Title

vec intro — Introduction to vector error-correction models

Description   Remarks and examples   References   Also see

Description

Stata has a suite of commands for fitting, forecasting, interpreting, and performing inference on vector error-correction models (VECMs) with cointegrating variables. After fitting a VECM, the irf commands can be used to obtain impulse–response functions (IRFs) and forecast-error variance decompositions (FEVDs). The table below describes the available commands.

Fitting a VECM

vec	[TS] **vec**	Fit vector error-correction models

Model diagnostics and inference

vecrank	[TS] **vecrank**	Estimate the cointegrating rank of a VECM
veclmar	[TS] **veclmar**	Perform LM test for residual autocorrelation after vec
vecnorm	[TS] **vecnorm**	Test for normally distributed disturbances after vec
vecstable	[TS] **vecstable**	Check the stability condition of VECM estimates
varsoc	[TS] **varsoc**	Obtain lag-order selection statistics for VARs and VECMs

Forecasting from a VECM

fcast compute	[TS] **fcast compute**	Compute dynamic forecasts after var, svar, or vec
fcast graph	[TS] **fcast graph**	Graph forecasts after fcast compute

Working with IRFs and FEVDs

irf	[TS] **irf**	Create and analyze IRFs and FEVDs

This manual entry provides an overview of the commands for VECMs; provides an introduction to integration, cointegration, estimation, inference, and interpretation of VECM models; and gives an example of how to use Stata's vec commands.

Remarks and examples

vec estimates the parameters of cointegrating VECMs. You may specify any of the five trend specifications in Johansen (1995, sec. 5.7). By default, identification is obtained via the Johansen normalization, but vec allows you to obtain identification by placing your own constraints on the parameters of the cointegrating vectors. You may also put more restrictions on the adjustment coefficients.

vecrank is the command for determining the number of cointegrating equations. vecrank implements Johansen's multiple trace test procedure, the maximum eigenvalue test, and a method based on minimizing either of two different information criteria.

Because Nielsen (2001) has shown that the methods implemented in `varsoc` can be used to choose the order of the autoregressive process, no separate `vec` command is needed; you can simply use `varsoc`. `veclmar` tests that the residuals have no serial correlation, and `vecnorm` tests that they are normally distributed.

All the `irf` routines described in [TS] **irf** are available for estimating, interpreting, and managing estimated IRFs and FEVDs for VECMs.

Remarks are presented under the following headings:

Introduction to cointegrating VECMs
What is cointegration?
The multivariate VECM specification
Trends in the Johansen VECM framework
VECM estimation in Stata
Selecting the number of lags
Testing for cointegration
Fitting a VECM
Fitting VECMs with Johansen's normalization
Postestimation specification testing
Impulse–response functions for VECMs
Forecasting with VECMs

Introduction to cointegrating VECMs

This section provides a brief introduction to integration, cointegration, and cointegrated vector error-correction models. For more details about these topics, see Hamilton (1994), Johansen (1995), Lütkepohl (2005), Watson (1994), and Becketti (2020).

What is cointegration?

Standard regression techniques, such as ordinary least squares (OLS), require that the variables be covariance stationary. A variable is covariance stationary if its mean and all its autocovariances are finite and do not change over time. Cointegration analysis provides a framework for estimation, inference, and interpretation when the variables are not covariance stationary.

Instead of being covariance stationary, many economic time series appear to be "first-difference stationary". This means that the level of a time series is not stationary but its first difference is. First-difference stationary processes are also known as integrated processes of order 1, or I(1) processes. Covariance-stationary processes are I(0). In general, a process whose dth difference is stationary is an integrated process of order d, or I(d).

The canonical example of a first-difference stationary process is the random walk. This is a variable x_t that can be written as

$$x_t = x_{t-1} + \epsilon_t \tag{1}$$

where the ϵ_t are independent and identically distributed with mean zero and a finite variance σ^2. Although $E[x_t] = 0$ for all t, $\text{Var}[x_t] = T\sigma^2$ is not time invariant, so x_t is not covariance stationary. Because $\Delta x_t = x_t - x_{t-1} = \epsilon_t$ and ϵ_t is covariance stationary, x_t is first-difference stationary.

These concepts are important because, although conventional estimators are well behaved when applied to covariance-stationary data, they have nonstandard asymptotic distributions and different rates of convergence when applied to I(1) processes. To illustrate, consider several variants of the model

$$y_t = a + bx_t + e_t \tag{2}$$

Throughout the discussion, we maintain the assumption that $E[e_t] = 0$.

If both y_t and x_t are covariance-stationary processes, e_t must also be covariance stationary. As long as $E[x_t e_t] = 0$, we can consistently estimate the parameters a and b by using OLS. Furthermore, the distribution of the OLS estimator converges to a normal distribution centered at the true value as the sample size grows.

If y_t and x_t are independent random walks and $b = 0$, there is no relationship between y_t and x_t, and (2) is called a spurious regression. Granger and Newbold (1974) performed Monte Carlo experiments and showed that the usual t statistics from OLS regression provide spurious results: given a large enough dataset, we can almost always reject the null hypothesis of the test that $b = 0$ even though b is in fact zero. Here the OLS estimator does not converge to any well-defined population parameter.

Phillips (1986) later provided the asymptotic theory that explained the Granger and Newbold (1974) results. He showed that the random walks y_t and x_t are first-difference stationary processes and that the OLS estimator does not have its usual asymptotic properties when the variables are first-difference stationary.

Because Δy_t and Δx_t are covariance stationary, a simple regression of Δy_t on Δx_t appears to be a viable alternative. However, if y_t and x_t cointegrate, as defined below, the simple regression of Δy_t on Δx_t is misspecified.

If y_t and x_t are I(1) and $b \neq 0$, e_t could be either I(0) or I(1). Phillips and Durlauf (1986) have derived the asymptotic theory for the OLS estimator when e_t is I(1), though it has not been widely used in applied work. More interesting is the case in which $e_t = y_t - a - bx_t$ is I(0). y_t and x_t are then said to be cointegrated. Two variables are cointegrated if each is an I(1) process but a linear combination of them is an I(0) process.

It is not possible for y_t to be a random walk and x_t and e_t to be covariance stationary. As Granger (1981) pointed out, because a random walk cannot be equal to a covariance-stationary process, the equation does not "balance". An equation balances when the processes on each side of the equal sign are of the same order of integration. Before attacking any applied problem with integrated variables, make sure that the equation balances before proceeding.

An example from Engle and Granger (1987) provides more intuition. Redefine y_t and x_t to be

$$y_t + \beta x_t = \epsilon_t, \qquad \epsilon_t = \epsilon_{t-1} + \xi_t \tag{3}$$
$$y_t + \alpha x_t = \nu_t, \qquad \nu_t = \rho\nu_{t-1} + \zeta_t, \quad |\rho| < 1 \tag{4}$$

where ξ_t and ζ_t are i.i.d. disturbances over time that are correlated with each other. Because ϵ_t is I(1), (3) and (4) imply that both x_t and y_t are I(1). The condition that $|\rho| < 1$ implies that ν_t and $y_t + \alpha x_t$ are I(0). Thus y_t and x_t cointegrate, and $(1, \alpha)$ is the cointegrating vector.

Using a bit of algebra, we can rewrite (3) and (4) as

$$\Delta y_t = \beta\delta z_{t-1} + \eta_{1t} \tag{5}$$
$$\Delta x_t = -\delta z_{t-1} + \eta_{2t} \tag{6}$$

where $\delta = (1-\rho)/(\alpha-\beta)$, $z_t = y_t + \alpha x_t$, and η_{1t} and η_{2t} are distinct, stationary, linear combinations of ξ_t and ζ_t. This representation is known as the vector error-correction model (VECM). One can think of $z_t = 0$ as being the point at which y_t and x_t are in equilibrium. The coefficients on z_{t-1} describe how y_t and x_t adjust to z_{t-1} being nonzero, or out of equilibrium. z_t is the "error" in the system, and (5) and (6) describe how system adjusts or corrects back to the equilibrium. As $\rho \to 1$, the system degenerates into a pair of correlated random walks. The VECM parameterization highlights this point, because $\delta \to 0$ as $\rho \to 1$.

If we knew α, we would know z_t, and we could work with the stationary system of (5) and (6). Although knowing α seems silly, we can conduct much of the analysis as if we knew α because there is an estimator for the cointegrating parameter α that converges to its true value at a faster rate than the estimator for the adjustment parameters β and δ.

The definition of a bivariate cointegrating relation requires simply that there exist a linear combination of the I(1) variables that is I(0). If y_t and x_t are I(1) and there are two finite real numbers $a \neq 0$ and $b \neq 0$, such that $ay_t + bx_t$ is I(0), then y_t and x_t are cointegrated. Although there are two parameters, a and b, only one will be identifiable because if $ay_t + bx_t$ is I(0), so is $cay_t + cbx_t$ for any finite, nonzero, real number c. Obtaining identification in the bivariate case is relatively simple. The coefficient on y_t in (4) is unity. This natural construction of the model placed the necessary identification restriction on the cointegrating vector. As we discuss below, identification in the multivariate case is more involved.

If $\mathbf{y}_t$ is a $K \times 1$ vector of I(1) variables and there exists a vector $\boldsymbol{\beta}$, such that $\boldsymbol{\beta}\mathbf{y}_t$ is a vector of I(0) variables, then $\mathbf{y}_t$ is said to be cointegrating of order $(1,0)$ with cointegrating vector $\boldsymbol{\beta}$. We say that the parameters in $\boldsymbol{\beta}$ are the parameters in the cointegrating equation. For a vector of length K, there may be at most $K-1$ distinct cointegrating vectors. Engle and Granger (1987) provide a more general definition of cointegration, but this one is sufficient for our purposes.

The multivariate VECM specification

In practice, most empirical applications analyze multivariate systems, so the rest of our discussion focuses on that case. Consider a VAR with p lags

$$\mathbf{y}_t = \mathbf{v} + \mathbf{A}_1\mathbf{y}_{t-1} + \mathbf{A}_2\mathbf{y}_{t-2} + \cdots + \mathbf{A}_p\mathbf{y}_{t-p} + \boldsymbol{\epsilon}_t \tag{7}$$

where $\mathbf{y}_t$ is a $K \times 1$ vector of variables, $\mathbf{v}$ is a $K \times 1$ vector of parameters, $\mathbf{A}_1$–$\mathbf{A}_p$ are $K \times K$ matrices of parameters, and $\boldsymbol{\epsilon}_t$ is a $K \times 1$ vector of disturbances. $\boldsymbol{\epsilon}_t$ has mean $\mathbf{0}$, has covariance matrix $\boldsymbol{\Sigma}$, and is i.i.d. normal over time. Any VAR(p) can be rewritten as a VECM. Using some algebra, we can rewrite (7) in VECM form as

$$\Delta\mathbf{y}_t = \mathbf{v} + \boldsymbol{\Pi}\mathbf{y}_{t-1} + \sum_{i=1}^{p-1}\boldsymbol{\Gamma}_i\Delta\mathbf{y}_{t-i} + \boldsymbol{\epsilon}_t \tag{8}$$

where $\boldsymbol{\Pi} = \sum_{j=1}^{j=p}\mathbf{A}_j - \mathbf{I}_k$ and $\boldsymbol{\Gamma}_i = -\sum_{j=i+1}^{j=p}\mathbf{A}_j$. The $\mathbf{v}$ and $\boldsymbol{\epsilon}_t$ in (7) and (8) are identical.

Engle and Granger (1987) show that if the variables $\mathbf{y}_t$ are I(1) the matrix $\boldsymbol{\Pi}$ in (8) has rank $0 \leq r < K$, where r is the number of linearly independent cointegrating vectors. If the variables cointegrate, $0 < r < K$ and (8) shows that a VAR in first differences is misspecified because it omits the lagged level term $\boldsymbol{\Pi}\mathbf{y}_{t-1}$.

Assume that $\boldsymbol{\Pi}$ has reduced rank $0 < r < K$ so that it can be expressed as $\boldsymbol{\Pi} = \boldsymbol{\alpha}\boldsymbol{\beta}'$, where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are both $r \times K$ matrices of rank r. Without further restrictions, the cointegrating vectors are not identified: the parameters $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ are indistinguishable from the parameters $(\boldsymbol{\alpha}\mathbf{Q}, \boldsymbol{\beta}\mathbf{Q}^{-1\prime})$ for any $r \times r$ nonsingular matrix $\mathbf{Q}$. Because only the rank of $\boldsymbol{\Pi}$ is identified, the VECM is said to identify the rank of the cointegrating space, or equivalently, the number of cointegrating vectors. In practice, the estimation of the parameters of a VECM requires at least r^2 identification restrictions. Stata's `vec` command can apply the conventional Johansen restrictions discussed below or use constraints that the user supplies.

The VECM in (8) also nests two important special cases. If the variables in $\mathbf{y}_t$ are I(1) but not cointegrated, $\boldsymbol{\Pi}$ is a matrix of zeros and thus has rank 0. If all the variables are I(0), $\boldsymbol{\Pi}$ has full rank K.

There are several different frameworks for estimation and inference in cointegrating systems. Although the methods in Stata are based on the maximum likelihood (ML) methods developed by Johansen (1988, 1991, 1995), other useful frameworks have been developed by Park and Phillips (1988, 1989); Sims, Stock, and Watson (1990); Stock (1987); and Stock and Watson (1988); among others. The ML framework developed by Johansen was independently developed by Ahn and Reinsel (1990). Maddala and Kim (1998) and Watson (1994) survey all of these methods. The cointegration methods in Stata are based on Johansen's maximum likelihood framework because it has been found to be particularly useful in several comparative studies, including Gonzalo (1994) and Hubrich, Lütkepohl, and Saikkonen (2001).

Trends in the Johansen VECM framework

Deterministic trends in a cointegrating VECM can stem from two distinct sources; the mean of the cointegrating relationship and the mean of the differenced series. Allowing for a constant and a linear trend and assuming that there are r cointegrating relations, we can rewrite the VECM in (8) as

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}\boldsymbol{\beta}'\mathbf{y}_{t-1} + \sum_{i=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \mathbf{v} + \boldsymbol{\delta} t + \epsilon_t \tag{9}$$

where $\boldsymbol{\delta}$ is a $K \times 1$ vector of parameters. Because (9) models the differences of the data, the constant implies a linear time trend in the levels, and the time trend $\boldsymbol{\delta}t$ implies a quadratic time trend in the levels of the data. Often we may want to include a constant or a linear time trend for the differences without allowing for the higher-order trend that is implied for the levels of the data. VECMs exploit the properties of the matrix $\boldsymbol{\alpha}$ to achieve this flexibility.

Because $\boldsymbol{\alpha}$ is a $K \times r$ rank matrix, we can rewrite the deterministic components in (9) as

$$\mathbf{v} = \boldsymbol{\alpha}\boldsymbol{\mu} + \boldsymbol{\gamma} \tag{10a}$$
$$\boldsymbol{\delta} t = \boldsymbol{\alpha}\boldsymbol{\rho} t + \boldsymbol{\tau} t \tag{10b}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\rho}$ are $r \times 1$ vectors of parameters and $\boldsymbol{\gamma}$ and $\boldsymbol{\tau}$ are $K \times 1$ vectors of parameters. $\boldsymbol{\gamma}$ is orthogonal to $\boldsymbol{\alpha}\boldsymbol{\mu}$, and $\boldsymbol{\tau}$ is orthogonal to $\boldsymbol{\alpha}\boldsymbol{\rho}$; that is, $\boldsymbol{\gamma}'\boldsymbol{\alpha}\boldsymbol{\mu} = \mathbf{0}$ and $\boldsymbol{\tau}'\boldsymbol{\alpha}\boldsymbol{\rho} = \mathbf{0}$, allowing us to rewrite (9) as

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}(\boldsymbol{\beta}'\mathbf{y}_{t-1} + \boldsymbol{\mu} + \boldsymbol{\rho} t) + \sum_{i=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \boldsymbol{\gamma} + \boldsymbol{\tau} t + \epsilon_t \tag{11}$$

Placing restrictions on the trend terms in (11) yields five cases.

CASE 1: Unrestricted trend

If no restrictions are placed on the trend parameters, (11) implies that there are quadratic trends in the levels of the variables and that the cointegrating equations are stationary around time trends (trend stationary).

CASE 2: Restricted trend, $\boldsymbol{\tau} = \mathbf{0}$

By setting $\boldsymbol{\tau} = \mathbf{0}$, we assume that the trends in the levels of the data are linear but not quadratic. This specification allows the cointegrating equations to be trend stationary.

CASE 3: Unrestricted constant, $\boldsymbol{\tau} = \mathbf{0}$ and $\boldsymbol{\rho} = \mathbf{0}$

By setting $\boldsymbol{\tau} = \mathbf{0}$ and $\boldsymbol{\rho} = \mathbf{0}$, we exclude the possibility that the levels of the data have quadratic trends, and we restrict the cointegrating equations to be stationary around constant means. Because $\boldsymbol{\gamma}$ is not restricted to zero, this specification still puts a linear time trend in the levels of the data.

CASE 4: Restricted constant, $\boldsymbol{\tau} = \mathbf{0}$, $\boldsymbol{\rho} = \mathbf{0}$, and $\gamma = \mathbf{0}$

By adding the restriction that $\gamma = \mathbf{0}$, we assume there are no linear time trends in the levels of the data. This specification allows the cointegrating equations to be stationary around a constant mean, but it allows no other trends or constant terms.

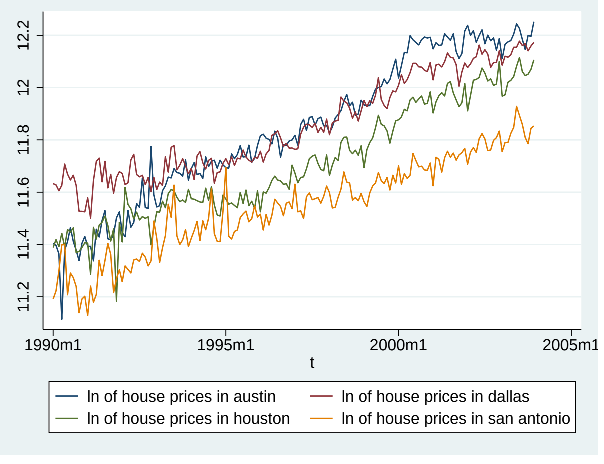
CASE 5: No trend, $\boldsymbol{\tau} = \mathbf{0}$, $\boldsymbol{\rho} = \mathbf{0}$, $\gamma = \mathbf{0}$, and $\boldsymbol{\mu} = \mathbf{0}$

This specification assumes that there are no nonzero means or trends. It also assumes that the cointegrating equations are stationary with means of zero and that the differences and the levels of the data have means of zero.

This flexibility does come at a price. Below we discuss testing procedures for determining the number of cointegrating equations. The asymptotic distribution of the LR for hypotheses about r changes with the trend specification, so we must first specify a trend specification. A combination of theory and graphical analysis will aid in specifying the trend before proceeding with the analysis.

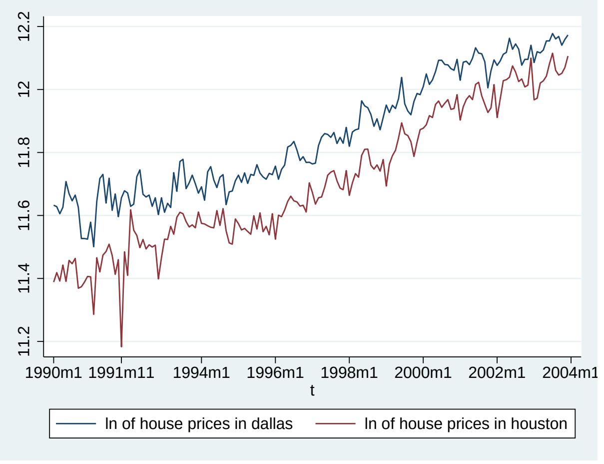
VECM estimation in Stata

We provide an overview of the `vec` commands in Stata through an extended example. We have monthly data on the average selling prices of houses in four cities in Texas: Austin, Dallas, Houston, and San Antonio. In the dataset, these average housing prices are contained in the variables `austin`, `dallas`, `houston`, and `sa`. The series begin in January of 1990 and go through December 2003, for a total of 168 observations. The following graph depicts our data.



The plots on the graph indicate that all the series are trending and potential I(1) processes. In a competitive market, the current and past prices contain all the information available, so tomorrow's price will be a random walk from today's price. Some researchers may opt to use [TS] **dfgls** to investigate the presence of a unit root in each series, but the test for cointegration we use includes the case in which all the variables are stationary, so we defer formal testing until we test for cointegration. The time trends in the data appear to be approximately linear, so we will specify `trend(constant)` when modeling these series, which is the default with `vec`.

The next graph shows just Dallas's and Houston's data, so we can more carefully examine their relationship.



Except for the crash at the end of 1991, housing prices in Dallas and Houston appear closely related. Although average prices in the two cities will differ because of resource variations and other factors, if the housing markets become too dissimilar, people and businesses will migrate, bringing the average housing prices back toward each other. We therefore expect the series of average housing prices in Houston to be cointegrated with the series of average housing prices in Dallas.

