Title

> **irf create** — Obtain IRFs, dynamic-multiplier functions, and FEVDs

Syntax   Menu   Description   Options
[Remarks and examples](Remarks)   [Methods and formulas](Methods)   References   [Also see](Also)

Syntax

After var

   irf <u>cr</u>eate *irfname* [, *var_options*]

After svar

   irf <u>cr</u>eate *irfname* [, *svar_options*]

After vec

   irf <u>cr</u>eate *irfname* [, *vec_options*]

After arima

   irf <u>cr</u>eate *irfname* [, *arima_options*]

After arfima

   irf <u>cr</u>eate *irfname* [, *arfima_options*]

irfname is any valid name that does not exceed 15 characters.

var_options	Description
Main	
<u>set</u>(*filename* [, replace])	make *filename* active
replace	replace *irfname* if it already exists
<u>step</u>(*#*)	set forecast horizon to *#*; default is step(8)
<u>order</u>(*varlist*)	specify Cholesky ordering of endogenous variables
<u>estim</u>ates(*estname*)	use previously stored results *estname*; default is to use active results
Std. errors	
nose	do not calculate standard errors
bs	obtain standard errors from bootstrapped residuals
bsp	obtain standard errors from parametric bootstrap
<u>nod</u>ots	do not display "." for each bootstrap replication
reps(*#*)	use *#* bootstrap replications; default is reps(200)
<u>bsav</u>ing(*filename* [, replace])	save bootstrap results in *filename*

svar_options	Description
[Main]	
set(*filename* [, replace])	make *filename* active
replace	replace *irfname* if it already exists
step(*#*)	set forecast horizon to *#*; default is step(8)
estimates(*estname*)	use previously stored results *estname*; default is to use active results
[Std. errors]	
nose	do not calculate standard errors
bs	obtain standard errors from bootstrapped residual
bsp	obtain standard errors from parametric bootstrap
nodots	do not display "." for each bootstrap replication
reps(*#*)	use *#* bootstrap replications; default is reps(200)
bsaving(*filename* [, replace])	save bootstrap results in *filename*

vec_options	Description
[Main]	
set(*filename* [, replace])	make *filename* active
replace	replace *irfname* if it already exists
step(*#*)	set forecast horizon to *#*; default is step(8)
estimates(*estname*)	use previously stored results *estname*; default is to use active results

arima_options	Description
[Main]	
set(*filename* [, replace])	make *filename* active
replace	replace *irfname* if it already exists
step(*#*)	set forecast horizon to *#*; default is step(8)
estimates(*estname*)	use previously stored results *estname*; default is to use active results
[Std. errors]	
nose	do not calculate standard errors

arfima_options	Description
[Main]	
set(*filename* [, replace])	make *filename* active
replace	replace *irfname* if it already exists
step(*#*)	set forecast horizon to *#*; default is step(8)
smemory	calculate short-memory IRFs
estimates(*estname*)	use previously stored results *estname*; default is to use active results
[Std. errors]	
nose	do not calculate standard errors

The default is to use asymptotic standard errors if no options are specified.

irf create is for use after fitting a model with the var, svar, vec, arima, or arfima command; see [TS] **var**, [TS] **var svar**, [TS] **vec**, [TS] **arima**, and [TS] **arfima**.

You must tsset your data before using var, svar, vec, arima, or arfima and, hence, before using irf create; see [TS] **tsset**.

Menu

Statistics > Multivariate time series > IRF and FEVD analysis > Obtain IRFs, dynamic-multiplier functions, and FEVDs

Description

irf create estimates multiple sets of impulse–response functions (IRFs), dynamic-multiplier functions, and forecast-error variance decompositions (FEVDs) after estimation by var, svar, or vec; see [TS] **var**, [TS] **var svar**, or [TS] **vec**. irf create also estimates multiple sets of IRFs after estimation by arima or arfima; see [TS] **arima** or [TS] **arfima**. All of these estimates and their standard errors are known collectively as IRF results and are saved in an IRF file under the specified *irfname*.

The following types of IRFs and dynamic-multiplier functions are saved:

simple IRFs	after var, svar, vec, arima, or arfima
orthogonalized IRFs	after var, svar, vec, arima, or arfima
dynamic multipliers	after var
cumulative IRFs	after var, svar, vec, arima, or arfima
cumulative orthogonalized IRFs	after var, svar, vec, arima, or arfima
cumulative dynamic multipliers	after var
structural IRFs	after svar, arima, or arfima

The following types of FEVDs are saved:

Cholesky FEVDs	after var, svar, or vec
structural FEVDs	after svar only

Once you have created a set of IRF results, use the other irf commands to analyze them.

Options

> Main

set(*filename*[, replace]) specifies the IRF file to be used. If set() is not specified, the active IRF file is used; see [TS] **irf set**.

If set() is specified, the specified file becomes the active file, just as if you had issued an irf set command.

replace specifies that the results saved under *irfname* may be replaced, if they already exist. IRF results are saved in files, and one file may contain multiple IRF results.

step(#) specifies the step (forecast) horizon; the default is eight periods.

order(*varlist*) is allowed only after estimation by var; it specifies the Cholesky ordering of the endogenous variables to be used when estimating the orthogonalized IRFs. By default, the order in which the variables were originally specified on the var command is used.

smemory is allowed only after estimation by arfima; it specifies that the IRFs are calculated based on a short-memory model with the fractional difference parameter d set to zero.

estimates(*estname*) specifies that estimation results previously estimated by var, svar, or vec, and stored by estimates, be used. This option is rarely specified; see [R] **estimates**.

⌐ Std. errors ⌐

nose, bs, and bsp are alternatives that specify how (whether) standard errors are to be calculated. If none of these options is specified, asymptotic standard errors are calculated, except in two cases: after estimation by vec and after estimation by svar in which long-run constraints were applied. In those two cases, the default is as if nose were specified, although in the second case, you could specify bs or bsp. After estimation by vec, standard errors are simply not available.

nose specifies that no standard errors be calculated.

bs specifies that standard errors be calculated by bootstrapping the residuals. bs may not be specified if there are gaps in the data.

bsp specifies that standard errors be calculated via a multivariate-normal parametric bootstrap. bsp may not be specified if there are gaps in the data.

nodots, reps(#), and bsaving(*filename*[, replace]) are relevant only if bs or bsp is specified.

nodots specifies that dots not be displayed each time irf create performs a bootstrap replication.

reps(#), # > 50, specifies the number of bootstrap replications to be performed. reps(200) is the default.

bsaving(*filename*[, replace]) specifies that file *filename* be created and that the bootstrap replications be saved in it. New file *filename* is just a .dta dataset than can be loaded later using use; see [D] **use**. If *filename* is specified without an extension, .dta is assumed.

Remarks and examples   stata.com

If you have not read [TS] **irf**, please do so. An introductory example using IRFs is presented there.

Remarks are presented under the following headings:

> Introductory examples
> Technical aspects of IRF files
> IRFs and FEVDs
> IRF results for VARs
>> An introduction to impulse–response functions for VARs
>> An introduction to dynamic-multiplier functions for VARs
>> An introduction to forecast-error variance decompositions for VARs
> IRF results for VECMs
>> An introduction to impulse–response functions for VECMs
>> An introduction to forecast-error variance decompositions for VECMs
> IRF results for ARIMA and ARFIMA

Introductory examples

▷ Example 1: After var

Below we compare bootstrap and asymptotic standard errors for a specific FEVD. We begin by
fitting a VAR(2) model to the Lütkepohl data (we use the `var` command). We next use the `irf create`
command twice, first to create results with asymptotic standard errors (saved under the name `asymp`)
and then to re-create the same results, this time with bootstrap standard errors (saved under the name
`bs`). Because bootstrapping is a random process, we set the random-number seed (`set seed 123456`)
before using `irf create` the second time; this makes our results reproducible. Finally, we compare
results by using the IRF analysis command `irf ctable`.

```
. use http://www.stata-press.com/data/r13/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(1961q2) & qtr<=tq(1978q4), lags(1/2)
 (output omitted )
. irf create asymp, step(8) set(results1)
(file results1.irf created)
(file results1.irf now active)
(file results1.irf updated)
. set seed 123456
. irf create bs, step(8) bs reps(250) nodots
(file results1.irf updated)
. irf ctable (asymp dln_inc dln_consump fevd)
> (bs dln_inc dln_consump fevd), noci stderror
```

step	(1) fevd	(1) S.E.	(2) fevd	(2) S.E.
0	0	0	0	0
1	.282135	.087373	.282135	.104073
2	.278777	.083782	.278777	.096954
3	.33855	.090006	.33855	.100452
4	.339942	.089207	.339942	.099085
5	.342813	.090494	.342813	.099326
6	.343119	.090517	.343119	.09934
7	.343079	.090499	.343079	.099325
8	.34315	.090569	.34315	.099368

```
(1) irfname = asymp, impulse = dln_inc, and response = dln_consump
(2) irfname = bs, impulse = dln_inc, and response = dln_consump
```

Point estimates are, of course, the same. The bootstrap estimates of the standard errors, however,
are larger than the asymptotic estimates, which suggests that the sample size of 71 is not large
enough for the distribution of the estimator of the FEVD to be well approximated by the asymptotic
distribution. Here we would expect the bootstrap confidence interval to be more reliable than the
confidence interval that is based on the asymptotic standard error.

◁

❑ Technical note

The details of the bootstrap algorithms are given in *Methods and formulas*. These algorithms are
conditional on the first p observations, where p is the order of the fitted VAR. (In an SVAR model, p
is the order of the VAR that underlies the SVAR.) The bootstrapped estimates are conditional on the
first p observations, just as the estimators of the coefficients in VAR models are conditional on the

first p observations. With bootstrap standard errors (option bs), the p initial observations are used with resampling the residuals to produce the bootstrap samples used for estimation. With the more parametric bootstrap (option bsp), the p initial observations are used with draws from a multivariate normal distribution with variance–covariance matrix $\widehat{\Sigma}$ to generate the bootstrap samples.

