Package ‘mfbvar’

February 10, 2021

Type Package

Title Mixed-Frequency Bayesian VAR Models

Version 0.5.6

Date 2021-02-09


License GPL-3

LazyData TRUE

URL https://github.com/ankargren/mfbvar

BugReports https://github.com/ankargren/mfbvar/issues

Imports Rcpp (>= 0.12.7), ggplot2 (>= 3.3.0), methods, lubridate, GIGrvg, stochvol (>= 2.0.3), RcppParallel, dplyr, magrittr, tibble, zoo

LinkingTo Rcpp, RcppArmadillo, RcppProgress, stochvol (>= 2.0.3), RcppParallel

Depends R (>= 3.5.0)

Suggests testthat, covr, knitr, ggridges, alfred, factorstochvol

RoxygenNote 7.1.1

Encoding UTF-8

SystemRequirements GNU make

VignetteBuilder knitr
estimate_mfbvar

Mixed-frequency Bayesian VAR

Description

The main function for estimating a mixed-frequency BVAR.

Usage

estimate_mfbvar(mfbvar_prior = NULL, prior, variance = "iw", ...)
estimate_mfbvar

Arguments

- `mfbvar_prior`: a mfbvar_prior object
- `prior`: either "ss" (steady-state prior), "ssng" (hierarchical steady-state prior with normal-gamma shrinkage) or "minn" (Minnesota prior)
- `variance`: form of the error variance-covariance matrix: "iw" for the inverse Wishart prior, "diffuse" for a diffuse prior, "csv" for common stochastic volatility or "fsv" for factor stochastic volatility
- Additional arguments to update_prior (if mfbvar_prior is NULL, the arguments are passed on to set_prior)

Value

An object of class mfbvar, mfbvar_<prior> and mfbvar_<prior>_<variance> containing posterior quantities as well as the prior object. For all choices of prior and variance, the returned object contains:

- **Pi**: Array of dynamic coefficient matrices; Pi[,r] is the rth draw
- **Z**: Array of monthly processes; Z[,r] is the rth draw
- **Z_fcst**: Array of monthly forecasts; Z_fcst[,r] is the rth forecast. The first n_lags rows are taken from the data to offer a bridge between observations and forecasts and for computing nowcasts (i.e. with ragged edges).

**Steady-state priors:** If prior = "ss", it also includes:
- **psi**: Matrix of steady-state parameter vectors; psi[r,] is the rth draw
- **roots**: The maximum eigenvalue of the lag polynomial (if check_roots = TRUE)

If prior = "ssng", it also includes:
- **psi**: Matrix of steady-state parameter vectors; psi[r,] is the rth draw
- **roots**: The maximum eigenvalue of the lag polynomial (if check_roots = TRUE)
- **lambda_psi**: Vector of draws of the global hyperparameter in the normal-Gamma prior
- **phi_psi**: Vector of draws of the auxiliary hyperparameter in the normal-Gamma prior
- **omega_psi**: Matrix of draws of the prior variances of psi; omega_psi[r,] is the rth draw, where diag(omega_psi[r,]) is used as the prior covariance matrix for psi

**Constant error covariances:** If variance = "iw" or variance = "diffuse", it also includes:
- **Sigma**: Array of error covariance matrices; Sigma[,r] is the rth draw

**Time-varying error covariances:** If variance = "csv", it also includes:
- **Sigma**: Array of error covariance matrices; Sigma[,r] is the rth draw
- **phi**: Vector of AR(1) parameters for the log-volatility regression; phi[r] is the rth draw
- **sigma**: Vector of error standard deviations for the log-volatility regression; sigma[r] is the rth draw
- **f**: Matrix of log-volatilities; f[r,] is the rth draw

If variance = "fsv", it also includes:
- **facload**: Array of factor loadings; facload[,r] is the rth draw
interval_to_moments

Description

Convert a matrix of 100*(1-alpha) % prior probability intervals for the steady states to prior moments.

