly sensitive towards the use of too few trees while increasing the number of covariates in the model, especially those of low frequencies.

Table 4: Nowcasting RMSFE for high frequency variables

	Horizon 1					Horizon 2					Horizon 3				
	BART	BVAR	SS	SSNG	AR(1)	BART	BVAR	SS	SSNG	AR(1)	BART	BVAR	SS	SSNG	AR(1)
Pre-Pandemic															
Unemployment	0.5088	0.5307	0.5377	0.5316	0.6303	0.4890	0.5366	0.5400	0.5367	0.6278	0.4865	0.5616	0.5603	0.5566	0.5278
CPIF	0.1736	0.2066	0.2197	0.2134	0.1622	0.2711	0.3760	0.3898	0.3912	0.2759	0.3189	0.5283	0.5328	0.5557	0.3450
IPI	2.2802	2.3501	2.3633	2.3029	2.7850	2.2532	2.4275	2.4963	2.32609	2.4658	2.3231	2.4887	2,6202	2.3631	2.3522
Bills 3m	0.1043	0.0952	0.0949	0.0919	0.0616	0.1451	0.1417	0.1403	0.1309	0.1108	0.1561	0.1597	0.1609	0.1379	0.1342
Bonds 5y	0.2181	0.1175	0.1151	0.1148	0.1107	0.3401	0.1865	0.1852	0.1906	0.1924	0.4208	0.2596	0.2632	0.2808	0.2646
Bonds 10y	0.3419	0.1214	0.1301	0.1226	0.1334	0.2522	0.1910	0.1978	0.1972	0.2070	0.2255	0.2755	0.2794	0.2967	0.2737
KIX	1.6744	1.6337	1.6651	1.6248	1.5931	2.4599	2.1784	2.1766	2.1327	2.2286	3.0522	2.2792	2.1428	2.1818	2.4337
Pandemic															
Unemployment	0.4785	0.4886	0.4883	0.4894	0.5286	0.6490	0.6319	0.6349	0.6295	0.6597	0.7598	0.7515	0.7645	0.7447	0.7372
CPIF	0.5444	0.4753	0.4704	0.4713	0.5056	0.7759	0.7415	0.7122	0.7272	0.7770	0.9349	0.9553	0.8977	0.9273	0.9437
IPI	6.4658	6.2974	6.3811	6.3860	5.7228	8.8609	9.0904	9.3388	9.2908	8.3311	10.0246	10.5662	10.8910	10.94404	9.7870
Bills 3m	0.1307	0.1148	0.1133	0.1209	0,0681	0.1973	0.2051	0.2149	0.2234	0.0892	0.2179	0.2648	0.2938	0.2949	0.0973
Bonds 5y	0.1442	0.0978	0,0993	0.1025	0.0737	0.2319	0.1510	0,1558	0.1610	0.1147	0.2762	0.1789	0.1821	0.1957	0.1396
Bonds 10y	0.2564	0.1077	0.1101	0.1146	0,0991	0.2354	0.1778	0.1819	0.1936	0.1564	0.2426	0.2090	0.2137	0.2347	0.1775
KIX	1.4519	1.3954	1.4202	1.4044	1.1701	2.4288	2.4068	2.5485	2.4245	1.8151	3.0479	2.9609	3.1542	2.9884	2.2544

The Pre-pandemic period covers the months January 2019 - December 2019. The pandemic period covers the months January 2020 - December 2021.

Table 5: Nowcasting RMSFE for low frequency variables

	Month 1					Month 2				Month 3			
	BART	BVAR	SS	SSNG	AR(1)	BART	BVAR	SS	SSNG	BART	BVAR	SS	SSNG
Pre-Pandemic													
GDP	0.9645	0.8907	0.9304	0.9328	1.1938	0.9210	0.6093	0.7339	0.5937	1.0206	0.4745	0.6685	0.4931
Capacity utilization	1.0812	1.2096	1.1532	1.2692	1.0450	0.9895	0.7768	0.7345	0.8385	1.1420	0.8777	0.8649	0.9432
Expected inflation	0.1176	0,1226	0.1355	0.1519	0.1032	0.1121	0.1054	0.0913	0.1343	0.0921	0.0777	0.0583	0.0880
Wage inflation	0.1157	0.0581	0.0412	0.0474	0.1128	0.1073	0.1286	0.1038	0.1240	0.1312	0.1939	0.1493	0.1959
Pandemic													
GDP	4.5756	4.6937	4,8176	4.7961	4.9645	4.2165	3.3020	3.4988	3.2390	3.8232	2.7710	3.0207	2.6841
Capacity utilization	2.7845	2.8445	2,9745	3.0376	2.7218	2.3124	2.2534	2.5043	2.3483	2.0069	2.0450	2.2833	2.1307
Expected inflation	0.6830	0.7567	0.6960	0.7097	0.6417	0.6986	0.7322	0.6959	0.6877	0.6601	0.7015	0.6677	0.6685
Wage inflation	0.5680	0.5901	0.6426	0.6265	0.3372	0.5438	0.5503	0.6119	0.5676	0.5615	0.5538	0.6282	0.5642

The Pre-pandemic period covers the months January 2019 - December 2019. The pandemic period covers the months January 2020 - December 2021.

Month 1 corresponds to the Nowcast being initiated in the first month in a quarter, Month 2 the second and Month 3 the third

6 Discussion and conclusions

The need for forecasting models with a higher degree of accuracy has been amplified by the pandemic. Bai et al. (2013) found that the use of MF-VARs can be less accurate if the state space model is miss-specified compared to MIDAS regression. The advantage of BART is that it's a nonparamteric approach which lends itself well to a wide variety of applications without much need for tuning of hyperparameters by using the default settings provided by Chipman et al. (2010). The attractiveness of using a nonparametric model may also be more pronounced during times of higher uncertainty as even experts may face increasing difficulties in formulating their models. This paper was devoted to using a nonparametric approach of forecasting by combining BART with a MF-VAR model using Swedish data where some of the comparisons where made during a time of high uncertainty. Performance was evaluated for both low and high frequency variables. The result was divided into two parts, where the first part was devoted to evaluating the model as a whole and also towards long term forecasting while only using three variables. The second part focused on adding more variables to assess accuracy in a Nowcasting setting. In contrast to Huber et al. (2020) who used a BVAR model with a horseshoe prior, the benchmark comparison model was a traditional BVAR with a Minnesota style prior in both long and short term forecasts. Two other common Bayesian models were also used as comparisons against BART as well as an AR(1) model.

