

lpirf postestimation — Postestimation tools for lpirf

[Postestimation commands](#)
[References](#)

[Remarks and examples](#)
[Also see](#)

[Methods and formulas](#)

Postestimation commands

The following postestimation command is of special interest after `lpirf`:

Command	Description
<code>irf</code>	create and analyze IRFs and FEVDs

The following standard postestimation commands are also available:

Command	Description
<code>estat summarize</code>	summary statistics for the estimation sample
<code>estat vce</code>	variance–covariance matrix of the estimators (VCE)
<code>estimates</code>	cataloging estimation results
<code>etable</code>	table of estimation results
<code>lincom</code>	point estimates, standard errors, testing, and inference for linear combinations of coefficients
<code>nlcom</code>	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
<code>test</code>	Wald tests of simple and composite linear hypotheses
<code>testnl</code>	Wald tests of nonlinear hypotheses

Remarks and examples

We assume that you have already read [TS] **lpirf**. In this entry, we illustrate some of the features of **irf** after using **lpirf** to estimate IRF and dynamic-multiplier functions.

The **irf** suite of commands creates and manages impulse–response functions. Results can be viewed as graphs or tables. **irf set filename** sets the file in which IRF results are stored. **irf create irfname** creates IRFs from the current estimation command. Results from multiple models may be stored in the same file so that IRFs derived from each can be compared.

► Example 1: Creating and viewing IRF results

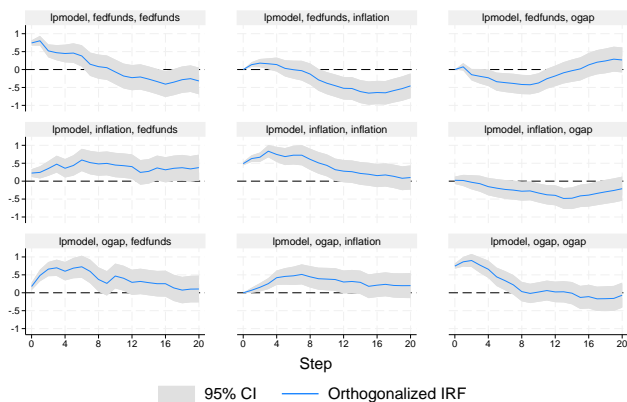
We first fit the local projections and estimate the impulse–response coefficients using **lpirf**. We name the results **lpmodel**.

```
. use https://www.stata-press.com/data/r18/usmacro
(Federal Reserve Economic Data - St. Louis Fed)
. lpirf inflation ogap fedfunds, lags(1/4) step(20)
(output omitted)
. irf set lpmodel.irf, replace
(file lpmodel.irf created)
(file lpmodel.irf now active)
. irf create lpmodel
(file lpmodel.irf updated)
```

The **lpirf** command reports the simple IRFs. The **irf** suite of commands provides both simple and orthogonalized IRFs, and they can be viewed as either a graph or a table.

Here we use the **irf graph oirf** command to produce a graph of the orthogonalized impulse–response functions. We use the **ylines(0)** option to add a horizontal reference line at 0 and use the **xlabel()** option to customize the *x*-axis label. The data are measured in quarters, so we use the **xlabel(0(4)20)** option to place a tick every four steps and indicate each year after impact.

```
. irf graph oirf, yline(0) xlabel(0(4)20)
```



Graphs by **irfname**, impulse variable, and response variable

The resulting graph produces orthogonalized impulse–response functions for each combination of impulse and response. In an OIRF, the *y* axis shows the change in levels of the response variable to a one-unit exogenous increase in the orthogonalized shock associated to the impulse variable. See [Methods and formulas](#) for more on the estimation and interpretation of OIRFs. Along the top row are the responses of the federal funds rate, inflation, and the output gap to a shock to the federal funds

rate. The responses to an inflation shock are presented in the middle row. Finally, the responses to an output gap shock are presented in the third row.

Focusing on the top row, we see in the second panel that an unexpected increase in the federal funds rate causes inflation to rise slightly in the year after the shock before declining afterward and reaching a trough about 16 steps (4 years) after the shock. The output gap falls on an interest rate shock, reaching its trough about 8 steps (2 years) after the shock. Output then recovers to trend and even rebounds above trend after 16 periods.