Selecting the number of lags

To test for cointegration or fit cointegrating VECMs, we must specify how many lags to include. Building on the work of Tsay (1984) and Paulsen (1984), Nielsen (2001) has shown that the methods implemented in `varsoc` can be used to determine the lag order for a VAR model with I(1) variables. As can be seen from (9), the order of the corresponding VECM is always one less than the VAR. `vec` makes this adjustment automatically, so we will always refer to the order of the underlying VAR. The output below uses `varsoc` to determine the lag order of the VAR of the average housing prices in Dallas and Houston.

```
. use https://www.stata-press.com/data/r17/txhprice
. varsoc dallas houston
Lag-order selection criteria
   Sample: 1990m5 thru 2003m12                          Number of obs = 164
```

Lag	LL	LR	df	p	FPE	AIC	HQIC	SBIC
0	299.525				.000091	-3.62835	-3.61301	-3.59055
1	577.483	555.92	4	0.000	3.2e-06	-6.9693	-6.92326	-6.85589
2	590.978	26.991*	4	0.000	2.9e-06*	-7.0851*	-7.00837*	-6.89608*
3	593.437	4.918	4	0.296	2.9e-06	-7.06631	-6.95888	-6.80168
4	596.364	5.8532	4	0.210	3.0e-06	-7.05322	-6.9151	-6.71299

```
   * optimal lag
   Endogenous: dallas houston
    Exogenous: _cons
```

We will use two lags for this bivariate model because the Hannan–Quinn information criterion (HQIC) method, Schwarz Bayesian information criterion (SBIC) method, and sequential likelihood-ratio (LR) test all chose two lags, as indicated by the “*” in the output.

The reader can verify that when all four cities' data are used, the LR test selects three lags, the HQIC method selects two lags, and the SBIC method selects one lag. We will use three lags in our four-variable model.

Testing for cointegration

The tests for cointegration implemented in `vecrank` are based on Johansen's method. If the log likelihood of the unconstrained model that includes the cointegrating equations is significantly different from the log likelihood of the constrained model that does not include the cointegrating equations, we reject the null hypothesis of no cointegration.

Here we use `vecrank` to determine the number of cointegrating equations:

```
. vecrank dallas houston

Johansen tests for cointegration
Trend: Constant                                 Number of obs  = 166
Sample: 1990m3 thru 2003m12                     Number of lags =   2
─────────────────────────────────────────────────────────────────────
                                                             Critical
Maximum                                                Trace    value
   rank  Params           LL   Eigenvalue   statistic       5%
      0       6    576.26444            .     46.8252    15.41
      1       9    599.58781      0.24498      0.1785*    3.76
      2      10    599.67706      0.00107
─────────────────────────────────────────────────────────────────────
* selected rank
```

Besides presenting information about the sample size and time span, the header indicates that test statistics are based on a model with two lags and a constant trend. The body of the table presents test statistics and their critical values of the null hypotheses of no cointegration (line 1) and one or fewer cointegrating equations (line 2). The eigenvalue shown on the last line is used to compute the trace statistic in the line above it. Johansen's testing procedure starts with the test for zero cointegrating equations (a maximum rank of zero) and then accepts the first null hypothesis that is not rejected.

In the output above, we strongly reject the null hypothesis of no cointegration and fail to reject the null hypothesis of at most one cointegrating equation. Thus we accept the null hypothesis that there is one cointegrating equation in the bivariate model.

Using all four series and a model with three lags, we find that there are two cointegrating relationships.

```
. vecrank austin dallas houston sa, lag(3)

Johansen tests for cointegration
Trend: Constant                                 Number of obs  = 165
Sample: 1990m4 thru 2003m12                     Number of lags =   3
─────────────────────────────────────────────────────────────────────
                                                             Critical
Maximum                                                Trace    value
   rank  Params           LL   Eigenvalue   statistic       5%
      0      36    1107.7833            .    101.6070    47.21
      1      43    1137.7484      0.30456     41.6768    29.68
      2      48    1153.6435      0.17524      9.8865*   15.41
      3      51    1158.4191      0.05624      0.3354     3.76
      4      52    1158.5868      0.00203
─────────────────────────────────────────────────────────────────────
* selected rank
```

Fitting a VECM

vec estimates the parameters of cointegrating VECMs. There are four types of parameters of interest:

1. The parameters in the cointegrating equations β
2. The adjustment coefficients α
3. The short-run coefficients
4. Some standard functions of β and α that have useful interpretations

Although all four types are discussed in [TS] **vec**, here we discuss only types 1–3 and how they appear in the output of vec.

Having determined that there is a cointegrating equation between the Dallas and Houston series, we now want to estimate the parameters of a bivariate cointegrating VECM for these two series by using vec.

```
. vec dallas houston

Vector error-correction model

Sample: 1990m3 thru 2003m12                     Number of obs     =        166
                                                AIC               =  -7.115516
Log likelihood =  599.5878                      HQIC              =   -7.04703
Det(Sigma_ml)  =  2.50e-06                      SBIC              =  -6.946794

Equation           Parms      RMSE     R-sq      chi2     P>chi2
----------------------------------------------------------------
D_dallas              4     .038546   0.1692   32.98959   0.0000
D_houston             4     .045348   0.3737   96.66399   0.0000
----------------------------------------------------------------

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
D_dallas     |
        _ce1 |
         L1. |  -.3038799   .0908504    -3.34   0.001    -.4819434   -.1258165
             |
      dallas |
         LD. |  -.1647304   .0879356    -1.87   0.061     -.337081    .0076202
             |
     houston |
         LD. |  -.0998368   .0650838    -1.53   0.125    -.2273988    .0277251
             |
       _cons |   .0056128   .0030341     1.85   0.064    -.0003339    .0115595
-------------+----------------------------------------------------------------
D_houston    |
        _ce1 |
         L1. |   .5027143   .1068838     4.70   0.000     .2932258    .7122028
             |
      dallas |
         LD. |  -.0619653   .1034547    -0.60   0.549    -.2647327    .1408022
             |
     houston |
         LD. |  -.3328437     .07657    -4.35   0.000    -.4829181   -.1827693
             |
       _cons |   .0033928   .0035695     0.95   0.342    -.0036034     .010389
------------------------------------------------------------------------------

Cointegrating equations

Equation           Parms    chi2     P>chi2
-------------------------------------------
_ce1                  1   1640.088   0.0000
-------------------------------------------
```

```
Identification:  beta is exactly identified
                 Johansen normalization restriction imposed
```

beta	Coefficient	Std. err.	z	P>|z|	[95% conf. interval]	
_ce1						
dallas	1	.	.	.	.	.
houston	-.8675936	.0214231	-40.50	0.000	-.9095821	-.825605
_cons	-1.688897	.	.	.	.	.

The header contains information about the sample, the fit of each equation, and overall model fit statistics. The first estimation table contains the estimates of the short-run parameters, along with their standard errors, z statistics, and confidence intervals. The two coefficients on `L._ce1` are the parameters in the adjustment matrix $\boldsymbol{\alpha}$ for this model. The second estimation table contains the estimated parameters of the cointegrating vector for this model, along with their standard errors, z statistics, and confidence intervals.

Using our previous notation, we have estimated

$$\widehat{\boldsymbol{\alpha}} = (-0.304, 0.503) \qquad \widehat{\boldsymbol{\beta}} = (1, -0.868) \qquad \widehat{\mathbf{v}} = (0.0056, 0.0034)$$

and

$$\widehat{\boldsymbol{\Gamma}} = \begin{pmatrix} -0.165 & -0.0998 \\ -0.062 & -0.333 \end{pmatrix}$$

Overall, the output indicates that the model fits well. The coefficient on `houston` in the cointegrating equation is statistically significant, as are the adjustment parameters. The adjustment parameters in this bivariate example are easy to interpret, and we can see that the estimates have the correct signs and imply rapid adjustment toward equilibrium. When the predictions from the cointegrating equation are positive, `dallas` is above its equilibrium value because the coefficient on `dallas` in the cointegrating equation is positive. The estimate of the coefficient `[D_dallas]L._ce1` is -0.3. Thus when the average housing price in Dallas is too high, it quickly falls back toward the Houston level. The estimated coefficient `[D_houston]L._ce1` of 0.5 implies that when the average housing price in Dallas is too high, the average price in Houston quickly adjusts toward the Dallas level at the same time that the Dallas prices are adjusting.

Fitting VECMs with Johansen's normalization

As discussed by Johansen (1995), if there are r cointegrating equations, then at least r^2 restrictions are required to identify the free parameters in $\boldsymbol{\beta}$. Johansen proposed a default identification scheme that has become the conventional method of identifying models in the absence of theoretically justified restrictions. Johansen's identification scheme is

$$\boldsymbol{\beta}' = (\mathbf{I}_r, \widetilde{\boldsymbol{\beta}}')$$

where $\mathbf{I}_r$ is the $r \times r$ identity matrix and $\widetilde{\boldsymbol{\beta}}$ is an $(K - r) \times r$ matrix of identified parameters. `vec` applies Johansen's normalization by default.

To illustrate, we fit a VECM with two cointegrating equations and three lags on all four series. We are interested only in the estimates of the parameters in the cointegrating equations, so we can specify the `noetable` option to suppress the estimation table for the adjustment and short-run parameters.

```
. vec austin dallas houston sa, lags(3) rank(2) noetable
Vector error-correction model
Sample: 1990m4 thru 2003m12                     Number of obs     =        165
                                                AIC               =  -13.40174
Log likelihood =  1153.644                      HQIC              =  -13.03496
Det(Sigma_ml)  =  9.93e-12                      SBIC              =  -12.49819
Cointegrating equations
Equation           Parms    chi2     P>chi2
-------------------------------------------
_ce1                   2   586.3044   0.0000
_ce2                   2   2169.826   0.0000
-------------------------------------------
Identification:  beta is exactly identified
                 Johansen normalization restrictions imposed
------------------------------------------------------------------------------
        beta | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
_ce1         |
      austin |          1          .        .       .            .           .
      dallas |          0  (omitted)
     houston |  -.2623782   .1893625    -1.39   0.166    -.6335219    .1087655
          sa |  -1.241805    .229643    -5.41   0.000    -1.691897   -.7917128
       _cons |   5.577099          .        .       .            .           .
-------------+----------------------------------------------------------------
_ce2         |
      austin |          0  (omitted)
      dallas |          1          .        .       .            .           .
     houston |  -1.095652   .0669898   -16.36   0.000     -1.22695   -.9643545
          sa |   .2883986   .0812396     3.55   0.000     .1291718    .4476253
       _cons |  -2.351372          .        .       .            .           .
------------------------------------------------------------------------------
```

The Johansen identification scheme has placed four constraints on the parameters in β: `[_ce1]austin` $= 1$, `[_ce1]dallas` $= 0$, `[_ce2]austin` $= 0$, and `[_ce2]dallas` $= 1$. We interpret the results of the first equation as indicating the existence of an equilibrium relationship between the average housing price in Austin and the average prices of houses in Houston and San Antonio.

The Johansen normalization restricted the coefficient on `dallas` to be unity in the second cointegrating equation, but we could instead constrain the coefficient on `houston`. Both sets of restrictions define just-identified models, so fitting the model with the latter set of restrictions will yield the same maximized log likelihood. To impose the alternative set of constraints, we use the `constraint` command.

```
. constraint define 1 [_ce1]austin = 1
. constraint define 2 [_ce1]dallas = 0
. constraint define 3 [_ce2]austin = 0
. constraint define 4 [_ce2]houston = 1
```

```
. vec austin dallas houston sa, lags(3) rank(2) noetable bconstraints(1/4)
Iteration 1:     log likelihood = 1148.8745
  (output omitted )
Iteration 25:    log likelihood = 1153.6435
Vector error-correction model
Sample: 1990m4 thru 2003m12                       Number of obs     =        165
                                                  AIC               =  -13.40174
Log likelihood =  1153.644                        HQIC              =  -13.03496
Det(Sigma_ml)  =  9.93e-12                        SBIC              =  -12.49819
Cointegrating equations
Equation           Parms    chi2     P>chi2
-------------------------------------------
_ce1                  2   586.3392   0.0000
_ce2                  2   3455.469   0.0000
-------------------------------------------

Identification:  beta is exactly identified
  ( 1)  [_ce1]austin = 1
  ( 2)  [_ce1]dallas = 0
  ( 3)  [_ce2]austin = 0
  ( 4)  [_ce2]houston = 1
------------------------------------------------------------------------------
        beta | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
_ce1         |
      austin |          1          .        .       .            .           .
      dallas |          0  (omitted)
     houston |  -.2623784   .1876727    -1.40   0.162    -.6302102    .1054534
          sa |  -1.241805   .2277537    -5.45   0.000    -1.688194   -.7954157
       _cons |   5.577099          .        .       .            .           .
-------------+----------------------------------------------------------------
_ce2         |
      austin |          0  (omitted)
      dallas |  -.9126985   .0595804   -15.32   0.000    -1.029474   -.7959231
     houston |          1          .        .       .            .           .
          sa |  -.2632209   .0628791    -4.19   0.000    -.3864617   -.1399802
       _cons |   2.146094          .        .       .            .           .
------------------------------------------------------------------------------
```

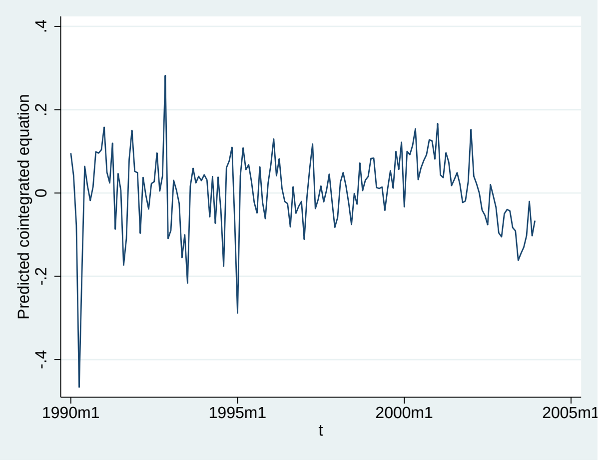
Only the estimates of the parameters in the second cointegrating equation have changed, and the new estimates are simply the old estimates divided by -1.095652 because the new constraints are just an alternative normalization of the same just-identified model. With the new normalization, we can interpret the estimates of the parameters in the second cointegrating equation as indicating an equilibrium relationship between the average house price in Houston and the average prices of houses in Dallas and San Antonio.

Postestimation specification testing

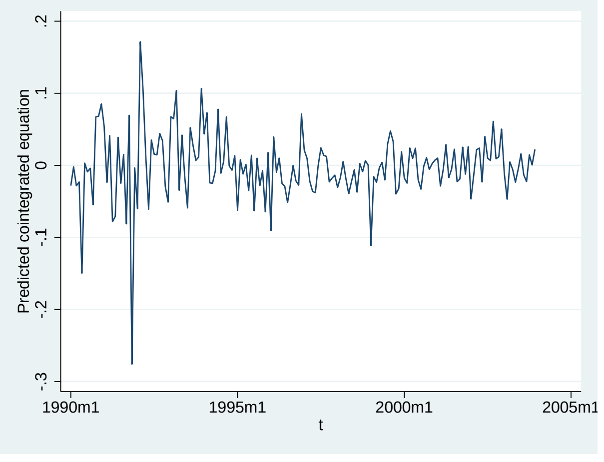
Inference on the parameters in α depends crucially on the stationarity of the cointegrating equations, so we should check the specification of the model. As a first check, we can predict the cointegrating equations and graph them over time.

```
. predict ce1, ce equ(#1)
. predict ce2, ce equ(#2)
```

```
. twoway line ce1 t
```




```
. twoway line ce2 t
```



Although the large shocks apparent in the graph of the levels have clear effects on the predictions from the cointegrating equations, our only concern is the negative trend in the first cointegrating equation since the end of 2000. The graph of the levels shows that something put a significant brake on the growth of housing prices after 2000 and that the growth of housing prices in San Antonio slowed during 2000 but then recuperated while Austin maintained slower growth. We suspect that this indicates that the end of the high-tech boom affected Austin more severely than San Antonio. This difference is what causes the trend in the first cointegrating equation. Although we could try to account for this effect with a more formal analysis, we will proceed as if the cointegrating equations are stationary.

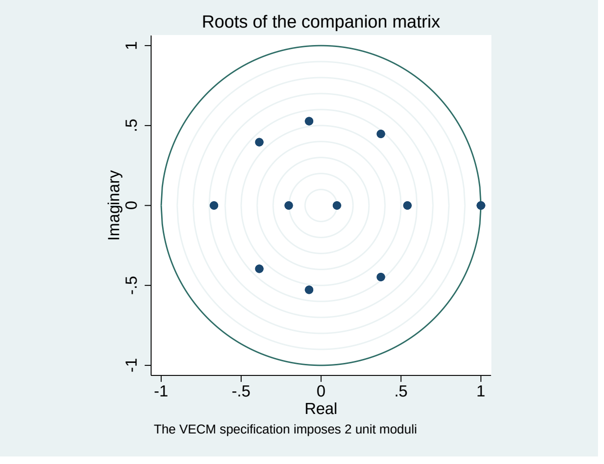
We can use `vecstable` to check whether we have correctly specified the number of cointegrating equations. As discussed in [TS] **vecstable**, the companion matrix of a VECM with K endogenous variables and r cointegrating equations has $K-r$ unit eigenvalues. If the process is stable, the moduli of the remaining r eigenvalues are strictly less than one. Because there is no general distribution theory for the moduli of the eigenvalues, ascertaining whether the moduli are too close to one can be difficult.

```
. vecstable, graph
```

Eigenvalue stability condition

Eigenvalue	Modulus
1	1
1	1
-.6698661	.669866
.3740191 + .4475996*i*	.583297
.3740191 - .4475996*i*	.583297
-.386377 + .395972*i*	.553246
-.386377 - .395972*i*	.553246
.540117	.540117
-.0749239 + .5274203*i*	.532715
-.0749239 - .5274203*i*	.532715
-.2023955	.202395
.09923966	.09924

The VECM specification imposes 2 unit moduli.



Because we specified the **graph** option, **vecstable** plotted the eigenvalues of the companion matrix. The graph of the eigenvalues shows that none of the remaining eigenvalues appears close to the unit circle. The stability check does not indicate that our model is misspecified.

Here we use **veclmar** to test for serial correlation in the residuals.

```
. veclmar, mlag(4)
```

Lagrange-multiplier test

lag	chi2	df	Prob > chi2
1	56.8757	16	0.00000
2	31.1970	16	0.01270
3	30.6818	16	0.01477
4	14.6493	16	0.55046

H0: no autocorrelation at lag order

The results clearly indicate serial correlation in the residuals. The results in Gonzalo (1994) indicate that underspecifying the number of lags in a VECM can significantly increase the finite-sample bias in the parameter estimates and lead to serial correlation. For this reason, we refit the model with five lags instead of three.

```
. vec austin dallas houston sa, lags(5) rank(2) noetable bconstraints(1/4)
Iteration 1:    log likelihood = 1200.5402
  (output omitted )
Iteration 20:   log likelihood = 1203.9465
Vector error-correction model
Sample: 1990m6 thru 2003m12                     Number of obs     =        163
                                                AIC               =  -13.79075
Log likelihood =  1203.946                      HQIC              =   -13.1743
Det(Sigma_ml)  =  4.51e-12                      SBIC              =  -12.27235
Cointegrating equations
Equation           Parms    chi2     P>chi2
-------------------------------------------
_ce1                   2   498.4682   0.0000
_ce2                   2   4125.926   0.0000
-------------------------------------------
Identification:  beta is exactly identified
 ( 1)  [_ce1]austin = 1
 ( 2)  [_ce1]dallas = 0
 ( 3)  [_ce2]austin = 0
 ( 4)  [_ce2]houston = 1
```

beta	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
_ce1						
austin	1	.	.	.	.	.
dallas	0	(omitted)				
houston	-.6525574	.2047061	-3.19	0.001	-1.053774	-.2513407
sa	-.6960166	.2494167	-2.79	0.005	-1.184864	-.2071688
_cons	3.846275	.	.	.	.	.
_ce2						
austin	0	(omitted)				
dallas	-.932048	.0564332	-16.52	0.000	-1.042655	-.8214409
houston	1	.	.	.	.	.
sa	-.2363915	.0599348	-3.94	0.000	-.3538615	-.1189215
_cons	2.065719	.	.	.	.	.

Comparing these results with those from the previous model reveals that

1. there is now evidence that the coefficient `[_ce1]houston` is not equal to zero,
2. the two sets of estimated coefficients for the first cointegrating equation are different, and
3. the two sets of estimated coefficients for the second cointegrating equation are similar.

The assumption that the errors are independent and are identically and normally distributed with zero mean and finite variance allows us to derive the likelihood function. If the errors do not come from a normal distribution but are just independent and identically distributed with zero mean and finite variance, the parameter estimates are still consistent, but they are not efficient.

