Package: hetiv (via r-universe)

October 27, 2026

Title Heteroskedasticity and Proxy-Based IV Methods

Version 0.1.3

Description Tools for identifying structural shocks using heteroskedasticity- and proxy-based instrumental variable (IV) methods in event-study settings. Supports local projection impulse response estimation of Jorda (2005) <doi:10.1257/0002828053828518>, generalised weak instrument testing of Lewis and Mertens (2025) <doi:10.1093/restud/rdaf103>, recursive heteroskedasticity-IV identification of Burri and Kaufmann (2026b) <doi:10.1016/j.econlet.2026.113091>, and Kalman-filter shock extraction of Burri and Kaufmann (2026a).

URL https://dankaufmann.github.io/hetiv/, https://github.com/dankaufmann/hetiv

BugReports https://github.com/dankaufmann/hetiv/issues

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1.0)

Roxygen list(markdown = TRUE)

Config/roxygen2/version 8.0.0

VignetteBuilder knitr

Imports dplyr, ggplot2, grid, gridExtra, ivreg, MASS, rlang, sandwich, zoo

Suggests cowplot, knitr, rmarkdown, testthat (>= 3.0.0)

Config/pak/sysreqs cmake make libicu-dev

Repository https://marcburri.r-universe.dev

Date/Publication 2026-10-27 14:32:13 UTC

RemoteUrl https://github.com/dankaufmann/hetiv

RemoteRef main

RemoteSha 5ae3b67a360e56000bba8c529905b4a4e97943f9

Contents

arrangeirf . . . 2
computeirf . . . 3
filllinear . . . 4
firstdiff . . . 5
gweakivtest . . . 5
hetiv . . . 7
kfpredict . . . 11
logdiff . . . 12
normalize . . . 13
plot2irf . . . 14
plotirf . . . 15
plotpval . . . 17
proxyiv . . . 17
recols . . . 21
simulatedata . . . 24
svariv . . . 25

Index **29**

`arrangeirf` *Arrange impulse response plots into one figure per shock*

Description

Arranges a list of IRF panels, as returned by `plotirf()` or `plot2irf()`, into a grid with one figure per shock dimension and a common x-axis label. Optionally saves each figure as a PDF.

Usage

```
arrangeirf(
  myGraphs,
  E = 1,
  N,
  nRows = 1,
  xLab = "Horizon",
  title = NULL,
  file = NA,
  figScaleW = 3.5,
  figScaleH = 2.5
)
```

Arguments

`myGraphs`	List of `ggplot` objects of length `E * N`, ordered by shock (outer loop) and then by variable (inner loop), as returned by `plotirf()` and `plot2irf()`.
`E`	Integer. Number of shock dimensions (one figure per shock). Defaults to 1

N	Integer. Number of variables (panels per figure).
nRows	Integer. Number of rows of panels in each figure. The number of columns is ceiling(N / nRows). Defaults to 1.
xLab	Character. Common x-axis label printed below the panels.
title	Character vector of length E (or 1, recycled) with a title printed above each figure, or NULL (default) for no title.
file	Character or NA. If not NA, file path stem; figure e is saved as paste0(file, "_", e, ".pdf"). If NA (default), nothing is saved.
figScaleW	Numeric. Width of one panel column in inches.
figScaleH	Numeric. Height of one panel row in inches. The saved figure has height 0.9 * nRows * figScaleH.

Value

Invisibly, a list of length E of arranged figures (gtable objects), which can be redrawn with grid::grid.draw(). Each figure is also drawn on the current graphics device as a side effect.

Examples

```
irf <- array(c(1, 0.5, 0.2, 0.1, 0, 0.3, 0.2, 0.1), dim = c(4, 2, 1))
dimnames(irf)[[1]] <- 0:3
se <- array(0.1, dim = dim(irf), dimnames = dimnames(irf))
g <- plotirf(irf, se, HTick = 1, Labels = c("Output", "Prices"))
figs <- arrangeirf(g, E = 1, N = 2, nRows = 1)


# Save to PDF (file is written as <stem>_1.pdf)
stem <- file.path(tempdir(), "irf")
arrangeirf(g, E = 1, N = 2, nRows = 1, file = stem)
```

---

computeirf *Compute impulse response functions for a VAR(P)*

Description

Recursively computes IRFs given an impact matrix and VAR coefficient array. Optionally cumulates responses for selected variables.

Usage

```
computeirf(Psi, Phi, H, cum)
```

Arguments

Psi	Impact matrix, dimension N x R.
Phi	Array of VAR coefficient matrices, dimension N x N x P.
H	Horizon (number of periods) for which to compute IRFs.
cum	Logical scalar or logical vector of length N. If TRUE for variable i, the IRF is cumulated via cumsum. A single value is applied to all variables.

Value

An array:

irf Array of impulse responses, dimension H x N x R. Row names are labelled 0 to H-1.

Examples

```
Psi <- matrix(c(1, 0.5), nrow = 2)
Phi <- array(0, dim = c(2, 2, 1))
Phi[, , 1] <- matrix(c(0.4, 0.1, 0, 0.3), 2, 2)
computeirf(Psi = Psi, Phi = Phi, H = 4, cum = FALSE)
```

---

filllinear *Linearly interpolate gaps in a data frame*

Description

Fills missing values (NA) in every column of a data frame using linear interpolation, up to a maximum of gap consecutive missing observations. Gaps longer than gap are left as NA.

Usage

```
filllinear(DF, gap)
```

Arguments

DF	A data frame or matrix with numeric columns.
gap	Integer. Maximum number of consecutive NAs to fill.

Value

The data frame DF with short gaps up to a maximum of gap consecutive missings linearly interpolated.

Examples

```
filllinear(data.frame(x = c(1, NA, 3), y = c(2, NA, 4)), gap = 1)
```

firstdiff *First-difference transformation*

Description

Computes the first difference of a numeric vector or time series, i.e. `x_t - x_{t-1}`. Gaps in dates are ignored; the difference is always taken with respect to the immediately preceding observation.

Usage

```
firstdiff(TS)
```

Arguments

`TS`	A numeric vector or time series.

Value

A numeric vector of the same length as `TS`, with `NA` in the first position.