❑

❑ Technical note

For var and svar e() results, irf uses $\widehat{\Sigma}$, the estimated variance matrix of the disturbances, in computing the asymptotic standard errors of all the functions. The point estimates of the orthogonalized impulse–response functions, the structural impulse–response functions, and all the variance decompositions also depend on $\widehat{\Sigma}$. As discussed in [TS] **var**, var and svar use the ML estimator of this matrix by default, but they have option dfk, which will instead use an estimator that includes a small-sample correction. Specifying dfk when the model is fit—when the var or svar command is given—changes the estimate of $\widehat{\Sigma}$ and will change the IRF results that depend on it.

❑

▷ Example 2: After var with exogenous variables

After fitting a VAR, irf create computes estimates of the dynamic multipliers, which describe the impact of a unit change in an exogenous variable on each endogenous variable. For instance, below we estimate and report the cumulative dynamic multipliers from a model in which changes in investment are exogenous. The results indicate that both of the cumulative dynamic multipliers are significant.

```
. var dln_inc dln_consump if qtr>=tq(1961q2) & qtr<=tq(1978q4), lags(1/2)
> exog(L(0/2).dln_inv)
  (output omitted)
. irf create dm, step(8)
(file results1.irf updated)
```

```
. irf table cdm, impulse(dln_inv) irf(dm)
```
<div align="center">Results from dm</div>

step	(1) cdm	(1) Lower	(1) Upper
0	.032164	-.027215	.091544
1	.096568	.003479	.189656
2	.140107	.022897	.257317
3	.150527	.032116	.268938
4	.148979	.031939	.26602
5	.151247	.033011	.269482
6	.150267	.033202	.267331
7	.150336	.032858	.267813
8	.150525	.033103	.267948

step	(2) cdm	(2) Lower	(2) Upper
0	.058681	.012529	.104832
1	.062723	-.005058	.130504
2	.126167	.032497	.219837
3	.136583	.038691	.234476
4	.146482	.04442	.248543
5	.146075	.045201	.24695
6	.145542	.044988	.246096
7	.146309	.045315	.247304
8	.145786	.045206	.246365

```
95% lower and upper bounds reported
(1) irfname = dm, impulse = dln_inv, and response = dln_inc
(2) irfname = dm, impulse = dln_inv, and response = dln_consump
```

◁

▷ Example 3: After vec

Although all IRFs and orthogonalized IRFs (OIRFs) from models with stationary variables will taper off to zero, some of the IRFs and OIRFs from models with first-difference stationary variables will not. This is the key difference between IRFs and OIRFs from systems of stationary variables fit by var or svar and those obtained from systems of first-difference stationary variables fit by vec. When the effect of the innovations dies out over time, the shocks are said to be transitory. In contrast, when the effect does not taper off, shocks are said to be permanent.

In this example, we look at the OIRF from one of the VECMs fit to the unemployment-rate data analyzed in example 2 of [TS] **vec**. We see that an orthogonalized shock to Indiana has a permanent effect on the unemployment rate in Missouri:

```
. use http://www.stata-press.com/data/r13/urates
. vec missouri indiana kentucky illinois, trend(rconstant) rank(2) lags(4)
 (output omitted )
. irf create vec1, set(vecirfs) step(50)
(file vecirfs.irf created)
(file vecirfs.irf now active)
(file vecirfs.irf updated)
```

Now we can use `irf graph` to graph the OIRF of interest:

    `. irf graph oirf, impulse(indiana) response(missouri)`
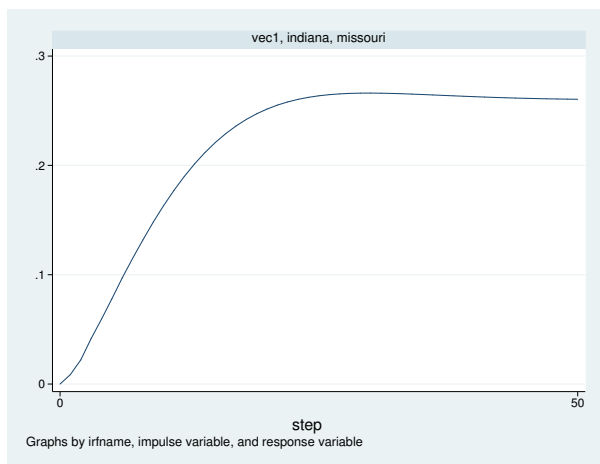
The graph shows that the estimated OIRF converges to a positive asymptote, which indicates that an orthogonalized innovation to the unemployment rate in Indiana has a permanent effect on the unemployment rate in Missouri.

◁

Technical aspects of IRF files

    This section is included for programmers wishing to extend the `irf` system.

    `irf create` estimates a series of impulse–response functions and their standard errors. Although these estimates are saved in an IRF file, most users will never need to look at the contents of this file. The IRF commands fill in, analyze, present, and manage IRF results.

    IRF files are just Stata datasets that have names ending in `.irf` instead of `.dta`. The dataset in the file has a nested panel structure.

    Variable `irfname` contains the *irfname* specified by the user. Variable `impulse` records the name of the endogenous variable whose innovations are the impulse. Variable `response` records the name of the endogenous variable that is responding to the innovations. In a model with K endogenous variables, there are K^2 combinations of `impulse` and `response`. Variable `step` records the periods for which these estimates were computed.

    Below is a catalog of the statistics that `irf create` estimates and the variable names under which they are saved in the IRF file.

Statistic	Name
impulse–response functions	`irf`
orthogonalized impulse–response functions	`oirf`
dynamic-multiplier functions	`dm`
cumulative impulse–response functions	`cirf`
cumulative orthogonalized impulse–response functions	`coirf`
cumulative dynamic-multiplier functions	`cdm`
Cholesky forecast-error decomposition	`fevd`
structural impulse–response functions	`sirf`
structural forecast-error decomposition	`sfevd`
standard error of the impulse–response functions	`stdirf`
standard error of the orthogonalized impulse–response functions	`stdoirf`
standard error of the cumulative impulse–response functions	`stdcirf`
standard error of the cumulative orthogonalized impulse–response functions	`stdcoirf`
standard error of the Cholesky forecast-error decomposition	`stdfevd`
standard error of the structural impulse–response functions	`stdsirf`
standard error of the structural forecast-error decomposition	`stdsfevd`

In addition to the variables, information is stored in _dta characteristics. Much of the following information is also available in r() after irf describe, where it is often more convenient to obtain the information. Characteristic _dta[version] contains the version number of the IRF file, which is currently 1.1. Characteristic _dta[irfnames] contains a list of all the *irfnames* in the IRF file. For each *irfname*, there are a series of additional characteristics:

Name	Contents
_dta[*irfname*_model]	var, sr var, lr var, vec, arima, or arfima
_dta[*irfname*_order]	Cholesky order used in IRF estimates
_dta[*irfname*_exog]	exogenous variables, and their lags, in VAR
_dta[*irfname*_exogvars]	exogenous variables in VAR
_dta[*irfname*_constant]	constant or noconstant, depending on whether noconstant was specified in var or svar
_dta[*irfname*_lags]	lags in model
_dta[*irfname*_exlags]	lags of exogenous variables in model
_dta[*irfname*_tmin]	minimum value of timevar in the estimation sample
_dta[*irfname*_tmax]	maximum value of timevar in the estimation sample
_dta[*irfname*_timevar]	name of tsset timevar
_dta[*irfname*_tsfmt]	format of timevar
_dta[*irfname*_varcns]	constrained or colon-separated list of constraints placed on VAR coefficients
_dta[*irfname*_svarcns]	constrained or colon-separated list of constraints placed on VAR coefficients
_dta[*irfname*_step]	maximum step in IRF estimates
_dta[*irfname*_stderror]	asymptotic, bs, bsp, or none, depending on the type of standard errors requested
_dta[*irfname*_reps]	number of bootstrap replications performed
_dta[*irfname*_version]	version of the IRF file that originally held *irfname* IRF results
_dta[*irfname*_rank]	number of cointegrating equations
_dta[*irfname*_trend]	trend() specified in vec
_dta[*irfname*_veccns]	constraints placed on VECM parameters
_dta[*irfname*_sind]	normalized seasonal indicators included in vec
_dta[*irfname*_d]	fractional difference parameter d in arfima

IRFs and FEVDs

irf create can estimate several types of IRFs and FEVDs for VARs and VECMs. irf create can also estimate IRFs and cumulative IRFs for ARIMA and ARFIMA models. We first discuss IRF results for VAR and SVAR models, and then we discuss them in the context of VECMs. Because the cointegrating VECM is an extension of the stationary VAR framework, the section that discusses the IRF results for VECMs draws on the earlier VAR material. We conclude our discussion with IRF results for ARIMA and ARFIMA models.

IRF results for VARs

An introduction to impulse–response functions for VARs

A pth-order vector autoregressive model (VAR) with exogenous variables is given by

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

where

$\mathbf{y}_t = (y_{1t}, \ldots, y_{Kt})'$ is a $K \times 1$ random vector,
the $\mathbf{A}_i$ are fixed $K \times K$ matrices of parameters,
$\mathbf{x}_t$ is an $R_0 \times 1$ vector of exogenous variables,
$\mathbf{B}$ is a $K \times R_0$ matrix of coefficients,
$\mathbf{v}$ is a $K \times 1$ vector of fixed parameters, and
$\mathbf{u}_t$ is assumed to be white noise; that is,
  $E(\mathbf{u}_t) = \mathbf{0}$
  $E(\mathbf{u}_t\mathbf{u}_t') = \mathbf{\Sigma}$
  $E(\mathbf{u}_t\mathbf{u}_s') = \mathbf{0}$ for $t \neq s$

As discussed in [TS] **varstable**, a VAR can be rewritten in moving-average form only if it is stable. Any exogenous variables are assumed to be covariance stationary. Because the functions of interest in this section depend only on the exogenous variables through their effect on the estimated $\mathbf{A}_i$, we can simplify the notation by dropping them from the analysis. All the formulas given below still apply, although the $\mathbf{A}_i$ are estimated jointly with $\mathbf{B}$ on the exogenous variables.