Usage

interval_to_moments(prior_psi_int, alpha = 0.05)

Arguments

prior_psi_int Matrix of size (n_determ*n_vars) * 2 with the prior 95 % prior probability intervals.
alpha 100*(1-alpha) is the prior probability of the interval

References


See Also

set_prior, update_prior, predict.mfbvar, plot.mfbvar_minn, plot.mfbvar_ss, varplot, summary.mfbvar

Examples

prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 20)
mod_minn <- estimate_mfbvar(prior_obj, prior = "minn")
Value

A list with two components:

prior_psi_mean  The prior mean of psi
prior_psi_Omega  The prior covariance matrix of psi

Examples

prior_intervals <- matrix(c(0.1, 0.2,
                            0.4, 0.6), ncol = 2, byrow = TRUE)
psi_moments <- interval_to_moments(prior_intervals)

Description

mdd estimates the (log) marginal data density.

Usage

mdd(x, ...)

Arguments

x  argument to dispatch on (of class mfbvar_ss or mfbvar_minn)
...  additional named arguments passed on to the methods

Details

This is a generic function. See the methods for more information.

The marginal data density is also known as the marginal likelihood.

Value

The logarithm of the marginal data density.

See Also

mdd.mfbvar_ss_iw, mdd.mfbvar_minn_iw
mdd.mfbvar_minn_iw  Marginal data density method for class mfbvar_minn

Description

Estimate the marginal data density for the model with a Minnesota prior.

Usage

```r
## S3 method for class 'mfbvar_minn_iw'
mdd(x, ...)
```

Arguments

- `x`: object of class `mfbvar_minn`
- `...`: additional arguments (currently only `p_trunc` for the degree of truncation is available)

Details

The method used for estimating the marginal data density is the proposal made by Schorfheide and Song (2015).

Value

The logarithm of the marginal data density.

References


See Also

- `mdd`, `mdd.mfbvar_ss_iw`
Marginal data density method for class mfbvar_ss

Description

Estimate the marginal data density for the model with a steady-state prior.

Usage

```r
## S3 method for class 'mfbvar_ss_iw'
mdd(x, method = 1, ...)
```

Arguments

- `x`: object of class `mfbvar_ss`
- `method`: option for which method to choose for computing the mdd (1 or 2)
- `...`: additional arguments (currently only `p_trunc` for the degree of truncation for method 2 is available)

Details

Two methods for estimating the marginal data density are implemented. Method 1 and 2 correspond to the two methods proposed by Fuentes-Albero and Melosi (2013) and Ankargren, Unosson and Yang (2018).

Value

The logarithm of the marginal data density.

References


See Also

`mdd`, `mdd.mfbvar_minn_iw`
mfbvar

*mfbvar* A package for mixed-frequency Bayesian vector autoregressive (VAR) models.

Description

The mfbvar package makes estimation of Bayesian VARs with a mix of monthly and quarterly data simple. The prior for the regression parameters is normal with Minnesota-style prior moments. The package supports either an inverse Wishart prior for the error covariance matrix, yielding a standard normal-inverse Wishart prior, or a time-varying error covariance matrix by means of a factor stochastic volatility model through the *factorstochvol-package* package.

Specifying the prior

The prior of the VAR model is specified using the function `set_prior`. The function creates a prior object, which can be further updated using `update_prior`. The model can be estimated using the steady-state prior, which requires the prior moments of the steady-state parameters. The function `interval_to_moments` is a helper function for obtaining these from prior intervals.

Estimating the model

The model is estimated using the function `estimate_mfbvar`. The error covariance matrix is given an inverse Wishart prior or modeled using factor stochastic volatility. If the former is used, `mdd` can be used to estimate the marginal data density (marginal likelihood).

Processing the output

Plots of the output can be obtained from calling the generic function `plot` (see `plot-mfbvar`). If factor stochastic volatility is used, the time-varying standard deviations can be plotted using `varplot`. Predictions can be obtained from `predict.mfbvar`.

mf_sweden

*mf_sweden* Real-time data set for Sweden.