Examples

```
firstdiff(c(1, 3, 6))
```

gweakivtest *Generalised weak instrument test*

Description

Tests for weak instruments with multiple endogenous regressors using the generalised minimum eigenvalue statistic of Lewis and Mertens (2025). The test is robust to heteroskedasticity and autocorrelation and nests the classical Stock-Yogo (2005) test as a special case. The function is a direct port from the Matlab codes by Lewis and Mertens (2025)

Usage

```
gweakivtest(
  y,
  Y,
  X,
  Z,
  cov_type = "EHW",
  alfa = 0.05,
  tau = 0.1,
  points = 1000L,
```

```
  target = "beta",
  crit = "abs",
  seed = 12345L
)
```

Arguments

y	Regressand (T x 1 numeric vector or matrix).
Y	Endogenous regressors (T x N numeric matrix).
X	Exogenous regressors (T x Nx numeric matrix). A constant column is added automatically if one is absent or if X has zero columns.
Z	Instruments (T x K numeric matrix). Requires K >= N.
cov_type	HAR covariance estimator: "EHW" (Eicker-Huber-White, default) or "NW" (Newey-West with Lazarus et al. (2018) bandwidth).
alfa	Nominal significance level (default 0.05).
tau	Bias tolerance: maximum acceptable relative (or absolute, see crit) bias of the 2SLS estimator (default 0.10).
points	Number of random starting points for the Stiefel manifold optimisation used to compute the sharp critical value when K > N + 1 (default 1000).
target	Either "beta" (default) to test the full coefficient vector, or a positive integer j <= N to target the single coefficient beta_j.
crit	Bias criterion: "abs" (absolute bias, default) or "rel" (relative bias). "abs" requires the error covariance matrix; "rel" does not.
seed	Integer seed used for the random Stiefel starting points when computing the sharp critical value. The default is fixed for reproducible critical values and the caller's RNG state is restored afterwards. Use NA to leave the RNG unmanaged.

Value

A named list with the following elements:

nobs Number of complete observations used.

beta_2SLS 2SLS point estimate(s).

target Description of the targeted parameter.

criterion Bias criterion used ("abs" or "rel").

gmin_generalized Generalised minimum eigenvalue test statistic (Lewis-Mertens).

gmin_generalized_critical_value Sharp critical value via Stiefel optimisation.

gmin_generalized_critical_value_simplified Conservative simplified critical value (closed-form bound).

stock_yogo_test_statistic Stock-Yogo test statistic under the Nagar approximation.

stock_yogo_critical_value_nagar Stock-Yogo critical value under the Nagar approximation.

References

Lazarus, E., Lewis, D. J., Stock, J. H. and Watson, M. W. (2018). HAR inference: recommendations for practice. *Journal of Business & Economic Statistics*, 36(4), 541-559.

Lewis, D. J. and Mertens, K. (2025). A robust test for weak instruments for 2SLS with multiple endogenous regressors. *Review of Economic Studies*, DOI: 10.1093/restud/rdaf103.

Stock, J. H. and Yogo, M. (2005). Testing for weak instruments in linear IV regression. In D. W. K. Andrews and J. H. Stock (Eds.), *Identification and inference for econometric models: essays in honor of Thomas Rothenberg*, pp. 80-108. Cambridge University Press.

Examples

```
set.seed(1)
n <- 80
Z <- matrix(rnorm(n), ncol = 1)
X <- matrix(1, n, 1)
Y <- 0.8 * Z + rnorm(n)
y <- 1.5 * Y + rnorm(n)
gweakivtest(y = y, Y = Y, X = X, Z = Z, points = 1)
```

`hetiv` *Estimate impulse responses via heteroskedasticity-based IV local projections*

Description

Estimates impulse response functions (IRFs) using recursive heteroskedasticity-IV identification (Rigobon, 2003; Rigobon and Sack, 2004; Lewis, 2022; Burri and Kaufmann, 2026a, 2026b) combined with local projections (Jorda, 2005). Identification exploits the difference in variance between policy event days and control days to construct instruments for the endogenous variables. It also returns weak instrument tests as well as shock predictions on event days.

Usage

```
hetiv(
  y,
  O,
  X = NULL,
  Ind,
  P,
  H,
  E = 1,
  norm = 1,
  interact = FALSE,
  cum = FALSE,
  cov_type = "HC3",
  hpredict = 0,
```

```
  pooled = FALSE,
  details = FALSE
)
```

Arguments

y	Numeric matrix of outcome variables (T x N). The effect on the first variable in each dimension is normalized to unity at horizon 0. These variables are also used to construct heteroskedasticity-based instruments and for recursive ordering to identify multiple dimensions.
O	Numeric matrix of information set variables (T x M). Should include at least y in case of non-stationary variables. Included as lags 1 through P.
X	Numeric matrix of deterministic variables (T x K). For example, time trend, seasonal dummies or other deterministic controls. Included as is (no lags). A constant is included by default.
Ind	Integer vector of length T, event indicator: • 0 Control day (no event) • 1 Policy day (event) • 2 Contaminated control day (excluded from estimation)
P	Integer. Maximum lag order for the information set. Set to 0 for no lags (regression on deterministic terms only).
H	Integer. Maximum horizon (in periods) up to which IRFs are estimated.
E	Integer. Number of shock dimensions to identify via recursive ordering.
norm	Numeric scalar. Normalize the impact response of the first variable to a specific value. Defaults to 1 for standard unit-effect normalization.
interact	Logical. If TRUE, lagged information set variables are interacted with event/non-event dummies.
cum	Logical vector of length N. For each variable in y, whether to report the cumulative impulse response instead of the level response. If only one provided, applied to all impulse responses.
cov_type	Covariance estimator for local-projection standard errors: "HC3" (default) for heteroskedasticity-robust standard errors or "NW" for Newey-West HAC standard errors. "HC3" is the default because Montiel Olea et al. (2025) show that heteroskedasticity-robust standard errors suffice for local-projection impulse responses under weak conditions, even though multi-step forecast errors are typically serially correlated. "NW" remains available as an optional HAC robustness check.
hpredict	Integer. Horizon of the responses and forecast errors used for shock prediction, on the same scale as the row labels of irf. Defaults to 0 (recommended; impact response and one-step-ahead residual are used) Must lie between 0 and H - 1.
pooled	Logical. If FALSE (default), the shock prediction uses the forecast errors and responses at horizon hpredict only. If TRUE, the forecast errors of all horizons 0, ..., hpredict are stacked and used jointly (pooled prediction); see kfpredict(). With hpredict = 0 both options coincide.

`details` Logical. If `TRUE`, saves detailed IV results, weak instrument tests, and shock prediction results, which is slightly slower. If `FALSE`, returns only impulse responses and standard errors.

Details

For `E > 1`, identification is recursive and order-dependent: the column order of `y` defines both the shock ordering and the normalization variable for each shock dimension.

Value

A named list. Always contains:

`irf` Array (H x N x E) of estimated impulse responses.

`se` Array (H x N x E) of local-projection standard errors.

`Method` Character string `"Heteroskedasticity-IV"`.

With `details = TRUE`, additionally contains:

`irf` Array (H x N x E) of estimated impulse responses.

`se` Array (H x N x E) of local-projection standard errors.

`Shocks` Predicted structural shocks on event days from `kfpredict()`, based on the responses at horizon `hpredict`.

`IVRes` List of `ivreg` model objects, one per horizon, variable, and shock dimension.

`OLSRes` List of the information-set regressions whose residuals enter the shock prediction, named `OLS.n<i>.h<h>` by outcome variable and horizon.

`ORTHRes` List of OLS models used to orthogonalize the variable underlying the instrument, one per shock dimension.