In the middle row, we see that the interest rate responds to the inflation shock by rising and remaining elevated even 20 steps (5 years) after the shock. The shock causes inflation itself to rise, falling back to trend after 20 steps. The inflation shock causes the output gap to decline slightly, reaching a trough about 12 steps (3 years) after the shock.

Finally, the output gap shock causes the interest rate to rise. Inflation rises in a delayed, hump-shaped manner, with inflation peaking 8 steps after the shock. The output gap rises but falls quickly back to trend within 8 steps (2 years).



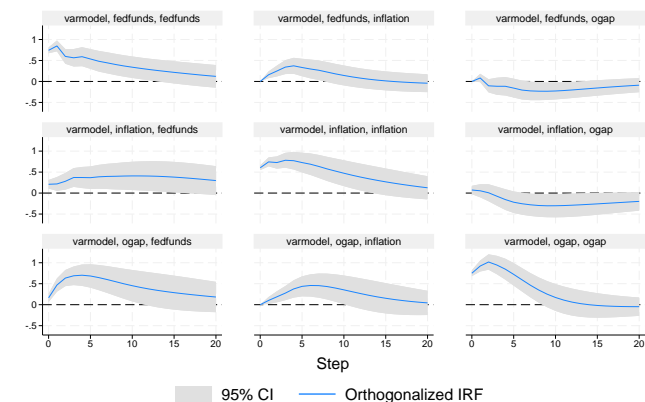
► Example 2: Comparing IRF results across models

We next compare the local-projection results with the IRF obtained when using a traditional vector autoregressive (VAR) model. We fit the VAR model with four lags and store its impulse–response functions under the name `varmodel`. We then add these results to the file `lpirf.irf`.

```
. var inflation ogap fedfunds, lags(1/4)
  (output omitted)
. irf create varmodel, step(20)
(file lpirf.irf updated)
```

We next graph the complete set of VAR IRFs:

```
. irf graph oirf, yline(0) irf(varmodel)
```

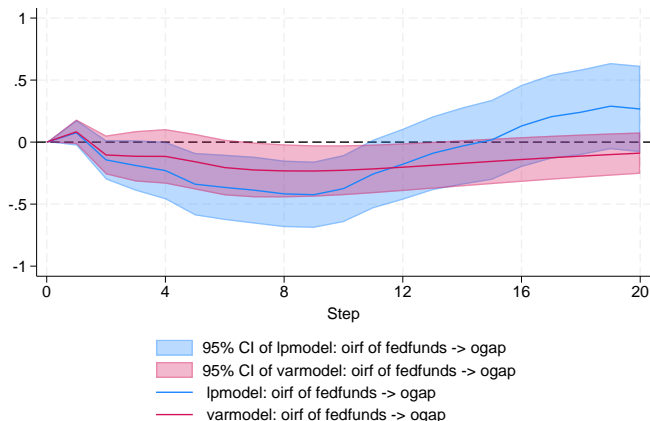


Graphs by irfname, impulse variable, and response variable

Compared with the local-projection IRFs, the VAR IRFs are smoother. In some instances, the two models show broadly similar patterns of responses. For example, the response to an output gap shock (row 3) is broadly similar across the two models. In other instances, the patterns of responses differ. For example, there are some differences in the response to a federal funds rate shock. We next use `irf ograph` to compare specific sets of results more closely.

The `irf ograph` command provides overlaid IRF results. In this example, we overlay the two models, `lpmodel` and `varmodel`, comparing the same impulse (to `fedfunds`), the same response (to `ogap`), and the same statistic (the orthogonalized IRF). We use the `ciopts(color())` option to adjust the color and transparency of the confidence intervals of the orthogonalized IRF.