Description

A dataset containing real-time data for mixed and quarterly frequencies.

Usage

mf_sweden
Format

A mixed-frequency data set of five Swedish macroeconomic variables.

- **unemp** harmonized unemployment rate (source: OECD)
- **infl** inflation rate (source: OECD)
- **ip** industrial production (source: OECD)
- **eti** economic tendency indicator (source: National Institute of Economic Research)
- **gdp** GDP growth (source: Statistics Sweden)

References

OECD (2016) MEI Archive: Revisions Analysis Dataset.
Statistics Sweden (2016) Revisions, expenditure approach and hours worked at each release.

---

**mf_usa**  
**US Macroeconomic Data Set**

Description

A dataset containing mixed-frequency data from FRED for three US macroeconomic variables.

Usage

**mf_usa**

Format

A list with components:

- **CPIAUCSL** inflation rate
- **UNRATE** unemployment rate
- **GDPC1** GDP growth rate
plot-mfbvar  
Plotting methods for posterior mfbvar objects

Description

Methods for plotting posterior mfbvar objects.

Usage

## S3 method for class 'mfbvar_ss'
plot(
x,  
aggregate_fcst = TRUE,  
plot_start = NULL,  
pred_bands = 0.8,  
nrow_facet = NULL,  
ss_bands = 0.95,  
...  
)

## S3 method for class 'mfbvar_ssng'
plot(
x,  
aggregate_fcst = TRUE,  
plot_start = NULL,  
pred_bands = 0.8,  
nrow_facet = NULL,  
ss_bands = 0.95,  
...  
)

## S3 method for class 'mfbvar_minn'
plot(
x,  
aggregate_fcst = TRUE,  
plot_start = NULL,  
pred_bands = 0.8,  
nrow_facet = NULL,  
...  
)

varplot(x, variables = colnames(x$Y), var_bands = 0.95, nrow_facet = NULL, ...)

Arguments

x  
object of class mfbvar_minn or mfbvar_ss
aggregate_fcst  Boolean indicating whether forecasts of the latent monthly series should be aggregated to the quarterly frequency.

plot_start  Time period (date or number) to start plotting from. Default is to use 5*n_fcst time periods if n_fcst exists, otherwise the entire sample.

pred_bands  Single number (between 0.0 and 1.0) giving the coverage level of forecast intervals.

nrow_facet  an integer giving the number of rows to use in the facet

ss_bands  (Steady-state prior only) Single number (between 0.0 and 1.0) giving the coverage level of posterior steady-state intervals.

...  Currently not in use.

variables  Vector of names or positions of variables to include in the plot of variances

var_bands  (varplot only) Single number (between 0.0 and 1.0) giving the coverage level of posterior intervals for the error standard deviations.

Value

A ggplot.

Examples

prior_obj <- set_prior(Y = mf_usa, d = "intercept",
  n_lags = 4, n_reps = 20,
  n_fcst = 4, n_fc = 1)

prior_intervals <- matrix(c(1, 3,
  4, 8,
  1, 3), ncol = 2, byrow = TRUE)

psi_moments <- interval_to_moments(prior_intervals)

prior_psi_mean <- psi_moments$prior_psi_mean

prior_psi_Omega <- psi_moments$prior_psi_Omega

prior_obj <- update_prior(prior_obj,
  prior_psi_mean = prior_psi_mean,
  prior_psi_Omega = prior_psi_Omega)

mod_ss <- estimate_mfbvar(prior_obj, prior = "ss", variance = "fsv")

plot(mod_ss)

varplot(mod_ss)
Usage

## S3 method for class 'mfbvar_prior'
plot(x, nrow_facet = NULL, ...)

Arguments

x object of class mfbvar_prior
nrow_facet number of rows in facet
... Currently not in use.

Details

The function plots the data. If the prior moments for the steady-state parameters are available in x, these are included.

Value

A ggplot.