We use `vecnorm` to test the null hypothesis that the errors are normally distributed.

```
. quietly vec austin dallas houston sa, lags(5) rank(2) bconstraints(1/4)
. vecnorm
```

Jarque-Bera test

Equation	chi2	df	Prob > chi2
D_austin	74.324	2	0.00000
D_dallas	3.501	2	0.17370
D_houston	245.032	2	0.00000
D_sa	8.426	2	0.01481
ALL	331.283	8	0.00000

Skewness test

Equation	Skewness	chi2	df	Prob > chi2
D_austin	.60265	9.867	1	0.00168
D_dallas	.09996	0.271	1	0.60236
D_houston	-1.0444	29.635	1	0.00000
D_sa	.38019	3.927	1	0.04752
ALL		43.699	4	0.00000

Kurtosis test

Equation	Kurtosis	chi2	df	Prob > chi2
D_austin	6.0807	64.458	1	0.00000
D_dallas	3.6896	3.229	1	0.07232
D_houston	8.6316	215.397	1	0.00000
D_sa	3.8139	4.499	1	0.03392
ALL		287.583	4	0.00000

The results indicate that we can strongly reject the null hypothesis of normally distributed errors. Most of the errors are both skewed and kurtotic.

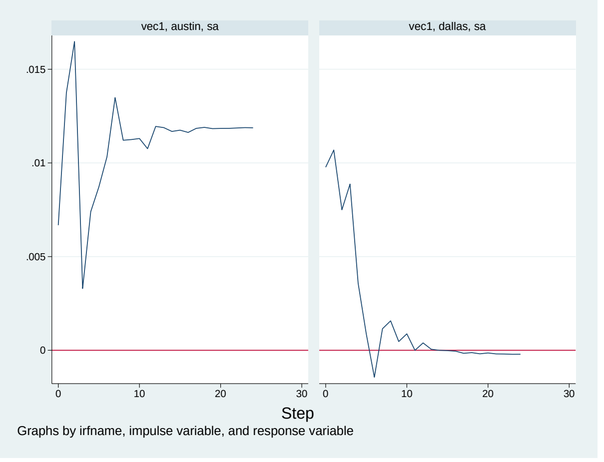
Impulse–response functions for VECMs

With a model that we now consider acceptably well specified, we can use the `irf` commands to estimate and interpret the IRFs. Whereas IRFs from a stationary VAR die out over time, IRFs from a cointegrating VECM do not always die out. Because each variable in a stationary VAR has a time-invariant mean and finite, time-invariant variance, the effect of a shock to any one of these variables must die out so that the variable can revert to its mean. In contrast, the I(1) variables modeled in a cointegrating VECM are not mean reverting, and the unit moduli in the companion matrix imply that the effects of some shocks will not die out over time.

These two possibilities gave rise to new terms. When the effect of a shock dies out over time, the shock is said to be transitory. When the effect of a shock does not die out over time, the shock is said to be permanent.

Below we use `irf create` to estimate the IRFs and `irf graph` to graph two of the orthogonalized IRFs.

```
. irf create vec1, set(vecintro, replace) step(24)
(file vecintro.irf created)
(file vecintro.irf now active)
(file vecintro.irf updated)
. irf graph oirf, impulse(austin dallas) response(sa) yline(0)
```



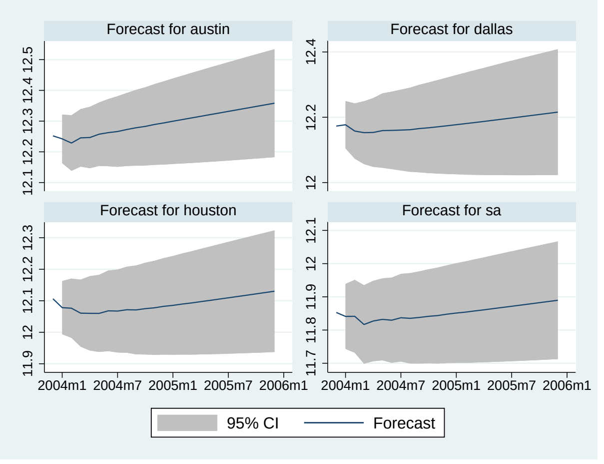
The graphs indicate that an orthogonalized shock to the average housing price in Austin has a permanent effect on the average housing price in San Antonio but that an orthogonalized shock to the average price of housing in Dallas has a transitory effect. According to this model, unexpected shocks that are local to the Austin housing market will have a permanent effect on the housing market in San Antonio, but unexpected shocks that are local to the Dallas housing market will have only a transitory effect on the housing market in San Antonio.

Forecasting with VECMs

Cointegrating VECMs are also used to produce forecasts of both the first-differenced variables and the levels of the variables. Comparing the variances of the forecast errors of stationary VARs with those from a cointegrating VECM reveals a fundamental difference between the two models. Whereas the variances of the forecast errors for a stationary VAR converge to a constant as the prediction horizon grows, the variances of the forecast errors for the levels of a cointegrating VECM diverge with the forecast horizon. (See sec. 6.5 of Lütkepohl [2005] for more about this result.) Because all the variables in the model for the first differences are stationary, the forecast errors for the dynamic forecasts of the first differences remain finite. In contrast, the forecast errors for the dynamic forecasts of the levels diverge to infinity.

We use `fcast compute` to obtain dynamic forecasts of the levels and `fcast graph` to graph these dynamic forecasts, along with their asymptotic confidence intervals.

```
. tsset
Time variable: t, 1990m1 to 2003m12
        Delta: 1 month
. fcast compute m1_, step(24)
. fcast graph m1_austin m1_dallas m1_houston m1_sa
```



As expected, the widths of the confidence intervals grow with the forecast horizon.

References

Ahn, S. K., and G. C. Reinsel. 1990. Estimation for partially nonstationary multivariate autoregressive models. *Journal of the American Statistical Association* 85: 813–823. https://doi.org/10.2307/2290020.

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

Du, K. 2017. Econometric convergence test and club clustering using Stata. *Stata Journal* 17: 882–900.

Engle, R. F., and C. W. J. Granger. 1987. Co-integration and error correction: Representation, estimation, and testing. *Econometrica* 55: 251–276. https://doi.org/10.2307/1913236.

Gonzalo, J. 1994. Five alternative methods of estimating long-run equilibrium relationships. *Journal of Econometrics* 60: 203–233. https://doi.org/10.1016/0304-4076(94)90044-2.

Granger, C. W. J. 1981. Some properties of time series data and their use in econometric model specification. *Journal of Econometrics* 16: 121–130. https://doi.org/10.1016/0304-4076(81)90079-8.

Granger, C. W. J., and P. Newbold. 1974. Spurious regressions in econometrics. *Journal of Econometrics* 2: 111–120. https://doi.org/10.1016/0304-4076(74)90034-7.

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

Hubrich, K., H. Lütkepohl, and P. Saikkonen. 2001. A review of systems cointegration tests. *Econometric Reviews* 20: 247–318. https://doi.org/10.1081/ETC-100104936.

Johansen, S. 1988. Statistical analysis of cointegration vectors. *Journal of Economic Dynamics and Control* 12: 231–254. https://doi.org/10.1016/0165-1889(88)90041-3.

——. 1991. Estimation and hypothesis testing of cointegration vectors in Gaussian vector autoregressive models. *Econometrica* 59: 1551–1580. https://doi.org/10.2307/2938278.

——. 1995. *Likelihood-Based Inference in Cointegrated Vector Autoregressive Models*. Oxford: Oxford University Press.

Lütkepohl, H. 2005. *New Introduction to Multiple Time Series Analysis*. New York: Springer.

Maddala, G. S., and I.-M. Kim. 1998. *Unit Roots, Cointegration, and Structural Change*. Cambridge: Cambridge University Press.

Nielsen, B. 2001. Order determination in general vector autoregressions. Working paper, Department of Economics, University of Oxford and Nuffield College. https://ideas.repec.org/p/nuf/econwp/0110.html.

Park, J. Y., and P. C. B. Phillips. 1988. Statistical inference in regressions with integrated processes: Part I. *Econometric Theory* 4: 468–497. https://doi.org/10.1017/S0266466600013402.

——. 1989. Statistical inference in regressions with integrated processes: Part II. *Econometric Theory* 5: 95–131. https://doi.org/10.1017/S0266466600012287.

Paulsen, J. 1984. Order determination of multivariate autoregressive time series with unit roots. *Journal of Time Series Analysis* 5: 115–127. https://doi.org/10.1111/j.1467-9892.1984.tb00381.x.

Phillips, P. C. B. 1986. Understanding spurious regressions in econometrics. *Journal of Econometrics* 33: 311–340. https://doi.org/10.1016/0304-4076(86)90001-1.

Phillips, P. C. B., and S. N. Durlauf. 1986. Multiple time series regressions with integrated processes. *Review of Economic Studies* 53: 473–495. https://doi.org/10.2307/2297602.

Sims, C. A., J. H. Stock, and M. W. Watson. 1990. Inference in linear time series models with some unit roots. *Econometrica* 58: 113–144. https://doi.org/10.2307/2938337.

Stock, J. H. 1987. Asymptotic properties of least squares estimators of cointegrating vectors. *Econometrica* 55: 1035–1056. https://doi.org/10.2307/1911260.

Stock, J. H., and M. W. Watson. 1988. Testing for common trends. *Journal of the American Statistical Association* 83: 1097–1107. https://doi.org/10.1080/01621459.1988.10478707.

Tsay, R. S. 1984. Order selection in nonstationary autoregressive models. *Annals of Statistics* 12: 1425–1433. https://doi.org/10.1214/aos/1176346801.

Watson, M. W. 1994. Vector autoregressions and cointegration. In Vol. 4 of *Handbook of Econometrics*, ed. R. F. Engle and D. L. McFadden. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4412(05)80016-9.

Also see

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

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

Title

vec — Vector error-correction models

Description	Quick start	Menu	Syntax
Options	Remarks and examples	Stored results	Methods and formulas
References	Also see		

Description

`vec` fits a type of vector autoregression in which some of the variables are cointegrated by using Johansen's (1995) maximum likelihood method. Constraints may be placed on the parameters in the cointegrating equations or on the adjustment terms. See [TS] **vec intro** for a list of commands that are used in conjunction with `vec`.

Quick start

Vector error-correction model for `y1`, `y2`, and `y3` using `tsset` data

```
vec y1 y2 y3
```

Use 4 lags for the underlying VAR model

```
vec y1 y2 y3, lags(4)
```

Use 2 cointegrating equations

```
vec y1 y2 y3, lags(4) rank(2)
```

Add a linear trend in the cointegrating equations and a quadratic trend in the undifferenced data

```
vec y1 y2 y3, lags(4) rank(2) trend(trend)
```

As above, but without a trend or a constant

```
vec y1 y2 y3, lags(4) rank(2) trend(none)
```

Menu

Statistics > Multivariate time series > Vector error-correction model (VECM)

Syntax

vec *depvarlist* [*if*] [*in*] [, *options*]

options	Description
Model	
rank(#)	use # cointegrating equations; default is rank(1)
lags(#)	use # for the maximum lag in underlying VAR model
trend(constant)	include an unrestricted constant in model; the default
trend(rconstant)	include a restricted constant in model
trend(trend)	include a linear trend in the cointegrating equations and a quadratic trend in the undifferenced data
trend(rtrend)	include a restricted trend in model
trend(none)	do not include a trend or a constant
bconstraints($constraints_{bc}$)	place $constraints_{bc}$ on cointegrating vectors
aconstraints($constraints_{ac}$)	place $constraints_{ac}$ on adjustment parameters
Adv. model	
sindicators($varlist_{si}$)	include normalized seasonal indicator variables $varlist_{si}$
noreduce	do not perform checks and corrections for collinearity among lags of dependent variables
Reporting	
level(#)	set confidence level; default is level(95)
nobtable	do not report parameters in the cointegrating equations
noidtest	do not report the likelihood-ratio test of overidentifying restrictions
alpha	report adjustment parameters in separate table
pi	report parameters in $\mathbf{\Pi} = \boldsymbol{\alpha}\boldsymbol{\beta}'$
noptable	do not report elements of $\mathbf{\Pi}$ matrix
mai	report parameters in the moving-average impact matrix
noetable	do not report adjustment and short-run parameters
dforce	force reporting of short-run, beta, and alpha parameters when the parameters in beta are not identified; advanced option
nocnsreport	do not display constraints
display_options	control columns and column formats, row spacing, and line width
Maximization	
maximize_options	control the maximization process; seldom used
coeflegend	display legend instead of statistics

vec does not allow gaps in the data.

You must tsset your data before using vec; see [TS] **tsset**.

varlist must contain at least two variables and may contain time-series operators; see [U] **11.4.4 Time-series varlists**.

by, collect, fp, rolling, statsby, and xi are allowed; see [U] **11.1.10 Prefix commands**.

coeflegend does not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

Model

rank(#) specifies the number of cointegrating equations; rank(1) is the default.

lags(#) specifies the maximum lag to be included in the underlying VAR model. The maximum lag in a VECM is one smaller than the maximum lag in the corresponding VAR in levels; the number of lags must be greater than zero but small enough so that the degrees of freedom used up by the model are fewer than the number of observations. The default is lags(2).

trend(*trend_spec*) specifies which of Johansen's five trend specifications to include in the model. These specifications are discussed in *Specification of constants and trends* below. The default is trend(constant).

bconstraints(*constraints*$_{\rm bc}$) specifies the constraints to be placed on the parameters of the cointegrating equations. When no constraints are placed on the adjustment parameters—that is, when the aconstraints() option is not specified—the default is to place the constraints defined by Johansen's normalization on the parameters of the cointegrating equations. When constraints are placed on the adjustment parameters, the default is not to place constraints on the parameters in the cointegrating equations.

aconstraints(*constraints*$_{\rm ac}$) specifies the constraints to be placed on the adjustment parameters. By default, no constraints are placed on the adjustment parameters.

Adv. model

sindicators(*varlist*$_{\rm si}$) specifies the normalized seasonal indicator variables to include in the model. The indicator variables specified in this option must be normalized as discussed in Johansen (1995). If the indicators are not properly normalized, the estimator of the cointegrating vector does not converge to the asymptotic distribution derived by Johansen (1995). More details about how these variables are handled are provided in *Methods and formulas*. sindicators() cannot be specified with trend(none) or with trend(rconstant).

noreduce causes vec to skip the checks and corrections for collinearity among the lags of the dependent variables. By default, vec checks to see whether the current lag specification causes some of the regressions performed by vec to contain perfectly collinear variables; if so, it reduces the maximum lag until the perfect collinearity is removed.

Reporting

level(#); see [R] **Estimation options**.

nobtable suppresses the estimation table for the parameters in the cointegrating equations. By default, vec displays the estimation table for the parameters in the cointegrating equations.

noidtest suppresses the likelihood-ratio test of the overidentifying restrictions, which is reported by default when the model is overidentified.

alpha displays a separate estimation table for the adjustment parameters, which is not displayed by default.

pi displays a separate estimation table for the parameters in $\mathbf{\Pi} = \alpha\beta'$, which is not displayed by default.

noptable suppresses the estimation table for the elements of the $\mathbf{\Pi}$ matrix, which is displayed by default when the parameters in the cointegrating equations are not identified.

mai displays a separate estimation table for the parameters in the moving-average impact matrix, which is not displayed by default.

`noetable` suppresses the main estimation table that contains information about the estimated adjustment parameters and the short-run parameters, which is displayed by default.

`dforce` displays the estimation tables for the short-run parameters and $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$—if the last two are requested—when the parameters in $\boldsymbol{\beta}$ are not identified. By default, when the specified constraints do not identify the parameters in the cointegrating equations, estimation tables are displayed only for $\mathbf{\Pi}$ and the MAI.

`nocnsreport`; see [R] **Estimation options**.

display_options: `noci`, `nopvalues`, `vsquish`, `cformat(`*%fmt*`)`, `pformat(`*%fmt*`)`, `sformat(`*%fmt*`)`, and `nolstretch`; see [R] **Estimation options**.

Maximization

maximize_options: `iterate(`*#*`)`, `[no]log`, `trace`, `toltrace`, `tolerance(`*#*`)`, `ltolerance(`*#*`)`, `afrom(`$matrix_a$`)`, and `bfrom(`$matrix_b$`)`; see [R] **Maximize**.

`toltrace` displays the relative differences for the log likelihood and the coefficient vector at every iteration. This option cannot be specified if no constraints are defined or if `nolog` is specified.

`afrom(`$matrix_a$`)` specifies a $1 \times (K * r)$ row vector with starting values for the adjustment parameters, where K is the number of endogenous variables and r is the number of cointegrating equations specified in the `rank()` option. The starting values should be ordered as they are reported in `e(alpha)`. This option cannot be specified if no constraints are defined.

`bfrom(`$matrix_b$`)` specifies a $1 \times (m_1 * r)$ row vector with starting values for the parameters of the cointegrating equations, where m_1 is the number of variables in the trend-augmented system and r is the number of cointegrating equations specified in the `rank()` option. (See *Methods and formulas* for more details about m_1.) The starting values should be ordered as they are reported in `e(betavec)`. As discussed in *Methods and formulas*, for some trend specifications, `e(beta)` contains parameter estimates that are not obtained directly from the optimization algorithm. `bfrom()` should specify only starting values for the parameters reported in `e(betavec)`. This option cannot be specified if no constraints are defined.

The following option is available with `vec` but is not shown in the dialog box:

`coeflegend`; see [R] **Estimation options**.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Specification of constants and trends
Collinearity

Introduction

VECMs are used to model the stationary relationships between multiple time series that contain unit roots. `vec` implements Johansen's approach for estimating the parameters of a VECM.

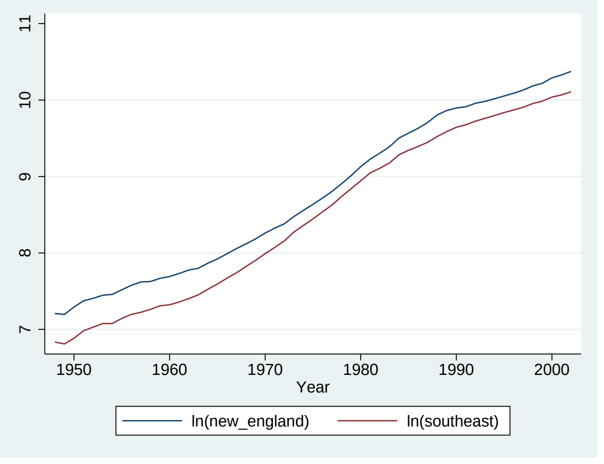
[TS] **vec intro** reviews the basics of integration and cointegration and highlights why we need special methods for modeling the relationships between processes that contain unit roots. This manual entry assumes familiarity with the material in [TS] **vec intro** and provides examples illustrating how to use the `vec` command. See Johansen (1995), Hamilton (1994), and Becketti (2020) for more in-depth introductions to cointegration analysis.

▷ Example 1

This example uses annual data on the average per-capita disposable personal income in the eight U.S. Bureau of Economic Analysis (BEA) regions of the United States. We use data from 1948–2002 in logarithms. Unit-root tests on these series fail to reject the null hypothesis that per-capita disposable income in each region contains a unit root. Because capital and labor can move easily between the different regions of the United States, we would expect that no one series will diverge from all the remaining series and that cointegrating relationships exist.

Below we graph the natural logs of average disposal income in the New England and the Southeast regions.

```
. use https://www.stata-press.com/data/r17/rdinc
. line ln_ne ln_se year
```



The graph indicates a differential between the two series that shrinks between 1960 and about 1980 and then grows until it stabilizes around 1990. We next estimate the parameters of a bivariate VECM with one cointegrating relationship.

```
. vec ln_ne ln_se
Vector error-correction model
Sample: 1950 thru 2002                          Number of obs     =         53
                                                AIC               =  -11.00462
Log likelihood =  300.6224                      HQIC              =  -10.87595
Det(Sigma_ml)  =  4.06e-08                      SBIC              =  -10.67004

Equation           Parms      RMSE     R-sq      chi2     P>chi2
----------------------------------------------------------------
D_ln_ne               4     .017896   0.9313   664.4668   0.0000
D_ln_se               4     .018723   0.9292   642.7179   0.0000
----------------------------------------------------------------

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
D_ln_ne      |
        _ce1 |
         L1. |  -.4337524   .0721365    -6.01   0.000    -.5751373   -.2923675
             |
       ln_ne |
         LD. |   .7168658   .1889085     3.79   0.000     .3466119     1.08712
             |
       ln_se |
         LD. |  -.6748754   .2117975    -3.19   0.001    -1.089991   -.2597599
             |
       _cons |  -.0019846   .0080291    -0.25   0.805    -.0177214    .0137521
-------------+----------------------------------------------------------------
D_ln_se      |
        _ce1 |
         L1. |  -.3543935   .0754725    -4.70   0.000    -.5023168   -.2064701
             |
       ln_ne |
         LD. |   .3366786   .1976448     1.70   0.088     -.050698    .7240553
             |
       ln_se |
         LD. |  -.1605811   .2215922    -0.72   0.469    -.5948939    .2737317
             |
       _cons |    .002429   .0084004     0.29   0.772    -.0140355    .0188936
------------------------------------------------------------------------------

Cointegrating equations
Equation           Parms    chi2     P>chi2
-------------------------------------------
_ce1                  1   29805.02   0.0000
-------------------------------------------

Identification:  beta is exactly identified
                 Johansen normalization restriction imposed
------------------------------------------------------------------------------
        beta | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
_ce1         |
       ln_ne |          1          .        .       .            .           .
       ln_se |  -.9433708   .0054643  -172.64   0.000    -.9540807   -.9326609
       _cons |  -.8964065          .        .       .            .           .
------------------------------------------------------------------------------
```

The default output has three parts. The header provides information about the sample, the model fit, and the identification of the parameters in the cointegrating equation. The main estimation table contains the estimates of the short-run parameters, along with their standard errors and confidence intervals. The second estimation table reports the estimates of the parameters in the cointegrating equation, along with their standard errors and confidence intervals.

