Title

stata.com

irf — Create and analyze IRFs, dynamic-multiplier functions, and FEVDs

Description

`irf` creates and manipulates IRF files that contain estimates of the IRFs, dynamic-multiplier functions, and forecast-error variance decompositions (FEVDs) created after estimation by `var`, `svar`, or `vec`; see [TS] **var**, [TS] **var svar**, or [TS] **vec**.

`irf` creates and manipulates IRF files that contain estimates of the IRFs created after estimation by `arima`, `arfima`, `dsge`, or `dsgenl`; see [TS] **arima**, [TS] **arfima**, [DSGE] **dsge**, or [DSGE] **dsgenl**.

IRFs and FEVDs are described below, and the process of analyzing them is outlined. After reading this entry, please see [TS] **irf create**.

Quick start

Fit a VAR model

```
var y1 y2 y3
```

Create impulse–response function `myirf` and IRF file `myirfs.irf`

```
irf create myirf, set(myirfs)
```

Graph orthogonalized impulse–response function for dependent variables `y1` and `y2` given a shock to `y1`

```
irf graph oirf, impulse(y1) response(y1 y2)
```

As above, but present results in a table

```
irf table oirf, impulse(y1) response(y1 y2)
```

Note: `irf` commands can be used after `var`, `svar`, `vec`, `arima`, `arfima`, `dsge`, or `dsgenl`; see [TS] **var**, [TS] **var svar**, [TS] **vec**, [TS] **arima**, [TS] **arfima**, [DSGE] **dsge**, or [DSGE] **dsgenl**.

See [TS] **irf add**, [TS] **irf cgraph**, [TS] **irf ctable**, [TS] **irf describe**, [TS] **irf drop**, [TS] **irf graph**, [TS] **irf ograph**, [TS] **irf rename**, [TS] **irf set**, and [TS] **irf table** for additional Quick starts.

Syntax

```
irf subcommand ... [ , ... ]
```

subcommand	Description
`create`	create IRF file containing IRFs, dynamic-multiplier functions, and FEVDs
`set`	set the active IRF file
`graph`	graph results from active file
`cgraph`	combine graphs of IRFs, dynamic-multiplier functions, and FEVDs
`ograph`	graph overlaid IRFs, dynamic-multiplier functions, and FEVDs
`table`	create tables of IRFs, dynamic-multiplier functions, and FEVDs from active file
`ctable`	combine tables of IRFs, dynamic-multiplier functions, and FEVDs
`describe`	describe contents of active file
`add`	add results from an IRF file to the active IRF file
`drop`	drop IRF results from active file
`rename`	rename IRF results within a file

IRF stands for impulse–response function; FEVD stands for forecast-error variance decomposition.

`irf` can be used after `var`, `svar`, `vec`, `arima`, `arfima`, `dsge`, or `dsgenl`; see [TS] **var**, [TS] **var svar**, [TS] **vec**, [TS] **arima**, [TS] **arfima**, [DSGE] **dsge**, or [DSGE] **dsgenl**.

Remarks and examples

stata.com

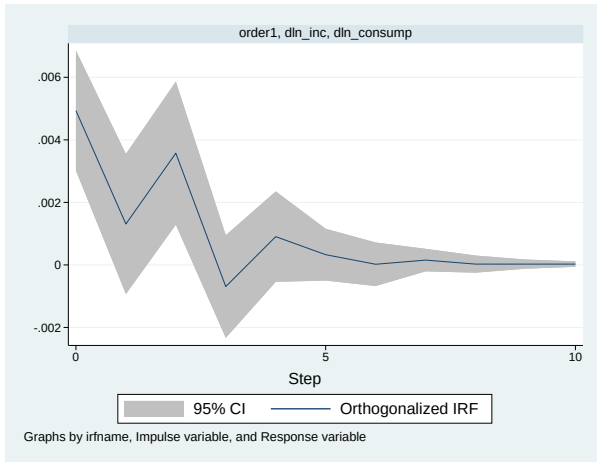
An IRF measures the effect of a shock to an endogenous variable on itself or on another endogenous variable; see Lütkepohl (2005, 51–63) and Hamilton (1994, 318–323) for formal definitions. Becketti (2020) provides an approachable, gentle introduction to IRF analysis. Of the many types of IRFs, `irf create` estimates the five most important: simple IRFs, orthogonalized IRFs, cumulative IRFs, cumulative orthogonalized IRFs, and structural IRFs.

A dynamic-multiplier function, or transfer function, measures the impact of a unit increase in an exogenous variable on the endogenous variables over time; see Lütkepohl (2005, chap. 10) for formal definitions. `irf create` estimates simple and cumulative dynamic-multiplier functions after `var`.

The forecast-error variance decomposition (FEVD) measures the fraction of the forecast-error variance of an endogenous variable that can be attributed to orthogonalized shocks to itself or to another endogenous variable; see Lütkepohl (2005, 63–66) and Hamilton (1994, 323–324) for formal definitions. Of the many types of FEVDs, `irf create` estimates the two most important: Cholesky and structural.