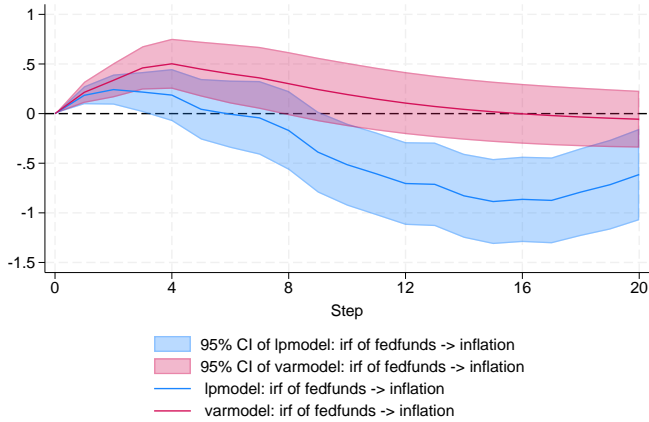
```
. irf ograph (lpmodel fedfunds ogap oirf, ci ciopts(color(stblue%30)))
> (varmodel fedfunds ogap oirf, ci ciopts(color(stred%30))),
> yline(0) xlabel(0(4)20)
```



Both models predict that the output gap will fall after an impulse to the federal funds rate. However, the local-projection model predicts a deeper decline in the output gap than the VAR model. In addition, the VAR model predicts that the output gap smoothly returns to trend, while the local-projection model predicts a rebound above trend after about 16 steps (4 years).

We also compare the response of inflation with a federal funds rate shock.

```
. irf ograph (lpmodel fedfunds inflation irf, ci ciopts(color(stblue%30)))
> (varmodel fedfunds inflation irf, ci ciopts(color(stred%30))),
> yline(0) xlabel(0(4)20)
```



Both models predict that inflation rises slightly in the steps immediately following an interest rate shock, but the VAR model predicts that inflation remains above trend throughout its response profile. By contrast, the local-projection model predicts that inflation eventually declines below trend, reaching a trough about 16 steps (4 years) after the shock, before returning to trend from below.



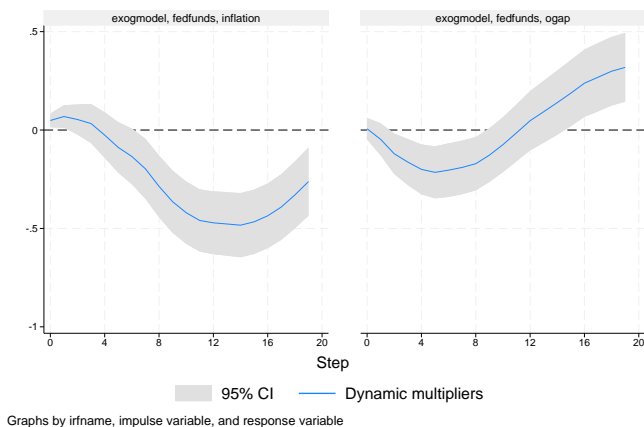
▶ Example 3: Model with exogenous shocks and dynamic multipliers

In this example, we repeat [example 3](#) of [\[TS\] lpirf](#) to fit a local-projection model of the output gap and inflation rate, treating the interest rate as exogenous. We then estimate the IRF and dynamic-multiplier function parameters and store them in the model `exogmodel`.

```
. lpirf ogap inflation, lags(1/4) step(20) exog(fedfunds)
(output omitted)
. irf create exogmodel
(file lpirf.irf updated)
```

We next graph the responses of the inflation rate and output gap to an exogenous increase in the interest rate.

```
. irf graph dm, irf(exogmodel) impulse(fedfunds) yline(0) xlabel(0(4)20)
```



The shock is a one-unit increase in the interest rate. Because the interest rate is measured in percentage points, the responses above trace out the paths of the inflation rate and output gap to a one-percentage-point increase in the interest rate. The steps on the horizontal axis correspond to the time unit of the model, in this case a quarterly frequency.

Inflation rises slightly immediately after the shock but declines in the medium run, hitting a trough of -0.5 percentage points about 12 periods after the shock. Inflation then begins to recover but is still considerably below its long-run value 20 periods after the shock. By contrast, the output gap falls initially, reaching a trough about one year after the shock. The output gap then recovers by 10 periods after the shock and even overshoots somewhat. Twenty periods after the shock, output remains above its long-run value.