Examples

prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 20, n_fcst = 4)
plot(prior_obj)

predict.mfbvar Predict method for class mfbvar

Description

Method for predicting mfbvar objects.

Usage

## S3 method for class 'mfbvar'
predict(object, aggregate_fcst = TRUE, pred_bands = 0.8, ...)

Arguments

object object of class mfbvar
aggregate_fcst If forecasts of quarterly variables should be aggregated back to the quarterly frequency.
pred_bands The level of the probability bands for the forecasts.
... Currently not in use.

Details

Note that this requires that forecasts were made in the original mfbvar call.
Value

A tibble with columns:

- variable Name of variable
- time Time index
- fcst_date Date of forecast

If the argument pred_bands is given as a numeric value between 0 and 1, the returned tibble also includes columns:

- lower The \((1-pred\_bands)/2\) lower quantiles of the predictive distributions
- median The medians of the predictive distributions
- upper The \((1+pred\_bands)/2\) upper quantiles of the predictive distributions

If pred_bands NULL or NA, the returned tibble also includes the columns:

- fcst MCMC samples from the predictive distributions
- iter Iteration indexes for the MCMC samples

Examples

```r
prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 20, n_fcst = 4)
mod_minn <- estimate_mfbvar(prior_obj, prior = "minn")
predict(mod_minn)
```

print.mfbvar

Printing method for class mfbvar

Description

Method for printing mfbvar objects.

Usage

```r
## S3 method for class 'mfbvar'
print(x, ...)
```

Arguments

- `x` object of class mfbvar
- `...` Currently not in use.

Value

No return value, called for side effects.
Examples

```r
prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 20)
mod_minn <- estimate_mfbvar(prior_obj, prior = "minn")
mod_minn
```

print.mfbvar_prior

Print method for mfbvar_prior

Description

Printing method for object of class mfbvar_prior, checking if information in the prior is sufficient for estimating models.

Usage

```r
## S3 method for class 'mfbvar_prior'
print(x, ...)
```

Arguments

- `x` prior object (class mfbvar_prior)
- `...` additional arguments (currently unused)

Details

The print method checks whether the steady-state and Minnesota priors can be used with the current specification. This check is minimal in the sense that it checks only prior elements with no defaults, and it only checks for estimation and not forecasting (for which the steady-state prior requires additional information).

Value

No return value, called for side effects.

See Also

`set_prior`, `update_prior`, `estimate_mfbvar`, `summary.mfbvar_prior`

Examples

```r
prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 100)
print(prior_obj)
```
set_prior

---

**set_prior**

*Set priors for mfbvar*

---

**Description**

The function creates an object storing all information needed for estimating a mixed-frequency BVAR. The object includes data as well as details for the model and its priors.

**Usage**

```r
set_prior(
  Y,
  aggregation = "average",
  prior_Pi_AR1 = 0,
  lambda1 = 0.2,
  lambda2 = 0.5,
  lambda3 = 1,
  lambda4 = 10000,
  block_exo = NULL,
  n_lags,
  n_fcst = 0,
  n_thin = 1,
  n_reps,
  n_burnin = n_reps,
  freq = NULL,
  d = NULL,
  d_fcst = NULL,
  prior_psi_mean = NULL,
  prior_psi_Omega = NULL,
  check_roots = FALSE,
  s = -1000,
  prior_ng = c(0.01, 0.01),
  prior_phi = c(0.9, 0.1),
  prior_sigma2 = c(0.01, 4),
  n_fac = NULL,
  n_cores = 1,
  verbose = FALSE,
  ...
)
```

```r
update_prior(prior_obj, ...)
```

**Arguments**

- **Y**
  - data input. For monthly-quarterly data, should be a list with components containing regularly spaced time series (that inherit from `ts` or `zoo`) or a component containing a single time series, the component itself must be named. If
a component contains multiple time series, each time series must be named. Monthly variables can only contain missing values at the end of the sample, and should precede quarterly variables in the list. Matrices in which quarterly variables are padded with NA and observations stored at the end of each quarter are also accepted, but then the frequency of each variable must be given in the argument freq. Weekly-monthly mixes can be provided using the matrix way, see examples.