`FirstStage` For `E = 1`: data frame with one row per horizon `h` giving the number of observations `Tobs`, the homoskedastic first-stage F statistic `F_hom` (the squared t statistic of the proxy; Stock and Yogo, 2005) and the robust effective F statistic `F_rob` of Montiel Olea and Pflueger (2013), always computed with the HAC (Newey-West) covariance estimator, irrespective of `cov_type`. The statistics do not depend on the outcome variable. `NULL` for `E > 1`.

`WeakTest` For `E > 1`: data frame with one row per shock dimension `target` (`e = 1, ..., E`), horizon `h` and outcome variable `i`, giving the results of `gweakivtest()` for the system with the first `e` policy variables as endogenous regressors and the first `e` instruments as instruments, targeting the coefficient on policy variable `e`: `Tobs`, `gmin` (generalized minimum eigenvalue statistic, which depends on the horizon only through the sample), `cv` and `cv_simplified` (sharp and simplified critical values, which may depend on the outcome and horizon), and the Stock-Yogo statistic `sy_stat` with its critical value `sy_cv` (`NaN` unless over-identified). The test always uses the HAC (Newey-West) covariance estimator, irrespective of `cov_type`, and the relative bias criterion (`crit = "rel"`), for comparability with Montiel Olea and Pflueger (2013). Rows where the outcome is a policy variable (`i <= E`) on impact are omitted. `NULL` for `E = 1`.

`Obs` Data frame with observation counts: `Tp` (policy days), `Tc` (control days), `To` (contaminated days), `Tt` (total used).

`Method` Character string `"Heteroskedasticity-IV"`.

`et` Data frame of forecast errors on event days used for shock prediction: residuals of the regression of `y_{t+h}` on the information set dated `t - 1`, one column per variable and horizon (`y1.h0`, ..., `yN.h0`, `y1.h1`, ...). One horizon (`hpredict`) if `pooled = FALSE`, horizons `0, ..., hpredict` if `pooled = TRUE`.

`Sig` Covariance matrix of `et` on event days.

`SigR` Covariance matrix of the same residuals on control days, or `NA` if unavailable. A control day enters horizon h only if no event occurs in its window `t + 1, ..., t + h`.

`Psi` Response matrix used for shock prediction, with rows matching the columns of `et`: `irf[hpredict + 1, , ]` (N x E) if `pooled = FALSE`, the rows `1, ..., hpredict + 1` of `irf` stacked (N(hpredict + 1) x E) if `pooled = TRUE`. With the default `hpredict = 0` this is the impact response.

`WeakData` Data frame of endogenous variables and instruments for the Lewis-Mertens (2025) weak instrument test.

References

Burri, M. and D. Kaufmann (2026a). Measuring monetary policy shocks. IRENE Working Papers 24-03, IRENE Institute of Economic Research, University of Neuchatel.

Burri, M. and D. Kaufmann (2026b). Multiple monetary policy shocks from daily data: A heteroskedasticity IV approach. *Economics Letters*, 268, 113091. doi:10.1016/j.econlet.2026.113091

Jorda, O. (2005). Estimation and inference of impulse responses by local projections. *American Economic Review*, 95(1), 161-182.

Lewis, D. J. (2022). Robust inference in models identified via heteroskedasticity. *Review of Economics and Statistics*, 104(3), 510-524.

Lewis, D. J. and Mertens, K. (2025). A robust test for weak instruments for 2SLS with multiple endogenous regressors. *Review of Economic Studies*, DOI: 10.1093/restud/rdaf103

Montiel Olea, J. L., M. Plagborg-Moller, E. Qian, and C. K. Wolf (2025). Local projections or VARs? A primer for macroeconomists. *NBER Working Paper* No. 33871.

Rigobon, R. (2003). Identification through heteroskedasticity. *Review of Economics and Statistics*, 85(4), 777-792.

Rigobon, R. and Sack, B. (2004). The impact of monetary policy on asset prices. *Journal of Monetary Economics*, 51(8), 1553-1575.

Examples

```
set.seed(1)
y <- matrix(rnorm(80), ncol = 2)
Ind <- rep(0L, nrow(y))
Ind[seq(5, nrow(y), by = 5)] <- 1L
res <- hetiv(y = y, O = y, Ind = Ind, P = 1, H = 3, details = TRUE)
dim(res$irf)
```

`kfpredict` *Extract monetary policy shocks via Kalman filter*

Description

Extracts structural shocks from reduced-form residuals using the Kalman filter formula. Given the response matrix `Psi` and the covariance matrices of residuals on event and control days, the function recovers the latent shocks for each observation. Optionally rescales shocks to unit variance.

Usage

```
kfpredict(Sig, SigR, Psi, et, tol = sqrt(.Machine$double.eps), scale = TRUE)
```

Arguments

`Sig`	Numeric matrix (M x M, with M = N for one horizon). Covariance matrix of the residuals in `et` on policy event days (e.g. `Sig` from `hetiv()`).
`SigR`	Numeric matrix (M x M). Covariance matrix of the residuals on control (non-event) days. Used to back out the implied shock variances when `scale = TRUE`.
`Psi`	Numeric matrix (M x E) of responses of the residuals in `et` to the E structural shocks (e.g. `Psi` from `hetiv()`), or an array N x E x M of responses at M horizons, which is stacked by rows.
`et`	Numeric matrix or data frame (T x M) of forecast errors on event days (e.g. `et` from `hetiv()`), or an array T x N x M of forecast errors at M horizons, which is stacked by columns. Rows correspond to time periods.
`tol`	Numeric scalar. Tolerance for the generalized inverse (`MASS::ginv()`). Clipped from below at `sqrt(.Machine$double.eps)`.
`scale`	Logical. If `TRUE` (default), shocks are rescaled to unit variance using the implied shock variances recovered from `Sig` and `SigR`. This scaling assumes that structural shock variances are diagonal, so that `Sig - SigR = sum_i sigma_i Psi_i Psi_i'`. If `FALSE`, the raw Kalman filter projection is returned.

Details

The prediction is the linear projection of the shocks on the residuals, `eps_t = Sig_eps Psi' Sig^{-1} e_t`, up to the scaling described under `scale`. It applies to any set of forecast errors that is linear in the shocks:

- One horizon: `Psi` is the N x E response at horizon h and `et` holds the T x N forecast errors of `y_{t+h}` given information dated `t - 1`. With h = 0 these are the impact responses and the one-step residuals.
- Pooled horizons: the forecast errors of several horizons are stacked, `Psi` has one block of N rows per horizon and `et` one block of N columns per horizon, in the same order. The pooled prediction weights every horizon by its signal-to-noise ratio and accounts for the correlation of the forecast errors across horizons. It has a weakly smaller mean squared error than the prediction from any single horizon, and coincides with the impact prediction when all responses are complete on impact.

For convenience, `Psi` may be supplied as an array N x E x M and `et` as an array T x N x M (one slice per horizon); both are then stacked internally. `Sig` and `SigR` must be of dimension N*M in that case.