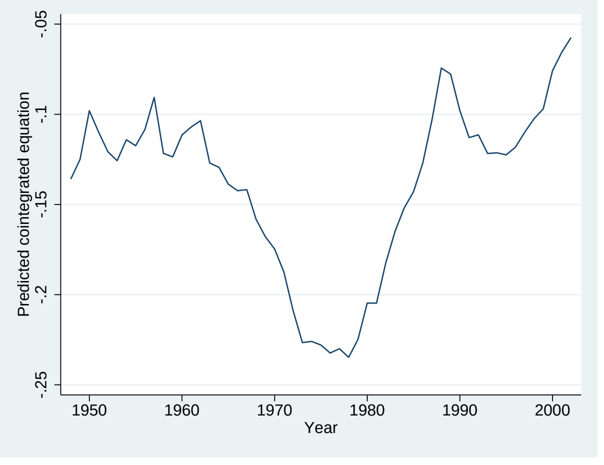
The results indicate strong support for a cointegrating equation such that

$$\texttt{ln\_ne} - 0.943\,\texttt{ln\_se} - 0.896$$

should be a stationary series. Identification of the parameters in the cointegrating equation is achieved by constraining some of them to be fixed, and fixed parameters do not have standard errors. In this example, the coefficient on `ln_ne` has been normalized to 1, so its standard error is missing. As discussed in *Methods and formulas*, the constant term in the cointegrating equation is not directly estimated in this trend specification but rather is backed out from other estimates. Not all the elements of the VCE that correspond to this parameter are readily available, so the standard error for the `_cons` parameter is missing.

To get a better idea of how our model fits, we predict the cointegrating equation and graph it over time:

```
. predict ce, ce
. line ce year
```



Although the predicted cointegrating equation has the right appearance for the time before the mid-1960s, afterward the predicted cointegrating equation does not look like a stationary series. A better model would account for the trends in the size of the differential.

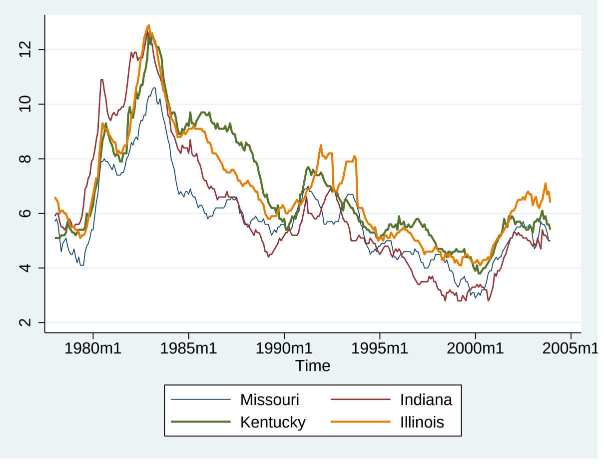
◁

As discussed in [TS] **vec intro**, simply normalizing one of the coefficients to be one is sufficient to identify the parameters of the single cointegrating vector. When there is more than one cointegrating equation, more restrictions are required.

▷ Example 2

We have data on monthly unemployment rates in Indiana, Illinois, Kentucky, and Missouri from January 1978 through December 2003. We suspect that factor mobility will keep the unemployment rates in equilibrium. The following graph plots the data.

```
. use https://www.stata-press.com/data/r17/urates, clear
. line missouri indiana kentucky illinois t
```



The graph shows that although the series do appear to move together, the relationship is not as clear as in the previous example. There are periods when Indiana has the highest rate and others when Indiana has the lowest rate. Although the Kentucky rate moves closely with the other series for most of the sample, there is a period in the mid-1980s when the unemployment rate in Kentucky does not fall at the same rate as the other series.

We will model the series with two cointegrating equations and no linear or quadratic time trends in the original series. Because we are focusing on the cointegrating vectors, we use the `noetable` option to suppress displaying the short-run estimation table.

```
. vec missouri indiana kentucky illinois, trend(rconstant) rank(2) lags(4)
> noetable
Vector error-correction model
Sample: 1978m5 thru 2003m12                     Number of obs     =        308
                                                AIC               =  -2.306048
Log likelihood =  417.1314                      HQIC              =  -2.005818
Det(Sigma_ml)  =  7.83e-07                      SBIC              =  -1.555184
Cointegrating equations
Equation           Parms    chi2     P>chi2
-------------------------------------------
_ce1                  2   133.3885   0.0000
_ce2                  2   195.6324   0.0000
-------------------------------------------

Identification:  beta is exactly identified
                 Johansen normalization restrictions imposed
------------------------------------------------------------------------------
        beta | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
_ce1         |
    missouri |          1          .        .       .            .           .
     indiana |          0  (omitted)
    kentucky |   .3493902   .2005537     1.74   0.081    -.0436879    .7424683
    illinois |  -1.135152   .2069063    -5.49   0.000    -1.540681   -.7296235
       _cons |  -.3880707   .4974323    -0.78   0.435     -1.36302    .5868787
-------------+----------------------------------------------------------------
_ce2         |
    missouri |  -1.11e-16          .        .       .            .           .
     indiana |          1          .        .       .            .           .
    kentucky |   .2059473   .2718678     0.76   0.449    -.3269038    .7387985
    illinois |   -1.51962   .2804792    -5.42   0.000    -2.069349   -.9698907
       _cons |    2.92857   .6743122     4.34   0.000     1.606942    4.250197
------------------------------------------------------------------------------
```

Except for the coefficients on kentucky in the two cointegrating equations and the constant term in the first, all the parameters are significant at the 5% level. We can refit the model with the Johansen normalization and the overidentifying constraint that the coefficient on kentucky in the second cointegrating equation is zero.

```
. constraint define 1 [_ce1]missouri = 1
. constraint define 2 [_ce1]indiana  = 0
. constraint define 3 [_ce2]missouri = 0
. constraint define 4 [_ce2]indiana  = 1
. constraint define 5 [_ce2]kentucky = 0
```

```
. vec missouri indiana kentucky illinois, trend(rconstant) rank(2)
> lags(4) noetable bconstraints(1/5)
Iteration 1:     log likelihood = 416.97177
  (output omitted )
Iteration 20:    log likelihood = 416.9744
Vector error-correction model
Sample: 1978m5 thru 2003m12                      Number of obs     =        308
                                                 AIC               =  -2.311522
Log likelihood =  416.9744                       HQIC              =  -2.016134
Det(Sigma_ml)  =  7.84e-07                       SBIC              =  -1.572769
Cointegrating equations
Equation           Parms    chi2     P>chi2
-------------------------------------------
_ce1                  2   145.233    0.0000
_ce2                  1  209.9344    0.0000
-------------------------------------------
Identification:  beta is overidentified
 ( 1)  [_ce1]missouri = 1
 ( 2)  [_ce1]indiana = 0
 ( 3)  [_ce2]missouri = 0
 ( 4)  [_ce2]indiana = 1
 ( 5)  [_ce2]kentucky = 0
------------------------------------------------------------------------------
        beta | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
_ce1         |
    missouri |          1          .        .       .            .           .
     indiana |          0  (omitted)
    kentucky |   .2521685   .1649653     1.53   0.126    -.0711576    .5754946
    illinois |  -1.037453   .1734165    -5.98   0.000    -1.377343   -.6975626
       _cons |  -.3891102   .4726968    -0.82   0.410    -1.315579    .5373586
-------------+----------------------------------------------------------------
_ce2         |
    missouri |          0  (omitted)
     indiana |          1          .        .       .            .           .
    kentucky |          0  (omitted)
    illinois |  -1.314265   .0907071   -14.49   0.000    -1.492048   -1.136483
       _cons |   2.937016   .6448924     4.55   0.000      1.67305    4.200982
------------------------------------------------------------------------------
LR test of identifying restrictions: chi2(1) = .3139       Prob > chi2 = 0.575
```

The test of the overidentifying restriction does not reject the null hypothesis that the restriction is valid, and the *p*-value on the coefficient on `kentucky` in the first cointegrating equation indicates that it is not significant. We will leave the variable in the model and attribute the lack of significance to whatever caused the `kentucky` series to temporarily rise above the others from 1985 until 1990, though we could instead consider removing `kentucky` from the model.

Next, we look at the estimates of the adjustment parameters. In the output below, we `replay` the previous results. We specify the `alpha` option so that `vec` will display an estimation table for the estimates of the adjustment parameters, and we specify `nobtable` to suppress the table for the parameters of the cointegrating equations because we have already looked at those.

```
. vec, alpha nobtable noetable
Vector error-correction model
Sample: 1978m5 thru 2003m12                     Number of obs     =        308
                                                AIC               =  -2.311522
Log likelihood =  416.9744                      HQIC              =  -2.016134
Det(Sigma_ml)  =  7.84e-07                      SBIC              =  -1.572769
Adjustment parameters
Equation           Parms    chi2     P>chi2
-------------------------------------------
D_missouri           2   19.39607   0.0001
D_indiana            2   6.426086   0.0402
D_kentucky           2   8.524901   0.0141
D_illinois           2   22.32893   0.0000
-------------------------------------------
```

alpha	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
D_missouri						
_ce1						
L1.	-.0683152	.0185763	-3.68	0.000	-.1047242	-.0319063
_ce2						
L1.	.0405613	.0112417	3.61	0.000	.018528	.0625946
D_indiana						
_ce1						
L1.	-.0342096	.0220955	-1.55	0.122	-.0775159	.0090967
_ce2						
L1.	.0325804	.0133713	2.44	0.015	.0063732	.0587877
D_kentucky						
_ce1						
L1.	-.0482012	.0231633	-2.08	0.037	-.0936004	-.0028021
_ce2						
L1.	.0374395	.0140175	2.67	0.008	.0099657	.0649133
D_illinois						
_ce1						
L1.	.0138224	.0227041	0.61	0.543	-.0306768	.0583215
_ce2						
L1.	.0567664	.0137396	4.13	0.000	.0298373	.0836955

```
LR test of identifying restrictions: chi2(1) = .3139        Prob > chi2 = 0.575
```

All the coefficients are significant at the 5% level, except those on Indiana and Illinois in the first cointegrating equation. From an economic perspective, the issue is whether the unemployment rates in Indiana and Illinois adjust when the first cointegrating equation is out of equilibrium. We could impose restrictions on one or both of those parameters and refit the model, or we could just decide to use the current results.

◁

❑ Technical note

vec can be used to fit models in which the parameters in $\boldsymbol{\beta}$ are not identified, in which case only the parameters in $\boldsymbol{\Pi}$ and the moving-average impact matrix $\mathbf{C}$ are identified. When the parameters in $\boldsymbol{\beta}$ are not identified, the values of $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\alpha}}$ can vary depending on the starting values. However, the estimates of $\boldsymbol{\Pi}$ and $\mathbf{C}$ are identified and have known asymptotic distributions. This method is valid because these additional normalization restrictions impose no restriction on $\boldsymbol{\Pi}$ or $\mathbf{C}$.

❑

Specification of constants and trends

As discussed in [TS] **vec intro**, allowing for a constant term and linear time trend allow us to write the VECM as

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}(\boldsymbol{\beta}\mathbf{y}_{t-1} + \boldsymbol{\mu} + \boldsymbol{\rho}t) + \sum_{i=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \boldsymbol{\gamma} + \boldsymbol{\tau}t + \epsilon_t$$

Five different trend specifications are available:

Option in `trend()`	Parameter restrictions	Johansen (1995) notation
`trend`	none	$H(r)$
`rtrend`	$\boldsymbol{\tau} = \mathbf{0}$	$H^*(r)$
`constant`	$\boldsymbol{\rho} = \mathbf{0}$, and $\boldsymbol{\tau} = \mathbf{0}$	$H_1(r)$
`rconstant`	$\boldsymbol{\rho} = \mathbf{0}$, $\boldsymbol{\gamma} = \mathbf{0}$ and $\boldsymbol{\tau} = \mathbf{0}$	$H_1^*(r)$
`none`	$\boldsymbol{\mu} = \mathbf{0}$, $\boldsymbol{\rho} = \mathbf{0}$, $\boldsymbol{\gamma} = \mathbf{0}$, and $\boldsymbol{\tau} = \mathbf{0}$	$H_2(r)$

`trend(trend)` allows for a linear trend in the cointegrating equations and a quadratic trend in the undifferenced data. A linear trend in the cointegrating equations implies that the cointegrating equations are assumed to be trend stationary.

`trend(rtrend)` defines a restricted trend model that excludes linear trends in the differenced data but allows for linear trends in the cointegrating equations. As in the previous case, a linear trend in a cointegrating equation implies that the cointegrating equation is trend stationary.

`trend(constant)` defines a model with an unrestricted constant. This allows for a linear trend in the undifferenced data and cointegrating equations that are stationary around a nonzero mean. This is the default.

`trend(rconstant)` defines a model with a restricted constant in which there is no linear or quadratic trend in the undifferenced data. A nonzero $\boldsymbol{\mu}$ allows for the cointegrating equations to be stationary around nonzero means, which provide the only intercepts for differenced data. Seasonal indicators are not allowed with this specification.

`trend(none)` defines a model that does not include a trend or a constant. When there is no trend or constant, the cointegrating equations are restricted to being stationary with zero means. Also, after adjusting for the effects of lagged endogenous variables, the differenced data are modeled as having mean zero. Seasonal indicators are not allowed with this specification.

❑ Technical note

vec uses a switching algorithm developed by Boswijk (1995) to maximize the log-likelihood function when constraints are placed on the parameters. The starting values affect both the ability of the algorithm to find a maximum and its speed in finding that maximum. By default, vec uses the parameter estimates that correspond to Johansen's normalization. Sometimes, other starting values will cause the algorithm to find a maximum faster.

To specify starting values for the parameters in $\boldsymbol{\alpha}$, we specify a $1 \times (K * r)$ matrix in the afrom() option. Specifying starting values for the parameters in $\boldsymbol{\beta}$ is slightly more complicated. As explained in *Methods and formulas*, specifying trend(constant), trend(rtrend), or trend(trend) causes some of the estimates of the trend parameters appearing in $\widehat{\boldsymbol{\beta}}$ to be "backed out". The switching algorithm estimates only the parameters of the cointegrating equations whose estimates are stored in e(betavec). For this reason, only the parameters stored in e(betavec) can have their initial values set via bfrom().

The table below describes which trend parameters in the cointegrating equations are estimated by the switching algorithm for each of the five specifications.

Trend specification	Trend parameters in cointegrating equations	Trend parameter estimated via switching algorithm
none	none	none
rconstant	_cons	_cons
constant	_cons	none
rtrend	_cons, _trend	_trend
trend	_cons, _trend	none

❑

Collinearity

As expected, collinearity among variables causes some parameters to be unidentified numerically. If vec encounters perfect collinearity among the dependent variables, it exits with an error.

In contrast, if vec encounters perfect collinearity that appears to be due to too many lags in the model, vec displays a warning message and reduces the maximum lag included in the model in an effort to find a model with fewer lags in which all the parameters are identified by the data. Specifying the noreduce option causes vec to skip over these additional checks and corrections for collinearity. Thus the noreduce option can be used to force the estimation to proceed when not all the parameters are identified by the data. When some parameters are not identified because of collinearity, the results cannot be interpreted but can be used to find the source of the collinearity.

Stored results

vec stores the following in e():

Scalars	
e(N)	number of observations
e(k_rank)	number of unconstrained parameters
e(k_eq)	number of equations in e(b)
e(k_dv)	number of dependent variables
e(k_ce)	number of cointegrating equations
e(n_lags)	number of lags
e(df_m)	model degrees of freedom
e(ll)	log likelihood
e(chi2_res)	value of test of overidentifying restrictions
e(df_lr)	degrees of freedom of the test of overidentifying restrictions
e(beta_iden)	1 if the parameters in $\boldsymbol{\beta}$ are identified and 0 otherwise
e(beta_icnt)	number of independent restrictions placed on $\boldsymbol{\beta}$
e(k_#)	number of variables in equation #
e(df_m#)	model degrees of freedom in equation #
e(r2_#)	R^2 of equation #
e(chi2_#)	$\chi 2$ statistic for equation #
e(rmse_#)	RMSE of equation #
e(aic)	value of AIC
e(hqic)	value of HQIC
e(sbic)	value of SBIC
e(tmin)	minimum time
e(tmax)	maximum time
e(detsig_ml)	determinant of the estimated covariance matrix
e(rank)	rank of e(V)
e(converge)	1 if the switching algorithm converged, 0 if it did not converge
Macros	
e(cmd)	vec
e(cmdline)	command as typed
e(trend)	trend specified
e(tsfmt)	format of the time variable
e(tvar)	variable denoting time within groups
e(endog)	endogenous variables
e(covariates)	list of covariates
e(eqnames)	equation names
e(cenames)	names of cointegrating equations
e(reduce_opt)	noreduce, if noreduce is specified
e(reduce_lags)	list of maximum lags to which the model has been reduced
e(title)	title in estimation output
e(aconstraints)	constraints placed on $\boldsymbol{\alpha}$
e(bconstraints)	constraints placed on $\boldsymbol{\beta}$
e(sindicators)	seasonal indicator variables
e(properties)	b V
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(marginsdefault)	default predict() specification for margins
Matrices	
e(b)	estimates of short-run parameters
e(V)	VCE of short-run parameter estimates
e(beta)	estimates of $\boldsymbol{\beta}$
e(V_beta)	VCE of $\widehat{\boldsymbol{\beta}}$
e(betavec)	directly obtained estimates of $\boldsymbol{\beta}$
e(pi)	estimates of $\widehat{\boldsymbol{\Pi}}$
e(V_pi)	VCE of $\widehat{\boldsymbol{\Pi}}$
e(alpha)	estimates of $\boldsymbol{\alpha}$
e(V_alpha)	VCE of $\widehat{\boldsymbol{\alpha}}$

`e(omega)`	estimates of $\widehat{\boldsymbol{\Omega}}$
`e(mai)`	estimates of $\mathbf{C}$
`e(V_mai)`	VCE of $\widehat{\mathbf{C}}$
Functions	
`e(sample)`	marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices	
`r(table)`	matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

Methods and formulas are presented under the following headings:

General specification of the VECM
The log-likelihood function
Unrestricted trend
Restricted trend
Unrestricted constant
Restricted constant
No trend
Estimation with Johansen identification
Estimation with constraints: β identified
Estimation with constraints: β not identified
Formulas for the information criteria
Formulas for predict

General specification of the VECM

`vec` estimates the parameters of a VECM that can be written as

$$\Delta\mathbf{y}_t = \boldsymbol{\alpha}\boldsymbol{\beta}'\mathbf{y}_{t-1} + \sum_{i=1}^{p-1}\boldsymbol{\Gamma}_i\Delta\mathbf{y}_{t-i} + \mathbf{v} + \boldsymbol{\delta}t + \mathbf{w}_1 s_1 + \cdots + \mathbf{w}_m s_m + \boldsymbol{\epsilon}_t \tag{1}$$

where

$\mathbf{y}_t$ is a $K \times 1$ vector of endogenous variables,

$\boldsymbol{\alpha}$ is a $K \times r$ matrix of parameters,

$\boldsymbol{\beta}$ is a $K \times r$ matrix of parameters,

$\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_{p-1}$ are $K \times K$ matrices of parameters,

$\mathbf{v}$ is a $K \times 1$ vector of parameters,

$\boldsymbol{\delta}$ is a $K \times 1$ vector of trend coefficients,

t is a linear time trend,

$s_1, \ldots, s_m$ are orthogonalized seasonal indicators specified in the `sindicators()` option, and

$\mathbf{w}_1, \ldots, \mathbf{w}_m$ are $K \times 1$ vectors of coefficients on the orthogonalized seasonal indicators.

There are two types of deterministic elements in (1): the trend, $\mathbf{v} + \boldsymbol{\delta}t$, and the orthogonalized seasonal terms, $\mathbf{w}_1 s_1 + \cdots + \mathbf{w}_m s_m$. Johansen (1995, chap. 11) shows that inference about the number of cointegrating equations is based on nonstandard distributions and that the addition of any term that generalizes the deterministic specification in (1) changes the asymptotic distributions of the statistics used for inference on the number of cointegrating equations and the asymptotic distribution of the ML estimator of the cointegrating equations. In fact, Johansen (1995, 84) notes that including event indicators causes the statistics used for inference on the number of cointegrating equations to have asymptotic distributions that must be computed case by case. For this reason, event indicators may not be specified in the present version of `vec`.

If seasonal indicators are included in the model, they cannot be collinear with a constant term. If they are collinear with a constant term, one of the indicator variables is omitted.