Below we discuss conditions under which the IRFs and forecast-error variance decompositions have a causal interpretation. Although estimation requires only that the exogenous variables be predetermined, that is, that $E(\mathbf{x}_{jt}u_{it}) = 0$ for all i, j, and t, assigning a causal interpretation to IRFs and FEVDs requires that the exogenous variables be strictly exogenous, that is, that $E(\mathbf{x}_{js}u_{it}) = 0$ for all i, j, s, and t.

IRFs describe how the innovations to one variable affect another variable after a given number of periods. For an example of how IRFs are interpreted, see Stock and Watson (2001). They use IRFs to investigate the effect of surprise shocks to the Federal Funds rate on inflation and unemployment. In another example, Christiano, Eichenbaum, and Evans (1999) use IRFs to investigate how shocks to monetary policy affect other macroeconomic variables.

Consider a VAR without exogenous variables:

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

The VAR represents the variables in $\mathbf{y}_t$ as functions of its own lags and serially uncorrelated innovations $\mathbf{u}_t$. All the information about contemporaneous correlations among the K variables in $\mathbf{y}_t$ is contained in $\mathbf{\Sigma}$. In fact, as discussed in [TS] **var svar**, a VAR can be viewed as the reduced form of a dynamic simultaneous-equation model.

To see how the innovations affect the variables in $\mathbf{y}_t$ after, say, i periods, rewrite the model in its moving-average form

$$\mathbf{y}_t = \boldsymbol{\mu} + \sum_{i=0}^{\infty} \boldsymbol{\Phi}_i \mathbf{u}_{t-i} \tag{2}$$

where $\boldsymbol{\mu}$ is the $K \times 1$ time-invariant mean of $\mathbf{y}_t$, and

$$\boldsymbol{\Phi}_i = \begin{cases} \mathbf{I}_K & \text{if } i = 0 \\ \sum_{j=1}^{i} \boldsymbol{\Phi}_{i-j} \mathbf{A}_j & \text{if } i = 1, 2, \dots \end{cases}$$

We can rewrite a VAR in the moving-average form only if it is stable. Essentially, a VAR is stable if the variables are covariance stationary and none of the autocorrelations are too high (the issue of stability is discussed in greater detail in [TS] **varstable**).

The $\boldsymbol{\Phi}_i$ are the simple IRFs. The j, k element of $\boldsymbol{\Phi}_i$ gives the effect of a 1–time unit increase in the kth element of $\mathbf{u}_t$ on the jth element of $\mathbf{y}_t$ after i periods, holding everything else constant. Unfortunately, these effects have no causal interpretation, which would require us to be able to answer the question, "How does an innovation to variable k, holding everything else constant, affect variable j after i periods?" Because the $\mathbf{u}_t$ are contemporaneously correlated, we cannot assume that everything else is held constant. Contemporaneous correlation among the $\mathbf{u}_t$ implies that a shock to one variable is likely to be accompanied by shocks to some of the other variables, so it does not make sense to shock one variable and hold everything else constant. For this reason, (2) cannot provide a causal interpretation.

This shortcoming may be overcome by rewriting (2) in terms of mutually uncorrelated innovations. Suppose that we had a matrix $\mathbf{P}$, such that $\boldsymbol{\Sigma} = \mathbf{P}\mathbf{P}'$. If we had such a $\mathbf{P}$, then $\mathbf{P}^{-1}\boldsymbol{\Sigma}\mathbf{P}'^{-1} = \mathbf{I}_K$, and

$$E\{\mathbf{P}^{-1}\mathbf{u}_t(\mathbf{P}^{-1}\mathbf{u}_t)'\} = \mathbf{P}^{-1}E\{(\mathbf{u}_t\mathbf{u}_t')\mathbf{P}'^{-1}\} = \mathbf{P}^{-1}\boldsymbol{\Sigma}\mathbf{P}'^{-1} = \mathbf{I}_K$$

We can thus use $\mathbf{P}^{-1}$ to orthogonalize the $\mathbf{u}_t$ and rewrite (2) as

$$\mathbf{y}_t = \boldsymbol{\mu} + \sum_{i=0}^{\infty} \boldsymbol{\Phi}_i \mathbf{P}\mathbf{P}^{-1} \mathbf{u}_{t-i}$$

$$= \boldsymbol{\mu} + \sum_{i=0}^{\infty} \boldsymbol{\Theta}_i \mathbf{P}^{-1} \mathbf{u}_{t-i}$$

$$= \boldsymbol{\mu} + \sum_{i=0}^{\infty} \boldsymbol{\Theta}_i \mathbf{w}_{t-i}$$

where $\boldsymbol{\Theta}_i = \boldsymbol{\Phi}_i \mathbf{P}$ and $\mathbf{w}_t = \mathbf{P}^{-1}\mathbf{u}_t$. If we had such a $\mathbf{P}$, the $\mathbf{w}_k$ would be mutually orthogonal, and no information would be lost in the holding-everything-else-constant assumption, implying that the $\boldsymbol{\Theta}_i$ would have the causal interpretation that we seek.

Choosing a $\mathbf{P}$ is similar to placing identification restrictions on a system of dynamic simultaneous equations. The simple IRFs do not identify the causal relationships that we wish to analyze. Thus we seek at least as many identification restrictions as necessary to identify the causal IRFs.

So, where do we get such a $\mathbf{P}$? Sims (1980) popularized the method of choosing $\mathbf{P}$ to be the Cholesky decomposition of $\widehat{\boldsymbol{\Sigma}}$. The IRFs based on this choice of $\mathbf{P}$ are known as the *orthogonalized IRFs*. Choosing $\mathbf{P}$ to be the Cholesky decomposition of $\widehat{\boldsymbol{\Sigma}}$ is equivalent to imposing a recursive structure for the corresponding dynamic structural equation model. The ordering of the recursive structure is the same as the ordering imposed in the Cholesky decomposition. Because this choice is arbitrary, some researchers will look at the OIRFs with different orderings assumed in the Cholesky decomposition. The `order()` option available with `irf create` facilitates this type of analysis.

The SVAR approach integrates the need to identify the causal IRFs into the model specification and estimation process. Sufficient identification restrictions can be obtained by placing either short-run or long-run restrictions on the model. The VAR in (1) can be rewritten as

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

Similarly, a short-run SVAR model can be written as

$$\mathbf{A}(\mathbf{y}_t - \mathbf{v} - \mathbf{A}_1\mathbf{y}_{t-1} - \cdots - \mathbf{A}_p\mathbf{y}_{t-p}) = \mathbf{A}\mathbf{u}_t = \mathbf{B}\mathbf{e}_t \tag{3}$$

where $\mathbf{A}$ and $\mathbf{B}$ are $K \times K$ nonsingular matrices of parameters to be estimated, $\mathbf{e}_t$ is a $K \times 1$ vector of disturbances with $\mathbf{e}_t \sim N(\mathbf{0}, \mathbf{I}_K)$, and $E(\mathbf{e}_t\mathbf{e}_s') = \mathbf{0}_K$ for all $s \neq t$. Sufficient constraints must be placed on $\mathbf{A}$ and $\mathbf{B}$ so that $\mathbf{P}$ is identified. One way to see the connection is to draw out the implications of the latter equality in (3). From (3) it can be shown that

$$\mathbf{\Sigma} = \mathbf{A}^{-1}\mathbf{B}(\mathbf{A}^{-1}\mathbf{B})'$$

As discussed in [TS] **var svar**, the estimates $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{B}}$ are obtained by maximizing the concentrated log-likelihood function on the basis of the $\widehat{\mathbf{\Sigma}}$ obtained from the underlying VAR. The short-run SVAR approach chooses $\mathbf{P} = \widehat{\mathbf{A}}^{-1}\widehat{\mathbf{B}}$ to identify the causal IRFs. The long-run SVAR approach works similarly, with $\mathbf{P} = \widehat{\mathbf{C}} = \widehat{\overline{\mathbf{A}}}^{-1}\widehat{\mathbf{B}}$, where $\widehat{\overline{\mathbf{A}}}^{-1}$ is the matrix of estimated long-run or accumulated effects of the reduced-form VAR shocks.

There is one important difference between long-run and short-run SVAR models. As discussed by Amisano and Giannini (1997, chap. 6), in the short-run model the constraints are applied directly to the parameters in $\mathbf{A}$ and $\mathbf{B}$. Then $\mathbf{A}$ and $\mathbf{B}$ interact with the estimated parameters of the underlying VAR. In contrast, in a long-run model, the constraints are placed on functions of the estimated VAR parameters. Although estimation and inference of the parameters in $\mathbf{C}$ is straightforward, obtaining the asymptotic standard errors of the structural IRFs requires untenable assumptions. For this reason, `irf create` does not estimate the asymptotic standard errors of the structural IRFs generated by long-run SVAR models. However, bootstrap standard errors are still available.

An introduction to dynamic-multiplier functions for VARs

A dynamic-multiplier function measures the effect of a unit change in an exogenous variable on the endogenous variables over time. Per Lütkepohl (2005, chap. 10), if the VAR with exogenous variables is stable, it can be rewritten as

$$\mathbf{y}_t = \sum_{i=0}^{\infty} \mathbf{D}_i x_{t-i} + \sum_{i=0}^{\infty} \mathbf{\Phi}_i u_{t-i}$$

where the $\mathbf{D}_i$ are the dynamic-multiplier functions. (See *Methods and formulas* for details.) Some authors refer to the dynamic-multiplier functions as transfer functions because they specify how a unit change in an exogenous variable is "transferred" to the endogenous variables.

