

Package ‘FARS’

May 22, 2025

Type Package

Title Factor-Augmented Regression Scenarios

Version 0.3.0

Maintainer Gian Pietro Bellocca <gbellocc@est-econ.uc3m.es>

Description Provides a comprehensive framework in R for modeling and forecasting economic scenarios based on multi-level dynamic factor model. The package enables users to: (i) extract global and block-specific factors using a flexible multilevel factor structure; (ii) compute asymptotically valid confidence regions for the estimated factors, accounting for uncertainty in the factor loadings; (iii) estimate factor-augmented quantile regressions; (iv) recover full predictive densities from these quantile forecasts; and (v) estimate the density when the factors are stressed.

Depends R (>= 3.5.0)

Imports ggplot2, plotly, sn, nloptr, ellipse, SyScSelection, quantreg,
tidyr, dplyr, forcats, MASS, reshape2, stringr,

Suggests devtools, knitr, rmarkdown, openxlsx, readxl, zoo

VignetteBuilder knitr

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Gian Pietro Bellocca [aut, cre],
Ignacio Garrón [aut],
Vladimir Rodríguez-Caballero [aut],
Esther Ruiz [aut]

Repository CRAN

Date/Publication 2025-05-22 10:50:05 UTC

Contents

compute_fars	2
correct_outliers	3

create_scenario	4
density	5
mldfm	6
mldfm_subsampling	8
nl_density	9
plot.fars	10
plot.fars_density	10
plot.mldfm	11
print.fars	12
print.fars_density	12
print.mldfm	13
quantile_risk	13
summary.fars	14
summary.fars_density	14
summary.mldfm	15
Index	16

compute_fars	<i>Compute Factor Augmented Quantile Regressions and Stressed Quantiles</i>
--------------	---

Description

Performs quantile regressions of a dependent variable on MLDFM-extracted factors. Optionally generates quantile forecasts under stressed scenarios using hyperellipsoids.

Usage

```
compute_fars(  
  dep_variable,  
  factors,  
  h = 1,  
  edge = 0.05,  
  scenario = NULL,  
  min = TRUE  
)
```

Arguments

dep_variable	A numeric vector representing the dependent variable (e.g., GDP growth, inflation).
factors	A matrix of factor estimates from a mldfm model.
h	Integer. Forecast horizon (in time steps) for the quantile regression. Default is 1.
edge	Numeric. Trimming amount applied to the outermost quantiles (default 0.05).
scenario	Optional list of matrices representing a stressed scenario, as returned by create_scenario().

min Logical. If TRUE (default), implement a stepwise minimization. If FALSE, implement a stepwise maximization.

Value

A list containing:

Quantiles Matrix of forecasted quantiles (rows = time, cols = quantile levels).

Scenario_Quantiles Matrix of stressed scenario quantiles (same format), returned only if `scenario` is provided.

Coeff Matrix of quantile regression coefficients for each quantile.

Std. Error Matrix of Std. Error for each regression coefficient.

Pvalue Matrix of p-values for each regression coefficient.

Examples

```
dep_variable <- rnorm(100) # A numeric vector
data <- matrix(rnorm(100*300), nrow = 100, ncol = 300)
block_ind <- c(150, 300) # Defines 2 blocks
global = 1
local <- c(1, 1)
mldfm_result <- mldfm(data, blocks = 2, block_ind = block_ind, global = global , local = local)
fars_result <- compute_fars(dep_variable, mldfm_result$Factors, h = 1, edge = 0.05, min = TRUE)
```

correct_outliers	<i>Correct Outliers in a Dataset</i>
------------------	--------------------------------------

Description

This function identifies and corrects outliers in a dataset using principal component analysis (PCA). It scales the data, performs PCA, computes idiosyncratic components, and replaces values that fall outside a defined outlier threshold with the median of 5 previous values. The outlier threshold is determined using the interquartile range (IQR) method.

Usage

```
correct_outliers(data, r)
```

Arguments

data A numeric matrix or data frame where rows represent observations and columns represent variables.

r An integer specifying the number of principal components to use for PCA.

Value

A list containing:

data	A matrix with corrected data where outliers are replaced by the median of previous values.
outliers	A binary matrix (same dimensions as the input data) indicating the position of outliers.

Examples

```
data <- matrix(rnorm(100), nrow = 10, ncol = 10)
result <- correct_outliers(data, r = 3)
corrected_data <- result$data
outliers_matrix <- result$outliers
```

create_scenario	<i>Create Stressed Scenarios</i>
-----------------	----------------------------------

Description

Constructs confidence regions (hyperellipsoids) for the factor space based on a central MLDFM estimate and a set of subsampled estimates. These regions capture estimation uncertainty and are used to simulate stresses scenarios.