As discussed in *Specification of constants and trends*, we can reparameterize the model as

$$\Delta\mathbf{y}_t = \boldsymbol{\alpha}(\boldsymbol{\beta}\mathbf{y}_{t-1} + \boldsymbol{\mu} + \boldsymbol{\rho}t) + \sum_{i=1}^{p-1}\boldsymbol{\Gamma}_i\Delta\mathbf{y}_{t-i} + \boldsymbol{\gamma} + \boldsymbol{\tau}t + \epsilon_t \tag{2}$$

The log-likelihood function

We can maximize the log-likelihood function much more easily by writing it in concentrated form. In fact, as discussed below, in the simple case with the Johansen normalization on $\boldsymbol{\beta}$ and no constraints on $\boldsymbol{\alpha}$, concentrating the log-likelihood function produces an analytical solution for the parameter estimates.

To concentrate the log likelihood, rewrite (2) as

$$\mathbf{Z}_{0t} = \boldsymbol{\alpha}\widetilde{\boldsymbol{\beta}}'\mathbf{Z}_{1t} + \boldsymbol{\Psi}\mathbf{Z}_{2t} + \boldsymbol{\epsilon}_t \tag{3}$$

where $\mathbf{Z}_{0t}$ is a $K \times 1$ vector of variables $\Delta\mathbf{y}_t$, $\boldsymbol{\alpha}$ is the $K \times r$ matrix of adjustment coefficients, and $\boldsymbol{\epsilon}_t$ is a $K \times 1$ vector of independent and identically distributed normal vectors with mean 0 and contemporaneous covariance matrix $\boldsymbol{\Omega}$. $\mathbf{Z}_{1t}$, $\mathbf{Z}_{2t}$, $\widetilde{\boldsymbol{\beta}}$, and $\boldsymbol{\Psi}$ depend on the trend specification and are defined below.

The log-likelihood function for the model in (3) is

$$\begin{aligned} L = -\frac{1}{2}\Big\{&TK\ln(2\pi) + T\ln(|\boldsymbol{\Omega}|) \\ &+ \sum_{t=1}^{T}(\mathbf{Z}_{0t} - \boldsymbol{\alpha}\widetilde{\boldsymbol{\beta}}'\mathbf{Z}_{1t} - \boldsymbol{\Psi}\mathbf{Z}_{2t})'\boldsymbol{\Omega}^{-1}(\mathbf{Z}_{0t} - \boldsymbol{\alpha}\widetilde{\boldsymbol{\beta}}'\mathbf{Z}_{1t} - \boldsymbol{\Psi}\mathbf{Z}_{2t})\Big\} \end{aligned} \tag{4}$$

with the constraints that $\boldsymbol{\alpha}$ and $\widetilde{\boldsymbol{\beta}}$ have rank r.

Johansen (1995, chap. 6), building on Anderson (1951), shows how the $\boldsymbol{\Psi}$ parameters can be expressed as analytic functions of $\boldsymbol{\alpha}$, $\widetilde{\boldsymbol{\beta}}$, and the data, yielding the concentrated log-likelihood function

$$\begin{aligned} L_c = -\frac{1}{2}\Big\{&TK\ln(2\pi) + T\ln(|\boldsymbol{\Omega}|) \\ &+ \sum_{t=1}^{T}(\mathbf{R}_{0t} - \boldsymbol{\alpha}\widetilde{\boldsymbol{\beta}}'\mathbf{R}_{1t})'\boldsymbol{\Omega}^{-1}(\mathbf{R}_{0t} - \boldsymbol{\alpha}\widetilde{\boldsymbol{\beta}}'\mathbf{R}_{1t})\Big\} \end{aligned} \tag{5}$$

where

$\mathbf{M}_{ij} = T^{-1}\sum_{t=1}^{T} \mathbf{Z}_{it}\mathbf{Z}'_{jt}, \qquad i,j \in \{0,1,2\};$

$\mathbf{R}_{0t} = \mathbf{Z}_{0t} - \mathbf{M}_{02}\mathbf{M}_{22}^{-1}\mathbf{Z}_{2t}$; and

$\mathbf{R}_{1t} = \mathbf{Z}_{1t} - \mathbf{M}_{12}\mathbf{M}_{22}^{-1}\mathbf{Z}_{2t}$.

The definitions of $\mathbf{Z}_{1t}$, $\mathbf{Z}_{2t}$, $\widetilde{\boldsymbol\beta}$, and $\boldsymbol\Psi$ change with the trend specifications, although some of their components stay the same.

Unrestricted trend

When the trend in the VECM is unrestricted, we can define the variables in (3) directly in terms of the variables in (1):

$\mathbf{Z}_{1t} = \mathbf{y}_{t-1}$ is $K \times 1$

$\mathbf{Z}_{2t} = (\Delta\mathbf{y}'_{t-1},\dots,\Delta\mathbf{y}'_{t-p+1},1,t,s_1,\dots,s_m)'$ is $\{K(p-1)+2+m\}\times 1$;

$\boldsymbol\Psi = (\boldsymbol\Gamma_1,\dots,\boldsymbol\Gamma_{p-1},\mathbf{v},\boldsymbol\delta,\mathbf{w}_1,\dots,\mathbf{w}_m)$ is $K \times \{K(p-1)+2+m\}$

$\widetilde{\boldsymbol\beta} = \boldsymbol\beta$ is the $K \times r$ matrix composed of the r cointegrating vectors.

In the unrestricted trend specification, $m_1 = K$, $m_2 = K(p-1)+2+m$, and there are $n_{\text{parms}} = Kr + Kr + K\{K(p-1)+2+m\}$ parameters in (3).

Restricted trend

When there is a restricted trend in the VECM in (2), $\boldsymbol\tau = 0$, but the intercept $\mathbf{v} = \boldsymbol\alpha\boldsymbol\mu + \boldsymbol\gamma$ is unrestricted. The VECM with the restricted trend can be written as

$$\Delta\mathbf{y}_t = \boldsymbol\alpha(\boldsymbol\beta',\boldsymbol\rho)\begin{pmatrix}\mathbf{y}_{t-1}\\ t\end{pmatrix} + \sum_{i=1}^{p-1}\boldsymbol\Gamma_i\Delta\mathbf{y}_{t-i} + \mathbf{v} + \mathbf{w}_1 s_1 + \cdots + \mathbf{w}_m s_m + \epsilon_t$$

This equation can be written in the form of (3) by defining

$\mathbf{Z}_{1t} = \left(\mathbf{y}'_{t-1},t\right)'$ is $(K+1)\times 1$

$\mathbf{Z}_{2t} = (\Delta\mathbf{y}'_{t-1},\dots,\Delta\mathbf{y}'_{t-p+1},1,s_1,\dots,s_m)'$ is $\{K(p-1)+1+m\}\times 1$

$\boldsymbol\Psi = (\boldsymbol\Gamma_1,\dots,\boldsymbol\Gamma_{p-1},\mathbf{v},\mathbf{w}_1,\dots,\mathbf{w}_m)$ is $K \times \{K(p-1)+1+m\}$

$\widetilde{\boldsymbol\beta} = \left(\boldsymbol\beta',\boldsymbol\rho\right)'$ is the $(K+1)\times r$ matrix composed of the r cointegrating vectors and the r trend coefficients $\boldsymbol\rho$

In the restricted trend specification, $m_1 = K+1$, $m_2 = \{K(p-1)+1+m\}$, and there are $n_{\text{parms}} = Kr + (K+1)r + K\{K(p-1)+1+m\}$ parameters in (3).

Unrestricted constant

An unrestricted constant in the VECM in (2) is equivalent to setting $\boldsymbol\delta = 0$ in (1), which can be written in the form of (3) by defining

$\mathbf{Z}_{1t} = \mathbf{y}_{t-1}$ is $(K \times 1)$

$\mathbf{Z}_{2t} = (\Delta\mathbf{y}'_{t-1},\dots,\Delta\mathbf{y}'_{t-p+1},1,s_1,\dots,s_m)'$ is $\{K(p-1)+1+m\}\times 1$;

$\boldsymbol\Psi = (\boldsymbol\Gamma_1,\dots,\boldsymbol\Gamma_{p-1},\mathbf{v},\mathbf{w}_1,\dots,\mathbf{w}_m)$ is $K \times \{K(p-1)+1+m\}$

$\widetilde{\boldsymbol\beta} = \boldsymbol\beta$ is the $K \times r$ matrix composed of the r cointegrating vectors

In the unrestricted constant specification, $m_1 = K$, $m_2 = \{K(p-1)+1+m\}$, and there are $n_{\rm parms} = Kr + Kr + K\{K(p-1)+1+m\}$ parameters in (3).

Restricted constant

When there is a restricted constant in the VECM in (2), it can be written in the form of (3) by defining

$\mathbf{Z}_{1t} = \left(\mathbf{y}'_{t-1}, 1\right)'$ is $(K+1)\times 1$

$\mathbf{Z}_{2t} = (\Delta\mathbf{y}'_{t-1}, \ldots, \Delta\mathbf{y}'_{t-p+1})'$ is $K(p-1)\times 1$

$\boldsymbol{\Psi} = (\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_{p-1})$ is $K \times K(p-1)$

$\widetilde{\boldsymbol{\beta}} = \left(\boldsymbol{\beta}', \boldsymbol{\mu}\right)'$ is the $(K+1)\times r$ matrix composed of the r cointegrating vectors and the r constants in the cointegrating relations.

In the restricted trend specification, $m_1 = K+1$, $m_2 = K(p-1)$, and there are $n_{\rm parms} = Kr + (K+1)r + K\{K(p-1)\}$ parameters in (3).

No trend

When there is no trend in the VECM in (2), it can be written in the form of (3) by defining

$\mathbf{Z}_{1t} = \mathbf{y}_{t-1}$ is $K \times 1$

$\mathbf{Z}_{2t} = (\Delta\mathbf{y}'_{t-1}, \ldots, \Delta\mathbf{y}'_{t-p+1})'$ is $K(p-1)+m \times 1$

$\boldsymbol{\Psi} = (\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_{p-1})$ is $K \times K(p-1)$

$\widetilde{\boldsymbol{\beta}} = \boldsymbol{\beta}$ is $K \times r$ matrix of r cointegrating vectors

In the no-trend specification, $m_1 = K$, $m_2 = K(p-1)$, and there are $n_{\rm parms} = Kr + Kr + K\{K(p-1)\}$ parameters in (3).

Estimation with Johansen identification

Not all the parameters in $\boldsymbol{\alpha}$ and $\widetilde{\boldsymbol{\beta}}$ are identified. Consider the simple case in which $\widetilde{\boldsymbol{\beta}}$ is $K \times r$ and let $\mathbf{Q}$ be a nonsingular $r \times r$ matrix. Then

$$\boldsymbol{\alpha}\widetilde{\boldsymbol{\beta}}' = \boldsymbol{\alpha}\mathbf{Q}\mathbf{Q}^{-1}\widetilde{\boldsymbol{\beta}}' = \boldsymbol{\alpha}\mathbf{Q}(\widetilde{\boldsymbol{\beta}}\mathbf{Q}'^{-1})' = \dot{\boldsymbol{\alpha}}\dot{\boldsymbol{\beta}}'$$

Substituting $\dot{\boldsymbol{\alpha}}\dot{\boldsymbol{\beta}}'$ into the log likelihood in (5) for $\boldsymbol{\alpha}\widetilde{\boldsymbol{\beta}}'$ would not change the value of the log likelihood, so some a priori identification restrictions must be found to identify $\boldsymbol{\alpha}$ and $\widetilde{\boldsymbol{\beta}}$. As discussed in Johansen (1995, chap. 5 and 6) and Boswijk (1995), if the restrictions exactly identify or overidentify $\widetilde{\boldsymbol{\beta}}$, the estimates of the unconstrained parameters in $\widetilde{\boldsymbol{\beta}}$ will be superconsistent, meaning that the estimates of the free parameters in $\widetilde{\boldsymbol{\beta}}$ will converge at a faster rate than estimates of the short-run parameters in $\boldsymbol{\alpha}$ and $\boldsymbol{\Gamma}_i$. This allows the distribution of the estimator of the short-run parameters to be derived conditional on the estimated $\widetilde{\boldsymbol{\beta}}$.

Johansen (1995, chap. 6) has proposed a normalization method for use when theory does not provide sufficient a priori restrictions to identify the cointegrating vector. This method has become widely adopted by researchers. Johansen's identification scheme is

$$\widetilde{\boldsymbol{\beta}}' = (\mathbf{I}_r, \breve{\boldsymbol{\beta}}') \tag{6}$$

where $\mathbf{I}_r$ is the $r \times r$ identity matrix and $\breve{\boldsymbol{\beta}}$ is a $(m_1 - r)\times r$ matrix of identified parameters.

Johansen's identification method places r^2 linearly independent constraints on the parameters in $\widetilde{\boldsymbol{\beta}}$, thereby defining an exactly identified model. The total number of freely estimated parameters is $n_{\rm parms} - r^2 = \{K + m_2 + (K + m_1 - r)r\}$, and the degrees of freedom d is calculated as the integer part of $(n_{\rm parms} - r^2)/K$.

When only the rank and the Johansen identification restrictions are placed on the model, we can further manipulate the log likelihood in (5) to obtain analytic formulas for the parameters in $\widetilde{\boldsymbol{\beta}}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\Omega}$. For a given value of $\widetilde{\boldsymbol{\beta}}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\Omega}$ can be found by regressing $\mathbf{R}_{0t}$ on $\widetilde{\boldsymbol{\beta}}'\mathbf{R}_{1t}$. This allows a further simplification of the problem in which

$$
\begin{aligned}
\boldsymbol{\alpha}(\widetilde{\boldsymbol{\beta}}) &= \mathbf{S}_{01}\widetilde{\boldsymbol{\beta}}(\widetilde{\boldsymbol{\beta}}'\mathbf{S}_{11}\widetilde{\boldsymbol{\beta}})^{-1}\\
\boldsymbol{\Omega}(\widetilde{\boldsymbol{\beta}}) &= \mathbf{S}_{00} - \mathbf{S}_{01}\widetilde{\boldsymbol{\beta}}(\widetilde{\boldsymbol{\beta}}'\mathbf{S}_{11}\widetilde{\boldsymbol{\beta}})^{-1}\widetilde{\boldsymbol{\beta}}'\mathbf{S}_{10}\\
\mathbf{S}_{ij} &= (1/T)\textstyle\sum_{t=1}^{T} R_{it}R'_{jt} \qquad\qquad i,j \in \{0,1\}
\end{aligned}
$$

Johansen (1995) shows that by inserting these solutions into equation (5), $\widehat{\boldsymbol{\beta}}$ is given by the r eigenvectors $\mathbf{v}_1, \dots, \mathbf{v}_r$ corresponding to the r largest eigenvalues $\lambda_1, \dots, \lambda_r$ that solve the generalized eigenvalue problem

$$
|\lambda_i \mathbf{S}_{11} - \mathbf{S}_{10}\mathbf{S}_{00}^{-1}\mathbf{S}_{01}| = 0 \tag{7}
$$

The eigenvectors corresponding to $\lambda_1, \dots, \lambda_r$ that solve (7) are the unidentified parameter estimates. To impose the identification restrictions in (6), we normalize the eigenvectors such that

$$
\lambda_i \mathbf{S}_{11}\mathbf{v}_i = \mathbf{S}_{01}\mathbf{S}_{00}^{-1}\mathbf{S}_{01}\mathbf{v}_i \tag{8}
$$

and

$$
\mathbf{v}_i'\mathbf{S}_{11}\mathbf{v}_j = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ otherwise} \end{cases} \tag{9}
$$

At the optimum the log-likelihood function with the Johansen identification restrictions can be expressed in terms of T, K, $\mathbf{S}_{00}$, and the r largest eigenvalues

$$
L_c = -\frac{1}{2}T\left\{K\ln(2\pi) + K + \ln(|\mathbf{S}_{00}|) + \sum_{i=1}^{r}\ln(1-\widehat{\lambda}_i)\right\}
$$

where the $\widehat{\lambda}_i$ are the eigenvalues that solve (7), (8), and (9).

Using the normalized $\widehat{\boldsymbol{\beta}}$, we can then obtain the estimates

$$
\widehat{\boldsymbol{\alpha}} = \mathbf{S}_{01}\widehat{\boldsymbol{\beta}}(\widehat{\boldsymbol{\beta}}'S_{11}\widehat{\boldsymbol{\beta}})^{-1} \tag{10}
$$

and

$$
\widehat{\boldsymbol{\Omega}} = \mathbf{S}_{00} - \widehat{\boldsymbol{\alpha}}\widehat{\boldsymbol{\beta}}'\mathbf{S}_{10}
$$

Let $\widehat{\boldsymbol{\beta}}_y$ be a $K \times r$ matrix that contains the estimates of the parameters in $\boldsymbol{\beta}$ in (1). $\widehat{\boldsymbol{\beta}}_y$ differs from $\widehat{\boldsymbol{\beta}}$ in that any trend parameter estimates are omitted from $\widehat{\boldsymbol{\beta}}$. We can then use $\widehat{\boldsymbol{\beta}}_y$ to obtain predicted values for the r nondemeaned cointegrating equations

$$
\widehat{\widehat{\mathbf{E}}}_t = \widehat{\boldsymbol{\beta}}_y'\mathbf{y}_t
$$

The r series in $\widehat{\widehat{E}}_t$ are called the predicted, nondemeaned cointegrating equations because they still contain the terms $\boldsymbol{\mu}$ and $\boldsymbol{\rho}$. We want to work with the predicted, demeaned cointegrating equations. Thus we need estimates of $\boldsymbol{\mu}$ and $\boldsymbol{\rho}$. In the `trend(rconstant)` specification, the algorithm directly produces the estimator $\widehat{\boldsymbol{\mu}}$. Similarly, in the `trend(rtrend)` specification, the algorithm directly produces the estimator $\widehat{\boldsymbol{\rho}}$. In the remaining cases, to back out estimates of $\boldsymbol{\mu}$ and $\boldsymbol{\rho}$, we need estimates of $\mathbf{v}$ and $\boldsymbol{\delta}$, which we can obtain by estimating the parameters of the following VAR:

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}\widehat{\widehat{\mathbf{E}}}_{t-1} + \sum_{i=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \mathbf{v} + \boldsymbol{\delta} t + \mathbf{w}_1 s_1 + \cdots + \mathbf{w}_m s_m + \boldsymbol{\epsilon}_t \tag{11}$$

Depending on the trend specification, we use $\widehat{\boldsymbol{\alpha}}$ to back out the estimates of

$$\widehat{\boldsymbol{\mu}} = (\widehat{\boldsymbol{\alpha}}'\widehat{\boldsymbol{\alpha}})^{-1}\widehat{\boldsymbol{\alpha}}'\widehat{\mathbf{v}} \tag{12}$$

$$\widehat{\boldsymbol{\rho}} = (\widehat{\boldsymbol{\alpha}}'\widehat{\boldsymbol{\alpha}})^{-1}\widehat{\boldsymbol{\alpha}}'\widehat{\boldsymbol{\delta}} \tag{13}$$

if they are not already in $\widehat{\boldsymbol{\beta}}$ and are included in the trend specification.

We then augment $\widehat{\boldsymbol{\beta}}_y$ to

$$\widehat{\boldsymbol{\beta}}_f' = (\widehat{\boldsymbol{\beta}}_y', \widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\rho}})$$

where the estimates of $\widehat{\boldsymbol{\mu}}$ and $\widehat{\boldsymbol{\rho}}$ are either obtained from $\widehat{\boldsymbol{\beta}}$ or backed out using (12) and (13). We next use $\widehat{\boldsymbol{\beta}}_f$ to obtain the r predicted, demeaned cointegrating equations, $\widehat{\mathbf{E}}_t$, via

$$\widehat{\mathbf{E}}_t = \widehat{\boldsymbol{\beta}}_f' (\mathbf{y}_t', 1, t)'$$

We last obtain estimates of all the short-run parameters from the VAR:

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}\widehat{\mathbf{E}}_{t-1} + \sum_{i=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \boldsymbol{\gamma} + \boldsymbol{\tau} t + \mathbf{w}_1 s_1 + \cdots + \mathbf{w}_m s_m + \boldsymbol{\epsilon}_t \tag{14}$$

Because the estimator $\widehat{\boldsymbol{\beta}}_f$ converges in probability to its true value at a rate faster than $T^{-\frac{1}{2}}$, we can take our estimated $\widehat{\mathbf{E}}_{t-1}$ as given data in (14). This allows us to estimate the variance–covariance (VCE) matrix of the estimates of the parameters in (14) by using the standard VAR VCE estimator. Equation (11) can be used to obtain consistent estimates of all the parameters and of the VCE of all the parameters, except $\mathbf{v}$ and $\boldsymbol{\delta}$. The standard VAR VCE of $\widehat{\mathbf{v}}$ and $\widehat{\boldsymbol{\delta}}$ is incorrect because these estimates converge at a faster rate. This is why it is important to use the predicted, demeaned cointegrating equations, $\widehat{\mathbf{E}}_{t-1}$, when estimating the short-run parameters and trend terms. In keeping with the cointegration literature, `vec` makes a small-sample adjustment to the VCE estimator so that the divisor is $(T-d)$ instead of T, where d represents the degrees of freedom of the model. d is calculated as the integer part of n_{parms}/K, where n_{parms} is the total number of freely estimated parameters in the model.