aggregation is the aggregation scheme used for relating latent high-frequency series to their low-frequency observations. The default is "average" for averaging within each low-frequency period (e.g., quarterly observations are averages of the constituent monthly observations). The alternative "triangular" can be used for monthly-quarterly mixes, and uses the Mariano-Murasawa triangular set of weights. See details for more information.

prior_Pi_AR1 is the prior means for the AR(1) coefficients.

lambda1 is the overall tightness.

lambda2 is (Only if variance is one of c(“diffuse”, “fsv”) The cross-variable tightness

lambda3 is The tightness of the intercept prior variance.

lambda4 is (Minnesota only) Prior variance of the intercept.

block_exo is (Only if variance is one of c(“diffuse”, “fsv”)) Vector of indexes/names of variables to be treated as block exogenous

n_lags is The number of lags.

n_fcst is The number of periods to forecast.

n_thin is Store every n_thinth draw

n_reps is The number of replications.

n_burnin is The number of burn-in replications.

freq is (Only used if Y is a matrix) Character vector with elements ’m’ (monthly) or ’q’ (quarterly) for sampling frequency. Monthly variables must precede all quarterly variables.

d is (Steady state only) Either a matrix with same number of rows as Y and n_determ number of columns containing the deterministic terms or a string “intercept” for requesting an intercept as the only deterministic term.

d_fcst is (Steady state only) The deterministic terms for the forecasting period (not used if d = ”intercept”).

prior_psi_mean is (Steady state only) Vector of length n_determ*n_vars with the prior means of the steady-state parameters.

prior_psi_Omega is (Steady state only) Matrix of size (n_determ*n_vars) * (n_determ*n_vars) with the prior covariance of the steady-state parameters.

check_roots is Logical, if roots of the companion matrix are to be checked to ensure stationarity.

s is (Hierarchical steady state only) scalar giving the tuning parameter for the Metropolis-Hastings proposal for the kurtosis parameter. If s < 0, then adaptive Metropolis-Hastings targeting an acceptance rate of 0.44 is used, where the scaling factor is restricted to the interval [-abs(s), abs(s)]
prior_ng  (Hierarchical steady state only) vector with two elements giving the parameters
   c(c0,c1) of the hyperprior for the global shrinkage parameter
prior_phi  (Only used with common stochastic volatility) Vector with two elements c(mean, variance)
   for the AR(1) parameter in the log-volatility regression
prior_sigma2 (Only used with common stochastic volatility) Vector with two elements c(mean, df)
   for the innovation variance of the log-volatility regression
n_fac     (Only used with factor stochastic volatility) Number of factors to use for the
   factor stochastic volatility model
n_cores   (Only used with factor stochastic volatility) Number of cores to use for drawing
   regression parameters in parallel
verbose   Logical, if progress should be printed to the console.
...       (Only used with factor stochastic volatility) Arguments to pass along to fsvsample.
   See details.
prior_obj  an object of class mfbvar_prior

Details

Some support is provided for single-frequency data sets, where Y contains variables sampled with
the same frequency.

The aggregation weights that can be used for aggregation are intra-quarterly averages (aggregation
   = "average"), where the quarterly observations y_{q,t} are assumed to relate to the underlying monthly
series z_{q,t} through:
   y_{q,t} = \frac{1}{3} (z_{q,t} + z_{q,t-1} + z_{q,t-2})

If aggregation = "triangular", then instead
   y_{q,t} = \frac{1}{9} (z_{q,t} + 2z_{q,t-1} + 3z_{q,t-2} + 2z_{q,t-3} + z_{q,t-4})

The latter is typically used when modeling growth rates, and the former when working with log-
levels.