❏ Technical note

irf create computes dynamic-multiplier functions only after var. After short-run SVAR models, the dynamic multipliers from the VAR are the same as those from the SVAR. The dynamic multipliers for long-run SVARs have not yet been worked out.

❏

An introduction to forecast-error variance decompositions for VARs

Another measure of the effect of the innovations in variable k on variable j is the FEVD. This method, which is also known as *innovation accounting*, measures the fraction of the error in forecasting variable j after h periods that is attributable to the orthogonalized innovations in variable k. Because deriving the FEVD requires orthogonalizing the $\mathbf{u}_t$ innovations, the FEVD is always predicated upon a choice of $\mathbf{P}$.

Lütkepohl (2005, sec. 2.2.2) shows that the h-step forecast error can be written as

$$\mathbf{y}_{t+h} - \widehat{\mathbf{y}}_t(h) = \sum_{i=0}^{h-1} \mathbf{\Phi}_i \mathbf{u}_{t+h-i} \tag{4}$$

where y_{t+h} is the value observed at time $t+h$ and $\widehat{y}_t(h)$ is the h-step-ahead predicted value for y_{t+h} that was made at time t.

Because the $\mathbf{u}_t$ are contemporaneously correlated, their distinct contributions to the forecast error cannot be ascertained. However, if we choose a $\mathbf{P}$ such that $\mathbf{\Sigma} = \mathbf{PP}'$, as above, we can orthogonalize the $\mathbf{u}_t$ into $\mathbf{w}_t = \mathbf{P}^{-1}\mathbf{u}_t$. We can then ascertain the relative contribution of the distinct elements of $\mathbf{w}_t$. Thus we can rewrite (4) as

$$\mathbf{y}_{t+h} - \widehat{\mathbf{y}}_t(h) = \sum_{i=0}^{h-1} \mathbf{\Phi}_i \mathbf{PP}^{-1} \mathbf{u}_{t+h-i}$$
$$= \sum_{i=0}^{h-1} \mathbf{\Theta}_i \mathbf{w}_{t+h-i}$$

Because the forecast errors can be written in terms of the orthogonalized errors, the forecast-error variance can be written in terms of the orthogonalized error variances. Forecast-error variance decompositions measure the fraction of the total forecast-error variance that is attributable to each orthogonalized shock.

❏ Technical note

The details in this note are not critical to the discussion that follows. A forecast-error variance decomposition is derived for a given $\mathbf{P}$. Per Lütkepohl (2005, sec. 2.3.3), letting $\theta_{mn,i}$ be the m, nth element of $\mathbf{\Theta}_i$, we can express the h-step forecast error of the jth component of $\mathbf{y}_t$ as

$$\mathbf{y}_{j,t+h} - \widehat{\mathbf{y}}_j(h) = \sum_{i=0}^{h-1} \theta_{j1,1} \mathbf{w}_{1,t+h-i} + \cdots + \theta_{jK,i} \mathbf{w}_{K,t+h-i}$$
$$= \sum_{k=1}^{K} \theta_{jk,0} \mathbf{w}_{k,t+h} + \cdots + \theta_{jk,h-1} \mathbf{w}_{k,t+1}$$

The $\mathbf{w}_t$, which were constructed using $\mathbf{P}$, are mutually orthogonal with unit variance. This allows us to compute easily the mean squared error (MSE) of the forecast of variable j at horizon h in terms of the contributions of the components of $\mathbf{w}_t$. Specifically,

$$E[\{y_{j,t+h} - y_{j,t}(h)\}^2] = \sum_{k=1}^{K} (\theta_{jk,0}^2 + \cdots + \theta_{jk,h-1}^2)$$

The kth term in the sum above is interpreted as the contribution of the orthogonalized innovations in variable k to the h-step forecast error of variable j. Note that the kth element in the sum above can be rewritten as

$$(\theta_{jk,0}^2 + \cdots + \theta_{jk,h-1}^2) = \sum_{i=0}^{h-1} \left(\mathbf{e}_j' \boldsymbol{\Theta}_k \mathbf{e}_k\right)^2$$

where $\mathbf{e}_i$ is the ith column of $\mathbf{I}_K$. Normalizing by the forecast error for variable j at horizon h yields

$$\omega_{jk,h} = \frac{\sum_{i=0}^{h-1} \left(\mathbf{e}_j' \boldsymbol{\Theta}_k \mathbf{e}_k\right)^2}{\text{MSE}\{y_{j,t}(h)\}}$$

where $\text{MSE}\{y_{j,t}(h)\} = \sum_{i=0}^{h-1} \sum_{k=1}^{K} \theta_{jk,i}^2$.

❑

Because the FEVD depends on the choice of $\mathbf{P}$, there are different forecast-error variance decompositions associated with each distinct $\mathbf{P}$. irf create can estimate the FEVD for a VAR or an SVAR. For a VAR, $\mathbf{P}$ is the Cholesky decomposition of $\widehat{\boldsymbol{\Sigma}}$. For an SVAR, $\mathbf{P}$ is the estimated structural decomposition, $\mathbf{P} = \widehat{\mathbf{A}}^{-1}\widehat{\mathbf{B}}$ for short-run models and $\mathbf{P} = \widehat{\mathbf{C}}$ for long-run SVAR models. Due to the same complications that arose with the structural impulse–response functions, the asymptotic standard errors of the structural FEVD are not available after long-run SVAR models, but bootstrap standard errors are still available.

IRF results for VECMs

An introduction to impulse–response functions for VECMs

As discussed in [TS] **vec intro**, the VECM is a reparameterization of the VAR that is especially useful for fitting VARs with cointegrating variables. This implies that the estimated parameters for the corresponding VAR model can be backed out from the estimated parameters of the VECM model. This relationship means we can use the VAR form of the cointegrating VECM to discuss the IRFs for VECMs.

Consider a cointegrating VAR with one lag with no constant or trend,

$$\mathbf{y}_t = \mathbf{A}\mathbf{y}_{t-1} + \mathbf{u}_t \tag{5}$$

where $\mathbf{y}_t$ is a $K \times 1$ vector of endogenous, first-difference stationary variables among which there are $1 \le r < K$ cointegration equations; $\mathbf{A}$ is $K \times K$ matrix of parameters; and $\mathbf{u}_t$ is a $K \times 1$ vector of i.i.d. disturbances.

We developed intuition for the IRFs from a stationary VAR by rewriting the VAR as an infinite-order vector moving-average (VMA) process. While the Granger representation theorem establishes the existence of a VMA formulation of this model, because the cointegrating VAR is not stable, the inversion is not nearly so intuitive. (See Johansen [1995, chapters 3 and 4] for more details.) For this reason, we use (5) to develop intuition for the IRFs from a cointegrating VAR.

Suppose that K is 3, that $\mathbf{u}_1 = (1, 0, 0)$, and that we want to analyze the time paths of the variables in $\mathbf{y}$ conditional on the initial values $\mathbf{y}_0 = \mathbf{0}$, $\mathbf{A}$, and the condition that there are no more shocks to the system, that is, $\mathbf{0} = \mathbf{u}_2 = \mathbf{u}_3 = \cdots$. These assumptions and (5) imply that

$$\mathbf{y}_1 = \mathbf{u}_1$$

$$\mathbf{y}_2 = \mathbf{A}\mathbf{y}_1 = \mathbf{A}\mathbf{u}_1$$

$$\mathbf{y}_3 = \mathbf{A}\mathbf{y}_2 = \mathbf{A}^2\mathbf{u}_1$$

and so on. The ith-row element of the first column of $\mathbf{A}^s$ contains the effect of the unit shock to the first variable after s periods. The first column of $\mathbf{A}^s$ contains the IRF of a unit impulse to the first variable after s periods. We could deduce the IRFs of a unit impulse to any of the other variables by administering the unit shock to one of them instead of to the first variable. Thus we can see that the (i, j)th element of $\mathbf{A}^s$ contains the unit IRF from variable j to variable i after s periods. By starting with orthogonalized shocks of the form $\mathbf{P}^{-1}\mathbf{u}_t$, we can use the same logic to derive the OIRFs to be $\mathbf{A}^s\mathbf{P}$.

For the stationary VAR, stability implies that all the eigenvalues of $\mathbf{A}$ have moduli strictly less than one, which in turn implies that all the elements of $\mathbf{A}^s \to \mathbf{0}$ as $s \to \infty$. This implies that all the IRFs from a stationary VAR taper off to zero as $s \to \infty$. In contrast, in a cointegrating VAR, some of the eigenvalues of $\mathbf{A}$ are 1, while the remaining eigenvalues have moduli strictly less than 1. This implies that in cointegrating VARs some of the elements of $\mathbf{A}^s$ are not going to zero as $s \to \infty$, which in turn implies that some of the IRFs and OIRFs are not going to zero as $s \to \infty$. The fact that the IRFs and OIRFs taper off to zero for stationary VARs but not for cointegrating VARs is one of the key differences between the two models.

When the IRF or OIRF from the innovation in one variable to another tapers off to zero as time goes on, the innovation to the first variable is said to have a transitory effect on the second variable. When the IRF or OIRF does not go to zero, the effect is said to be permanent.

Note that, because some of the IRFs and OIRFs do not taper off to zero, some of the cumulative IRFs and OIRFs diverge over time.

An introduction to forecast-error variance decompositions for VECMs

The results from *An introduction to impulse–response functions for VECMs* can be used to show that the interpretation of FEVDs for a finite number of steps in cointegrating VARs is essentially the same as in the stationary case. Because the MSE of the forecast is diverging, this interpretation is valid only for a finite number of steps. (See [TS] **vec intro** and [TS] **fcast compute** for more information on this point.)