The pooled prediction requires a nonsingular covariance of the stacked forecast errors. A sufficient condition is that at least as many non-policy shocks as variables drive the data (the one-day non-policy covariance has full rank). If fewer shocks are present, as in a stylised simulation with fewer regular shocks than variables, `Sig` is singular for stacked horizons, and the generalized inverse yields a regularized prediction that is sensitive to estimation error in `Psi`. A warning is issued when `Sig` is (nearly) singular, including when the number of complete rows of `et` does not exceed its number of columns.

Value

A numeric matrix (T x E) of extracted structural shocks, with the same number of rows as `et` and one column per shock dimension. Rows with missing residuals give missing shocks.

References

Burri, M. and Kaufmann, D. (2026a). Measuring monetary policy shocks. IRENE Working Papers 24-03, IRENE Institute of Economic Research, University of Neuchatel.

Examples

```
Sig <- diag(2)
SigR <- diag(c(0.5, 0.5))
Psi <- matrix(c(1, 0.5), nrow = 2)
et <- matrix(rnorm(20), ncol = 2)
kfpredict(Sig = Sig, SigR = SigR, Psi = Psi, et = et)

# Pooled prediction from two horizons (arrays with one slice per horizon)
Psi2 <- array(c(Psi, 0.5 * Psi), dim = c(2, 1, 2))
et2 <- array(rnorm(40), dim = c(10, 2, 2))
et2m <- matrix(et2, nrow = 10)
kfpredict(Sig = var(et2m), SigR = 0.5 * var(et2m), Psi = Psi2, et = et2)
```

---

logdiff *Log-difference transformation*

Description

Computes the first log-difference of a numeric vector or time series, i.e. `log(x_t) - log(x_{t-1})`. Gaps in dates are ignored; the difference is always taken with respect to the immediately preceding observation.

Usage

```
logdiff(TS)
```

Arguments

TS	A numeric vector or time series.

Value

A numeric vector of the same length as TS, with NA in the first position.

Examples

```
logdiff(c(1, 2, 4))
```

---

normalize *Normalize a time series*

Description

Standardizes a numeric vector to zero mean and unit standard deviation (z-score normalization), ignoring missing values.

Usage

```
normalize(x)
```

Arguments

x	A numeric vector or time series.

Value

A numeric vector of the same length as x, with mean 0 and standard deviation 1.

Examples

```
normalize(c(1, 2, 3, 4, 5))
```

`plot2irf` *Plot and compare impulse responses from two estimation approaches*

Description

Produces a panel of IRF plots comparing two sets of impulse responses with normal confidence bands at level `ci`. This function is kept for backward compatibility and is a thin wrapper around `plotirf()`, which accepts up to three sets of impulse responses as well as bootstrap draws.

Usage

```
plot2irf(IRF1, IRF1se, IRF2, IRF2se, HTick, Labels, ci = 0.9)
```

Arguments

`IRF1`	Array (H x N x E) of impulse responses for the first approach. Dimnames on the first dimension must be the horizon labels (numeric).
`IRF1se`	Array (H x N x E) of standard errors for the first approach (set to 0 to suppress confidence intervals).
`IRF2`	Array (H x N x E) of impulse responses for the second approach.
`IRF2se`	Array (H x N x E) of standard errors for the second approach (set to 0 to suppress confidence intervals).
`HTick`	Integer. Step size for x-axis tick marks.
`Labels`	Character vector of length N. Variable names used as panel titles.
`ci`	Numeric scalar in (0, 1). Confidence level for the shaded bands. Defaults to `0.90` (90% CI).

Value

A list of `ggplot` objects, one per variable-shock combination, ordered by shock (outer loop) then variable (inner loop).

See Also

`plotirf()`

Examples

```
irf1 <- array(c(1, 0.5, 0.2, 0.1), dim = c(4, 1, 1))
dimnames(irf1)[[1]] <- 0:3
irf2 <- irf1 * 0.8
se <- array(0.1, dim = dim(irf1), dimnames = dimnames(irf1))
plot2irf(irf1, se, irf2, se * 0, HTick = 1, Labels = "Output")
```

plotirf *Plot impulse responses from one or several estimation approaches*

Description

Produces a panel of IRF plots, one per variable-shock combination, for up to three sets of impulse responses drawn in the same panels. Returns a list of ggplot objects that can be arranged with arrangeirf() (or e.g. cowplot::plot_grid()).

Usage

```
plotirf(
  IRF,
  IRFse = NULL,
  HTick,
  Labels,
  ci = 0.9,
  names = NULL,
  colors = c("steelblue", "darkred", "darkgreen"),
  boot_type = c("recentred", "percentile"),
  legend = FALSE
)
```

Arguments

IRF	Impulse responses to plot. One of • a numeric array (H x N x E, or H x N for a single shock) of point estimates, with numeric horizon labels in dimnames()[[1]]; • an estimator result as returned by hetiv(), proxyiv(), recols() or svariv(), i.e. a list with elements irf and optionally se, boot (array H x N x E x B of bootstrap draws) and Method; • an (unnamed) list of up to three such objects, which are overlaid in the same panels.
IRFse	Array of standard errors with the same dimensions as IRF, or NULL. Only used when IRF is a numeric array (kept for backward compatibility). For list inputs, supply the standard errors as element se of each list.
HTick	Integer. Step size for x-axis tick marks.
Labels	Character vector of length N. Variable names used as panel titles.
ci	Numeric vector of one or two confidence levels in (0, 1). One shaded band is drawn per level, with wider bands rendered at lower opacity. Defaults to 0.90 (a single 90% band); supply two levels, e.g. c(0.90, 0.95), to draw two bands.
names	Character vector with one name per set of impulse responses, used in the legend. Defaults to the Method element of each result if available and to "IRF 1", "IRF 2", ... otherwise.

colors	Character vector of colours, one per set of impulse responses. Defaults to c("steelblue", "darkred", "darkgreen").
boot_type	Character. How bootstrap draws are turned into bands: "recentred" (default) or "percentile". See Details. Ignored when no bootstrap draws are supplied.
legend	Logical. If TRUE, a legend with names is added to each panel when more than one set of impulse responses is plotted. Default FALSE.

Details

Each set of impulse responses is plotted in one of three ways, depending on what is supplied:

- Point estimate only: a line without confidence bands.
- Point estimate with standard errors: shaded normal bands irf +/- qnorm((1 + ci) / 2) * se for up to two confidence levels.
- Point estimate with bootstrap draws: shaded percentile bands for up to two confidence levels. With boot_type = "recentred" (the default) the quantiles are computed from the deviations of the draws from their bootstrap mean and added to the point estimate, so the (possibly asymmetric) band is centred on the reported point estimate. With boot_type = "percentile" the raw quantiles of the draws are used; such bands can be off-centre when the bootstrap distribution is biased relative to the point estimate.

Bootstrap draws take precedence over standard errors if both are present (as for svariv() with nboot > 0). Standard errors that are all zero suppress the bands.