If the steady-state prior is to be used, the deterministic matrix needs to be supplied, or a string
indicating that the intercept should be the only deterministic term (d = "intercept"). If the latter,
d_fcst is automatically set to be intercept only. Otherwise, if forecasts are requested (n_fcst > 0)
also d_fcst must be provided. Finally, the prior means of the steady-state parameters must (at the
very minimum) also be provided in prior_psi_mean. The steady-state prior involves inverting the
lag polynomial. For this reason, draws in which the largest eigenvalue (in absolute value) of the lag
polynomial is greater than 1 are discarded and new draws are made if check_roots = TRUE. The
maximum number of attempts is 1,000.

For modeling stochastic volatility by the factor stochastic volatility model, the number of factors
to use must be supplied. Further arguments can be passed along, but are not included as formal
arguments. If the default settings are not overridden, the defaults used are as follows (see fsvsample
for descriptions):
   • priormu = c(0,10)
   • priorphiidi = c(10,3)
• priorphifac = c(10,3)
• priorsigmaidi = 1
• priorsigmafac = 1
• priorfacload = 1
• restrict = "none"

The function `update_prior` can be used to update an existing prior object. See the examples.

Value
An object of class `mfbvar_prior` that is used as input to `estimate_mfbvar`.

See Also
`estimate_mfbvar`, `update_prior`, `interval_to_moments`, `print.mfbvar_prior`, `summary.mfbvar_prior`, `fsvsample`

Examples

```r
# Standard list-based way
prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 100)
prior_obj <- update_prior(prior_obj, n_fcst = 4)

# Weekly-monthly mix of data, four weeks per month
Y <- matrix(rnorm(400), 100, 4)
Y[setdiff(1:100, seq(4, 100, by = 4)), 4] <- NA
prior_obj <- set_prior(Y = Y, freq = c(rep("w", 3), "m"),
                        n_lags = 4, n_reps = 10)
```

---

summary.mfbvar

Summary method for class `mfbvar`

Description
Method for summarizing `mfbvar` objects.

Usage
```r
## S3 method for class 'mfbvar'
summary(object, ...)
```

Arguments

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>object of class <code>mfbvar</code></td>
</tr>
<tr>
<td>...</td>
<td>Currently not in use.</td>
</tr>
</tbody>
</table>
Examples

```r
prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 20)
mod_minn <- estimate_mfbvar(prior_obj, prior = "minn")
summary(mod_minn)
```

Description

summary method for object of class mfbvar_prior, showing some basic information regarding the contents of the prior.

Usage

```r
## S3 method for class 'mfbvar_prior'
summary(object, ...)
```

Arguments

- `object`: prior object (class mfbvar_prior)
- `...`: additional arguments (currently unused)

See Also

- `set_prior`, `update_prior`, `estimate_mfbvar`, `print.mfbvar_prior`

Examples

```r
prior_obj <- set_prior(Y = mf_usa, n_lags = 4, n_reps = 100)
summary(prior_obj)
```
Index

* datasets
  mf_sweden, 8
  mf_usa, 9

estimate_mfbvar, 2, 8, 14, 18, 19

fsvsample, 17, 18

ggplot, 11, 12

interval_to_moments, 4, 8, 18

mdd, 5, 6-8
mdd.mfbvar_minn_iw, 5, 6, 7
mdd.mfbvar_ss_iw, 5, 6, 7
mf_sweden, 8
mf_usa, 9
mfbvar, 8

plot-mfbvar, 10
plot.mfbvar_minn, 4
plot.mfbvar_minn(plot-mfbvar), 10
plot.mfbvar_prior, 11
plot.mfbvar_ss, 4
plot.mfbvar_ss(plot-mfbvar), 10
plot.mfbvar_sssng(plot-mfbvar), 10
predict.mfbvar, 4, 8, 12
print.mfbvar, 13
print.mfbvar_prior, 14, 18, 19

set_prior, 4, 8, 14, 15, 19
summary.mfbvar, 4, 18
summary.mfbvar_prior, 14, 18, 19

tibble, 13

update_prior, 4, 8, 14, 18, 19
update_prior(set_prior), 15

varplot, 4, 8
varplot(plot-mfbvar), 10