The results here suggests that using BART with only a few variables for the purpose of long term forecasting is not ideal and that there are other competing BVAR models that can produce more accurate results at a much greater speed. As a concrete comparison, the BART model used to produce the RMSFE values with 36 horizons in section 5.1 ran for over 111 minutes (141 minutes using stochastic volatility) on a computer with an intel core i7 processor, 16 GB RAM and 8 logical cores. This can be compared to the simple benchmark BVAR model which only needed roughly 8 minutes to produce its output, so while there has been promising results in previous studies and implementations of BART as in Huber and Rossini (2020), Huber et al. (2020) and Clark et al. (2021), the algorithm can be quite slow. If the number of observations is of small to moderate size, the speed of the algorithm may not be of such importance. But it may pose an issue if the sample size is large. Increasing the number of VAR equations, and thus also the covariates used, quickly puts a large burden on the speed of the algorithm. The total runtime for both type of models seem to come closer (roughly 12 hours for BART and over 5 hours for the BVAR model which was used to produce the results in section 5.2). This can be contrasted against the simple AR(1) model which only needed less than 30 seconds for all variables combined. To address the computational burden by the BART algorithm, Lakshminarayanan et al. (2015) proposed a Particle Gibbs sampler to speed up the algorithm while Jingyu and Hahn (2021) put forward Accelerated Bayesian Regression Trees also known as XBART, which in their simulation studies showed of computational benefits compared to BART and other rivaling algorithms such as a neural network one and XGBoost. Although, if the user has access to a very high performing computer, the speed may not impose an issue as long as the sample size is kept moderate and the covariates likewise.

When it comes to nowcasting, the BART model seems to perform just as well, or close to the other Bayesian models for the high frequency variables. It's also clear that the performance of the low frequency variables using BART, as well as the other Bayesian models seem to make more improvements as the nowcast is initiated at a later month in a given quarter during the pandemic period. Although this was only prominent in the

pandemic period, it may indicate that adding more high frequency observations can have a positive impact on performance, which is similar to what was found in Schorfheide and Song (2015).

Huber et al. (2020) argued that the Nowcasting strength in BART may be due to the algorithm's ability to include possible outliers as nodes in the trees and that the hyper-parameter $\sigma_{k,\mu}$ in (3.34) also allows for greater flexibility as it helps adjust for extreme values in the data. While possible outliers can be incorporated in the trees, the findings here showed that BART didn't produce any major difference in accuracy compared to the more traditional Bayesian models. As seen in section 3.1, BART is good at capturing nonlinear relationships in particular where there are several interactions between variables. It's possible that the variable used here exhibit more linear relationships.

The overall result shows that the AR(1) model performs quite well and is sometimes more accurate than all the other mixed frequency models which could be a sign of the more complex models overfitting to the training data. Additionally, the results also indicates that performance gains using mixed frequencies are higher for some variables than others. Studies devoted to mixed frequency forecasting are often focused on the performance gains in specifically GDP (as in Schorfheide and Song (2015), Mariano and Murasawa (2003) Mariano and Murasawa (2010) and Huber et al. (2020)). The results here also indicated that out of all the variables, GDP seemed to benefit the most by using a mixed frequency approach, even if BART in particular didn't outperform the other Bayesian models. Using a mixed frequency Bayesian approach could therefore be beneficial Since GDP is often an important variable used in macroeconomic forecasting. It may also therefore be of interest in future studies to use several other low frequency variables to further explore if predictions generally improve by using a mixed frequency VAR approach.

Furthermore, the results in section 5.1 clearly showed that the standard BVAR model outperformed BART for every single horizon. This might also indicate that BART performs better when more variables are added to the equation and could also give an indication that the horseshoe prior may be too restrictive and inappropriate in sparse models. It's an aggressive prior and the results might have been more in favor of BART had a Minnesota or a steady state prior been implemented instead.

If there's a desire to use a nonparametric approach then a possibility for future research could be to use Gaussian process regression in a mixed frequency setting. Another possibility could be to combine BART with a mixed data sampling (MIDAS) regression. Although the univariate outcome BART algorithm is quite robust and can handle parameter-rich models with the number of parameters grossly exceeding the number of observations as shown in simulation experiments in Chipman et al. (2010), this might not hold in a MF-VAR-setting. The findings here suggests that using too few trees not only hurts predictive performance, but may also cause failure of convergence. This may pose an issue since the algorithm is very costly, especially compared to other competing methods that may yield good results. BART also seems to be more unstable as more variables are added, especially with lower frequencies. The implication of using two different priors in the covariance matrix (an inverse-χ^2 for the diagonal elements in $\mathbf{H}$ and a horseshoe prior for each element in $\mathbf{c}_k$ given by (3.25)) on the stability of the model may also be explored to a wider extent.

References

Ankargren, S (2019). *VAR Models, Cointegration and Mixed-Frequency Data*. Uppsala University, Uppsala.

Ankargren, S, Unosson, M, and Yang, Y (2020). "A flexible mixed-frequency vector autoregression with a steady-state prior". In: *Journal of Time Series Econometrics* 12(2), pp. 289–328. DOI: 10.1515/jtse-2018-0034.

Ankargren, S and Yang, Y (2019). *Mixed-Frequency Bayesian VAR Models in R: The mfbvar Package*. https://cran.r-project.org/web/packages/mfbvar/vignettes/mfbvar_jss.pdf. Accessed: 2022-03-11.

Bai, J, Ghysels, E, and Wright, J (2013). "State space models and MIDAS regression". In: *Econometric Reviews* 32, pp. 779–813. DOI: 10.1080/07474938.2012.690675.

Carriero, A, Clark, T E, and Marcellino, M (2019). "Large Bayesian vector autoregressions with stochastic volatility and non-conjugate priors". In: *Journal of Econometrics* 212(1), pp. 137–154. DOI: 10.1016/j.jeconom.2019.04.024.

Carter, C K and Kohn, R (1994). "On Gibbs Sampling for State Space Models". In: *Biometrika* 81(3), pp. 541–553. DOI: https://doi.org/10.2307/2337125.

