# Package ‘BVARverse’

October 12, 2022

- **Type**: Package
- **Title**: Tidy Bayesian Vector Autoregression
- **Version**: 0.0.1
- **Date**: 2020-09-07
- **Author**: Lukas Vashold [aut, cre] (<https://orcid.org/0000-0002-3562-3414>), Nikolas Kuschnig [aut] (<https://orcid.org/0000-0002-6642-2543>)
- **Maintainer**: Lukas Vashold <lukas.vashold@wu.ac.at>
- **Description**: Functions to prepare tidy objects from estimated models via 'BVAR' (see Kuschnig & Vashold, 2019 <doi:10.13140/RG.2.2.25541.60643>) and visualisation thereof. Bridges the gap between estimating models with 'BVAR' and plotting the results in a more sophisticated way with 'ggplot2' as well as passing them on in a tidy format.

- **URL**: https://github.com/nk027/bvarverse
- **BugReports**: https://github.com/nk027/bvarverse/issues
- **Depends**: R (>= 3.5.0), BVAR
- **Imports**: ggplot2, tidyr, generics, rlang
- **Suggests**: tinytest
- **License**: GPL-3
- **Encoding**: UTF-8
- **LazyData**: true
- **RoxygenNote**: 7.1.1
- **NeedsCompilation**: no
- **Repository**: CRAN
- **Date/Publication**: 2020-09-22 08:20:12 UTC

### R topics documented:

- `augment.bvar` .......................................................... 2
- `bv_ggplot` ............................................................... 3
- `tidy.bvar` ............................................................... 4
**Description**

Turn the outputs of a Bayesian VAR (see `bvar`) into a an augmented tibble. Methods are available for `bvar` objects (will yield coefficients and their quantiles), `bvar_fcast` objects (with predictions, their quantiles and optionally real datapoints), and `bvar_irf` objects (with impulse responses).

**Usage**

```r
## S3 method for class 'bvar'
augment(x, conf_bands = 0.16, ...)

## S3 method for class 'bvar_fcast'
augment(x, t_back = 0L, ...)

## S3 method for class 'bvar_irf'
augment(x, ...)
```

**Arguments**

- `x` A `bvar` or derived object to turn into a tibble.
- `conf_bands` Numeric vector. Credible intervals of coefficients to include in the tibble.
- `...` Not used.
- `t_back` Integer scalar. Whether to include actual datapoints in the tidied forecast.

**Value**

Returns a tibble with relevant information; quantiles can be found in the columns.

**Examples**

```r
# Access a subset of the fred_qd dataset
data <- fred_qd[, c("CPIAUCSL", "UNRATE", "FEDFUNDS")]

# Transform it to be stationary
data <- fred_transform(data, codes = c(5, 5, 1), lag = 4)

# Estimate a BVAR using one lag, default settings and very few draws
x <- bvar(data, lags = 1, n_draw = 1000L, n_burn = 200L, verbose = FALSE)

# Create tibbles from the outputs
augment(x)
augment(irf(x))
augment(predict(x))
```
Quick ggplot2 plots for Bayesian VARs

Description

Function to quickly plot outputs from bvar and derived objects. Supported plots include traces and densities, forecasts, and impulse response functions. For more flexible plots one may use the outputs of tidy.bvar and augment.bvar.

Usage

`bv_ggplot(x, ...)`

## Default S3 method:
`bv_ggplot(x, ...)`

## S3 method for class 'bvar_chains'
`bv_ggplot(x, ...)`

## S3 method for class 'bvar'
`bv_ggplot(x, type = c("trace", "density"), vars = NULL, vars_response = NULL, vars_impulse = NULL, orientation = c("horizontal", "vertical"), chains = list(), ...)

## S3 method for class 'bvar_irf'
`bv_ggplot(x, vars_response = NULL, vars_impulse = NULL, col = "#737373", ...)`

## S3 method for class 'bvar_fcast'
`bv_ggplot(x, vars = NULL, col = "#737373", t_back = 1L, ...)`

Arguments

- **x**: A bvar or derived object to turn into a dataframe.
- **...**: Not used.
- **type**: A string with the type (trace or density) of plot desired.
- **vars**: Character vector used to select variables. Elements are matched to hyperparameters or coefficients. Coefficients may be matched based on the dependent variable (by providing the name or position) or the explanatory variables (by providing the name and the desired lag). See the example section for a demonstration. Defaults to NULL, i.e. all hyperparameters.
tidy.bvar

Tidy BVAR outputs and convert into a tibble

Description

Turn the outputs of a Bayesian VAR (see bvar) into a tidy tibble. Methods are available for bvar objects (will yield a subset of coefficient and/or hyperparameter draws), bvar_coefs objects (with the coefficients and their quantiles) bvar_fcast objects (with predictions, their quantiles and optionally real datapoints), and bvar_irf objects (with impulse responses).
tidy.bvar

Usage

## S3 method for class 'bvar'
tidy(
  x,
  vars = NULL,
  vars_response = NULL,
  vars_impulse = NULL,
  chains = list(),
  ...
)

## S3 method for class 'bvar_coefs'
tidy(x, ...)

## S3 method for class 'bvar_fcast'
tidy(x, t_back = 0L, ...)

## S3 method for class 'bvar_irf'
tidy(x, ...)

Arguments

- **x**: A bvar or derived object to turn into a dataframe.
- **vars**: Character vector used to select variables. Elements are matched to hyperparameters or coefficients. Coefficients may be matched based on the dependent variable (by providing the name or position) or the explanatory variables (by providing the name and the desired lag). See the example section for a demonstration. Defaults to NULL, i.e. all hyperparameters.
- **vars_impulse, vars_response**: Optional character or integer vectors used to select coefficients. Dependent variables are specified with **vars_response**, explanatory ones with **vars_impulse**. Defaults to NULL, indicating that no coefficients will be processed. draws.
- **chains**: List of bvar objects. Contents of multiple runs are added to the output, in order to help in assessing coverage.
- **t_back**: Integer scalar. Whether to include actual datapoints in the tidied forecast.

Value

Returns a tidy tibble with relevant information for further processing.

Examples

# Access a subset of the fred_qd dataset
data <- fred_qd[, c("CPIAUCSL", "UNRATE", "FEDFUNDS")]
# Transform it to be stationary
data <- fred_transform(data, codes = c(5, 5, 1), lag = 4)
# Estimate a BVAR using one lag, default settings and very few draws
x <- bvar(data, lags = 1, n_draw = 1000L, n_burn = 200L, verbose = FALSE)

# Create tidy tibbles from the outputs
 tidy(x)
tidy(irf(x))
tidy(predict(x))
Index

augment.bvar, 2, 3
augment.bvar_fcast (augment.bvar), 2
augment.bvar_irf (augment.bvar), 2

bv_ggplot, 3
bvar, 2, 4

tibble, 2, 5
tidy.bvar, 3, 4
tidy.bvar_coefs (tidy.bvar), 4
tidy.bvar_fcast (tidy.bvar), 4
tidy.bvar_irf (tidy.bvar), 4