The size of the rebound in the output gap is substantial. We can even test whether the rise in the output gap at the last step is equal to the trough in the output gap at its lowest point:

```
. test [fedfunds]f5.ogap = -[fedfunds]f19.ogap
( 1) [fedfunds]F5.ogap + [fedfunds]F19.ogap = 0
      chi2( 1) =      0.97
      Prob > chi2 =      0.3249
```

We cannot reject the hypothesis that the 2 coefficients are equal, meaning that the rebound after 19 periods is just as significant in magnitude as the trough after 5 periods.

Methods and formulas

Three statistics are produced by `irf create` after `lpirf`: impulse–response functions (`irf`), orthogonalized impulse–response functions (`oirf`), and dynamic-multiplier functions (`dm`). The `irf` and `dm` results are taken directly from `lpirf`. This section describes the calculation of `oirf` and its standard errors.

When there are k variables in the local-projection model, there are k^2 combinations of impulses and responses at each horizon h . These impulse–response coefficients are collected in the $k \times k$ matrix \mathbf{B}_1^h in the projection

$$\mathbf{y}_{t+h-1} = \mathbf{B}_1^h \mathbf{y}_{t-1} + \cdots + \mathbf{B}_p^h \mathbf{y}_{t-p} + \mathbf{u}_{t+h-1}$$

The coefficients in \mathbf{B}_1^h are the simple IRFs.

To orthogonalize these impulse–response coefficients, we must compute an orthogonalizing matrix \mathbf{P} . To do this, we fit an auxiliary VAR model

$$\mathbf{y}_t = \mathbf{A}_1 \mathbf{y}_{t-1} + \cdots + \mathbf{A}_p \mathbf{y}_{t-p} + \mathbf{u}_t$$

Let \mathbf{U} be the $k \times T$ matrix of residuals from the VAR model. Then the covariance matrix of the residuals is the $k \times k$ matrix

$$\boldsymbol{\Sigma}_u = \frac{1}{T - \bar{m}} \mathbf{U} \mathbf{U}'$$

where \bar{m} is a degrees-of-freedom adjustment that is applied when the `dfk` option is specified and is 0 otherwise.

The orthogonalized impulse–response coefficients use $\mathbf{P} = \text{cholesky}(\boldsymbol{\Sigma}_u)$. That is, the orthogonalized local-projection IRFs use the same orthogonalization as the traditional VAR model. This convention follows Jordà (2005) and Kilian and Kim (2011).

With \mathbf{P} in hand, compute the orthogonalized IRFs at horizon h as

$$\mathbf{B}^{\text{oirf},h} = \mathbf{B}_1^h \mathbf{P}$$

and standard errors are square roots of the diagonals of

$$\mathbf{V} = (\mathbf{P}' \otimes \mathbf{I}) \mathbf{V}_b (\mathbf{P} \otimes \mathbf{I}) + (\mathbf{I} \otimes \mathbf{B}_1^h) \mathbf{V}_P (\mathbf{I} \otimes \mathbf{B}_1^h)'$$

The first piece captures uncertainty in the estimate of \mathbf{B}_1^h , and the second piece captures uncertainty in the estimate of \mathbf{P} . The $k^2 \times k^2$ matrix \mathbf{V}_b is the covariance matrix of the parameters $\text{vec}(\mathbf{B}_1^h)$, and the $k^2 \times k^2$ matrix \mathbf{V}_P is the covariance matrix of the parameters $\text{vec}(\mathbf{P})$.

References

- Jordà, Ò. 2005. Estimation and inference of impulse responses by local projections. *American Economic Review* 95: 161–182. <https://doi.org/10.1257/0002828053828518>.
- Kilian, L., and Y. J. Kim. 2011. How reliable are local projection estimators of impulse responses? *Review of Economics and Statistics* 93: 1460–1466. https://doi.org/10.1162/REST_a_00143.

Also see

[TS] **lpirf** — Local-projection impulse–response functions

[TS] **irf** — Create and analyze IRFs, dynamic-multiplier functions, and FEVDs

[U] **20 Estimation and postestimation commands**

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on [citing Stata documentation](#).