IRF results for ARIMA and ARFIMA

A covariance-stationary additive ARMA(p, q) model can be written as

$$\boldsymbol{\rho}(L^p)(y_t - \mathbf{x}_t\boldsymbol{\beta}) = \boldsymbol{\theta}(L^q)\epsilon_t$$

where

$$\boldsymbol{\rho}(L^p) = 1 - \rho_1 L - \rho_2 L^2 - \cdots - \rho_p L^p$$
$$\boldsymbol{\theta}(L^q) = 1 + \theta_1 L + \theta_2 L^2 + \cdots + \theta_q L^q$$

and $L^j y_t = y_{t-j}$.

We can rewrite the above model as an infinite-order moving-average process

$$y_t = \mathbf{x}_t\boldsymbol{\beta} + \boldsymbol{\psi}(L)\epsilon_t$$

where

$$\boldsymbol{\psi}(L) = \frac{\boldsymbol{\theta}(L)}{\boldsymbol{\rho}(L)} = 1 + \psi_1 L + \psi_2 L^2 + \cdots \tag{6}$$

This representation shows the impact of the past innovations on the current y_t. The ith coefficient describes the response of y_t to a one-time impulse in ϵ_{t-i}, holding everything else constant. The ψ_i coefficients are collectively referred to as the impulse–response function of the ARMA model. For a covariance-stationary series, the ψ_i coefficients decay exponentially.

A covariance-stationary multiplicative seasonal ARMA model, often abbreviated SARMA, of order $(p, q) \times (P, Q)_s$ can be written as

$$\rho(L^p)\boldsymbol{\rho}_s(L^P)(y_t - \mathbf{x}_t\boldsymbol{\beta}) = \boldsymbol{\theta}(L^q)\boldsymbol{\theta}_s(L^Q)\epsilon_t$$

where

$$\boldsymbol{\rho}_s(L^P) = (1 - \rho_{s,1}L^s - \rho_{s,2}L^{2s} - \cdots - \rho_{s,P}L^{Ps})$$
$$\boldsymbol{\theta}_s(L^Q) = (1 + \theta_{s,1}L^s + \theta_{s,2}L^{2s} + \cdots + \theta_{s,Q}L^{Qs})$$

with $\boldsymbol{\rho}(L^p)$ and $\boldsymbol{\theta}(L^q)$ defined as above.

We can express this model as an additive ARMA model by multiplying the terms and imposing nonlinear constraints on multiplied coefficients. For example, consider the SARMA model given by

$$(1 - \rho_1 L)(1 - \rho_{4,1}L^4)y_t = \epsilon_t$$

Expanding the above equation and solving for y_t yields

$$y_t = \rho_1 y_{t-1} + \rho_{4,1}y_{t-4} - \rho_1\rho_{4,1}y_{t-5} + \epsilon_t$$

or, in ARMA terms,

$$y_t = \rho_1 y_{t-1} + \rho_4 y_{t-4} + \rho_5 y_{t-5} + \epsilon_t$$

subject to the constraint $\rho_5 = -\rho_1\rho_{4,1}$.

Once we have obtained an ARMA representation of a SARMA process, we obtain the IRFs from (6).

An ARFIMA(p, d, q) model can be written as

$$\boldsymbol{\rho}(L^p)(1 - L)^d(y_t - \mathbf{x}_t\boldsymbol{\beta}) = \boldsymbol{\theta}(L^q)\epsilon_t$$

with $(1 - L)^d$ denoting a fractional integration operation.

Solving for y_t, we obtain

$$y_t = \mathbf{x}_t\boldsymbol{\beta} + (1 - L)^{-d}\boldsymbol{\psi}(L)\epsilon_t$$

This makes it clear that the impulse–response function for an ARFIMA model corresponds to a fractionally differenced impulse–response function for an ARIMA model. Because of the fractional differentiation, the ψ_i coefficients decay very slowly; see *Remarks and examples* in [TS] **arfima**.

Methods and formulas

Methods and formulas are presented under the following headings:

> *Impulse–response function formulas for VARs*
> *Dynamic-multiplier function formulas for VARs*
> *Forecast-error variance decomposition formulas for VARs*
> *Impulse–response function formulas for VECMs*
> *Algorithms for bootstrapping the VAR IRF and FEVD standard errors*
> *Impulse–response function formulas for ARIMA and ARFIMA*

Impulse–response function formulas for VARs

The previous discussion implies that there are three different choices of $\mathbf{P}$ that can be used to obtain distinct $\boldsymbol{\Theta}_i$. $\mathbf{P}$ is the Cholesky decomposition of $\boldsymbol{\Sigma}$ for the OIRFs. For the structural IRFs, $\mathbf{P} = \mathbf{A}^{-1}\mathbf{B}$ for short-run models, and $\mathbf{P} = \mathbf{C}$ for long-run models. We will distinguish between the three by defining $\boldsymbol{\Theta}_i^o$ to be the OIRFs, $\boldsymbol{\Theta}_i^{\text{sr}}$ to be the short-run structural IRFs, and $\boldsymbol{\Theta}_i^{lr}$ to be the long-run structural IRFs.

We also define $\widehat{\mathbf{P}}_c$ to be the Cholesky decomposition of $\widehat{\boldsymbol{\Sigma}}$, $\widehat{\mathbf{P}}_{\text{sr}} = \widehat{\mathbf{A}}^{-1}\widehat{\mathbf{B}}$ to be the short-run structural decomposition, and $\widehat{\mathbf{P}}_{lr} = \widehat{\mathbf{C}}$ to be the long-run structural decomposition.

Given estimates of the $\widehat{\mathbf{A}}_i$ and $\widehat{\boldsymbol{\Sigma}}$ from var or svar, the estimates of the simple IRFs and the OIRFs are, respectively,

$$\widehat{\boldsymbol{\Phi}}_i = \sum_{j=1}^{i} \widehat{\boldsymbol{\Phi}}_{i-j}\widehat{\mathbf{A}}_j$$

and

$$\widehat{\boldsymbol{\Theta}}_i^o = \widehat{\boldsymbol{\Phi}}_i\widehat{\mathbf{P}}_c$$

where $\widehat{\mathbf{A}}_j = \mathbf{0}_K$ for $j > p$.

Given the estimates $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{B}}$, or $\widehat{\mathbf{C}}$, from svar, the estimates of the structural IRFs are either

$$\widehat{\boldsymbol{\Theta}}_i^{\text{sr}} = \widehat{\boldsymbol{\Phi}}_i\widehat{\mathbf{P}}_{\text{sr}}$$

or

$$\widehat{\boldsymbol{\Theta}}_i^{lr} = \widehat{\boldsymbol{\Phi}}_i\widehat{\mathbf{P}}_{lr}$$

The estimated structural IRFs stored in an IRF file with the variable name `sirf` may be from either a short-run model or a long-run model, depending on the estimation results used to create the IRFs. As discussed in [TS] **irf describe**, you can easily determine whether the structural IRFs were generated from a short-run or a long-run SVAR model using `irf describe`.

Following Lütkepohl (2005, sec. 3.7), estimates of the cumulative IRFs and the cumulative orthogonalized impulse–response functions (COIRFs) at period n are, respectively,

$$\widehat{\boldsymbol{\Psi}}_n = \sum_{i=0}^{n} \widehat{\boldsymbol{\Phi}}_i$$

and

$$\widehat{\boldsymbol{\Xi}}_n = \sum_{i=0}^{n} \widehat{\boldsymbol{\Theta}}_i$$

The asymptotic standard errors of the different impulse–response functions are obtained by applications of the delta method. See Lütkepohl (2005, sec. 3.7) and Amisano and Giannini (1997, chap. 4) for the derivations. See Serfling (1980, sec. 3.3) for a discussion of the delta method. In presenting the variance–covariance matrix estimators, we make extensive use of the vec() operator, where vec($\mathbf{X}$) is the vector obtained by stacking the columns of $\mathbf{X}$.

Lütkepohl (2005, sec. 3.7) derives the asymptotic VCEs of vec($\boldsymbol{\Phi}_i$), vec($\boldsymbol{\Theta}_i^o$), vec($\widehat{\boldsymbol{\Psi}}_n$), and vec($\widehat{\boldsymbol{\Xi}}_n$). Because vec($\boldsymbol{\Phi}_i$) is $K^2 \times 1$, the asymptotic VCE of vec($\boldsymbol{\Phi}_i$) is $K^2 \times K^2$, and it is given by

$$\mathbf{G}_i \widehat{\boldsymbol{\Sigma}}_{\widehat{\alpha}} \mathbf{G}_i'$$

where

$$\mathbf{G}_i = \sum_{m=0}^{i-1} \mathbf{J}(\widehat{\mathbf{M}}')^{(i-1-m)} \otimes \widehat{\boldsymbol{\Phi}}_m \qquad \mathbf{G}_i \text{ is } K^2 \times K^2 p$$

$$\mathbf{J} = (\mathbf{I}_K, \mathbf{0}_K, \ldots, \mathbf{0}_K) \qquad \mathbf{J} \text{ is } K \times K p$$

$$\widehat{\mathbf{M}} = \begin{bmatrix} \widehat{\mathbf{A}}_1 & \widehat{\mathbf{A}}_2 & \ldots & \widehat{\mathbf{A}}_{p-1} & \widehat{\mathbf{A}}_p \\ \mathbf{I}_K & \mathbf{0}_K & \ldots & \mathbf{0}_K & \mathbf{0}_K \\ \mathbf{0}_K & \mathbf{I}_K & & \mathbf{0}_K & \mathbf{0}_K \\ \vdots & & \ddots & \vdots & \vdots \\ \mathbf{0}_K & \mathbf{0}_K & \ldots & \mathbf{I}_K & \mathbf{0}_K \end{bmatrix} \qquad \widehat{\mathbf{M}} \text{ is } K p \times K p$$

The $\widehat{\mathbf{A}}_i$ are the estimates of the coefficients on the lagged variables in the VAR, and $\widehat{\boldsymbol{\Sigma}}_{\widehat{\alpha}}$ is the VCE matrix of $\widehat{\boldsymbol{\alpha}} = \text{vec}(\widehat{\mathbf{A}}_1, \ldots, \widehat{\mathbf{A}}_p)$. $\widehat{\boldsymbol{\Sigma}}_{\widehat{\alpha}}$ is a $K^2 p \times K^2 p$ matrix whose elements come from the VCE of the VAR coefficient estimator. As such, this VCE is the VCE of the constrained estimator if there are any constraints placed on the VAR coefficients.