In the `trend(rconstant)` specification, the estimation procedure directly estimates $\boldsymbol{\mu}$. For `trend(constant)`, `trend(rtrend)`, and `trend(trend)`, the estimates of $\boldsymbol{\mu}$ are backed out using (12). In the `trend(rtrend)` specification, the estimation procedure directly estimates $\boldsymbol{\rho}$. In the `trend(trend)` specification, the estimates of $\boldsymbol{\rho}$ are backed out using (13). Because the elements of the estimated VCE are readily available only when the estimates are obtained directly, when the trend parameter estimates are backed out, their elements in the VCE for $\widehat{\boldsymbol{\beta}}_f$ are missing.

Under the Johansen identification restrictions, `vec` obtains $\widehat{\boldsymbol\beta}$, the estimates of the parameters in the $r \times m_1$ matrix $\widetilde{\boldsymbol\beta}'$ in (5). The VCE of vec($\widehat{\boldsymbol\beta}$) is $rm_1 \times rm_1$. Per Johansen (1995), the asymptotic distribution of $\widehat{\boldsymbol\beta}$ is mixed Gaussian, and its VCE is consistently estimated by

$$\left(\frac{1}{T-d}\right)(\mathbf{I}_r \otimes \mathbf{H}_J)\left\{(\widehat{\boldsymbol\alpha}'\boldsymbol\Omega^{-1}\widehat{\boldsymbol\alpha}) \otimes (\mathbf{H}_J'\mathbf{S}_{11}\mathbf{H}_J)\right\}^{-1}(\mathbf{I}_r \otimes \mathbf{H}_J)' \tag{15}$$

where $\mathbf{H}_J$ is the $m_1 \times (m_1 - r)$ matrix given by $\mathbf{H}_J = (\mathbf{0}'_{r\times(m_1-r)}, \mathbf{I}_{m_1-r})'$. The VCE reported in `e(V_beta)` is the estimated VCE in (15) augmented with missing values to account for any backed-out estimates of $\boldsymbol\mu$ or $\boldsymbol\rho$.

The parameter estimates $\widehat{\boldsymbol\alpha}$ can be found either as a function of $\widehat{\boldsymbol\beta}$, using (10) or from the VAR in (14). The estimated VCE of $\widehat{\boldsymbol\alpha}$ reported in `e(V_alpha)` is given by

$$\frac{1}{(T-d)}\widehat{\boldsymbol\Omega} \otimes \widehat{\boldsymbol\Sigma}_B$$

where $\widehat{\boldsymbol\Sigma}_B = (\widehat{\boldsymbol\beta}'\mathbf{S}_{11}\widehat{\boldsymbol\beta})^{-1}$.

As we would expect, the estimator of $\Pi = \boldsymbol\alpha\boldsymbol\beta'$ is

$$\widehat{\Pi} = \widehat{\boldsymbol\alpha}\widehat{\boldsymbol\beta}'$$

and its estimated VCE is given by

$$\frac{1}{(T-d)}\widehat{\boldsymbol\Omega} \otimes (\widehat{\boldsymbol\beta}\widehat{\boldsymbol\Sigma}_B\widehat{\boldsymbol\beta}')$$

The moving-average impact matrix $\mathbf{C}$ is estimated by

$$\widehat{\mathbf{C}} = \widehat{\boldsymbol\beta}_\perp(\widehat{\boldsymbol\alpha}_\perp\widehat{\boldsymbol\Gamma}\widehat{\boldsymbol\beta}_\perp)^{-1}\widehat{\boldsymbol\alpha}'_\perp$$

where $\widehat{\boldsymbol\beta}_\perp$ is the orthogonal complement of $\widehat{\boldsymbol\beta}_y$, $\widehat{\boldsymbol\alpha}_\perp$ is the orthogonal complement of $\widehat{\boldsymbol\alpha}$, and $\widehat{\boldsymbol\Gamma} = \mathbf{I_K} - \sum_{i=1}^{p-1}\boldsymbol\Gamma_i$. The orthogonal complement of a $K \times r$ matrix $\mathbf{Q}$ that has rank r is a matrix $\mathbf{Q}_\perp$ of rank $K - r$, such that $\mathbf{Q}'\mathbf{Q}_\perp = \mathbf{0}$. Although this operation is not uniquely defined, the results used by `vec` do not depend on the method of obtaining the orthogonal complement. `vec` uses the following method: the orthogonal complement of $\mathbf{Q}$ is given by the r eigenvectors with the highest eigenvalues from the matrix $\mathbf{Q}'(\mathbf{Q}'\mathbf{Q})^{-1}\mathbf{Q}'$.

Per Johansen (1995, chap. 13) and Drukker (2004), the VCE of $\widehat{\mathbf{C}}$ is estimated by

$$\frac{T-d}{T}\widehat{\mathbf{S}}_q\widehat{\mathbf{V}}_{\widehat{\boldsymbol\nu}}\widehat{\mathbf{S}}'_q \tag{16}$$

where

$$\widehat{\mathbf{S}}_q = \widehat{\mathbf{C}} \otimes \widehat{\boldsymbol\xi}$$

$$\widehat{\boldsymbol\xi} = \begin{cases} (\widehat{\boldsymbol\xi}_1, \widehat{\boldsymbol\xi}_2) & \text{if } \ p > 1 \\ \widehat{\boldsymbol\xi}_1 & \text{if } \ p = 1 \end{cases}$$

$$\widehat{\boldsymbol\xi}_1 = (\widehat{\mathbf{C}}'\widehat{\boldsymbol\Gamma}' - \mathbf{I}_K)\bar{\boldsymbol\alpha}$$

$$\bar{\boldsymbol\alpha} = \widehat{\boldsymbol\alpha}(\widehat{\boldsymbol\alpha}'\widehat{\boldsymbol\alpha})^{-1}$$

$$\widehat{\boldsymbol\xi}_2 = \boldsymbol\iota_{p-1} \otimes \widehat{\mathbf{C}}$$

$\boldsymbol\iota_{p-1}$ is a $(p-1) \times 1$ vector of ones

$\widehat{\mathbf{V}}_{\widehat{\boldsymbol\nu}}$ is the estimated VCE of $\widehat{\boldsymbol\nu} = (\widehat{\boldsymbol\alpha}, \widehat{\boldsymbol\Gamma}_1, \ldots \widehat{\boldsymbol\Gamma}_{p-1})$

Estimation with constraints: β identified

vec can also fit models in which the adjustment parameters are subject to homogeneous linear constraints and the cointegrating vectors are subject to general linear restrictions. Mathematically, vec allows for constraints of the form

$$\mathbf{R}'_{\boldsymbol{\alpha}}\text{vec}(\boldsymbol{\alpha}) = \mathbf{0} \tag{17}$$

where $\mathbf{R}_{\boldsymbol{\alpha}}$ is a known $Kr \times n_{\boldsymbol{\alpha}}$ constraint matrix, and

$$\mathbf{R}'_{\widetilde{\boldsymbol{\beta}}}\text{vec}(\widetilde{\boldsymbol{\beta}}) = \mathbf{b} \tag{18}$$

where $\mathbf{R}_{\widetilde{\boldsymbol{\beta}}}$ is a known $m_1 r \times n_{\boldsymbol{\beta}}$ constraint matrix and $\mathbf{b}$ is a known $n_{\boldsymbol{\beta}} \times 1$ vector of constants. Although (17) and (18) are intuitive, they can be rewritten in a form to facilitate computation. Specifically, (17) can be written as

$$\text{vec}(\boldsymbol{\alpha}') = \mathbf{Ga} \tag{19}$$

where $\mathbf{G}$ is $Kr \times n_{\boldsymbol{\alpha}}$ and $\mathbf{a}$ is $n_{\boldsymbol{\alpha}} \times 1$. Equation (18) can be rewritten as

$$\text{vec}(\widetilde{\boldsymbol{\beta}}) = \mathbf{Hb} + \mathbf{h}_0 \tag{20}$$

where $\mathbf{H}$ is a known $n_1 r \times n_{\boldsymbol{\beta}}$ matrix, $\mathbf{b}$ is an $n_{\boldsymbol{\beta}} \times 1$ matrix of parameters, and $\mathbf{h}_0$ is a known $n_1 r \times 1$ matrix. See [P] **makecns** for a discussion of the different ways of specifying the constraints.

When constraints are specified via the aconstraints() and bconstraints() options, the Boswijk (1995) rank method determines whether the parameters in $\widetilde{\boldsymbol{\beta}}$ are underidentified, exactly identified, or overidentified.

Boswijk (1995) uses the Rothenberg (1971) method to determine whether the parameters in $\widetilde{\boldsymbol{\beta}}$ are identified. Thus the parameters in $\widetilde{\boldsymbol{\beta}}$ are exactly identified if $\rho_{\boldsymbol{\beta}} = r^2$, and the parameters in $\widetilde{\boldsymbol{\beta}}$ are overidentified if $\rho_{\boldsymbol{\beta}} > r^2$, where

$$\rho_{\boldsymbol{\beta}} = \text{rank}\left\{\mathbf{R}_{\widetilde{\boldsymbol{\beta}}}(\mathbf{I}_r \otimes \ddot{\boldsymbol{\beta}})\right\}$$

and $\ddot{\boldsymbol{\beta}}$ is a full-rank matrix with the same dimensions as $\widetilde{\boldsymbol{\beta}}$. The computed $\rho_{\boldsymbol{\beta}}$ is stored in e(beta_icnt).

Similarly, the number of freely estimated parameters in $\boldsymbol{\alpha}$ and $\widetilde{\boldsymbol{\beta}}$ is given by ρ_{jacob}, where

$$\rho_{\text{jacob}} = \text{rank}\left\{(\widehat{\boldsymbol{\alpha}} \otimes \mathbf{I}_{m_1})\mathbf{H}, (\mathbf{I}_K \otimes \widehat{\boldsymbol{\beta}})\mathbf{G}\right\}$$

Using ρ_{jacob}, we can calculate several other parameter counts of interest. In particular, the degrees of freedom of the overidentifying test are given by $(K + m_1 - r)r - \rho_{\text{jacob}}$, and the number of freely estimated parameters in the model is $n_{\text{parms}} = Km_2 + \rho_{\text{jacob}}$.

Although the problem of maximizing the log-likelihood function in (4), subject to the constraints in (17) and (18), could be handled by the algorithms in [R] **ml**, the switching algorithm of Boswijk (1995) has proven to be more convergent. For this reason, vec uses the Boswijk (1995) switching algorithm to perform the optimization.

Given starting values $(\widehat{\mathbf{b}}_0, \widehat{\mathbf{a}}_0, \widehat{\mathbf{\Omega}}_0)$, the algorithm iteratively updates the estimates until convergence is achieved, as follows:

$\widehat{\boldsymbol{\alpha}}_j$ is constructed from (19) and $\widehat{\mathbf{a}}_j$

$\widehat{\boldsymbol{\beta}}_j$ is constructed from (20) and $\widehat{\mathbf{b}}_j$

$\widehat{\mathbf{b}}_{j+1} = \{\mathbf{H}'(\widehat{\boldsymbol{\alpha}}_j'\widehat{\mathbf{\Omega}}_j^{-1}\widehat{\boldsymbol{\alpha}}_j \otimes \mathbf{S}_{11})\mathbf{H}\}^{-1}\mathbf{H}'(\widehat{\boldsymbol{\alpha}}_j\widehat{\mathbf{\Omega}}_j^{-1} \otimes \mathbf{S}_{11})\{\mathrm{vec}(\widehat{\mathbf{P}}) - (\widehat{\boldsymbol{\alpha}}_j \otimes \mathbf{I}_{n_{Z1}})\mathbf{h}_0\}$

$\widehat{\mathbf{a}}_{j+1} = \{\mathbf{G}(\widehat{\mathbf{\Omega}}_j^{-1} \otimes \widehat{\boldsymbol{\beta}}_j\mathbf{S}_{11}\widehat{\boldsymbol{\beta}}_j)\mathbf{G}\}^{-1}\mathbf{G}'(\widehat{\mathbf{\Omega}}_j^{-1} \otimes \widehat{\boldsymbol{\beta}}_j\mathbf{S}_{11})\mathrm{vec}(\widehat{\mathbf{P}})$

$\widehat{\Omega}_{j+1} = \mathbf{S}_{00} - \mathbf{S}_{01}\widehat{\boldsymbol{\beta}}_j\widehat{\boldsymbol{\alpha}}_j' - \widehat{\boldsymbol{\alpha}}_j\widehat{\boldsymbol{\beta}}_j'\mathbf{S}_{10} + \widehat{\boldsymbol{\alpha}}_j\widehat{\boldsymbol{\beta}}_j'\mathbf{S}_{11}\widehat{\boldsymbol{\beta}}_j\widehat{\boldsymbol{\alpha}}_j'$

The estimated VCE of $\widehat{\boldsymbol{\beta}}$ is given by

$$\frac{1}{(T-d)}\mathbf{H}\{\mathbf{H}'(\mathbf{W} \otimes \mathbf{S}_{11})\mathbf{H}\}^{-1}\mathbf{H}'$$

where $\mathbf{W}$ is $\widehat{\boldsymbol{\alpha}}'\widehat{\mathbf{\Omega}}^{-1}\widehat{\boldsymbol{\alpha}}$. As in the case without constraints, the estimated VCE of $\widehat{\boldsymbol{\alpha}}$ can be obtained either from the VCE of the short-run parameters, as described below, or via the formula

$$\widehat{V}_{\widehat{\boldsymbol{\alpha}}} = \frac{1}{(T-d)}\mathbf{G}\left[\mathbf{G}'\left\{\widehat{\mathbf{\Omega}}^{-1} \otimes (\widehat{\boldsymbol{\beta}}'\mathbf{S}_{11}\widehat{\boldsymbol{\beta}})\mathbf{G}\right\}^{-1}\right]\mathbf{G}'$$

Boswijk (1995) notes that, as long as the parameters of the cointegrating equations are exactly identified or overidentified, the constrained ML estimator produces superconsistent estimates of $\widetilde{\boldsymbol{\beta}}$. This implies that the method of estimating the short-run parameters described above applies in the presence of constraints, as well, albeit with a caveat: when there are constraints placed on $\boldsymbol{\alpha}$, the VARs must be estimated subject to these constraints.

With these estimates and the estimated VCE of the short-run parameter matrix $\widehat{\mathbf{V}}_{\widehat{\boldsymbol{\nu}}}$, Drukker (2004) shows that the estimated VCE for $\widehat{\Pi}$ is given by

$$(\widehat{\boldsymbol{\beta}} \otimes \mathbf{I}_K)\widehat{V}_{\widehat{\boldsymbol{\alpha}}}(\widehat{\boldsymbol{\beta}} \otimes \mathbf{I}_K)'$$

Drukker (2004) also shows that the estimated VCE of $\widehat{\mathbf{C}}$ can be obtained from (16) with the extension that $\widehat{V}_{\widehat{\boldsymbol{\nu}}}$ is the estimated VCE of $\widehat{\nu}$ that takes into account any constraints on $\widehat{\boldsymbol{\alpha}}$.

Estimation with constraints: β not identified

When the parameters in $\boldsymbol{\beta}$ are not identified, only the parameters in $\mathbf{\Pi} = \boldsymbol{\alpha}\boldsymbol{\beta}$ and $\mathbf{C}$ are identified. The estimates of $\mathbf{\Pi}$ and $\mathbf{C}$ would not change if more identification restrictions were imposed to achieve exact identification. Thus the VCE matrices for $\widehat{\mathbf{\Pi}}$ and $\widehat{\mathbf{C}}$ can be derived as if the model exactly identified $\boldsymbol{\beta}$.

Formulas for the information criteria

The AIC, SBIC, and HQIC are calculated according to their standard definitions, which include the constant term from the log likelihood; that is,

$$\text{AIC} = -2\left(\frac{L}{T}\right) + \frac{2n_{\text{parms}}}{T}$$

$$\text{SBIC} = -2\left(\frac{L}{T}\right) + \frac{\ln(T)}{T}n_{\text{parms}}$$

$$\text{HQIC} = -2\left(\frac{L}{T}\right) + \frac{2\ln\{\ln(T)\}}{T}n_{\text{parms}}$$

where n_{parms} is the total number of parameters in the model and L is the value of the log likelihood at the optimum.

Formulas for predict

`xb`, `residuals` and `stdp` are standard and are documented in [R] **predict**. `ce` causes predict to compute $\widehat{E}_t = \widehat{\boldsymbol{\beta}}_f \mathbf{y}_t$ for the requested cointegrating equation.

`levels` causes `predict` to compute the predictions for the levels of the data. Let $\widehat{y}_t^d$ be the predicted value of Δy_t. Because the computations are performed for a given equation, y_t is a scalar. Using $\widehat{y}_t^d$, we can predict the level by $\widehat{y}_t = \widehat{y}_t^d + y_{t-1}$.

Because the residuals from the VECM for the differences and the residuals from the corresponding VAR in levels are identical, there is no need for an option for predicting the residuals in levels.

References

Anderson, T. W. 1951. Estimating linear restrictions on regression coefficients for multivariate normal distributions. *Annals of Mathematical Statistics* 22: 327–351. https://doi.org/10.1214/aoms/1177729580.

Baum, C. F., and S. Hurn. 2021. *Environmental Econometrics Using Stata*. College Station, TX: Stata Press.

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

Boswijk, H. P. 1995. Identifiability of cointegrated systems. Discussion Paper #95-78, Tinbergen Institute. http://hdl.handle.net/11245/1.112017.

Boswijk, H. P., and J. A. Doornik. 2004. Identifying, estimating and testing restricted cointegrating systems: An overview. *Statistica Neerlandica* 58: 440–465. https://doi.org/10.1111/j.1467-9574.2004.00270.x.

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.

Drukker, D. M. 2004. Some further results on estimation and inference in the presence of constraints on alpha in a cointegrating VECM. Working paper, StataCorp.

Du, K. 2017. Econometric convergence test and club clustering using Stata. *Stata Journal* 17: 882–900.

Engle, R. F., and C. W. J. Granger. 1987. Co-integration and error correction: Representation, estimation, and testing. *Econometrica* 55: 251–276. https://doi.org/10.2307/1913236.

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

Johansen, S. 1988. Statistical analysis of cointegration vectors. *Journal of Economic Dynamics and Control* 12: 231–254. https://doi.org/10.1016/0165-1889(88)90041-3.

——. 1991. Estimation and hypothesis testing of cointegration vectors in Gaussian vector autoregressive models. *Econometrica* 59: 1551–1580. https://doi.org/10.2307/2938278.

———. 1995. *Likelihood-Based Inference in Cointegrated Vector Autoregressive Models*. Oxford: Oxford University Press.

Jordan, S., and A. Q. Philips. 2018. Cointegration testing and dynamic simulations of autoregressive distributed lag models. *Stata Journal* 18: 902–923.

Maddala, G. S., and I.-M. Kim. 1998. *Unit Roots, Cointegration, and Structural Change*. Cambridge: Cambridge University Press.

Park, J. Y., and P. C. B. Phillips. 1988. Statistical inference in regressions with integrated processes: Part I. *Econometric Theory* 4: 468–497. https://doi.org/10.1017/S0266466600013402.

———. 1989. Statistical inference in regressions with integrated processes: Part II. *Econometric Theory* 5: 95–131. https://doi.org/10.1017/S0266466600012287.

Phillips, P. C. B. 1986. Understanding spurious regressions in econometrics. *Journal of Econometrics* 33: 311–340. https://doi.org/10.1016/0304-4076(86)90001-1.

Phillips, P. C. B., and S. N. Durlauf. 1986. Multiple time series regressions with integrated processes. *Review of Economic Studies* 53: 473–495. https://doi.org/10.2307/2297602.

Pickup, M. 2015. *Introduction to Time Series Analysis*. Thousand Oaks, CA: SAGE.

Rothenberg, T. J. 1971. Identification in parametric models. *Econometrica* 39: 577–591. https://doi.org/10.2307/1913267.

Sims, C. A., J. H. Stock, and M. W. Watson. 1990. Inference in linear time series models with some unit roots. *Econometrica* 58: 113–144. https://doi.org/10.2307/2938337.

Stock, J. H. 1987. Asymptotic properties of least squares estimators of cointegrating vectors. *Econometrica* 55: 1035–1056. https://doi.org/10.2307/1911260.

Stock, J. H., and M. W. Watson. 1988. Testing for common trends. *Journal of the American Statistical Association* 83: 1097–1107. https://doi.org/10.1080/01621459.1988.10478707.

Watson, M. W. 1994. Vector autoregressions and cointegration. In Vol. 4 of *Handbook of Econometrics*, ed. R. F. Engle and D. L. McFadden. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4412(05)80016-9.

Also see

[TS] **vec postestimation** — Postestimation tools for vec

[TS] **tsset** — Declare data to be time-series data

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

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

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

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

Title

vec postestimation — Postestimation tools for vec

Postestimation commands predict margins Remarks and examples
Also see

Postestimation commands

The following postestimation commands are of special interest after `vec`:

Command	Description
`fcast compute`	obtain dynamic forecasts
`fcast graph`	graph dynamic forecasts obtained from `fcast compute`
`irf`	create and analyze IRFs and FEVDs
`veclmar`	LM test for autocorrelation in residuals
`vecnorm`	test for normally distributed residuals
`vecstable`	check stability condition of estimates

The following standard postestimation commands are also available:

Command	Description
`estat ic`	Akaike's and Schwarz's Bayesian information criteria (AIC and BIC)
`estat summarize`	summary statistics for the estimation sample
`estat vce`	variance–covariance matrix of the estimators (VCE)
`estimates`	cataloging estimation results
`forecast`	dynamic forecasts and simulations
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of coefficients
`lrtest`	likelihood-ratio test
`margins`	marginal means, predictive margins, marginal effects, and average marginal effects
`marginsplot`	graph the results from margins (profile plots, interaction plots, etc.)
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
`predict`	linear predictions and their SEs; residuals
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

predict

Description for predict

`predict` creates a new variable containing predictions such as expected values, residuals, and cointegrating equations.