Usage

```
create_scenario(model, subsamples, data, block_ind, alpha = 0.95)
```

Arguments

model	An object of class <code>mldfm</code> , containing the factor estimates.
subsamples	A list of <code>mldfm</code> objects returned from <code>mldfm_subsampling</code> .
data	A numeric matrix or data frame containing the time series data. Rows represent time points; columns represent observed variables.
block_ind	A vector of integers indicating the end index of each block. Must be of length blocks and in increasing order. Required if blocks > 1.
alpha	Numeric. Confidence level (level of stress) for the hyperellipsoid (e.g., 0.95).

Value

A list of matrices representing the hyperellipsoid points for each time observation.

Examples

```

data <- matrix(rnorm(100*300), nrow = 100, ncol = 300)
block_ind <- c(150, 300) # Defines 2 blocks
global = 1
local <- c(1, 1)
mldfm_result <- mldfm(data, blocks = 2, block_ind = block_ind,
global = global, local = local)
mldfm_subsampling_result <- mldfm_subsampling(data, blocks = 2,
block_ind = block_ind, global = global,
local = local, n_samples = 100, sample_size = 0.9)
scenario <- create_scenario(mldfm_result, mldfm_subsampling_result,
data, block_ind, alpha = 0.95)

```

density

*Compute Skew-t Densities from Forecasted Quantiles***Description**

Fits a skew-t distribution to a set of quantile forecasts using linear optimization

Usage

```

density(
  quantiles,
  levels = c(0.05, 0.25, 0.5, 0.75, 0.95),
  est_points = 512,
  random_samples = 5000,
  seed = NULL
)

```

Arguments

quantiles	A matrix of forecasted quantiles. Each row is a time observation; each column a quantile level.
levels	A numeric vector of the quantile levels corresponding to the columns of the quantile matrix (default: c(0.05, 0.25, 0.50, 0.75, 0.95)).
est_points	Integer. Number of evaluation points for the estimated density (default: 512).
random_samples	Integer. Number of random samples to draw from the fitted skew-t distribution (default: 5000).
seed	Optional integer to set the random seed for reproducibility (default: NULL).

Value

An object of class "fars_density", which is a list containing the following components:

density A matrix of estimated densities for each time period (rows) across estimation points (columns).

distribution A matrix of random draws from the fitted skew-t distribution for each time period.

x_vals The sequence of evaluation points used to compute the density. Useful for plotting.

Examples

```
Quantiles <- matrix(rnorm(500, mean = 0, sd = 1), nrow = 100, ncol = 5)
Levels <- c(0.05, 0.25, 0.5, 0.75, 0.95)
density_result <- density(Quantiles,
                           levels = Levels,
                           est_points = 512,
                           random_samples = 100000,
                           seed = 42)
```

mldfm

Estimate Multilevel Dynamic Factor Model

Description

Estimates a multilevel dynamic factor model from time series data. Supports both single-block and hierarchical multi-block structures with customizable factor extraction settings.

Usage

```
mldfm(
  data,
  blocks = 1,
  block_ind = NULL,
  global = 1,
  local = NULL,
  middle_layer = NULL,
  method = 0,
  tol = 1e-06,
  max_iter = 1000,
  verbose = TRUE
)
```

Arguments

<code>data</code>	A numeric matrix or data frame containing the time series data. Rows represent time points; columns represent observed variables.
<code>blocks</code>	Integer. Number of blocks into which the data is divided.
<code>block_ind</code>	Integer vector. End column indices for each block. Must be of length <code>blocks</code> and in increasing order.
<code>global</code>	Integer. Number of global factors extracted from the entire dataset.
<code>local</code>	Integer vector of length <code>blocks</code> . Specifies the number of local factors for each block.
<code>middle_layer</code>	Named list. Each name is a string specifying a group of blocks (e.g., "1-3" or "2-3"), and each value is the number of factors to extract.
<code>method</code>	Integer. Method used to initialize the factors: 0 for Canonical Correlation Analysis (CCA), 1 for Principal Component Analysis (PCA).
<code>tol</code>	Numeric. The tolerance level for the residual sum of squares (RSS) minimization process. Used as a convergence criterion.
<code>max_iter</code>	Integer. The maximum number of iterations allowed for the RSS minimization process.
<code>verbose</code>	Logical. If TRUE (default), print a summary of the mldfm.

Value

An object of class `mldfm`, which is a list containing the following components:

Factors Matrix of estimated factors.

Lambda Matrix of factor loadings.

Residuals Matrix of residuals.

Iterations Number of iterations before convergence.

Factors_list List of estimated factors for each node.