The $K^2 \times K^2$ asymptotic VCE matrix for vec($\widehat{\boldsymbol{\Psi}}_n$) after n periods is given by

$$\mathbf{F}_n \widehat{\boldsymbol{\Sigma}}_{\widehat{\alpha}} \mathbf{F}_n'$$

where

$$\mathbf{F}_n = \sum_{i=1}^{n} \mathbf{G}_i$$

The $K^2 \times K^2$ asymptotic VCE matrix of the vectorized, orthogonalized, IRFs at horizon i, vec($\boldsymbol{\Theta}_i^o$), is

$$\mathbf{C}_i \widehat{\boldsymbol{\Sigma}}_{\widehat{\alpha}} \mathbf{C}_i' + \overline{\mathbf{C}}_i \widehat{\boldsymbol{\Sigma}}_{\widehat{\sigma}} \overline{\mathbf{C}}_i'$$

where

$$\mathbf{C}_0 = \mathbf{0} \qquad\qquad \mathbf{C}_0 \text{ is } K^2 \times K^2 p$$

$$\mathbf{C}_i = (\widehat{\mathbf{P}}_c' \otimes \mathbf{I}_K)\mathbf{G}_i, \quad i = 1, 2, \dots \qquad\qquad \mathbf{C}_i \text{ is } K^2 \times K^2 p$$

$$\overline{\mathbf{C}}_i = (\mathbf{I}_K \otimes \boldsymbol{\Phi}_i)\mathbf{H}, \quad i = 0, 1, \dots \qquad\qquad \overline{\mathbf{C}}_i \text{ is } K^2 \times K^2$$

$$\mathbf{H} = \mathbf{L}_K' \left\{ \mathbf{L}_K \mathbf{N}_K (\widehat{\mathbf{P}}_c \otimes \mathbf{I}_K)\mathbf{L}_K' \right\}^{-1} \qquad\qquad \mathbf{H} \text{ is } K^2 \times K\frac{(K+1)}{2}$$

$\mathbf{L}_K$ solves
$$\text{vech}(\mathbf{F}) = \mathbf{L}_K \text{ vec}(\mathbf{F}) \qquad\qquad \mathbf{L}_K \text{ is } K\frac{(K+1)}{2} \times K^2$$

for $\mathbf{F}$ $K \times K$ and symmetric

$\mathbf{K}_K$ solves $\quad \mathbf{K}_K \text{vec}(\mathbf{G}) = \text{vec}(\mathbf{G}')$ for any $K \times K$ matrix $\mathbf{G}$ $\qquad \mathbf{K}_K \text{ is } K^2 \times K^2$

$$\mathbf{N}_K = \frac{1}{2}\left(\mathbf{I}_{K^2} + \mathbf{K}_K\right) \qquad\qquad \mathbf{N}_K \text{ is } K^2 \times K^2$$

$$\widehat{\boldsymbol{\Sigma}}_{\widehat{\sigma}} = 2\mathbf{D}_K^+(\widehat{\boldsymbol{\Sigma}} \otimes \widehat{\boldsymbol{\Sigma}})\mathbf{D}_K^+ \qquad\qquad \widehat{\boldsymbol{\Sigma}}_{\widehat{\sigma}} \text{ is } K\frac{(K+1)}{2} \times K\frac{(K+1)}{2}$$

$$\mathbf{D}_K^+ = \left(\mathbf{D}_K'\mathbf{D}_K\right)^{-1}\mathbf{D}_K' \qquad\qquad \mathbf{D}_K^+ \text{ is } K\frac{(K+1)}{2} \times K^2$$

$\mathbf{D}_K$ solves $\quad \mathbf{D}_K \text{vech}(\mathbf{F}) = \text{vec}(\mathbf{F}) \quad$ for $\mathbf{F}$ $K \times K$ and symmetric $\qquad \mathbf{D}_K \text{ is } K^2 \times K\frac{(K+1)}{2}$

$$\text{vech}(\mathbf{X}) = \begin{bmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{K1} \\ x_{22} \\ \vdots \\ x_{K2} \\ \vdots \\ x_{KK} \end{bmatrix} \qquad \text{for } \mathbf{X} \ K \times K \qquad\qquad \text{vech}(\mathbf{X}) \text{ is } K\frac{(K+1)}{2} \times 1$$

Note that $\widehat{\boldsymbol{\Sigma}}_{\widehat{\sigma}}$ is the VCE of vech($\widehat{\boldsymbol{\Sigma}}$). More details about $\mathbf{L}_K$, $\mathbf{K}_K$, $\mathbf{D}_K$ and vech() are available in Lütkepohl (2005, sec. A.12). Finally, as Lütkepohl (2005, 113–114) discusses, $\mathbf{D}_K^+$ is the Moore–Penrose inverse of $\mathbf{D}_K$.

As discussed in Amisano and Giannini (1997, chap. 6), the asymptotic standard errors of the structural IRFs are available for short-run SVAR models but not for long-run SVAR models. Following Amisano and Giannini (1997, chap. 5), the asymptotic $K^2 \times K^2$ VCE of the short-run structural IRFs after i periods, when a maximum of h periods are estimated, is the i, i block of

$$\widehat{\boldsymbol{\Sigma}}(h)_{ij} = \widetilde{\mathbf{G}}_i \widehat{\boldsymbol{\Sigma}}_{\widehat{\alpha}} \widetilde{\mathbf{G}}_j' + \left\{ \mathbf{I}_K \otimes (\mathbf{J}\widehat{\mathbf{M}}^i\mathbf{J}') \right\} \boldsymbol{\Sigma}(0) \left\{ \mathbf{I}_K \otimes (\mathbf{J}\widehat{\mathbf{M}}^j\mathbf{J}') \right\}'$$

where

$$\widetilde{\mathbf{G}}_0 = \mathbf{0}_K \qquad\qquad \mathbf{G}_0 \text{ is } K^2 \times K^2 p$$

$$\widetilde{\mathbf{G}}_i = \sum_{k=0}^{i-1}\left\{\widehat{\mathbf{P}}'_{\mathrm{sr}}\mathbf{J}(\widehat{\mathbf{M}}')^{i-1-k}\otimes\left(\mathbf{J}\widehat{\mathbf{M}}^k\mathbf{J}'\right)\right\} \qquad \mathbf{G}_i \text{ is } K^2 \times K^2 p$$

$$\widehat{\mathbf{\Sigma}}(0) = \mathbf{Q}_2\widehat{\mathbf{\Sigma}}_W\mathbf{Q}'_2 \qquad\qquad \widehat{\mathbf{\Sigma}}(0) \text{ is } K^2 \times K^2$$

$$\widehat{\mathbf{\Sigma}}_W = \mathbf{Q}_1\widehat{\mathbf{\Sigma}}_{AB}\mathbf{Q}'_1 \qquad\qquad \widehat{\mathbf{\Sigma}}_W \text{ is } K^2 \times K^2$$

$$\mathbf{Q}_2 = \widehat{\mathbf{P}}'_{\mathrm{sr}}\otimes\widehat{\mathbf{P}}_{\mathrm{sr}} \qquad\qquad \mathbf{Q}_2 \text{ is } K^2 \times K^2$$

$$\mathbf{Q}_1 = \left\{(\mathbf{I}_K\otimes\widehat{\mathbf{B}}^{-1}), (-\widehat{\mathbf{P}}'^{-1}_{\mathrm{sr}}\otimes\mathbf{B}^{-1})\right\} \qquad \mathbf{Q}_1 \text{ is } K^2 \times 2K^2$$

and $\widehat{\mathbf{\Sigma}}_{AB}$ is the $2K^2 \times 2K^2$ VCE of the estimator of $\mathrm{vec}(\mathbf{A},\mathbf{B})$.

Dynamic-multiplier function formulas for VARs

This section provides the details of how irf create estimates the dynamic-multiplier functions and their asymptotic standard errors.

A pth order vector autoregressive model (VAR) with exogenous variables may be written as

$$\mathbf{y}_t = \mathbf{v} + \mathbf{A}_1\mathbf{y}_{t-1} + \cdots + \mathbf{A}_p\mathbf{y}_{t-p} + \mathbf{B}_0\mathbf{x}_t + \mathbf{B}_1\mathbf{x}_{t-1} + \cdots + \mathbf{B}_s\mathbf{x}_{t-s} + \mathbf{u}_t$$

where all the notation is the same as above except that the s $K \times R$ matrices $\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_s$ are explicitly included and s is the number of lags of the R exogenous variables in the model.