Menu for predict

Statistics > Postestimation

Syntax for predict

`predict` [*type*] *newvar* [*if*] [*in*] [, *statistic* `equation(`*eqno* | *eqname*`)`]

statistic	Description
Main	
`xb`	fitted value for the specified equation; the default
`stdp`	standard error of the linear prediction
`residuals`	residuals
`ce`	the predicted value of specified cointegrating equation
`levels`	one-step prediction of the level of the endogenous variable
`usece(`*varlist*$_{ce}$`)`	compute the predictions using previously predicted cointegrating equations

These statistics are available both in and out of sample; type `predict ... if e(sample) ...` if wanted only for the estimation sample.

Options for predict

Main

`xb`, the default, calculates the fitted values for the specified equation. The form of the VECM implies that these fitted values are the one-step predictions for the first-differenced variables.

`stdp` calculates the standard error of the linear prediction for the specified equation.

`residuals` calculates the residuals from the specified equation of the VECM.

`ce` calculates the predicted value of the specified cointegrating equation.

`levels` calculates the one-step prediction of the level of the endogenous variable in the requested equation.

`usece(`*varlist*$_{ce}$`)` specifies that previously predicted cointegrating equations saved under the names in *varlist*$_{ce}$ be used to compute the predictions. The number of variables in the *varlist*$_{ce}$ must equal the number of cointegrating equations specified in the model.

`equation(`*eqno* | *eqname*`)` specifies to which equation you are referring.

`equation()` is filled in with one *eqno* or *eqname* for `xb`, `residuals`, `stdp`, `ce`, and `levels` options. `equation(#1)` would mean that the calculation is to be made for the first equation, `equation(#2)` would mean the second, and so on. You could also refer to the equation by its name. `equation(D_income)` would refer to the equation named D_income and `equation(_ce1)`, to the first cointegrating equation, which is named _ce1 by `vec`.

If you do not specify `equation()`, the results are as if you specified `equation(#1)`.

For more information on using `predict` after multiple-equation estimation commands, see [R] **predict**.

margins

Description for margins

margins estimates margins of response for linear predictions.

Menu for margins

Statistics > Postestimation

Syntax for margins

margins [*marginlist*] [, *options*]

margins [*marginlist*] , predict(*statistic* ...) [predict(*statistic* ...) ...] [*options*]

statistic	Description
default	linear predictions for each equation
xb	linear prediction for a specified equation
stdp	not allowed with margins
residuals	not allowed with margins
ce	not allowed with margins
levels	not allowed with margins
usece(*varlist*$_{ce}$)	not allowed with margins

xb defaults to the first equation.

Statistics not allowed with margins are functions of stochastic quantities other than e(b).

For the full syntax, see [R] **margins**.

Remarks and examples

Remarks are presented under the following headings:

Model selection and inference
Forecasting

Model selection and inference

See the following sections for information on model selection and inference after vec.

[TS] **irf** — Create and analyze IRFs, dynamic-multiplier functions, and FEVDs
[TS] **varsoc** — Obtain lag-order selection statistics for VARs and VECMs
[TS] **veclmar** — LM test for residual autocorrelation after vec
[TS] **vecnorm** — Test for normally distributed disturbances after vec
[TS] **vecrank** — Estimate the cointegrating rank of a VECM
[TS] **vecstable** — Check the stability condition of VECM estimates

Forecasting

See the following sections for information on obtaining forecasts after `vec`:

[TS] **fcast compute** — Compute dynamic forecasts after var, svar, or vec
[TS] **fcast graph** — Graph forecasts after fcast compute

Also see

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

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

[U] 20 Estimation and postestimation commands

Title

veclmar — LM test for residual autocorrelation after vec

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
Reference | Also see

Description

`veclmar` implements a Lagrange multiplier (LM) test for autocorrelation in the residuals of vector error-correction models (VECMs).

Quick start

Test of residual autocorrelation for the first two lags of the residuals after `vec`

```
veclmar
```

As above, but test the first 5 lags

```
veclmar, mlag(5)
```

As above, but perform test using stored estimates `myest` from a VECM

```
veclmar, mlag(5) estimates(myest)
```

Menu

Statistics > Multivariate time series > VEC diagnostics and tests > LM test for residual autocorrelation

Syntax

veclmar [, *options*]

options	Description
mlag(#)	use # for the maximum order of autocorrelation; default is mlag(2)
estimates(*estname*)	use previously stored results *estname*; default is to use active results
separator(#)	draw separator line after every # rows

veclmar can be used only after vec; see [TS] **vec**.
You must tsset your data before using veclmar; see [TS] **tsset**.
collect is allowed; see [U] **11.1.10 Prefix commands**.

Options

mlag(#) specifies the maximum order of autocorrelation to be tested. The integer specified in mlag() must be greater than 0; the default is 2.

estimates(*estname*) requests that veclmar use the previously obtained set of vec estimates stored as *estname*. By default, veclmar uses the active results. See [R] **estimates** for information on manipulating estimation results.

separator(#) specifies how many rows should appear in the table between separator lines. By default, separator lines do not appear. For example, separator(1) would draw a line between each row, separator(2) between every other row, and so on.

Remarks and examples

Estimation, inference, and postestimation analysis of VECMs is predicated on the errors' not being autocorrelated. veclmar implements the LM test for autocorrelation in the residuals of a VECM discussed in Johansen (1995, 21–22). The test is performed at lags $j = 1, \ldots,$ mlag(). For each j, the null hypothesis of the test is that there is no autocorrelation at lag j.

▷ Example 1

We fit a VECM using the regional income data described in [TS] **vec** and then call veclmar to test for autocorrelation.

```
. use https://www.stata-press.com/data/r17/rdinc
. vec ln_ne ln_se
  (output omitted)
. veclmar, mlag(4)
  Lagrange-multiplier test
```

lag	chi2	df	Prob > chi2
1	8.9586	4	0.06214
2	4.9809	4	0.28926
3	4.8519	4	0.30284
4	0.3270	4	0.98801

```
H0: no autocorrelation at lag order
```

At the 5% level, we cannot reject the null hypothesis that there is no autocorrelation in the residuals for any of the orders tested. Thus this test finds no evidence of model misspecification.

◁

Stored results

`veclmar` stores the following in `r()`:

Matrices
`r(lm)` χ^2, df, and p-values

Methods and formulas

Consider a VECM without any trend:

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}\boldsymbol{\beta}\mathbf{y}_{t-1} + \sum_{i=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \epsilon_t$$

As discussed in [TS] **vec**, as long as the parameters in the cointegrating vectors, $\boldsymbol{\beta}$, are exactly identified or overidentified, the estimates of these parameters are superconsistent. This implies that the $r \times 1$ vector of estimated cointegrating relations

$$\widehat{\mathbf{E}}_t = \widehat{\boldsymbol{\beta}}\mathbf{y}_t \tag{1}$$

can be used as data with standard estimation and inference methods. When the parameters of the cointegrating equations are not identified, (1) does not provide consistent estimates of $\widehat{\mathbf{E}}_t$; in these cases, `veclmar` exits with an error message.

The VECM above can be rewritten as

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}\widehat{\mathbf{E}}_t + \sum_{i=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \epsilon_t$$

which is just a VAR with $p-1$ lags where the endogenous variables have been first-differenced and is augmented with the exogenous variables $\widehat{\mathbf{E}}$. `veclmar` fits this VAR and then calls `varlmar` to compute the LM test for autocorrelation.

The above discussion assumes no trend and implicitly ignores constraints on the parameters in $\boldsymbol{\alpha}$. As discussed in `vec`, the other four trend specifications considered by Johansen (1995, sec. 5.7) complicate the estimation of the free parameters in $\boldsymbol{\beta}$ but do not alter the basic result that the $\widehat{\mathbf{E}}_t$ can be used as data in the subsequent VAR. Similarly, constraints on the parameters in $\boldsymbol{\alpha}$ imply that the subsequent VAR must be estimated with these constraints applied, but $\widehat{\mathbf{E}}_t$ can still be used as data in the VAR.

See [TS] **varlmar** for more information on the Johansen LM test.

Reference

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

Also see

[TS] **varlmar** — LM test for residual autocorrelation after var or svar

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

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

Title

vecnorm — Test for normally distributed disturbances after vec

Description
Quick start
Menu
Syntax
Options
Remarks and examples
Stored results
Methods and formulas
References
Also see

Description

`vecnorm` computes and reports a series of statistics against the null hypothesis that the disturbances in a VECM are normally distributed.

Quick start

Compute Jarque–Bera, skewness, and kurtosis statistics after `vec` to test the null hypothesis that the residuals are normally distributed

```
vecnorm
```

As above, but only report the Jarque–Bera statistic

```
vecnorm, jbera
```

As above, but only report kurtosis

```
vecnorm, kurtosis
```

Menu

Statistics > Multivariate time series > VEC diagnostics and tests > Test for normally distributed disturbances

Syntax

`vecnorm` [, *options*]

options	Description
jbera	report Jarque–Bera statistic; default is to report all three statistics
skewness	report skewness statistic; default is to report all three statistics
kurtosis	report kurtosis statistic; default is to report all three statistics
estimates(*estname*)	use previously stored results *estname*; default is to use active results
dfk	make small-sample adjustment when computing the estimated variance–covariance matrix of the disturbances
separator(#)	draw separator line after every # rows

`vecnorm` can be used only after `vec`; see [TS] **vec**.

`collect` is allowed; see [U] **11.1.10 Prefix commands**.

Options

`jbera` requests that the Jarque–Bera statistic and any other explicitly requested statistic be reported. By default, the Jarque–Bera, skewness, and kurtosis statistics are reported.

`skewness` requests that the skewness statistic and any other explicitly requested statistic be reported. By default, the Jarque–Bera, skewness, and kurtosis statistics are reported.

`kurtosis` requests that the kurtosis statistic and any other explicitly requested statistic be reported. By default, the Jarque–Bera, skewness, and kurtosis statistics are reported.

`estimates(`*estname*`)` requests that `vecnorm` use the previously obtained set of `vec` estimates stored as *estname*. By default, `vecnorm` uses the active results. See [R] **estimates** for information on manipulating estimation results.

`dfk` requests that a small-sample adjustment be made when computing the estimated variance–covariance matrix of the disturbances.

`separator(`#`)` specifies how many rows should appear in the table between separator lines. By default, separator lines do not appear. For example, `separator(1)` would draw a line between each row, `separator(2)` between every other row, and so on.

Remarks and examples

`vecnorm` computes a series of test statistics of the null hypothesis that the disturbances in a VECM are normally distributed. For each equation and all equations jointly, up to three statistics may be computed: a skewness statistic, a kurtosis statistic, and the Jarque–Bera statistic. By default, all three statistics are reported; if you specify only one statistic, the others are not reported. The Jarque–Bera statistic tests skewness and kurtosis jointly. The single-equation results are against the null hypothesis that the disturbance for that particular equation is normally distributed. The results for all the equations are against the null that all K disturbances have a K-dimensional multivariate normal distribution. Failure to reject the null hypothesis indicates lack of model misspecification.

As noted by Johansen (1995, 141), the log likelihood for the VECM is derived assuming the errors are independent and identically distributed normal, though many of the asymptotic properties can be derived under the weaker assumption that the errors are merely independent and identically distributed.

Many researchers still prefer to test for normality. vecnorm uses the results from vec to produce a series of statistics against the null hypothesis that the K disturbances in the VECM are normally distributed.

▷ Example 1

This example uses vecnorm to test for normality after estimating the parameters of a VECM using the regional income data.

```
. use https://www.stata-press.com/data/r17/rdinc
. vec ln_ne ln_se
  (output omitted )
. vecnorm
  Jarque-Bera test
```

Equation	chi2	df	Prob > chi2
D_ln_ne	0.094	2	0.95417
D_ln_se	0.586	2	0.74608
ALL	0.680	4	0.95381

```
  Skewness test
```

Equation	Skewness	chi2	df	Prob > chi2
D_ln_ne	.05982	0.032	1	0.85890
D_ln_se	.243	0.522	1	0.47016
ALL		0.553	2	0.75835

```
  Kurtosis test
```

Equation	Kurtosis	chi2	df	Prob > chi2
D_ln_ne	3.1679	0.062	1	0.80302
D_ln_se	2.8294	0.064	1	0.79992
ALL		0.126	2	0.93873

The Jarque–Bera results present test statistics for each equation and for all equations jointly against the null hypothesis of normality. For the individual equations, the null hypothesis is that the disturbance term in that equation has a univariate normal distribution. For all equations jointly, the null hypothesis is that the K disturbances come from a K-dimensional normal distribution. In this example, the single-equation and overall Jarque–Bera statistics do not reject the null of normality.

The single-equation skewness test statistics are of the null hypotheses that the disturbance term in each equation has zero skewness, which is the skewness of a normally distributed variable. The row marked ALL shows the results for a test that the disturbances in all equations jointly have zero skewness. The skewness results shown above do not suggest nonnormality.

The kurtosis of a normally distributed variable is three, and the kurtosis statistics presented in the table test the null hypothesis that the disturbance terms have kurtosis consistent with normality. The results in this example do not reject the null hypothesis.

◁

The statistics computed by vecnorm are based on the estimated variance–covariance matrix of the disturbances. vec saves the ML estimate of this matrix, which vecnorm uses by default. Specifying the dfk option instructs vecnorm to make a small-sample adjustment to the variance–covariance matrix before computing the test statistics.

Stored results

`vecnorm` stores the following in `r()`:

Macros

`r(dfk)`	`dfk`, if specified

Matrices

`r(jb)`	Jarque–Bera χ^2, df, and p-values
`r(skewness)`	skewness χ^2, df, and p-values
`r(kurtosis)`	kurtosis χ^2, df, and p-values

Methods and formulas

As discussed in *Methods and formulas* of [TS] **vec**, a cointegrating VECM can be rewritten as a VAR in first differences that includes the predicted cointegrating equations as exogenous variables. `vecnorm` computes the tests discussed in [TS] **varnorm** for the corresponding augmented VAR in first differences. See *Methods and formulas* of [TS] **veclmar** for more information on this approach.

When the parameters of the cointegrating equations are not identified, the consistent estimates of the cointegrating equations are not available, and, in these cases, `vecnorm` exits with an error message.

References

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

Jarque, C. M., and A. K. Bera. 1987. A test for normality of observations and regression residuals. *International Statistical Review* 2: 163–172. https://doi.org/10.2307/1403192.

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

Lütkepohl, H. 2005. *New Introduction to Multiple Time Series Analysis*. New York: Springer.

Also see

[TS] **varnorm** — Test for normally distributed disturbances after var or svar

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

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

Title

vecrank — Estimate the cointegrating rank of a VECM

Description Quick start Menu Syntax
Options Remarks and examples Stored results Methods and formulas
References Also see

Description

`vecrank` produces statistics used to determine the number of cointegrating equations in a vector error-correction model (VECM).

Quick start

Estimate the cointegrating rank for a VECM of `y1`, `y2`, and `y3` using `tsset` data

```
vecrank y1 y2 y3
```

As above, but specify that the underlying VAR model has 6 lags

```
vecrank y1 y2 y3, lags(6)
```

As above, but specify that the model includes a linear trend in the cointegrating equations and a quadratic trend in the undifferenced data

```
vecrank y1 y2 y3, lags(6) trend(trend)
```

As above, and report information criteria

```
vecrank y1 y2 y3, lags(6) trend(trend) ic
```

Menu

Statistics > Multivariate time series > Cointegrating rank of a VECM

Syntax

`vecrank` *depvarlist* [*if*] [*in*] [, *options*]

options	Description
Model	
`lags(#)`	use # for the maximum lag in underlying VAR model
`trend(constant)`	include an unrestricted constant in model; the default
`trend(rconstant)`	include a restricted constant in model
`trend(trend)`	include a linear trend in the cointegrating equations and a quadratic trend in the undifferenced data
`trend(rtrend)`	include a restricted trend in model
`trend(none)`	do not include a trend or a constant
Adv. model	
`sindicators(`$varlist_{si}$`)`	include normalized seasonal indicator variables $varlist_{si}$
`noreduce`	do not perform checks and corrections for collinearity among lags of dependent variables
Reporting	
`notrace`	do not report the trace statistic
`max`	report maximum-eigenvalue statistic
`ic`	report information criteria
`level99`	report 1% critical values instead of 5% critical values
`levela`	report both 1% and 5% critical values

You must `tsset` your data before using `vecrank`; see [TS] **tsset**.

depvar may contain time-series operators; see **[U] 11.4.4 Time-series varlists**.

`by`, `collect`, `rolling`, and `statsby` are allowed; see **[U] 11.1.10 Prefix commands**.

`vecrank` does not allow gaps in the data.

Options

Model

`lags(#)` specifies the number of lags in the VAR representation of the model. The VECM will include one fewer lag of the first differences. The number of lags must be greater than zero but small enough so that the degrees of freedom used by the model are less than the number of observations.

`trend(`*trend_spec*`)` specifies one of five trend specifications to include in the model. See [TS] **vec intro** and [TS] **vec** for descriptions. The default is `trend(constant)`.

Adv. model

`sindicators(`$varlist_{si}$`)` specifies normalized seasonal indicator variables to be included in the model. The indicator variables specified in this option must be normalized as discussed in Johansen (1995, 84). If the indicators are not properly normalized, the likelihood-ratio–based tests for the number of cointegrating equations do not converge to the asymptotic distributions derived by Johansen. For details, see *Methods and formulas* of [TS] **vec**. `sindicators()` cannot be specified with `trend(none)` or `trend(rconstant)` `noreduce` causes `vecrank` to skip the checks and corrections for collinearity among the lags of the dependent variables. By default, `vecrank` checks whether the current lag specification causes some of the regressions performed by `vecrank` to contain perfectly collinear variables and reduces the maximum lag until the perfect collinearity is removed. See *Collinearity* in [TS] **vec** for more information.

Reporting

`notrace` requests that the output for the trace statistic not be displayed. The default is to display the trace statistic.

`max` requests that the output for the maximum-eigenvalue statistic be displayed. The default is to not display this output.

`ic` causes the output for the information criteria to be displayed. The default is to not display this output.

`level99` causes the 1% critical values to be displayed instead of the default 5% critical values.

`levela` causes both the 1% and the 5% critical values to be displayed.

Remarks and examples

Remarks are presented under the following headings:

Introduction
The trace statistic
The maximum-eigenvalue statistic
Minimizing an information criterion

Introduction

Before estimating the parameters of a VECM models, you must choose the number of lags in the underlying VAR, the trend specification, and the number of cointegrating equations. `vecrank` offers several ways of determining the number of cointegrating vectors conditional on a trend specification and lag order.

`vecrank` implements three types of methods for determining r, the number of cointegrating equations in a VECM. The first is Johansen's "trace" statistic method. The second is his "maximum eigenvalue" statistic method. The third method chooses r to minimize an information criterion.

All three methods are based on Johansen's maximum likelihood (ML) estimator of the parameters of a cointegrating VECM. The basic VECM is

$$\Delta \mathbf{y}_t = \boldsymbol{\alpha}\boldsymbol{\beta}'\mathbf{y}_{t-1} + \sum_{t=1}^{p-1} \boldsymbol{\Gamma}_i \Delta \mathbf{y}_{t-i} + \epsilon_t$$

where $\mathbf{y}$ is a $(K \times 1)$ vector of I(1) variables, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are $(K \times r)$ parameter matrices with rank $r < K$, $\boldsymbol{\Gamma}_1, \ldots, \Gamma_{p-1}$ are $(K \times K)$ matrices of parameters, and ϵ_t is a $(K \times 1)$ vector of normally distributed errors that is serially uncorrelated but has contemporaneous covariance matrix $\boldsymbol{\Omega}$.

Building on the work of Anderson (1951), Johansen (1995) derives an ML estimator for the parameters and two likelihood-ratio (LR) tests for inference on r. These LR tests are known as the trace statistic and the maximum-eigenvalue statistic because the log likelihood can be written as the log of the determinant of a matrix plus a simple function of the eigenvalues of another matrix.

Let $\lambda_1, \ldots, \lambda_K$ be the K eigenvalues used in computing the log likelihood at the optimum. Furthermore, assume that these eigenvalues are sorted from the largest λ_1 to the smallest λ_K. If there are $r < K$ cointegrating equations, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ have rank r and the eigenvalues $\lambda_{r+1}, \ldots, \lambda_K$ are zero.

The trace statistic

The null hypothesis of the trace statistic is that there are no more than r cointegrating relations. Restricting the number of cointegrating equations to be r or less implies that the remaining $K - r$ eigenvalues are zero. Johansen (1995, chap. 11 and 12) derives the distribution of the trace statistic

$$-T \sum_{i=r+1}^{K} \ln(1 - \widehat{\lambda}_i)$$

where T is the number of observations and the $\widehat{\lambda}_i$ are the estimated eigenvalues. For any given value of r, large values of the trace statistic are evidence against the null hypothesis that there are r or fewer cointegrating relations in the VECM.