Examples

```
data <- matrix(rnorm(1000), nrow = 100, ncol = 519)
block_ind <- c(63, 311, 519) # Defines 3 blocks
local <- c(1, 1, 1)          # One local factor per block
middle_layer <- list("1-3" = 1)
result <- mldfm(data, blocks = 3, block_ind = block_ind, global = 1,
  local = local, middle_layer = middle_layer)
summary(result)
```

mldfm_subsampling	<i>Subsampling Procedure for MLDFM Estimation</i>
-------------------	---

Description

Repeatedly applies the MLDFM estimation to randomly drawn subsamples of the input data.

Usage

```
mldfm_subsampling(
  data,
  blocks = 1,
  block_ind = NULL,
  global = 1,
  local = NULL,
  middle_layer = NULL,
  method = 0,
  tol = 1e-06,
  max_iter = 1000,
  n_samples = 10,
  sample_size = 0.9,
  seed = NULL
)
```

Arguments

data	A numeric matrix or data frame containing the time series data. Rows represent time points; columns represent observed variables.
blocks	Integer. The number of blocks into which the data is divided.
block_ind	A vector of integers indicating the end index of each block. Must be of length blocks and in increasing order. Required if blocks > 1.
global	Integer. Number of global factors extracted from the entire dataset.
local	Integer vector of length blocks. Specifies the number of local factors for each block.
middle_layer	Named list. Each name is a string specifying a group of blocks (e.g., "1-3" or "2-3"), and each value is the number of factors to extract.
method	Integer. The method used to initialize the factors: 0 for Canonical Correlation Analysis (CCA), 1 for Principal Component Analysis (PCA).
tol	Numeric. The tolerance level for the residual sum of squares (RSS) minimization process. Used as a convergence criterion.
max_iter	Integer. The maximum number of iterations allowed for the RSS minimization process.
n_samples	Number of subsamples to generate.
sample_size	Proportion of the original sample to retain (e.g., 0.9 for 90%).
seed	Optional integer. Seed for reproducibility of the subsampling process. If NULL, random draws will differ each run.

Value

A list of mldfm objects, one for each subsample.

Examples

```
data <- matrix(rnorm(1000), nrow = 100, ncol = 100)
block_ind <- c(50,100) # Defines 3 blocks
local <- c(1, 1)
result <- mldfm_subsampling(data, blocks = 2, block_ind = block_ind, global = 1,
local = local, n_samples = 100, sample_size = 0.9)
```

nl_density	<i>Compute Skew-t Densities from Forecasted Quantiles (Nonlinear Optimization)</i>
------------	--

Description

Fits a skew-t distribution to a set of quantile forecasts using nonlinear optimization

Usage

```
nl_density(
  quantiles,
  levels = c(0.05, 0.25, 0.5, 0.75, 0.95),
  est_points = 512,
  random_samples = 5000,
  seed = NULL
)
```

Arguments

quantiles	A matrix of forecasted quantiles. Each row is a time observation; each column a quantile level.
levels	A numeric vector of the quantile levels corresponding to the columns of the quantile matrix (default: c(0.05, 0.25, 0.50, 0.75, 0.95)).
est_points	Integer. Number of evaluation points for the estimated density (default: 512).
random_samples	Integer. Number of random samples to draw from the fitted skew-t distribution (default: 5000).
seed	Optional integer to set the random seed for reproducibility (default: NULL).

Value

An object of class "fars_density", which is a list containing the following components:

density A matrix of estimated densities for each time period (rows) across estimation points (columns).
distribution A matrix of random draws from the fitted skew-t distribution for each time period.
x_vals The sequence of evaluation points used to compute the density. Useful for plotting.

Examples

```
Quantiles <- matrix(rnorm(500, mean = 0, sd = 1), nrow = 100, ncol = 5)
Levels <- c(0.05, 0.25, 0.5, 0.75, 0.95)
density_result <- nl_density(Quantiles,
                             levels = Levels,
                             est_points = 512,
                             random_samples = 100000,
                             seed = 42)
```

plot.fars	<i>Plot Method for fars Object</i>
-----------	------------------------------------

Description

Generates line plots of forecasted quantiles from a FARS object. If a stressed scenario is available, it is plotted in a separate panel.

Usage

```
## S3 method for class 'fars'
plot(x, dates = NULL, ...)
```

Arguments

x	An object of class fars.
dates	Optional vector of dates (as Date or zoo::yearqtr) to use for the x-axis. If not provided, a simple index is used.
...	Additional arguments (currently ignored).

Value

No return value. Called for plot generation.

plot.fars_density	<i>Plot method for fars_density objects</i>
-------------------	---

Description

Plots the evolution of the estimated density over time as a 3D surface.