Carvalho, C M, Polson, N G, and Scott, J G (2010). "The horseshoe estimator for sparse signals". In: *Biometrika* 97(2), pp. 465–480. DOI: 10.2307/25734098.

Chan, J C C (2021). "Minnesota-type adaptive hierarchical priors for large Bayesian VARs". In: *International Journal of Forecasting* 37(3), pp. 1212–1226. DOI: 10.1016/j.ijforecast.2021.01.002..

Chipman, H A, George, E I, and McCulloch, R E (1998). "Bayesian CART model search". In: *Journal of the American Statistical Association* 93, pp. 935–948. DOI: https://doi.org/10.1080/01621459.1998.10473750.

Chipman, H A, George, E I, and McCulloch, R E (2010). "Bart: Bayesian Additive Regression Trees". In: *The Annals of Applied Statistics* 4(1), pp. 266–298. DOI: 10.2307/27801587.

Clark, T E, Huber, F, Koop, G, Marcellino, M, and Pfarrhofer, M (2021). *Tail Forecasting with Multivariate Bayesian Additive Regression Trees*. Working Paper No. 21-08. Accessed: 2022-02-03. URL: https://doi.org/10.26509/frbc-wp-202108.

Cole, T J and Altman, D G (2017). "Statistics Notes: Percentage differences, symmetry, and natural logarithms". In: *British Medical Journal* 358(August). DOI: 10.1136/bmj.j3683.

Commandeur, J and Koopman, S J (2007). *An Introduction to State Space Time Series Analysis*. Oxford university press.

Crawford, L, Flaxman, S, Runcie, D, and West, M (2019). "Variable prioritization in nonlinear black box methods: A genetic association case study". In: *Annals of Applied Statistics* 13, pp. 958–989. DOI: 10.1214/18-aoas1222.

Crawford, L, Wood, K, Zhou, X, and Mukherjee, S (2018). "Bayesian Approximate Kernel Regression With Variable Selection". In: *Journal of the American Statistical Association* 113, pp. 1710–1721. DOI: 10.1080/01621459.2017.1361830.

Diebold, F X and Mariano, R S (1995). "Comparing predictive accuracy". In: *Journal of Business and Economic Statistics* 13, pp. 253–263.

Dieppe, A, Legrand, R, and van Roye, B (2016). *The BEAR Toolbox*. https://www.ecb.europa.eu/pub/pdf/scpwps/ecbwp1934.en.pdf. Working Paper No. 1934, European Central Bank. Accessed: 2022-04-13.

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

Ekonomifakta (2022). *Real BNP*. https://www.ekonomifakta.se/fakta/ekonomi/tillvaxt/real-bnp/. Accessed: 2022-05-08.

Engle, R F (1982). "Autoregressive Conditional Heteroskedasticity With Estimates of the Variance of U.K. Inflation". In: *Econometrica* 50, pp. 987–1008. DOI: https://doi.org/10.2307/1912773.

Eurostat (2021). *Industrial production (volume) index overview*. https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Industrial_production_(volume)_index_overview#General_overview. Accessed: 2022-05-08.

Frühwirth-Schnatter, S (1994). "Data augmentation and dynamic linear models". In: *Journal of Time series analysis* 15(2), pp. 183–202. DOI: https://doi.org/10.1111/j.1467-9892.1994.tb00184.x.

Gelman, A, Carlin, J B, Stern, H S, Dunson, D B, Vehtari, A, and Rubin, D B (2021). *Bayesian data analysis*. Vol. 3rd edition. CRC press.

Ghysels, E, Santa-Clara, P, and Valkanov, R (2002). *The MIDAS touch: mixed data sampling regression*. https://rady.ucsd.edu/faculty/directory/valkanov/pub/docs/midas-touch.pdf. Discussion Paper UCLA and UNC, Accessed: 2022-05-03.

Ghysels, E, Santa-Clara, P, and Valkanov, R (2005). "There is a risk-return tradeoff after all". In: *Journal of Financial Economics* 76, pp. 509–548.

Hamilton, J D (1994). *Time Series Analysis*. Princeton, NJ: Princeton University Press.

Huber, F, Koop, G, L, L Onorante, Pfarrhofer, M, and Schreiner, J (2020). "Nowcasting in a pandemic using non-parametric mixed frequency VARs. Journal of Econometrics". In: *Journal of Econometrics*. ISSN: 0304-4076. DOI: https://doi.org/10.1016/j.jeconom.2020.11.006.

Huber, F and Rossini, L (2020). *Inference in Bayesian Additive Vector Autoregressive Tree Models*. Preprint available on arXiv. DOI: 10.48550/ARXIV.2006.16333.

Hyndman, R J and Athanasopoulos, G (2018). *Forecasting: principles and practice*. Vol. 2nd Edition. Melbourne, Australia: OTexts. URL: https://otexts.com/fpp2/ (visited on 03/03/2022).

Jingyu, H and Hahn, R P (2021). "Stochastic tree ensembles for regularized nonlinear regression". In: *Journal of the American Statistical Association*, pp. 1–20. DOI: 10.1080/01621459.2021.1942012.

Kalman, R E (1960). "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82, pp. 35–45. DOI: doi:10.1115/1.3662552.

Kapelner, A and Bleich, J (2016). "bartMachine: Machine Learning with Bayesian Additive Regression Trees". In: *Journal of Statistical Software* 70, pp. 1–40. DOI: https://doi.org/10.18637/jss.v070.i04.

Kastner, G (2016). "Dealing with Stochastic Volatility in Time Series Using the R Package stochvol". In: *Journal of Statistical Software* 69(5), pp. 1–30. DOI: 10.18637/jss.v069.i05.

Kastner, G and Frühwirth-Schnatter, S (2014). "Ancillarity-Sufficiency Interweaving Strategy (ASIS) for Boosting MCMC Estimation of Stochastic Volatility Models". In: *Computational Statistics and Data Analysis* 76, pp. 408–423. DOI: 10.1016/j.csda.2013.01.002.

Kowarik, A, Meraner, A, Templ, M, and Schopfhauser, D (2014). "Seasonal Adjustment with the R Packages x12 and x12GUI". In: *Journal of Statistical Software* 62(1), pp. 1–21. DOI: 10.18637/jss.v062.i02.

Kuzin, V, Marcellino, M, and Schumacher, C (2011). "MIDAS vs. mixed-frequency VAR: Nowcasting GDP in the euro area". In: *International Journal of Forecasting* 27(2), pp. 529–542. DOI: 10.1016/j.ijforecast.2010.02.006.