Value

A list of ggplot objects, one per variable-shock combination, ordered by shock (outer loop) then variable (inner loop).

Examples

```
irf <- array(c(1, 0.5, 0.2, 0.1), dim = c(4, 1, 1))
dimnames(irf)[[1]] <- 0:3
se <- array(0.1, dim = dim(irf), dimnames = dimnames(irf))

# One IRF with normal bands at two confidence levels
plotirf(irf, se, HTick = 1, Labels = "Output")

# Two IRFs: the second without bands, the first with bootstrap draws
set.seed(1)
boot <- array(irf, dim = c(dim(irf), 200)) + rnorm(4 * 200, sd = 0.1)
plotirf(
  list(list(irf = irf, boot = boot), list(irf = irf * 0.8)),
  HTick = 1, Labels = "Output", ci = 0.90,
  names = c("Bootstrap", "Alternative"), legend = TRUE
)
```

plotpval *Plot p-values across horizons*

Description

Produces a panel of p-value plots across IRF horizons, with optional horizontal reference lines at user-supplied significance levels. Returns a list of `ggplot` objects (one per variable-shock combination) that can be arranged with e.g. `cowplot::plot_grid()`.

Usage

```
plotpval(pvals, HTick, Labels, sigLevels = c(0.05, 0.1))
```

Arguments

`pvals`	Array (H x N x E) of p-values. Dimnames on the first dimension must be the horizon labels (numeric).
`HTick`	Integer. Step size for x-axis tick marks.
`Labels`	Character vector of length N. Variable names used as panel titles.
`sigLevels`	Numeric vector of significance levels at which to draw horizontal reference lines. The first value is drawn as a solid line, the second as dashed, and any further values as dotted. Defaults to `c(0.05, 0.10)`.

Value

A list of `ggplot` objects, one per variable-shock combination, ordered by shock (outer loop) then variable (inner loop).

Examples

```
pvals <- array(c(0.2, 0.08, 0.04), dim = c(3, 1, 1))
dimnames(pvals)[[1]] <- 0:2
plotpval(pvals, HTick = 1, Labels = "Output")
```

proxyiv *Estimate impulse responses via proxy-IV local projections*

Description

Estimates impulse response functions (IRFs) using external instruments (proxies) combined with local projections (Jorda, 2005, Stock and Watson, 2018). The proxy variables serve as instruments for the endogenous variables. Optionally imposes recursive zero restrictions across shock dimensions. It also returns weak instrument tests as well as shock predictions on event days, and supports deterministic controls following the same conventions as `hetiv()`.

Usage

```
proxyiv(
  y,
  O,
  Z,
  X = NULL,
  Ind,
  P,
  H,
  E = 1,
  norm = 1,
  cum = FALSE,
  cov_type = "HC3",
  recursive = FALSE,
  hpredict = 0,
  pooled = FALSE,
  details = FALSE
)
```

Arguments

y	Numeric matrix of outcome variables (T x N). The effect on the first variable in each dimension is normalized to norm at horizon 0. These variables are also used as the endogenous regressors instrumented by the columns of Z.
O	Numeric matrix of information set variables (T x M). For non-stationary variables y, it should at least include y. Included as lags 1 through P.
Z	Numeric matrix of external instruments (T x E). Column e is used as the proxy for shock dimension e for the corresponding endogenous variable. Following the proxy-SVAR literature (Mertens and Ravn, 2013), the recommended convention is to set the proxy to zero on control days (Ind == 0). These days then remain in the IV sample and sharpen the estimates of the control coefficients, while the identifying moments are unaffected. Alternatively, missing values control days are treated as contaminated and excluded from estimation, which reproduces the event-day sample of the event-study literature (Nakamura and Steinsson, 2018). With zeros, always prefer the robust first-stage statistics (F_rob, WeakTest) over the homoskedastic F, which treats the zero observations as informative. Missing values on policy days (Ind == 1) remain labelled as events for residual-based shock prediction, but are necessarily dropped from IV regressions that use the missing proxy.
X	Numeric matrix of deterministic variables (T x K), or NULL (default). May include a constant, time trend, or seasonal dummies. Included as-is (no lags).
Ind	Integer vector of length T, event indicator: • 0 Control day (no event) • 1 Policy day (event) • 2 Contaminated control day (excluded from estimation)
P	Integer. Maximum lag order for the information set. Set to 0 for no lags (regression on constant only).

H	Integer. Maximum horizon (in periods) up to which IRFs are estimated.
E	Integer. Number of shock dimensions to identify. Default 1. Needs to be equal to the number of columns in Z
norm	Numeric scalar. Normalize the impact response of the first variable to this value. Defaults to 1 for standard unit-effect normalization.
cum	Logical scalar or vector of length N. If TRUE for variable i, the cumulative IRF is reported. A single value is applied to all variables. Default FALSE.
cov_type	Covariance estimator for local-projection standard errors: "HC3" (default) for heteroskedasticity-robust standard errors or "NW" for Newey-West HAC standard errors. "HC3" is the default because Montiel Olea et al. (2025) show that heteroskedasticity-robust standard errors suffice for local-projection impulse responses under weak conditions, even though multi-step forecast errors are typically serially correlated. "NW" remains available as an optional HAC robustness check.
recursive	Logical. If TRUE, imposes recursive zero restrictions across shock dimensions: for shock e > 1, the variables and instruments from dimensions 1, ..., e-1 are added as controls. This implies that the shock identified for dimension e has no impact effect on the endogenous variables from dimensions 1, ..., e-1. Default FALSE.
hpredict	Integer. Horizon of the responses and forecast errors used for shock prediction, on the same scale as the row labels of irf. Defaults to 0 (recommended; the impact response and one-step-ahead residual are used). Must lie between 0 and H - 1.
pooled	Logical. If FALSE (default), the shock prediction uses the forecast errors and responses at horizon hpredict only. If TRUE, the forecast errors of all horizons 0, ..., hpredict are stacked and used jointly (pooled prediction); see kfpredict(). With hpredict = 0 both options coincide.
details	Logical. If TRUE, saves detailed IV results, weak instrument tests, and shock prediction results, which is slightly slower. If FALSE, returns only impulse responses and standard errors.

Details

For E > 1, identification can be order-dependent: the column order of y defines the endogenous shock variables and normalizations, the column order of Z assigns proxies to shock dimensions, and recursive = TRUE imposes zero restrictions in that order. If recursive = FALSE the instruments are assumed to be valid across dimensions.

Value

A named list. Always contains:

irf Array (H x N x E) of estimated impulse responses.

se Array (H x N x E) of local-projection standard errors.

Method Character string "Proxy-IV".

With details = TRUE, additionally contains:

`Shocks` Predicted structural shocks on event days from `kfpredict()`, based on the responses at horizon `hpredict`.

`IVRes` List of `ivreg` model objects, one per horizon, variable, and shock dimension.

`OLSRes` List of the information-set regressions whose residuals enter the shock prediction, named `OLS.n<i>.h<h>` by outcome variable and horizon.