Lütkepohl (2005) shows that the dynamic-multipliers $\mathbf{D}_i$ are consistently estimated by

$$\widehat{\mathbf{D}}_i = \mathbf{J}_x\widetilde{\mathbf{A}}^i_x\widehat{\mathbf{B}}_x \qquad i \in \{0, 1, \ldots\}$$

where

$$\mathbf{J}_x = (\mathbf{I}_K, \mathbf{0}_K, \ldots, \mathbf{0}_K) \qquad\qquad \mathbf{J} \text{ is } K \times (Kp+Rs)$$

$$\widetilde{\mathbf{A}}_x = \begin{bmatrix} \widehat{\mathbf{M}} & \widehat{\mathbf{B}} \\ \widetilde{\mathbf{0}} & \widetilde{\mathbf{I}} \end{bmatrix} \qquad\qquad \widetilde{\mathbf{A}}_x \text{ is } (Kp+Rs) \times (Kp+Rs)$$

$$\widehat{\mathbf{B}} = \begin{bmatrix} \widehat{\mathbf{B}}_1 & \widehat{\mathbf{B}}_2 & \ldots & \widehat{\mathbf{B}}_s \\ \ddot{\mathbf{0}} & \ddot{\mathbf{0}} & \ldots & \ddot{\mathbf{0}} \\ \vdots & \vdots & \ddots & \vdots \\ \ddot{\mathbf{0}} & \ddot{\mathbf{0}} & \ldots & \ddot{\mathbf{0}} \end{bmatrix} \qquad\qquad \widehat{\mathbf{B}} \text{ is } Kp \times Rs$$

$$\widetilde{\mathbf{I}} = \begin{bmatrix} \mathbf{0}_R & \mathbf{0}_R & \ldots & \mathbf{0}_R & \mathbf{0}_R \\ \mathbf{I}_R & \mathbf{0}_R & \ldots & \mathbf{0}_R & \mathbf{0}_R \\ \mathbf{0}_R & \mathbf{I}_R & & \mathbf{0}_R & \mathbf{0}_R \\ \vdots & & \ddots & \vdots & \vdots \\ \mathbf{0}_R & \mathbf{0}_R & \ldots & \mathbf{I}_R & \mathbf{0}_R \end{bmatrix} \qquad \widetilde{\mathbf{I}} \text{ is } Rs \times Rs$$

$$\widehat{\mathbf{B}}'_x = \begin{bmatrix} \widetilde{\mathbf{B}}' & \ddot{\mathbf{I}}' \end{bmatrix} \qquad\qquad \widehat{\mathbf{B}}'_x \text{ is } R \times (Kp+Rs)$$

$$\widetilde{\mathbf{B}}' = \begin{bmatrix} \widehat{\mathbf{B}}'_0 & \ddot{\mathbf{0}}' \cdots \ddot{\mathbf{0}}' \end{bmatrix} \qquad\qquad \widetilde{\mathbf{B}} \text{ is } R \times Kp$$

$$\ddot{\mathbf{I}}' = \begin{bmatrix} \mathbf{I}_R & \mathbf{0}_R \cdots \mathbf{0}_R \end{bmatrix} \qquad\qquad \ddot{\mathbf{I}} \text{ is } R \times Rs$$

and $\ddot{\mathbf{0}}$ is a $K \times R$ matrix of 0s and $\widetilde{\mathbf{0}}$ is a $Rs \times Kp$ matrix of 0s.

Consistent estimators of the cumulative dynamic-multiplier functions are given by

$$\overline{\mathbf{D}}_i = \sum_{j=0}^{i} \widehat{\mathbf{D}}_j$$

Letting $\boldsymbol{\beta}_x = \text{vec}[\mathbf{A}_1 \mathbf{A}_2 \cdots \mathbf{A}_p \mathbf{B}_1 \mathbf{B}_2 \cdots \mathbf{B}_s \mathbf{B}_0]$ and letting $\boldsymbol{\Sigma}_{\widehat{\boldsymbol{\beta}}_x}$ be the asymptotic variance–covariance estimator (VCE) of $\widehat{\boldsymbol{\beta}}_x$, Lütkepohl shows that an asymptotic VCE of $\widehat{\mathbf{D}}_i$ is $\widetilde{\mathbf{G}}_i \boldsymbol{\Sigma}_{\widehat{\boldsymbol{\beta}}_x} \widetilde{\mathbf{G}}_i'$ where

$$\widetilde{\mathbf{G}}_i = \left[\sum_{j=0}^{i-1} \mathbf{B}_x' \widetilde{\mathbf{A}}_x^{i-1-j} \otimes \mathbf{J}_x \widetilde{\mathbf{A}}_x^j \mathbf{J}_x', \ \mathbf{I}_R \otimes \mathbf{J}_x \widetilde{\mathbf{A}}_x^j \mathbf{J}_x \right]$$

Similarly, an asymptotic VCE of $\bar{\mathbf{D}}_i$ is $\left(\sum_{j=0}^{i} \widetilde{\mathbf{G}}_j \right) \boldsymbol{\Sigma}_{\widehat{\boldsymbol{\beta}}_x} \left(\sum_{j=0}^{i} \widetilde{\mathbf{G}}_j' \right)$.

Forecast-error variance decomposition formulas for VARs

This section provides details of how irf create estimates the Cholesky FEVD, the structural FEVD, and their standard errors. Beginning with the Cholesky-based forecast-error decompositions, the fraction of the h-step-ahead forecast-error variance of variable j that is attributable to the Cholesky orthogonalized innovations in variable k can be estimated as

$$\widehat{\omega}_{jk,h} = \frac{\sum_{i=0}^{h-1} (\mathbf{e}_j' \widehat{\boldsymbol{\Theta}}_i \mathbf{e}_k)^2}{\widehat{\text{MSE}}_j(h)}$$

where $\text{MSE}_j(h)$ is the jth diagonal element of

$$\sum_{i=0}^{h-1} \widehat{\boldsymbol{\Phi}}_i \widehat{\boldsymbol{\Sigma}} \widehat{\boldsymbol{\Phi}}_i'$$

(See Lütkepohl [2005, 109] for a discussion of this result.) $\widehat{\omega}_{jk,h}$ and $\text{MSE}_j(h)$ are scalars. The square of the standard error of $\widehat{\omega}_{jk,h}$ is

$$\mathbf{d}_{jk,h} \widehat{\boldsymbol{\Sigma}}_\alpha \mathbf{d}_{jk,h}' + \overline{\mathbf{d}}_{jk,h} \widehat{\boldsymbol{\Sigma}}_\sigma \overline{\mathbf{d}}_{jk,h}$$

where

$$
\begin{aligned}
\mathbf{d}_{jk,h} \ &= \ \frac{2}{\text{MSE}_j(h)^2} \sum_{i=0}^{h-1} \Bigg\{ \text{MSE}_j(h) (\mathbf{e}_j' \widehat{\boldsymbol{\Phi}}_i \widehat{\mathbf{P}}_c \mathbf{e}_k)(\mathbf{e}_k' \widehat{\mathbf{P}}_c' \otimes \mathbf{e}_j') \mathbf{G}_i \\
&\quad - (\mathbf{e}_j' \boldsymbol{\Phi}_i \widehat{\mathbf{P}}_c \mathbf{e}_k)^2 \sum_{m=0}^{h-1} (\mathbf{e}_j' \widehat{\boldsymbol{\Phi}}_m \widehat{\boldsymbol{\Sigma}} \otimes \mathbf{e}_j') \mathbf{G}_m \Bigg\} \qquad \mathbf{d}_{jk,h} \text{ is } 1 \times K^2 p
\end{aligned}
$$

$$
\begin{aligned}
\overline{\mathbf{d}}_{jk,h} \ &= \ \sum_{i=0}^{h-1} \Bigg\{ \text{MSE}_j(h)(\mathbf{e}_j' \widehat{\boldsymbol{\Phi}}_i \mathbf{P}_c \mathbf{e}_k)(\mathbf{e}_k' \otimes \mathbf{e}_j' \widehat{\boldsymbol{\Phi}}_i) \mathbf{H} \\
&\quad - (\mathbf{e}_j' \widehat{\boldsymbol{\Phi}}_i \widehat{\mathbf{P}}_c \mathbf{e}_k)^2 \sum_{m=0}^{h-1} (\mathbf{e}_j' \widehat{\boldsymbol{\Phi}}_m \otimes \mathbf{e}_j \widehat{\boldsymbol{\Phi}}_m) \mathbf{D}_K \Bigg\} \frac{1}{\text{MSE}_j(h)^2} \qquad \overline{\mathbf{d}}_{jk,h} \text{ is } 1 \times K \frac{(K+1)}{2}
\end{aligned}
$$

$$\mathbf{G}_0 \ = \ \mathbf{0} \qquad\qquad\qquad \mathbf{G}_0 \text{ is } K^2 \times K^2 p$$

and $\mathbf{D}_K$ is the $K^2 \times K\{(K+1)/2\}$ duplication matrix defined previously.

For the structural forecast-error decompositions, we follow Amisano and Giannini (1997, sec. 5.2). They define the matrix of structural forecast-error decompositions at horizon s, when a maximum of h periods are estimated, as

$$\widehat{\mathbf{W}}_s = \widehat{\mathbf{F}}_s^{-1} \widehat{\widetilde{\mathbf{M}}}_s \qquad \text{for } s = 1, \ldots, h+1$$

$$\widehat{\mathbf{F}}_s = \left(\sum_{i=0}^{s-1} \widehat{\mathbf{\Theta}}_i^{\mathrm{sr}} \widehat{\mathbf{\Theta}}_i^{\mathrm{sr}\prime} \right) \odot \mathbf{I}_K$$

$$\widehat{\widetilde{\mathbf{M}}}_s = \sum_{i=0}^{s-1} \widehat{\mathbf{\Theta}}_i^{\mathrm{sr}} \odot \widehat{\mathbf{\Theta}}_i^{\mathrm{sr}}$$

where $\odot$ is the Hadamard, or element-by-element, product.