Lakshminarayanan, B, Roy, D M, and Teh, Y W (2015). *Particle Gibbs for Bayesian Additive Regression Trees*. Preprint available on arXiv. DOI: 10.48550/ARXIV.1502.04622.

Lindholm, A, Wahlström, N, Lindsten, F, and Schön, T B (2022). *Machine Learning: A First Course for Engineers and Scientists*. Cambridge University Press.

Lindholm, U, Mossfeldt, M, and Stockhammar, P (2020). "Forecasting inflation in Sweden". In: *Economia Politica: Journal of Analytical and Institutional Economics* 37(1), pp. 39–68. DOI: 10.1007/s40888-019-00161-9.

Litterman, R B (1986). "Forecasting with Bayesian Vector Autoregressions – Five years of experience". In: *Journal of Business and Economic Statistics* 4, pp. 25–38. DOI: https://doi.org/10.1080/07350015.1986.10509491.

Louzis, D P (2019). "Steady-state modeling and macroeconomic forecasting quality". In: *Journal of applied econometrics* 34(2), pp. 285–314. DOI: https://doi.org/10.1002/jae.2657.

Lütkepohl, H (2006). *New Introduction to Multiple Time Series Analysis*. Springer-Verlag Berlin and Heidelberg GmbH Co. K.

Mariano, R and Murasawa, Y (2010). "A Coincident Index, Common Factors, and Monthly Real GDP". In: *Oxford Bulletin of Economics and Statistics* 72(1), pp. 27–46. DOI: https://doi.org/10.1111/j.1468-0084.2009.00567.x.

Mariano, R S and Murasawa, Y (2003). "A New Coincident Index of Business Cycles Based on Monthly and Quarterly Series". In: *Journal of Applied Econometrics* 18, pp. 427–443. DOI: https://doi.org/10.1002/jae.695.

MathWorks (2022). *jbtest*. https://se.mathworks.com/help/stats/jbtest.html;jsessionid=35ed756ec0e2ab93cb33d82c843d. Accessed: 2022-06-05.

Meinhold, R J and Singpurwalla, N D (1983). "Understanding the Kalman Filter". In: *The American Statistician* 37(2), pp. 123–127. DOI: 10.2307/2685871.

Schorfheide, F and Song, D (2015). "Real-Time Forecasting With a Mixed-Frequency VAR". In: *Journal of Business Economic Statistics* 33(3), pp. 366–380. DOI: 10.1080/07350015.2014.954707.

Sims, C A (1980). "Macroeconomics and Reality". In: *Econometrica* 48(1), pp. 1–48. DOI: 10.2307/1912017.

Statistics Sweden (2022). *Real BNP tar hänsyn till köpkraftsförändringar*. https://www.scb.se/hitta-statistik/statistik-efter-amne/nationalrakenskaper/nationalrakenskaper/nationalrakenskaper-kvartals-och-arsberakningar/produktrelaterat/Fordjupad-information/real-bnp-tar-hansyn-till-kopkraftsforandringar/. Accessed: 2022-05-08.

Swedish Riksbank (2022). *How is inflation measured?* https://www.riksbank.se/en-gb/monetary-policy/the-inflation-target/how-is-inflation-measured/. Accessed: 2022-03-13.

Tan, Y V and Roy, J (2019). "Bayesian additive regression trees and the General BART model". In: *Statistics in medicine* 38(25), pp. 5048–5069. DOI: 10.1002/sim.8347.

Villani, M (2009). "Steady-State Priors for Vector Autoregressions". In: *Journal of Applied Econometrics* 24(4), pp. 630–650. DOI: 10.1002/jae.1065.

A Appendix

A.1 Description of data

Real GDP originally in millions of Swedish krona consists of constant prices of GDP. It measures the total production value in a year while excluding the impact of inflation on these volumes. It's also possible to adjust GDP for the terms-of-trade effect, meaning that the net value of all exports and imports in a given year are also considered. Although the terms of trade effect was not included here, it's normally what's denoted as real GDP in Swedish economic terms (Statistics Sweden, 2022), but in this paper, real GDP will refer to the former measure. Both series exhibit very similar patterns, with the latter one considering the terms-of-trade effect being at a slightly lower level (Ekonomifakta, 2022).

Fixed consumer price index, with 1987 = 100, is used since it's been formally employed by the Swedish Riksbank as of September 2017. It's a consumer price index, excluding the mortgage rates' effects on the inflation. They previously used CPI along with some other series but made the final change due to the sometime misleading effect monetary policies could have on CPI (Swedish Riksbank, 2022).

The Unemployment rate is a survey based measure of the total share of unemployed in the working force aged 16-64. The series suffers from a potential major time series break around January 2021 since the method used to collect the data has been changed since that time point. It has since been retrieved from statistics Sweden's site and current work is devoted to linking the series.

Government bonds and Treasury Bills All the series are comprised of monthly averages, produced by the Riksbank. No transformation or adjustment was done on any of them. They fluctuate more than for instance the Swedish repo rate which has been negative or steady at 0 since 2015-2022 (April) and might therefore give a more nuanced picture about the state of development in the Swedish economy. Treasury bills comes in 1, 3, 6 and 12 months. But they all have quite similar patterns. The 3 month version was chosen because it fluctuates more than the more lengthy bonds.

Effective exchange rate index Denoted as KIX (Krona index) with November 1992 = 100, is a geometric index forming a type of average index compared to the Swedish krona with the rest of the world. It also takes into account the impact of imports and exports. An increase in the index means that the Krona has depreciated while a decrease means that it has appreciated.

Expected inflation It's a survey measure of what the inflation is expected to be by the total business sector in Sweden. It's also usually considered as stationary in the long run in Swedish settings so it's usually given in its level form.

Wage inflation series comes from the Swedish National Mediation Office (Medlingsinstitutet). It's quarterly data. Wages are generally of huge importance in driving the inflation, but with some delay because of contract negotiations and their duration. The series used here only covers the increase in the public sector since the total sector was not available for that long time span.