To analyze IRFs and FEVDs in Stata, you first fit a model, then use `irf create` to estimate the IRFs and FEVDs and save them in a file, and finally use `irf graph` or any of the other `irf` analysis commands to examine results:

```
. use https://www.stata-press.com/data/r17/lutkepohl2
(Quarterly SA West German macro data, Bil DM, from Lutkepohl 1993 Table E.1)
. var dln_inv dln_inc dln_consump if qtr<=tq(1978q4), lags(1/2) dfk
  (output omitted)
. irf create order1, step(10) set(myirf1, new)
(file myirf1.irf created)
(file myirf1.irf now active)
(file myirf1.irf updated)
. irf graph oirf, impulse(dln_inc) response(dln_consump)
```

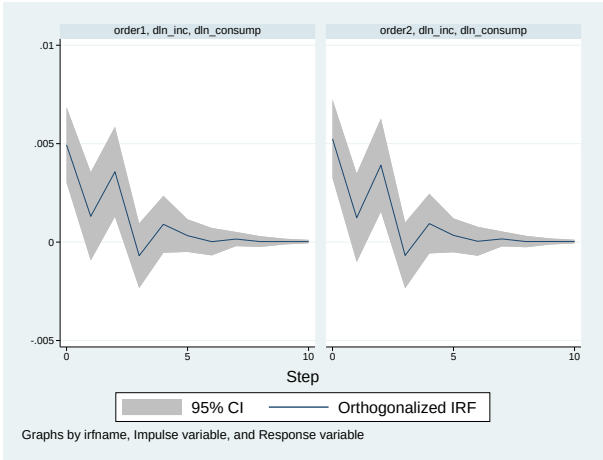


Multiple sets of IRFs and FEVDs can be placed in the same file, with each set of results in a file bearing a distinct name. The `irf create` command above created file `myirf1.irf` and put one set of results in it, named `order1`. The `order1` results include estimates of the simple IRFs, orthogonalized IRFs, cumulative IRFs, cumulative orthogonalized IRFs, and Cholesky FEVDs.

IRF files are just files: they can be erased by `erase`, listed by `dir`, and copied by `copy`; see [D] **erase**, [D] **dir**, and [D] **copy**.

Below we use the same estimated `var` but use a different Cholesky ordering to create a second set of IRF results, which we will save as `order2` in the same file, and then we will graph both results:

```
. irf create order2, step(10) order(dln_inc dln_inv dln_consump)
(file myirf1.irf updated)
. irf graph oirf, irf(order1 order2) impulse(dln_inc) response(dln_consump)
```



We have compared results for one model under two different identification schemes. We could just as well have compared results of two different models. We now use `irf table` to display the results tabularly:

```
. irf table oirf, irf(order1 order2) impulse(dln_inc) response(dln_consump)
Results from order1 and order2
```

Step	(1) oirf	(1) Lower	(1) Upper
0	.004934	.003016	.006852
1	.001309	-.000931	.003549
2	.003573	.001285	.005862
3	-.000692	-.002333	.00095
4	.000905	-.000541	.002351
5	.000328	-.0005	.001156
6	.000021	-.000675	.000717
7	.000154	-.000206	.000515
8	.000026	-.000248	.0003
9	.000026	-.000121	.000174
10	.000026	-.000061	.000113

Step	(2) oirf	(2) Lower	(2) Upper
0	.005244	.003252	.007237
1	.001235	-.001011	.003482
2	.00391	.001542	.006278
3	-.000677	-.002347	.000993
4	.00094	-.000576	.002456
5	.000341	-.000518	.001201
6	.000042	-.000693	.000777
7	.000161	-.000218	.00054
8	.000027	-.000261	.000315
9	.00003	-.000125	.000184
10	.000027	-.000065	.00012

95% lower and upper bounds reported.
(1) irfname = order1, impulse = dln_inc, and response = dln_consump.
(2) irfname = order2, impulse = dln_inc, and response = dln_consump.

Both the table and the graph show that the two orthogonalized IRFs are essentially the same. In both functions, an increase in the orthogonalized shock to `dln_inc` causes a short series of increases in `dln_consump` that dies out after four or five periods.

References

Becketti, S. 2020. *Introduction to Time Series Using Stata*. Rev. ed. College Station, TX: Stata Press.

Box-Steffensmeier, J. M., J. R. Freeman, M. P. Hitt, and J. C. W. Pevehouse. 2014. *Time Series Analysis for the Social Sciences*. New York: Cambridge University Press.

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

Lütkepohl, H. 1993. *Introduction to Multiple Time Series Analysis*. 2nd ed. New York: Springer.

——. 2005. *New Introduction to Multiple Time Series Analysis*. New York: Springer.

Rajbhandari, A. 2016. Vector autoregression–simulation, estimation, and inference in Stata. *The Stata Blog: Not Elsewhere Classified*. http://blog.stata.com/2016/02/18/vector-autoregressionsimulation-estimation-and-inference-in-stata/.

Schenck, D. 2016a. Long-run restrictions in a structural vector autoregression. *The Stata Blog: Not Elsewhere Classified*. http://blog.stata.com/2016/10/27/long-run-restrictions-in-a-structural-vector-autoregression/.

——. 2016b. Vector autoregressions in Stata. *The Stata Blog: Not Elsewhere Classified*. http://blog.stata.com/2016/08/09/vector-autoregressions-in-stata/.

Also see

[TS] **arfima** — Autoregressive fractionally integrated moving-average models

[TS] **arima** — ARIMA, ARMAX, and other dynamic regression models

[TS] **var** — Vector autoregressive models

[TS] **var intro** — Introduction to vector autoregressive models

[TS] **var svar** — Structural vector autoregressive models

[TS] **varbasic** — Fit a simple VAR and graph IRFs or FEVDs

[TS] **vec** — Vector error-correction models

[TS] **vec intro** — Introduction to vector error-correction models

[DSGE] **dsge** — Linear dynamic stochastic general equilibrium models

[DSGE] **dsgenl** — Nonlinear dynamic stochastic general equilibrium models