The $K^2 \times K^2$ asymptotic VCE of $\mathrm{vec}(\widehat{\mathbf{W}}_s)$ is given by

$$\widetilde{\mathbf{Z}}_s \mathbf{\Sigma}(h) \widetilde{\mathbf{Z}}_s'$$

where $\widehat{\mathbf{\Sigma}}(h)$ is as derived previously, and

$$\widetilde{\mathbf{Z}}_s = \left\{ \frac{\partial \mathrm{vec}(\widehat{\mathbf{W}}_s)}{\partial \mathrm{vec}(\widehat{\mathbf{\Theta}}_0^{\mathrm{sr}})}, \frac{\partial \mathrm{vec}(\widehat{\mathbf{W}}_s)}{\partial \mathrm{vec}(\widehat{\mathbf{\Theta}}_1^{\mathrm{sr}})}, \cdots, \frac{\partial \mathrm{vec}(\widehat{\mathbf{W}}_s)}{\partial \mathrm{vec}(\widehat{\mathbf{\Theta}}_h^{\mathrm{sr}})} \right\}$$

$$\frac{\partial \mathrm{vec}(\widehat{\mathbf{W}}_s)}{\partial \mathrm{vec}(\widehat{\mathbf{\Theta}}_j^{\mathrm{sr}})} = 2 \left\{ (\mathbf{I}_K \otimes \widehat{\mathbf{F}}_s^{-1}) \widetilde{\mathbf{D}}(\widehat{\mathbf{\Theta}}_j^{\mathrm{sr}}) - (\widehat{\mathbf{W}}_s' \otimes \widehat{\mathbf{F}}_s^{-1}) \widetilde{\mathbf{D}}(\mathbf{I}_K) \mathbf{N}_K (\widehat{\mathbf{\Theta}}_j^{\mathrm{sr}} \otimes I_K) \right\}$$

If $\mathbf{X}$ is an $n \times n$ matrix, then $\widetilde{\mathbf{D}}(\mathbf{X})$ is the $n^2 \times n^2$ matrix with $\mathrm{vec}(\mathbf{X})$ on the diagonal and zeros in all the off-diagonal elements, and $\mathbf{N}_K$ is as defined previously.

Impulse–response function formulas for VECMs

We begin by providing the formulas for backing out the estimates of the $\mathbf{A}_i$ from the $\mathbf{\Gamma}_i$ estimated by vec. As discussed in [TS] **vec intro**, the VAR in (1) can be rewritten as a VECM:

$$\Delta y_t = v + \mathbf{\Pi} y_{t-1} + \Gamma_1 \Delta y_{t-1} + \Gamma_{p-1} \Delta y_{p-2} + \epsilon_t$$

vec estimates $\mathbf{\Pi}$ and the $\mathbf{\Gamma}_i$. Johansen (1995, 25) notes that

$$\mathbf{\Pi} = \sum_{i=1}^{p} \mathbf{A}_i - \mathbf{I}_K \tag{6}$$

where $\mathbf{I}_K$ is the K-dimensional identity matrix, and

$$\mathbf{\Gamma}_i = -\sum_{j=i+1}^{p} \mathbf{A}_j \tag{7}$$

Defining

$$\mathbf{\Gamma} = \mathbf{I}_K - \sum_{i=1}^{p-1} \mathbf{\Gamma}_i$$

and using (6) and (7) allow us to solve for the $\mathbf{A}_i$ as

$$\mathbf{A}_1 = \mathbf{\Pi} + \mathbf{\Gamma}_1 + \mathbf{I}_K$$

$$\mathbf{A}_i = \mathbf{\Gamma}_i - \mathbf{\Gamma}_{i-1} \quad \text{for } i = \{2, \ldots, p-1\}$$

and

$$\mathbf{A}_p = -\mathbf{\Gamma}_{p-1}$$

Using these formulas, we can back out estimates of $\mathbf{A}_i$ from the estimates of the $\mathbf{\Gamma}_i$ and $\mathbf{\Pi}$ produced by `vec`. Then we simply use the formulas for the IRFs and OIRFs presented in *Impulse–response function formulas for VARs*.

The running sums of the IRFs and OIRFs over the steps within each impulse–response pair are the cumulative IRFs and OIRFs.

Algorithms for bootstrapping the VAR IRF and FEVD standard errors

`irf create` offers two bootstrap algorithms for estimating the standard errors of the various IRFs and FEVDs. Both `var` and `svar` contain estimators for the coefficients in a VAR that are conditional on the first p observations. The two bootstrap algorithms are also conditional on the first p observations.

Specifying the `bs` option calculates the standard errors by bootstrapping the residuals. For a bootstrap with R repetitions, this method uses the following algorithm:

1. Fit the model and save the estimated parameters.

2. Use the estimated coefficients to calculate the residuals.

3. Repeat steps 3a to 3c R times.

    3a. Draw a simple random sample of size T with replacement from the residuals. The random samples are drawn over the $K \times 1$ vectors of residuals. When the tth vector is drawn, all K residuals are selected. This preserves the contemporaneous correlations among the residuals.

    3b. Use the p initial observations, the sampled residuals, and the estimated coefficients to construct a new sample dataset.

    3c. Fit the model and calculate the different IRFs and FEVDs.

    3d. Save these estimates as observation r in the bootstrapped dataset.

4. For each IRF and FEVD, the estimated standard deviation from the R bootstrapped estimates is the estimated standard error of that impulse–response function or forecast-error variance decomposition.

Specifying the `bsp` option estimates the standard errors by a multivariate normal parametric bootstrap. The algorithm for the multivariate normal parametric bootstrap is identical to the one above, with the exception that 3a is replaced by 3a(bsp):

    3a(bsp). Draw T pseudovariates from a multivariate normal distribution with covariance matrix $\widehat{\mathbf{\Sigma}}$.

Impulse–response function formulas for ARIMA and ARFIMA

The previous discussion showed that a SARMA process can be rewritten as an ARMA process and that for an ARMA process, we can express $\boldsymbol{\psi}(L)$ in terms of $\boldsymbol{\theta}(L)$ and $\boldsymbol{\rho}(L)$,

$$\boldsymbol{\psi}(L) = \frac{\boldsymbol{\theta}(L)}{\boldsymbol{\rho}(L)}$$

Expanding the above, we obtain

$$\psi_0 + \psi_1 L + \psi_2 L^2 + \cdots = \frac{1 + \theta_1 L + \theta_2 L^2 + \cdots}{1 - \rho_1 L - \rho_2 L^2 - \cdots}$$

Given the estimate of the autoregressive terms $\widehat{\boldsymbol{\rho}}$ and the moving-average terms $\widehat{\boldsymbol{\theta}}$, the IRF is obtained by solving the above equation for the ψ weights. The ψ_i are calculated using the recursion

$$\widehat{\psi}_i = \widehat{\theta}_i + \sum_{j=1}^{p} \widehat{\phi}_j \widehat{\psi}_{i-j}$$

with $\psi_0 = 1$ and $\theta_i = 0$ for $i > \max(p, q+1)$.

The asymptotic standard errors for the IRF for ARMA are calculated using the delta method; see Serfling (1980, sec. 3.3) for a discussion of the delta method. Let $\widehat{\boldsymbol{\Sigma}}$ be the estimate of the variance–covariance matrix for $\widehat{\boldsymbol{\rho}}$ and $\widehat{\boldsymbol{\theta}}$, and let $\boldsymbol{\Psi}$ be a matrix of derivatives of ψ_i with respect to $\widehat{\boldsymbol{\rho}}$ and $\widehat{\boldsymbol{\theta}}$. Then the standard errors for $\widehat{\psi}_i$ are calculated as

$$\boldsymbol{\Psi}_i \widehat{\boldsymbol{\Sigma}} \boldsymbol{\Psi}_i'$$

The IRF for the ARFIMA(p, d, q) model is obtained by applying the filter $(1 - L)^{-d}$ to $\boldsymbol{\psi}(\mathrm{L})$. The filter is given by Hassler and Kokoszka (2010) as

$$(1 - L)^{-d} = \sum_{i=0}^{\infty} b_i L^i$$

with $b_0 = 1$ and subsequent b_i calculated by the recursion

$$\widehat{b}_i = \frac{\widehat{d} + i - 1}{i} \widehat{b}_{i-1}$$

The resulting IRF is then given by

$$\widehat{\phi}_i = \sum_{j=0}^{i} \widehat{\psi}_j \widehat{b}_{i-j}$$

The asymptotic standard errors for the IRF for ARFIMA are calculated using the delta method. Let $\widehat{\boldsymbol{\Sigma}}$ be the estimate of the variance–covariance matrix for $\widehat{\boldsymbol{\rho}}$, $\widehat{\boldsymbol{\theta}}$, and $\widehat{d}$, and let $\boldsymbol{\Phi}$ be a matrix of derivatives of ϕ_i with respect to $\widehat{\boldsymbol{\rho}}$, $\widehat{\boldsymbol{\theta}}$, and $\widehat{d}$. Then the standard errors for $\widehat{\phi}_i$ are calculated as

$$\boldsymbol{\Phi}_i \widehat{\boldsymbol{\Sigma}} \boldsymbol{\Phi}_i'$$

References

Amisano, G., and C. Giannini. 1997. *Topics in Structural VAR Econometrics*. 2nd ed. Heidelberg: Springer.

Christiano, L. J., M. Eichenbaum, and C. L. Evans. 1999. Monetary policy shocks: What have we learned and to what end? In *Handbook of Macroeconomics: Volume 1A*, ed. J. B. Taylor and M. Woodford. New York: Elsevier.

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

Hassler, U., and P. Kokoszka. 2010. Impulse responses of fractionally integrated processes with long memory. *Econometric Theory* 26: 1855–1861.

Johansen, S. 1995. *Likelihood-Based Inference in Cointegrated Vector Autoregressive Models*. Oxford: Oxford 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.

Serfling, R. J. 1980. *Approximation Theorems of Mathematical Statistics*. New York: Wiley.

Sims, C. A. 1980. Macroeconomics and reality. *Econometrica* 48: 1–48.

Stock, J. H., and M. W. Watson. 2001. Vector autoregressions. *Journal of Economic Perspectives* 15: 101–115.

Also see

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

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

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