`Obs` Data frame with observation counts: `Tp` (policy days), `Tc` (control days), `To` (contaminated days), `Tt` (labelled event or control days), and `Tiv` (complete observations available to the IV regression of the first shock at the impact horizon).

`FirstStage` For `E = 1`: data frame with one row per horizon `h` giving the number of observations `Tobs`, the homoskedastic first-stage F statistic `F_hom` (the squared t statistic of the proxy; Stock and Yogo, 2005) and the robust effective F statistic `F_rob` of Montiel Olea and Pflueger (2013), always computed with the HAC (Newey-West) covariance estimator, irrespective of `cov_type`. The statistics do not depend on the outcome variable. `NULL` for `E > 1`.

`WeakTest` For `E > 1`: data frame with one row per shock dimension `target` (`e = 1, ..., E`), horizon `h` and outcome variable `i`, giving the results of `gweakivtest()` for the system with the first `e` policy variables as endogenous regressors and the first `e` proxies as instruments, targeting the coefficient on policy variable `e`: `Tobs`, `gmin` (generalized minimum eigenvalue statistic, which depends on the horizon only through the sample), `cv` and `cv_simplified` (sharp and simplified critical values, which may depend on the outcome and horizon), and the Stock-Yogo statistic `sy_stat` with its critical value `sy_cv` (`NaN` unless over-identified). The test always uses the HAC (Newey-West) covariance estimator, irrespective of `cov_type`, and the relative bias criterion (`crit = "rel"`), for comparability with Montiel Olea and Pflueger (2013). Rows where the outcome is a policy variable (`i <= E`) on impact are omitted. `NULL` for `E = 1`.

`et` Data frame of forecast errors on event days used for shock prediction, one column per variable and horizon (`y1.h0`, ..., `yN.h0`, `y1.h1`, ...): horizon `hpredict` only if `pooled = FALSE`, horizons `0, ..., hpredict` if `pooled = TRUE`.

`Sig` Covariance matrix of `et` on event days.

`SigR` Covariance matrix of the same residuals on control days, or `NA` if unavailable. A control day enters horizon h only if no event occurs in its window `t + 1, ..., t + h`.

`Psi` Response matrix used for shock prediction, with rows matching the columns of `et`: `irf[hpredict + 1, , ]` (N x E) if `pooled = FALSE`, the rows `1, ..., hpredict + 1` of `irf` stacked if `pooled = TRUE`. With the default `hpredict = 0` this is the impact response.

`WeakData` Data frame of endogenous variables and instruments for the Lewis-Mertens (2025) weak instrument test.

References

Jorda, O. (2005). Estimation and inference of impulse responses by local projections. *American Economic Review*, 95(1), 161-182.

Lewis, D. J. and Mertens, K. (2025). A robust test for weak instruments for 2SLS with multiple endogenous regressors. *Review of Economic Studies*, DOI: 10.1093/restud/rdaf103

Mertens, K. and Ravn, M. O. (2013). The dynamic effects of personal and corporate income tax changes in the United States. *American Economic Review*, 103(4), 1212-1247.

Montiel Olea, J. L., M. Plagborg-Moller, E. Qian, and C. K. Wolf (2025). Local projections or VARs? A primer for macroeconomists. *NBER Working Paper* No. 33871.

Nakamura, E. and Steinsson, J. (2018). High-frequency identification of monetary non-neutrality: The information effect. *Quarterly Journal of Economics*, 133(3), 1283-1330.

Stock, J. H. and Watson, M. W. (2018). Identification and estimation of dynamic causal effects in macroeconomics using external instruments. *Economic Journal*, 128(610), 917-948.

Examples

```
set.seed(1)
y <- matrix(rnorm(80), ncol = 2)
Ind <- rep(0L, nrow(y))
Ind[seq(5, nrow(y), by = 5)] <- 1L
Z <- matrix(Ind * y[, 1] + rnorm(nrow(y)), ncol = 1)
res <- proxyiv(
  y = y, O = y, Z = Z, Ind = Ind, P = 1, H = 3,
  details = TRUE
)
dim(res$irf)
```

`recols` *Estimate impulse responses via recursive (Cholesky) local projections*

Description

Estimates impulse response functions (IRFs) using recursive zero restrictions combined with local projections (Jorda, 2005). The shock of dimension `e` is identified by regressing the outcome at horizon `h` by OLS on the contemporaneous value of the `e`-th variable in `y`, controlling for the contemporaneous values of variables `1, ..., e - 1` in `y`, lags of the information set, and deterministic terms. This is the local-projection analogue of a Cholesky decomposition and serves as a benchmark for the heteroskedasticity-based estimator `hetiv()`.

Usage

```
recols(
  y,
  O,
  X = NULL,
  Ind,
  P,
  H,
  E = 1,
  norm = 1,
  cum = FALSE,
  cov_type = "HC3",
  hpredict = 0,
```

```
  pooled = FALSE,
  details = FALSE
)
```

Arguments

y	Numeric matrix of stationary outcome variables (T x N). The effect on the e-th variable in each dimension e is normalized to norm at horizon 0. The first E variables are also used to impose the recursive zero restrictions.
O	Numeric matrix of information set variables (T x M). May be identical to y. Included as lags 1 through P.
X	Numeric matrix of deterministic variables (T x K). For example, time trend, seasonal dummies or other deterministic controls. Included as is (no lags). A constant is included by default.
Ind	Integer vector of length T, event indicator: • 0 Control day (no event) • 1 Policy day (event) • 2 Contaminated control day (excluded from estimation) At least one policy day is required.
P	Integer. Maximum lag order for the information set. Set to 0 for no lags (regression on deterministic terms only).
H	Integer. Maximum horizon (in periods) up to which IRFs are estimated.
E	Integer. Number of shock dimensions to identify via recursive ordering.
norm	Numeric scalar. Normalize the impact response of the e-th variable to shock e to a specific value. Set to 1 for standard unit-effect normalization.
cum	Logical vector of length N. For each variable in y, whether to report the cumulative impulse response instead of the level response. If only one provided, applied to all impulse responses.
cov_type	Covariance estimator for local-projection standard errors: "HC3" (default) for heteroskedasticity-robust standard errors or "NW" for Newey-West HAC standard errors. "HC3" is the default because Montiel Olea et al. (2025) show that heteroskedasticity-robust standard errors suffice for local-projection impulse responses under weak conditions, even though multi-step forecast errors are typically serially correlated. "NW" remains available as an optional HAC robustness check.
hpredict	Integer. Horizon of the responses and forecast errors used for shock prediction, on the same scale as the row labels of irf. Defaults to 0 (impact response and one-step-ahead residual). Must lie between 0 and H - 1.
pooled	Logical. If FALSE (default), the shock prediction uses the forecast errors and responses at horizon hpredict only. If TRUE, the forecast errors of all horizons 0, ..., hpredict are stacked and used jointly (pooled prediction), which weights every horizon by its signal-to-noise ratio; see kfpredict(). With hpredict = 0 both options coincide.
details	Logical. If TRUE, saves detailed OLS results and extracts shocks, which is slightly slower. If FALSE, returns only impulse responses and standard errors (e.g. for bootstrap).