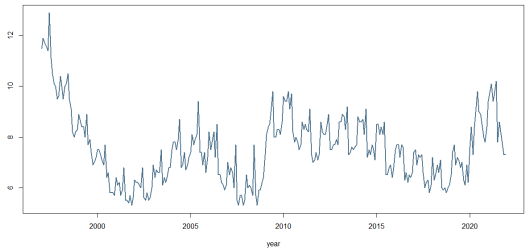
Industry capacity utilization is a survey based measure sent out to companies in the industrial sector with at least 10 employees. The series basically covers the

% of resources utilized in all industries and can be viewed as a measure of demand in the economy. Higher values indicates an overall larger pressure on production of goods.

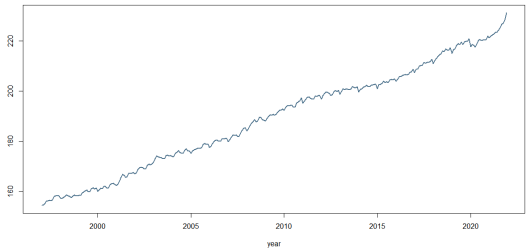
Production of total industry is an index measure of changes in the total output from all Swedish industries adjusted for inflation, with year 2015 = 100. Fluctuations in the IPI can often give good hints on the short term future developments of GDP (Eurostat, 2021). The main production items involved include intermediate goods (i.e. goods used in production for other goods), Energy, capital goods, durable consumer goods (goods with a lifespan longer than three years) and non-durable consumer goods (e.g. food, beverages or gasoline).

The objective was also to retain data on the production value index but there was no unbroken series available for the time span 1996-2021.

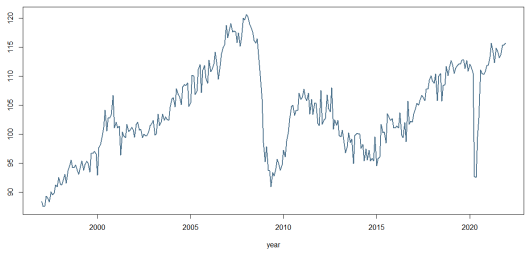
This concludes the description of the series used here. Figure 7 and 8 give a visual illustration of the data before any transformation and Figure 9 after the transformations. Table 6 shows the correlation between all the variables where the monthly variables have been aggregated to quarterly.



(a) Unemployment

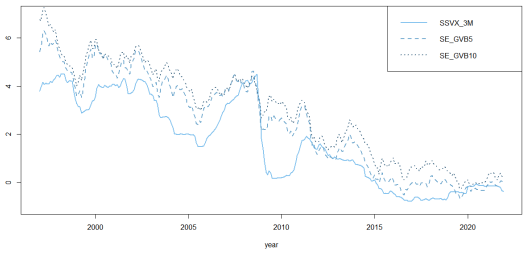


(b) CPIF



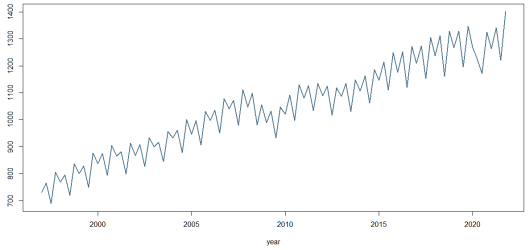
(c) IPI

(d) KIX

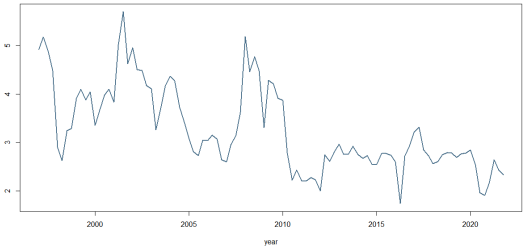


(e) Government bonds and Bills

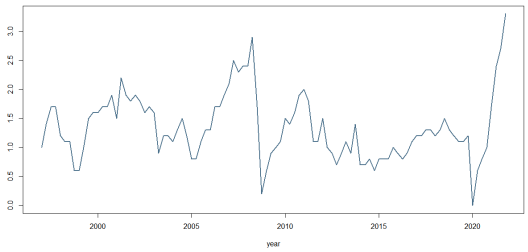
Figure 7: The untransformed monthly series



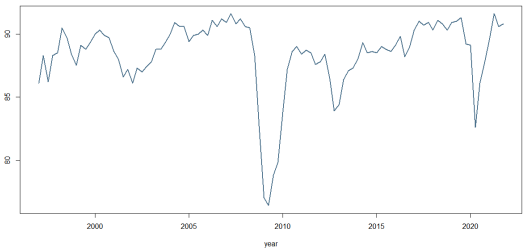
(a) GDP



(b) Wage inflation

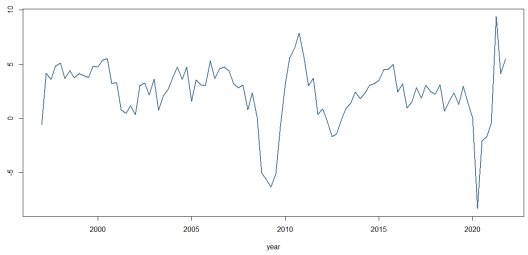


(c) Expected inflation

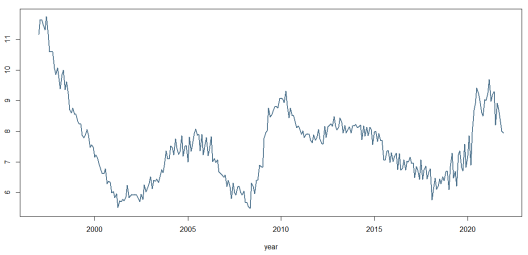


(d) Capacity utilization

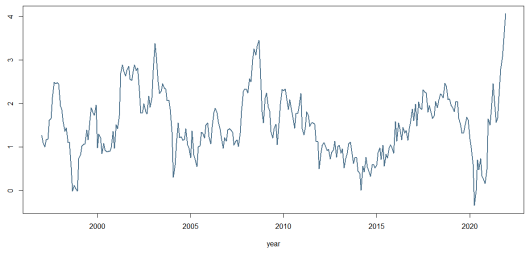
Figure 8: The untransformed quarterly series, GDP has been divided by 1000 for visual purposes