Usage

```
## S3 method for class 'fars_density'
plot(x, time_index = NULL, ...)
```

Arguments

x	An object of class <code>fars_density</code> .
time_index	Optional vector for the time axis (default is <code>1:nrow</code>).
...	Additional arguments passed to the plot function. (ignored)

Value

An interactive plot of class `plotly`.

<code>plot.mldfm</code>	<i>Plot method for MLDFM object</i>
-------------------------	-------------------------------------

Description

Dispatches to specific plot functions for factors, loadings, or residuals.

Usage

```
## S3 method for class 'mldfm'
plot(x, which = "factors", dates = NULL, var_names = NULL, ...)
```

Arguments

x	An object of class <code>mldfm</code> .
which	What to plot: one of "factors" (default), "loadings", or "residuals".
dates	Optional vector of dates (as <code>Date</code> or <code>zoo::yearqtr</code>) to use for the x-axis. If not provided, a simple index (1:N) is used.
var_names	Optional vector of variable names to label loadings and residual axis.
...	Additional arguments (ignored)

Value

No return value. Called for plots generation.

print.fars	<i>Print method for fars object</i>
------------	-------------------------------------

Description

Prints a short summary of the fars object

Usage

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

Arguments

x	An object of class fars_quantiles.
...	Additional arguments (ignored).

Value

The input object x, returned invisibly.

print.fars_density	<i>Print method for fars_density objects</i>
--------------------	--

Description

Displays a brief summary of the density estimation object produced by the densi ty() or nl_densi ty() function.

Usage

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

Arguments

x	An object of class fars_density.
...	Additional arguments (ignored).

Value

The input object x, returned invisibly.

print.mldfm	<i>Print Method for MLDFM Object</i>
-------------	--------------------------------------

Description

Prints a short summary of the multilevel dynamic factor model

Usage

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

Arguments

x	An object of class mldfm.
...	Additional arguments (ignored).

Value

The input object x, invisibly.

quantile_risk	<i>Extract Conditional Quantile from Simulated Densities</i>
---------------	--

Description

Computes the conditional quantile (e.g., 5th percentile) from a simulated skew-t distribution, generated via `density()` or `nl_density()`. The result corresponds to the risk measure (e.g., Growth-at-Risk, Inflation-at-Risk, Groth-in-Stress etc.).

Usage

```
quantile_risk(density, QTAU = 0.05)
```

Arguments

density	An object of class <code>fars_density</code> returned by <code>density()</code> or <code>nl_density()</code> .
QTAU	A numeric value between 0 and 1 indicating the quantile to extract (e.g., 0.05 for 5th percentile).

Value

A numeric vector of conditional quantiles (one observation for each time period).

Examples

```
Quantiles <- matrix(rnorm(500), ncol = 5)
fars_density <- density(Quantiles, levels = c(0.05,0.25,0.50,0.75,0.95),
  est_points = 512, random_samples = 1000)
GaR <- quantile_risk(fars_density, QTAU = 0.05)
```

summary.fars	<i>Summary Method for fars Object</i>
--------------	---------------------------------------

Description

Prints a complete summary of the fars object.

Usage

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

Arguments

object	An object of class fars_quantiles.
...	Additional arguments (ignored).

Value

The input object object, returned invisibly.

summary.fars_density	<i>Summary method for fars_density objects</i>
----------------------	--

Description

Provides summary statistics of the density estimation for each time observation, including the mean, median, and standard deviation of the simulated distribution.

Usage

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

Arguments

object	An object of class fars_density.
...	Additional arguments (ignored).

Value

A data frame with one row per time observation and columns: Observation, Mean, Median, and StdDev. The object is also printed to the console and returned invisibly.

summary.mldfm	<i>Summary Method for MLDFM Object</i>
---------------	--

Description

Provides a complete summary of the multilevel dynamic factor model

Usage

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

Arguments

- object An object of class mldfm.
- ... Additional arguments (ignored).

Value

The input object object, invisibly.

Index

`compute_fars`, [2](#)
`correct_outliers`, [3](#)
`create_scenario`, [4](#)

`density`, [5](#)

`mldfm`, [6](#)
`mldfm_subsampling`, [8](#)

`nl_density`, [9](#)

`plot.fars`, [10](#)
`plot.fars_density`, [10](#)
`plot.mldfm`, [11](#)
`print.fars`, [12](#)
`print.fars_density`, [12](#)
`print.mldfm`, [13](#)

`quantile_risk`, [13](#)

`summary.fars`, [14](#)
`summary.fars_density`, [14](#)
`summary.mldfm`, [15](#)