Details

For `E > 1`, identification is recursive and order-dependent: the column order of `y` defines both the shock ordering and the normalization variable for each shock dimension. By construction, the impact response of variable `e` to shock `e` equals `norm`, and the impact responses of variables `1, ..., e - 1` to shock `e` are zero.

Unlike `hetiv()`, the distinction between policy days (`Ind == 1`) and control days (`Ind == 0`) does not affect the IRF estimates: both enter the local projections, and only contaminated days (`Ind == 2`) are dropped. The distinction matters only for the shock extraction returned when `details = TRUE`.

Value

A named list. If `details = FALSE`, it contains `irf`, `se`, and `Method`. If `details = TRUE`, it contains:

`irf` Array (H x N x E) of estimated impulse responses.

`se` Array (H x N x E) of local-projection standard errors.

`Shocks` Output of `kfpredict()`: predicted structural shocks.

`OLSRes` List of `lm` model objects, one per horizon, variable, and shock dimension.

`Obs` Data frame with observation counts: `Tp` (policy days), `Tc` (control days), `To` (contaminated days), `Tt` (total used).

`Method` Character string `"Recursive-OLS"`.

`et` Matrix of forecast errors on event days used for shock prediction, one column per variable and horizon (`y1.h0`, ..., `yN.h0`, `y1.h1`, ...): horizon `hpredict` only if `pooled = FALSE`, horizons `0, ..., hpredict` if `pooled = TRUE`.

`Sig` Covariance matrix of `et` on event days.

`SigR` Covariance matrix of the same residuals on control days, or `NA` if unavailable. A control day enters horizon h only if no event occurs in its window `t + 1, ..., t + h`.

`Psi` Response matrix used for shock prediction, with rows matching the columns of `et`: `irf[hpredict + 1, , ]` (N x E) if `pooled = FALSE`, the rows `1, ..., hpredict + 1` of `irf` stacked if `pooled = TRUE`. With the default `hpredict = 0` this is the impact response.

References

Jorda, O. (2005). Estimation and inference of impulse responses by local projections. *American Economic Review*, 95(1), 161-182.

Montiel Olea, J. L., M. Plagborg-Moller, E. Qian, and C. K. Wolf (2025). Local projections or VARs? A primer for macroeconomists. *NBER Working Paper* No. 33871.

Plagborg-Moller, M. and C. K. Wolf (2021). Local projections and VARs estimate the same impulse responses. *Econometrica*, 89(2), 955-980.

See Also

`hetiv()` for heteroskedasticity-based identification and `proxyiv()` for proxy-based identification.

Examples

```
set.seed(1)
y <- matrix(rnorm(80), ncol = 2)
Ind <- rep(0L, nrow(y))
Ind[seq(5, nrow(y), by = 5)] <- 1L
res <- recols(y = y, O = y, Ind = Ind, P = 1, H = 3, E = 2)
res$irf[1, , ] # impact responses: unit diagonal, zero above it
```

`simulatedata` *Simulate VAR data with heteroskedastic shocks and given parameters*

Description

Simulates data from a VAR(P) model with two types of structural shocks: regular shocks (always present) and event shocks (occurring every `Nevn` periods). The shock distribution can be normal, Student-t, or GARCH(1,1).

Usage

```
simulatedata(Phi, SigE, PsiE, PsiR, Nobs, Nbin, N, R, E, Nevn, P, eDist, seed)
```

Arguments

`Phi`	Array of VAR coefficient matrices, dimension `N x N x P`.
`SigE`	Variance of the event shocks. This applies to all E shocks.
`PsiE`	Impact matrix for event shocks, dimension `N x E`.
`PsiR`	Impact matrix for regular shocks, dimension `N x R`.
`Nobs`	Number of observations to retain after burn-in. The returned data have exactly `Nobs` rows.
`Nbin`	Number of burn-in observations. These are simulated to initialise the VAR but are discarded before returning.
`N`	Number of variables in the VAR.
`R`	Number of regular shocks.
`E`	Number of event shocks. For GARCH shock distributions (`eDist = c(alpha, beta)`), only `E = 1` is supported.
`Nevn`	Event frequency: an event shock occurs every `Nevn` periods. Set to `0` to suppress event shocks entirely (no heteroskedasticity).
`P`	VAR lag order.
`eDist`	Shock distribution. Use `0` for standard normal; a positive integer for Student-t with that many degrees of freedom; or a numeric vector `c(alpha, beta)` for GARCH(1,1) with ARCH effect `alpha` and persistence `beta`.
`seed`	Integer seed passed to `set.seed()` for reproducibility. Use `NA` to skip seeding.

Details

Unless `seed = NA`, the function calls `set.seed()` and does not restore the previous state of the random number generator. Any random draws made after the call (e.g. a simulated proxy) therefore follow deterministically from `seed`; set a new seed explicitly if independent draws are required.

Value

A named list with components:

y Simulated VAR data, dimension `Nobs x N` (burn-in discarded).

IndE Event indicator matrix, dimension `Nobs x 1`.

eR Simulated regular shocks, dimension `Nobs x R`.

eE Simulated event shocks, dimension `Nobs x E`.

e Composite structural shocks, dimension `Nobs x N`.

Phi VAR coefficient array (returned unchanged).

PsiE Event shock impact matrix (returned unchanged).

PsiR Regular shock impact matrix (returned unchanged).

SigE Event shock variance (returned unchanged).

Examples

```
N <- 2
Phi <- array(0, dim = c(N, N, 1))
Phi[, , 1] <- diag(c(0.4, 0.2))
simulatedata(
  Phi = Phi, SigE = 2, PsiE = matrix(c(1, 0.5), N, 1),
  PsiR = diag(N), Nobs = 20, Nbin = 5, N = N, R = N, E = 1,
  Nevn = 5, P = 1, eDist = 0, seed = 1
)
```

`svariv` *Estimate impulse responses via a proxy SVAR (SVAR-IV)*

Description

Estimates a reduced-form VAR(P) by OLS, identifies the impact matrix of the first `E` structural shocks with external instruments (proxies) following Mertens and Ravn (2013) and Stock and Watson (2018), and computes impulse responses by recursion through the VAR. Optionally computes confidence bands with a wild bootstrap following Mertens and Ravn (2013).