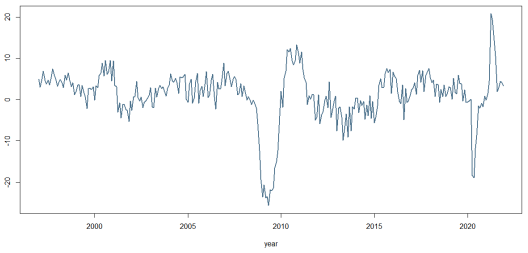
(a) GDP



(b) Unemployment



(c) CPIF



(d) IPI

Figure 9: All the transformed series

Table 6: Correlation table of data aggregated into quarterly frequencies

	GDP	Capacity Utilization	Expected Inflation	Wage inflation	KIX	Treasury Bills	Bonds 5Y	Bonds 10Y	Unemployment	CPIF	IPI
GDP	1										
Capacity utilization	0.705	1									
Expected inflation	0.409	0.365	1								
Wage inflation	-0.100	-0.235	0.226	1							
KIX	-0.313	-0.282	-0.007	0.309	1						
Treasury Bills	0.175	0.049	0.366	0.669	-0.058	1					
Bonds 5Y	0.214	-0.040	0.348	0.717	-0.024	0.945	1				
Bonds 10Y	0.203	-0.076	0.292	0.738	-0.026	0.919	0.993	1			
Unemployment	0.004	-0.290	-0.252	-0.132	-0.218	-0.078	0.011	0.052	1		
CPIF	0.067	0.042	0.584	0.418	0.367	0.183	0.195	0.184	-0.337	1	
IPI	0.851	0.801	0.391	-0.134	-0.288	0.125	0.105	0.089	-0.005	0.082	1

Below are Phillips Perron tests against unit roots for monthly and quarterly variables.

Table 7: Phillips Perron test on the original dataset

	Constant		Trend	
	Test Statistic	Critical Value	Test Statistic	Critical value
Monthly Variables				
Unemployment	-4.6458***	-2.8714	-4.5960***	-3.4263
CPIF	1.2758	-2.8714	-2.1659	-3.4263
IPI	-2.6660*	-2.8714	-2.9032	-3.4263
Bills 3m	-1.2331	-2.8714	-2.7528	-3.4263
Bonds 5y	-1.1855	-2.8714	-1.1855	-2.8714
Bonds 10y	-1.3938	-2.8714	-3.6519**	-3.4263
KIX	-2.7627*	-2.8714	-2.7597	-3.4263
Quarterly Variables				
GDP	-1.485697	-2.8906	-13.7174***	-3.4552
Capacity utilization	-3.4000**	-2.8906	-3.3926*	-3.4552
Expected inflation	-2.9342**	-2.8906	-2.8193	-3.4552
Wage inflation	-3.0409**	-2.8906	-4.0829***	-3.4552

Constant = testing for stationarity, Trend = testing for trend stationarity.

The critical values are based on a 5 % significance level.

***, **, * = 1%, 5% 10% significance levels respectively

Table 8: Phillips Perron tests for level stationarity on the transformed variables

	Unemployment	CPIF	IPI	GDP
Test statistic	-3.0924**	-3.2659**	-4.8134***	-4.7338***
Critical value	-2.8714	-2.8714	-2.8714	-3.4552

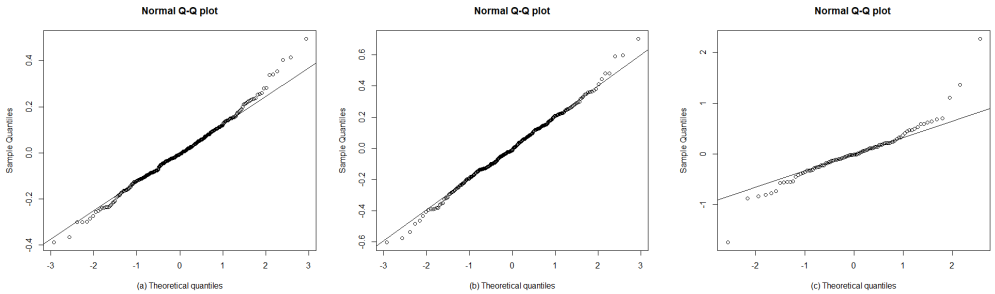
The critical values are based on a 5 % significance level.

***, **, * = 1%, 5% 10% significance levels respectively

A.2 Additional results

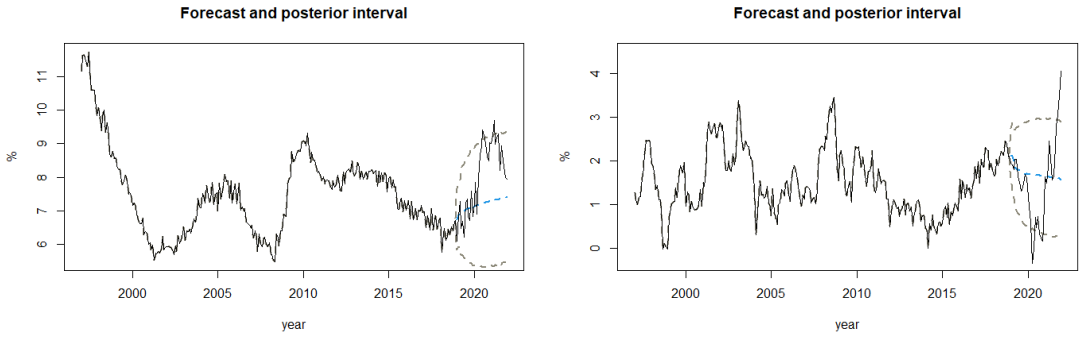
Table 9: Residual mean and 95% credible interval

		95% CI	
	Mean	Low	High
Unemployment	-0.0003	-0.2654	0.2648
CPIF	0.0002	-0.4040	0.4043
GDP	0.0078	-0.9474	0.9630

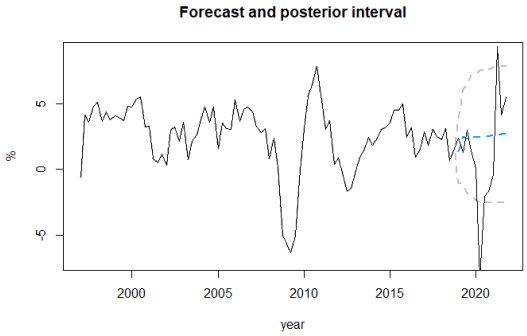


(a) Unemployment **(b)** CPIF **(c)** GDP

Figure 10: QQ-plot of residuals. Deviations from the straight line indicates non-normality