Usage

```
svariv(
  y,
  Z,
  X = NULL,
  P,
  H,
  E = 1,
  norm = 1,
  cum = FALSE,
  nboot = 0,
  ci = c(0.9, 0.95),
  cov_type = "HC0",
  details = FALSE
)
```

Arguments

y	Numeric matrix of VAR variables (T x N) without missing values. The first E columns are the policy variables on which the responses are normalized.
Z	Numeric matrix of external instruments (T x M, M >= E). Missing values are allowed and drop the corresponding observation from the identification step. Set the proxy to zero (not NA) on days without an event.
X	Numeric matrix of deterministic variables (T x K), or NULL (default). Included as-is (no lags) in addition to the constant, which is always included.
P	Integer. Lag order of the VAR (P >= 1).
H	Integer. Number of horizons: IRFs are reported for horizons 0, ..., H - 1.
E	Integer. Number of shocks to identify (E < N). Default 1.
norm	Numeric scalar. Normalize the impact response of variable e to shock e to this value. Default 1.
cum	Logical scalar or vector of length N. If TRUE for variable i, the cumulative IRF is reported. A single value is recycled to all variables. Default FALSE.
nboot	Integer. Number of wild-bootstrap replications. 0 (default) skips the bootstrap, in which case se, lower, and upper are NULL.
ci	Numeric vector of confidence levels in (0, 1) for the percentile bands. Default c(0.90, 0.95).
cov_type	Covariance estimator for the robust first-stage F statistic reported with details = TRUE: "HC0" (default, Eicker-Huber-White), "HC1", "HC3", or "NW" for Newey-West HAC.
details	Logical. If TRUE, additionally returns the VAR estimates, residuals and first-stage statistics.

Details

The first E columns of y are the policy variables: shock e is the structural shock to variable e, and the impact response of variable e to shock e is normalized to norm. For E > 1 the impact matrix

of the policy block is identified up to a lower-triangular (Cholesky) rotation, so shock `e` has no impact effect on policy variables `1, ..., e-1`; identification is therefore order-dependent, as in `proxyiv()` with `recursive = TRUE`.

All columns of `Z` are used jointly as instruments in the first-stage regressions of the policy residuals on the proxies. Rows of `Z` with missing values are excluded from the identification step only: the VAR and the reduced-form residual covariance `Sigma` are estimated on the full sample, as in Mertens and Ravn (2013), whereas the 2SLS step that recovers the relative impact responses uses only the observations with non-missing proxies (the identification sample).

With `details = TRUE`, first-stage diagnostics are reported for every policy residual: the homoskedastic F statistic of the proxies (`FStat`, equal to the squared t statistic when `M = 1`; Stock and Yogo, 2005) and the robust effective F statistic of Montiel Olea and Pflueger (2013) (`FStatRobust`), computed with the covariance estimator in `cov_type`. For `E > 1` these are equation-by-equation statistics; use `gweakivtest()` for a joint test.

The bootstrap follows Mertens and Ravn (2013): in every replication, the VAR residuals and the proxies are multiplied by the same Rademacher draw, artificial data are generated recursively from the first `P` observations, and the VAR is re-estimated and re-identified. Each draw is normalized in the same way as the point estimate. Percentile bands are reported for every level in `ci`; `se` is the bootstrap standard deviation and `boot` holds the raw draws, so the result can be passed directly to `plotirf()`, which draws recentred percentile bands from the draws. Replications in which identification fails (e.g. a non-positive-definite policy block) are dropped with a warning.

Value

A named list. Always contains:

`irf` Array (H x N x E) of impulse responses, horizons labelled `0` to `H - 1`.

`se` Array (H x N x E) of bootstrap standard deviations, or `NULL` if `nboot = 0`.

`lower`, `upper` Arrays (H x N x E x length(ci)) of percentile bands, one slice per level in `ci` (labelled in the fourth dimension), or `NULL` if `nboot = 0`.

`boot` Array (H x N x E x nboot) of bootstrap draws (`NA` for failed replications), or `NULL` if `nboot = 0`.

`Method` Character string `"SVAR-IV"`.

With `details = TRUE`, additionally contains:

`Psi` Normalized impact matrix (N x E), equal to `irf[1, , ]` before cumulation.

`Phi` Array (N x N x P) of VAR slope coefficients.

`C` Vector of length N with the VAR intercepts.

`D` Matrix (N x K) of coefficients on `X`, or `NULL`.

`U` Matrix ((T - P) x N) of reduced-form residuals.

`FStat`, `FStatRobust`, `R2` Vectors of length E with the homoskedastic first-stage F statistic, the robust effective F statistic (Montiel Olea and Pflueger, 2013) and the R-squared of the regressions of each policy residual on the proxies (see Details).

`Obs` Data frame with `T` (sample size), `Test` (VAR estimation sample), and `Tiv` (identification sample).

`nfail` Number of failed bootstrap replications.

`ci` The confidence levels used for `lower` and `upper`.

References

Mertens, K. and Ravn, M. O. (2013). The dynamic effects of personal and corporate income tax changes in the United States. *American Economic Review*, 103(4), 1212-1247.

Montiel Olea, J. L. and Pflueger, C. (2013). A robust test for weak instruments. *Journal of Business & Economic Statistics*, 31(3), 358-369.

Stock, J. H. and Watson, M. W. (2018). Identification and estimation of dynamic causal effects in macroeconomics using external instruments. *Economic Journal*, 128(610), 917-948.

Stock, J. H. and Yogo, M. (2005). Testing for weak instruments in linear IV regression. In D. W. K. Andrews and J. H. Stock (Eds.), *Identification and inference for econometric models: Essays in honor of Thomas Rothenberg* (pp. 80-108). Cambridge University Press.

See Also

`proxyiv()` for the local-projection counterpart, `computeirf()`, `plotirf()`.

Examples

```
set.seed(1)
Phi <- array(c(0.5, 0.1, -0.1, 0.4), dim = c(2, 2, 1))
Psi <- matrix(c(1, 0.5, 0, 1), 2, 2)
Nobs <- 400
eps <- matrix(rnorm(2 * Nobs), ncol = 2)
y <- matrix(0, Nobs, 2)
for (t in 2:Nobs) y[t, ] <- Phi[, , 1] %*% y[t - 1, ] + Psi %*% eps[t, ]
Z <- matrix(eps[, 1] + rnorm(Nobs), ncol = 1)
res <- svariv(y = y, Z = Z, P = 1, H = 6, nboot = 50, ci = 0.90)
dim(res$irf)
res$irf[1, , 1] # impact responses, normalized to 1 on the first variable
```

Index

arrangeirf, 2
arrangeirf(), *15*

computeirf, 3
computeirf(), *28*

filllinear, 4
firstdiff, 5

grid::grid.draw(), *3*
gweakivtest, 5
gweakivtest(), *9*, *20*, *27*

hetiv, 7
hetiv(), *11*, *15*, *17*, *21*, *23*

kfpredict, 11
kfpredict(), *8, 9*, *19, 20*, *22, 23*

logdiff, 12

MASS::ginv(), *11*

normalize, 13

plot2irf, 14
plot2irf(), *2*
plotirf, 15
plotirf(), *2*, *14*, *27, 28*
plotpval, 17
proxyiv, 17
proxyiv(), *15*, *23*, *27, 28*

recols, 21
recols(), *15*

set.seed(), *24, 25*
simulatedata, 24
svariv, 25
svariv(), *15, 16*