(a) Unemployment **(b)** CPIF



(c) GDP

Figure 11: Final 36 month forecast values (ranging from January 2018 - December 2021) of Unemployment (a), CPIF (b) and GDP (c) using BART with 95 % posterior bands (in lighter dashed lines) plotted along with the actual values. Monthly forecast values for GDP have been aggregated to quarterly for comparison

Table 10: Unconditional mean for the steady state prior

	Unemployment	CPIF	IPI	Bills 3m	Bonds 5y	Bonds 10y	KIX	GDP	Capacity Utilization	Expected inflation	Wage inflation
value	8.1634	1.6399	1.8081	2.6795	3.1947	-4.3211	113.7905	0.3470	85.2638	1.5843	3.5525

Table 11: Diebold Mariano test on high frequency variables

	Horizon 1	Horizon 2	Horizon 3
Pre-pandemic			
Unemployment	0.3463	0.2722	0.0852
CPIF	0.1988	0.1175	0.0375
IPI	0.3858	0.2410	0.3808
Bills 3m	0.6831	0.5671	0.4518
Bonds 5y	0.9894	0.9519	0.8776
Bonds 10y	0.9978	0.7153	0.3279
KIX	0.5897	0.7256	0.8140
Pandemic			
Unemployment	0.4064	0.5982	0.5343
CPIF	0.9383	0.6494	0.4206
IPI	0.7036	0.3361	0.2232
Bills 3m	0.9161	0.3828	0.0992
Bonds 5y	0.9871	0.8787	0.8511
Bonds 10y	0.9994	0.8041	0.6559
KIX	0.6983	0.5306	0.5716

p-values for BART vs the benchmark BVAR model using a one sided test where H_A: BVAR is less accurate than BART.

Table 12: Diebold Mariano test on low frequency variables

	Month 1	Month 2	Month 3
Pre-pandemic			
GDP	0.6136	0.7935	0.8675
Capacity Utilization	0.3418	0.9584	0.9686
Expected inflation	0.3421	0.5867	0.6857
Wage inflation	0.8871	0.0458	0.0441
Pandemic			
GDP	0.2863	0.8682	0.8841
Capacity Utilization	0.3698	0.6564	0.4496
Expected inflation	0.1021	0.2859	0.2490
Wage inflation	0.3865	0.4512	0.5537

p-values for BART vs the benchmark BVAR model using a one sided test where H_A: BVAR is less accurate than BART.

B Appendix

B.1 The Gibbs sampler

The Gibbs sampler can be seen as a special case of the Metropolis Hastings algorithm. There's a wide variety of extensions of the algorithm but for a simple Gibbs sampler the basic setup is to initialize starting values for the parameters, then draw the conditional distribution of the first parameter given the data and all other S-1 parameters. Followed by a draw of the second parameter conditional on the first obtained parameter, the data

and all other S-2 parameters and so on. This just concludes one iteration, where the iterations are repeated until convergence (Gelman et al., 2021, pp. 276). The procedure may take long if the parameter space is large.

The algorithm can be summarized as below.

A simple Gibbs sampler

Input:
Data: $\boldsymbol{y}$
Parameters: $\mu_1, \mu_2, ..., \mu_n$

Initialize starting values for all the parameters $\mu_1^{(0)}, \mu_2^{(0)}, ..., \mu_S^{(0)}$

For i = 1,2,...N+ burn-in
draw from

$$Pr(\mu_1^{(i)}|\mu_2^{(i-1)}, \mu_3^{(i-1)}, ..., \mu_S^{(i-1)}, \boldsymbol{y})$$

$$Pr(\mu_2^{(i)}|\mu_1^{(i)}, \mu_3^{(i-1)}, ..., \mu_S^{(i-1)}, \boldsymbol{y})$$

$$\vdots$$

$$Pr(\mu_S^{(i)}|\mu_1^{(i)}, \mu_2^{(i)}, ..., \mu_{S-1}^{(i)}, \boldsymbol{y})$$

Output: N posterior draws $(\boldsymbol{\mu}_1', \boldsymbol{\mu}_2', ..., \boldsymbol{\mu}_S')'$

B.2 The Kalman filter recursions

The filtering recursions works as updates of the system, when new observations appear.

Assume that we have a system represented by (3.20) and by following the derivations by Durbin and Koopman (2012, pp. 82) in a much more condensed form assuming that $Pr(\theta_t|\mathbf{Y}_{t-1}) \sim N(\boldsymbol{a}_t, \boldsymbol{P}_t)$ and that both $\boldsymbol{a}_t$ and $\boldsymbol{P}_t$ are known to us initially (an assumption that can be relaxed if needed) and that $Pr(\theta_t|\mathbf{Y}_t) \sim N(\boldsymbol{a}_{t|t}, \boldsymbol{P}_{t|t})$ and that $Pr(\theta_{t+1}|\mathbf{Y}_t) \sim N(\boldsymbol{a}_{t+1}, \boldsymbol{P}_{t+1})$ and also

$$\boldsymbol{a}_{t|t} = E(\boldsymbol{\theta}_t|\boldsymbol{Y}_t) = E(\boldsymbol{\theta}_t|\boldsymbol{Y}_{t-1}, \boldsymbol{\omega}_t) \tag{B.1}$$

$$\boldsymbol{a}_{t+1} = E(\boldsymbol{\theta}_{t+1}|\boldsymbol{Y}_t) = E(\boldsymbol{\theta}_{t+1}|\boldsymbol{Y}_{t-1}, \boldsymbol{\omega}_t) \tag{B.2}$$

where $\boldsymbol{\omega}_t = \boldsymbol{y}_t - E(\boldsymbol{y}_t|\boldsymbol{Y}_{t-1})$ is the difference between the observed vector $\boldsymbol{y}_t$ and the expected value of the observed vector given the previous values where $\boldsymbol{Y}_{t-1} = (\boldsymbol{y}_1', \boldsymbol{y}_2', ..., \boldsymbol{y}_{t-1}')'$ for $t = 2, 3, ...$, so that

$$\begin{aligned}\boldsymbol{\omega}_t &= \boldsymbol{y}_t - E(\boldsymbol{y}_t|\boldsymbol{Y}_{t-1}) \\ &= \boldsymbol{y}_t - E(\boldsymbol{V}_t\boldsymbol{\theta}_t + \boldsymbol{\varepsilon}_t|\boldsymbol{Y}_{t-1}) \\ &= \boldsymbol{y}_t - \boldsymbol{V}_t\boldsymbol{a}_t \end{aligned} \tag{B.3}$$

and assume that $\mathbf{F}_t$ is the variance of the prediction error $\boldsymbol{\omega}_t$ in the multivariate case

defined such that

$$\begin{aligned}\boldsymbol{F}_t &= Var(\boldsymbol{\omega}_t|\boldsymbol{Y}_{t-1}) \\ &= Var(\boldsymbol{V}_t\boldsymbol{\theta}_t + \boldsymbol{\varepsilon}_t - \boldsymbol{V}_t\boldsymbol{a}_t|\boldsymbol{Y}_{t-1}) \\ &= \boldsymbol{V}_t\boldsymbol{P}_t\boldsymbol{V}_t' + \boldsymbol{\Sigma}_{\boldsymbol{\eta}_t}\end{aligned} \tag{B.4}$$

And that the Kalman gain is given by

$$\boldsymbol{K}_t = \boldsymbol{W}_t\boldsymbol{P}_t\boldsymbol{V}_t'\boldsymbol{F}_t^{-1} \tag{B.5}$$

Then the recursions of the Kalman filer for the general, linear Gaussian model is given by

$$\begin{aligned}\boldsymbol{a}_{t|t} &= \boldsymbol{a}_t + \boldsymbol{P}_t\boldsymbol{V}_t'\boldsymbol{F}_t^{-1}\boldsymbol{\omega}_t \\ \mathbf{P}_{t|t} &= \mathbf{P}_t - \mathbf{P}_t\mathbf{V}_t'\mathbf{F}_t^{-1}\mathbf{V}_t\mathbf{P}_t \\ \boldsymbol{a}_{t+1} &= \boldsymbol{W}_t\boldsymbol{a}_t + \boldsymbol{K}_t\boldsymbol{\omega}_t \\ \mathbf{P}_{t+1} &= \mathbf{W}_t\mathbf{P}_t(\boldsymbol{W}_t - \boldsymbol{K}_t\boldsymbol{V}_t)' + \boldsymbol{M}_t\boldsymbol{\Sigma}_{\boldsymbol{\varepsilon}_t}\boldsymbol{M}_t\end{aligned} \tag{B.6}$$

where $t = 1, 2, ..., T$. The filtering step allows the user to look at the state at a specific point in time, which is ideal in nowcasting. When it comes to smoothing, all the sample time points T are used instead of a specific point in time. This makes smoothing generally better suited for evaluating a model as a whole.

B.3 Forward filtering backward sampling

Forward filtering backward sampling (FFBS) can be viewed as a stochastic version of the Kalman filter using a Gibbs sampler approach and it will only be treated very shallowly here in order to provide some intuition of the procedure. The goal is to simulate all the state variables based on all data. To ease notation, consider the case where y_t is univariate outcome and $Y_{1:t}$ is a vector of observations up to time t. Then $Y_{1:T}$ represents a vector of all observations and θ_t can either be univariate or some vector. The procedure for drawing the latent states can be formulated as

$$Pr(\theta|Y_{1:T}) \propto Pr(\theta_T|Y_{1:T}) \prod_{t=1}^{T-1} Pr(\theta_t|Y_{1:t}, \theta_{t+1}) \tag{B.7}$$

By assuming a Gaussian model, the Kalman filter is run in the forward recursions by simulating the first state conditional on the second state and all observed values up until time t ($Y_{1:t}$), The second state is simulated after this, conditional on the third state and all the observed ($Y_{1:t}$). This is continued until the final state is reached, which is only conditioned on the observed vector ($Y_{1:T}$). In order to do this, the mean and variance of the state vector needs to be obtained for each point in time. This requires setting a starting value for these for the first point in time.[9] Then the same procedure is run in reverse, starting from the last state. A draw of the state variable θ_T is produced and it's then possible to include this draw into the conditional distribution coming right before it i.e. θ_{T-1} and so on. There are thus fewer values to condition on compared to the conventional Gibbs sampler as seen for instance in Appendix B.1, which in practice should improve the speed.

The algorithm goes back to Frühwirth-Schnatter (1994) used for her dynamic linear model and also Carter and Kohn (1994) who used it for a linear state space model.

[9] Sometimes the value is known, but many times it's unknown and there are a few techniques to initialize these values.

BART is a non-linear model but a linear approximation is given by first calculating the projection matrix $\boldsymbol{A}$. In linear regression $\boldsymbol{A} = (\boldsymbol{X'X})^{-1}\boldsymbol{X'Y}$. In this setting

$$\boldsymbol{A} = Proj(\boldsymbol{X}^{\dagger}\boldsymbol{F}) \tag{B.8}$$

instead, where $\boldsymbol{F}$ is given as in the state equation in (3.24) and $\boldsymbol{X}^{\dagger}$ is a Moore-Penrose matrix inverse. The non-linear state equation can thus be denoted as

$$\boldsymbol{y}_t = \boldsymbol{A'X}_t + \boldsymbol{\varepsilon}_t \tag{B.9}$$

Huber et al. (2020) argue that this approach, which is used in a similar manner as Crawford et al. (2019, 2018) produces a good approximation and enables the use of the FFBS algorithm.

C Appendix

Abbreviations

ANOVA - Analysis of variance

AR(p) - Autoregressive of order p

ARCH-LM - Autoregressive conditional heteroskedasticity-Lagrange multiplier

ARIMA - Autoregressive integrated moving average

BART - Bayesian additive regression trees

BAVART - Bayesian additive vector autoregressive trees

BVAR - Bayesian Vector autoregression

CART - Classification and regression trees

CPIF - Fixed consumer price index

FFBS - Forward filtering backward sampling

FRED - Federal reserve economic data

GARCH - Generalized autoregressive conditional heteroskedasticity

GDP - Gross domestic product

IPI - Production of total industry index

KIX - Effective rate index

MF - Mixed frequency

MIDAS - Mixed data sampling

QF - Quarterly frequency

RMSE - Root mean square error

RMSFE - Root mean square forecast error

SS - Steady state prior

SSNG - Steady state prior with hierarchical shrinkage

SV - Stochastic volatility

VAR - Vector autoregression