One of the problems in determining the number of cointegrating equations is that the process involves more than one statistical test. Johansen (1995, chap. 6, 11, and 12) derives a method based on the trace statistic that has nominal coverage despite evaluating multiple tests. This method can be interpreted as being an estimator $\widehat{r}$ of the true number of cointegrating equations r_0. The method starts testing at $r = 0$ and accepts as $\widehat{r}$ the first value of r for which the trace statistic fails to reject the null.

▷ Example 1

We have quarterly data on the natural logs of aggregate consumption, investment, and GDP in the United States from the first quarter of 1959 through the fourth quarter of 1982. As discussed in King et al. (1991), the balanced-growth hypothesis in economics implies that we would expect to find two cointegrating equations among these three variables. In the output below, we use `vecrank` to determine the number of cointegrating equations using Johansen's multiple-trace test method.

```
. use https://www.stata-press.com/data/r17/balance2
(macro data for VECM/balance study)
. vecrank y i c, lags(5)
Johansen tests for cointegration
Trend: Constant                                    Number of obs  = 91
Sample: 1960q2 thru 1982q4                         Number of lags =  5
───────────────────────────────────────────────────────────────────────────
                                                                 Critical
Maximum                                                 Trace       value
   rank  Params          LL  Eigenvalue    statistic          5%
      0      39   1231.1041           .      46.1492       29.68
      1      44   1245.3882     0.26943      17.5810       15.41
      2      47   1252.5055     0.14480       3.3465*       3.76
      3      48   1254.1787     0.03611
───────────────────────────────────────────────────────────────────────────
* selected rank
```

The header produces information about the sample, the trend specification, and the number of lags included in the model. The main table contains a separate row for each possible value of r, the number of cointegrating equations. When $r = 3$, all three variables in this model are stationary.

In this example, because the trace statistic at $r = 0$ of 46.1492 exceeds its critical value of 29.68, we reject the null hypothesis of no cointegrating equations. Similarly, because the trace statistic at $r = 1$ of 17.581 exceeds its critical value of 15.41, we reject the null hypothesis that there is one or fewer cointegrating equation. In contrast, because the trace statistic at $r = 2$ of 3.3465 is less than its critical value of 3.76, we cannot reject the null hypothesis that there are two or fewer cointegrating equations. Because Johansen's method for estimating r is to accept as $\widehat{r}$ the first r for which the null hypothesis is not rejected, we accept $r = 2$ as our estimate of the number of cointegrating equations between these three variables. The "*" by the trace statistic at $r = 2$ indicates that this is the value of r selected by Johansen's multiple-trace test procedure. The eigenvalue shown in the last line of output computes the trace statistic in the preceding line.

◁

▷ Example 2

In the previous example, we used the default 5% critical values. We can estimate r with 1% critical values instead by specifying the `level99` option.

```
. vecrank y i c, lags(5) level99
Johansen tests for cointegration
Trend: Constant                                    Number of obs  = 91
Sample: 1960q2 thru 1982q4                         Number of lags =  5
───────────────────────────────────────────────────────────────────────────
                                                                 Critical
Maximum                                                 Trace       value
   rank  Params          LL  Eigenvalue    statistic          1%
      0      39   1231.1041           .      46.1492       35.65
      1      44   1245.3882     0.26943      17.5810*      20.04
      2      47   1252.5055     0.14480       3.3465        6.65
      3      48   1254.1787     0.03611
───────────────────────────────────────────────────────────────────────────
* selected rank
```

The output indicates that switching from the 5% to the 1% level changes the resulting estimate from $r = 2$ to $r = 1$.

◁

The maximum-eigenvalue statistic

The alternative hypothesis of the trace statistic is that the number of cointegrating equations is strictly larger than the number r assumed under the null hypothesis. Instead, we could assume a given r under the null hypothesis and test this against the alternative that there are $r+1$ cointegrating equations. Johansen (1995, chap. 6, 11, and 12) derives an LR test of the null of r cointegrating relations against the alternative of $r+1$ cointegrating relations. Because the part of the log likelihood that changes with r is a simple function of the eigenvalues of a $(K \times K)$ matrix, this test is known as the maximum-eigenvalue statistic. This method is used less often than the trace statistic method because no solution to the multiple-testing problem has yet been found.

▷ Example 3

In the output below, we reexamine the balanced-growth hypothesis. We use the `levela` option to obtain both the 5% and 1% critical values, and we use the `notrace` option to suppress the table of trace statistics.

```
. vecrank y i c, lags(5) max levela notrace

Johansen tests for cointegration
Trend: Constant                                          Number of obs  = 91
Sample: 1960q2 thru 1982q4                               Number of lags =  5
─────────────────────────────────────────────────────────────────────────────
Maximum                              ───────Eigenvalue──────     ───Critical value───
   rank  Params          LL                          Maximum          5%           1%
      0      39   1231.1041                          28.5682       20.97        25.52
      1      44   1245.3882       0.26943            14.2346       14.07        18.63
      2      47   1252.5055       0.14480             3.3465        3.76         6.65
      3      48   1254.1787       0.03611
─────────────────────────────────────────────────────────────────────────────
```

We can reject $r = 1$ in favor of $r = 2$ at the 5% level but not at the 1% level. As with the trace statistic method, whether we choose to specify one or two cointegrating equations in our VECM will depend on the significance level we use here.

◁

Minimizing an information criterion

Many multiple-testing problems in the time-series literature have been solved by defining an estimator that minimizes an information criterion with known asymptotic properties. Selecting the lag length in an autoregressive model is probably the best-known example. Gonzalo and Pitarakis (1998) and Aznar and Salvador (2002) have shown that this approach can be applied to determining the number of cointegrating equations in a VECM. As in the lag-length selection problem, choosing the number of cointegrating equations that minimizes either the Schwarz Bayesian information criterion (SBIC) or the Hannan and Quinn information criterion (HQIC) provides a consistent estimator of the number of cointegrating equations.

▷ Example 4

We use these information-criteria methods to estimate the number of cointegrating equations in our balanced-growth data.

```
. vecrank y i c, lags(5) ic notrace
Johansen tests for cointegration
Trend: Constant                                    Number of obs  = 91
Sample: 1960q2 thru 1982q4                         Number of lags =  5

Maximum
   rank  Params         LL   Eigenvalue       SBIC        HQIC         AIC
      0      39  1231.1041              -25.12401   -25.76596   -26.20009
      1      44  1245.3882     0.26943  -25.19009   -25.91435   -26.40414
      2      47  1252.5055     0.14480  -25.19781*  -25.97144*  -26.49463
      3      48  1254.1787     0.03611  -25.18501   -25.97511   -26.50942

* selected rank
```

Both the SBIC and the HQIC estimators suggest that there are two cointegrating equations in the balanced-growth data.

◁

Stored results

`vecrank` stores the following in `e()`:

Scalars

`e(N)`	number of observations
`e(k_eq)`	number of equations in `e(b)`
`e(k_dv)`	number of dependent variables
`e(tmin)`	minimum time
`e(tmax)`	maximum time
`e(n_lags)`	number of lags
`e(k_ce95)`	number of cointegrating equations chosen by multiple trace tests with `level(95)`
`e(k_ce99)`	number of cointegrating equations chosen by multiple trace tests with `level(99)`
`e(k_cesbic)`	number of cointegrating equations chosen by minimizing SBIC
`e(k_cehqic)`	number of cointegrating equations chosen by minimizing HQIC

Macros

`e(cmd)`	`vecrank`
`e(cmdline)`	command as typed
`e(trend)`	trend specified
`e(reduced_lags)`	list of maximum lags to which the model has been reduced
`e(reduce_opt)`	`noreduce`, if `noreduce` is specified
`e(tsfmt)`	format for current time variable

Matrices

`e(max)`	vector of maximum-eigenvalue statistics
`e(trace)`	vector of trace statistics
`e(ll)`	vector of model log likelihoods
`e(lambda)`	vector of eigenvalues
`e(k_rank)`	vector of numbers of unconstrained parameters
`e(hqic)`	vector of HQIC values
`e(sbic)`	vector of SBIC values
`e(aic)`	vector of AIC values

Methods and formulas

As shown in *Methods and formulas* of [TS] **vec**, given a lag, trend, and seasonal specification when there are $0 \leq r \leq K$ cointegrating equations, the log likelihood with the Johansen identification restrictions can be written as

$$L = -\frac{1}{2}T\left[K\left\{\ln\left(2\pi\right)+1\right\}+\ln\left(|S_{00}|\right)+\sum_{i=1}^{r}\ln\left(1-\widehat{\lambda}_i\right)\right] \tag{1}$$

where the $(K \times K)$ matrix S_{00} and the eigenvalues $\widehat{\lambda}_i$ are defined in *Methods and formulas* of [TS] **vec**.

The trace statistic compares the null hypothesis that there are r or fewer cointegrating relations with the alternative hypothesis that there are more than r cointegrating equations. Under the alternative hypothesis, the log likelihood is

$$L_A = -\frac{1}{2}T\left[K\left\{\ln\left(2\pi\right)+1\right\}+\ln\left(|S_{00}|\right)+\sum_{i=1}^{K}\ln\left(1-\widehat{\lambda}_i\right)\right] \tag{2}$$

Thus the LR test that compares the unrestricted model in (2) with the restricted model in (1) is given by

$$LR_{\text{trace}} = -T\sum_{i=r+1}^{K}\ln\left(1-\widehat{\lambda}_i\right)$$

As discussed by Johansen (1995), the trace statistic has a nonstandard distribution under the null hypothesis because the null hypothesis places restrictions on the coefficients on $\mathbf{y}_{t-1}$, which is assumed to have $K-r$ random-walk components. `vecrank` reports the Osterwald-Lenum (1992) critical values.

The maximum-eigenvalue statistic compares the null model containing r cointegrating relations with the alternative model that has $r+1$ cointegrating relations. Thus using these two values for r in (1) and a few lines of algebra implies that the LR test of this hypothesis is

$$LR_{\max} = -T\ln\left(1-\widehat{\lambda}_{r+1}\right)$$

As for the trace statistic, because this test involves restrictions on the coefficients on a vector of I(1) variables, the test statistic's distribution will be nonstandard. `vecrank` reports the Osterwald-Lenum (1992) critical values.

The formulas for the AIC, SBIC, and HQIC are given in *Methods and formulas* of [TS] **vec**.

Søren Johansen (1939–) earned degrees in mathematical statistics at the University of Copenhagen, where he is now based. In addition to making contributions to mathematical statistics, probability theory, and medical statistics, he has worked mostly in econometrics—in particular, on the theory of cointegration.

References

Anderson, T. W. 1951. Estimating linear restrictions on regression coefficients for multivariate normal distributions. *Annals of Mathematical Statistics* 22: 327–351. https://doi.org/10.1214/aoms/1177729580.

Aznar, A., and M. Salvador. 2002. Selecting the rank of the cointegration space and the form of the intercept using an information criterion. *Econometric Theory* 18: 926–947. https://doi.org/10.1017/S0266466602184064.

Du, K. 2017. Econometric convergence test and club clustering using Stata. *Stata Journal* 17: 882–900.

Engle, R. F., and C. W. J. Granger. 1987. Co-integration and error correction: Representation, estimation, and testing. *Econometrica* 55: 251–276. https://doi.org/10.2307/1913236.

Gonzalo, J., and J.-Y. Pitarakis. 1998. Specification via model selection in vector error correction models. *Economics Letters* 60: 321–328. https://doi.org/10.1016/S0165-1765(98)00129-3.

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

Hubrich, K., H. Lütkepohl, and P. Saikkonen. 2001. A review of systems cointegration tests. *Econometric Reviews* 20: 247–318. https://doi.org/10.1081/ETC-100104936.

Johansen, S. 1988. Statistical analysis of cointegration vectors. *Journal of Economic Dynamics and Control* 12: 231–254. https://doi.org/10.1016/0165-1889(88)90041-3.

———. 1991. Estimation and hypothesis testing of cointegration vectors in Gaussian vector autoregressive models. *Econometrica* 59: 1551–1580. https://doi.org/10.2307/2938278.

———. 1995. *Likelihood-Based Inference in Cointegrated Vector Autoregressive Models*. Oxford: Oxford University Press.

Jordan, S., and A. Q. Philips. 2018. Cointegration testing and dynamic simulations of autoregressive distributed lag models. *Stata Journal* 18: 902–923.

King, R. G., C. I. Plosser, J. H. Stock, and M. W. Watson. 1991. Stochastic trends and economic fluctuations. *American Economic Review* 81: 819–840.

Lütkepohl, H. 2005. *New Introduction to Multiple Time Series Analysis*. New York: Springer.

Maddala, G. S., and I.-M. Kim. 1998. *Unit Roots, Cointegration, and Structural Change*. Cambridge: Cambridge University Press.

Osterwald-Lenum, M. G. 1992. A note with quantiles of the asymptotic distribution of the maximum likelihood cointegration rank test statistics. *Oxford Bulletin of Economics and Statistics* 54: 461–472. https://doi.org/10.1111/j.1468-0084.1992.tb00013.x.

Park, J. Y., and P. C. B. Phillips. 1988. Statistical inference in regressions with integrated processes: Part I. *Econometric Theory* 4: 468–497. https://doi.org/10.1017/S0266466600013402.

———. 1989. Statistical inference in regressions with integrated processes: Part II. *Econometric Theory* 5: 95–131. https://doi.org/10.1017/S0266466600012287.

Phillips, P. C. B. 1986. Understanding spurious regressions in econometrics. *Journal of Econometrics* 33: 311–340. https://doi.org/10.1016/0304-4076(86)90001-1.

Phillips, P. C. B., and S. N. Durlauf. 1986. Multiple time series regressions with integrated processes. *Review of Economic Studies* 53: 473–495. https://doi.org/10.2307/2297602.

Sims, C. A., J. H. Stock, and M. W. Watson. 1990. Inference in linear time series models with some unit roots. *Econometrica* 58: 113–144. https://doi.org/10.2307/2938337.

Stock, J. H. 1987. Asymptotic properties of least squares estimators of cointegrating vectors. *Econometrica* 55: 1035–1056. https://doi.org/10.2307/1911260.

Stock, J. H., and M. W. Watson. 1988. Testing for common trends. *Journal of the American Statistical Association* 83: 1097–1107. https://doi.org/10.1080/01621459.1988.10478707.

Watson, M. W. 1994. Vector autoregressions and cointegration. In Vol. 4 of *Handbook of Econometrics*, ed. R. F. Engle and D. L. McFadden. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4412(05)80016-9.

Also see

[TS] **tsset** — Declare data to be time-series data

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

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

Title

vecstable — Check the stability condition of VECM estimates

Description Quick start Menu Syntax
Options Remarks and examples Stored results Methods and formulas
References Also see

Description

vecstable checks the eigenvalue stability condition in a vector error-correction model (VECM) fit using vec.

Quick start

Check eigenvalue stability condition after vec

```
vecstable
```

As above, and graph the eigenvalues of the companion matrix

```
vecstable, graph
```

As above, and label each eigenvalue with its distance from the unit circle

```
vecstable, graph dlabel
```

As above, but label the eigenvalues with their moduli

```
vecstable, graph modlabel
```

Menu

Statistics > Multivariate time series > VEC diagnostics and tests > Check stability condition of VEC estimates

Syntax

`vecstable` [, *options*]

options	Description
Main	
estimates(*estname*)	use previously stored results *estname*; default is to use active results
amat(*matrix_name*)	save the companion matrix as *matrix_name*
graph	graph eigenvalues of the companion matrix
dlabel	label eigenvalues with the distance from the unit circle
modlabel	label eigenvalues with the modulus
marker_options	change look of markers (color, size, etc.)
rlopts(*cline_options*)	affect rendition of reference unit circle
nogrid	suppress polar grid circles
pgrid([...])	specify radii and appearance of polar grid circles; see *Options* for details
Add plots	
addplot(*plot*)	add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than by() documented in [G-3] ***twoway_options***

vecstable can be used only after vec; see [TS] **vec**.

collect is allowed; see [U] **11.1.10 Prefix commands**.

Options

Main

estimates(*estname*) requests that vecstable use the previously obtained set of vec estimates stored as *estname*. By default, vecstable uses the active results. See [R] **estimates** for information on manipulating estimation results.

amat(*matrix_name*) specifies a valid Stata matrix name by which the companion matrix can be saved. The companion matrix is referred to as the **A** matrix in Lütkepohl (2005) and [TS] **varstable**. The default is not to save the companion matrix.

graph causes vecstable to draw a graph of the eigenvalues of the companion matrix.

dlabel labels the eigenvalues with their distances from the unit circle. dlabel cannot be specified with modlabel.

modlabel labels the eigenvalues with their moduli. modlabel cannot be specified with dlabel.

marker_options specify the look of markers. This look includes the marker symbol, the marker size, and its color and outline; see [G-3] ***marker_options***.

rlopts(*cline_options*) affects the rendition of the reference unit circle; see [G-3] ***cline_options***.

nogrid suppresses the polar grid circles.

pgrid([*numlist*] [, *line_options*]) [pgrid([*numlist*] [, *line_options*]) ...

pgrid([*numlist*] [, *line_options*])] determines the radii and appearance of the polar grid circles. By default, the graph includes nine polar grid circles with radii 0.1, 0.2, ..., 0.9 that have the grid linestyle. The *numlist* specifies the radii for the polar grid circles. The *line_options* determine the appearance of the polar grid circles; see [G-3] ***line_options***. Because the pgrid() option can be repeated, circles with different radii can have distinct appearances.

Add plots

addplot(*plot*) adds specified plots to the generated graph; see [G-3] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding by(). These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

Remarks and examples

Inference after vec requires that the cointegrating equations be stationary and that the number of cointegrating equations be correctly specified. Although the methods implemented in vecrank identify the number of stationary cointegrating equations, they assume that the individual variables are I(1). vecstable provides indicators of whether the number of cointegrating equations is misspecified or whether the cointegrating equations, which are assumed to be stationary, are not stationary.

vecstable is analogous to varstable. vecstable uses the coefficient estimates from the previously fitted VECM to back out estimates of the coefficients of the corresponding VAR and then compute the eigenvalues of the companion matrix. See [TS] **varstable** for details about how the companion matrix is formed and about how to interpret the resulting eigenvalues for covariance-stationary VAR models.

If a VECM has K endogenous variables and r cointegrating vectors, there will be $K - r$ unit moduli in the companion matrix. If any of the remaining moduli computed by vecrank are too close to one, either the cointegrating equations are not stationary or there is another common trend and the rank() specified in the vec command is too high. Unfortunately, there is no general distribution theory that allows you to determine whether an estimated root is too close to one for all the cases that commonly arise in practice.

▷ Example 1

In example 1 of [TS] **vec**, we estimated the parameters of a bivariate VECM of the natural logs of the average disposable incomes in two of the economic regions created by the U.S. Bureau of Economic Analysis. In that example, we concluded that the predicted cointegrating equation was probably not stationary. Here we continue that example by refitting that model and using vecstable to analyze the eigenvalues of the companion matrix of the corresponding VAR.

```
. use https://www.stata-press.com/data/r17/rdinc
. vec ln_ne ln_se
  (output omitted)
. vecstable
  Eigenvalue stability condition
```

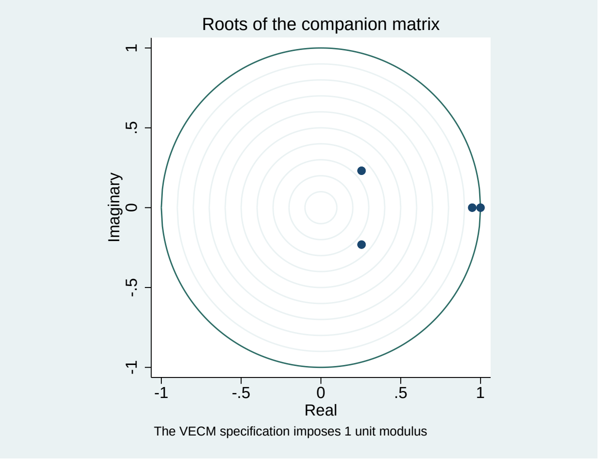
Eigenvalue	Modulus
1	1
.9477854	.947785
.2545357 + .2312756*i*	.343914
.2545357 - .2312756*i*	.343914

```
  The VECM specification imposes a unit modulus.
```

The output contains a table showing the eigenvalues of the companion matrix and their associated moduli. The table shows that one of the roots is 1. The table footer reminds us that the specified VECM imposes one unit modulus on the companion matrix.

The output indicates that there is a real root at about 0.95. Although there is no distribution theory to measure how close this root is to one, per other discussions in the literature (for example, Johansen [1995, 137–138]), we conclude that the root of 0.95 supports our earlier analysis, in which we concluded that the predicted cointegrating equation is probably not stationary.

If we had included the `graph` option with `vecstable`, the following graph would have been displayed:



The graph plots the eigenvalues of the companion matrix with the real component on the x axis and the imaginary component on the y axis. Although the information is the same as in the table, the graph shows visually how close the root with modulus 0.95 is to the unit circle.

◁

Stored results

`vecstable` stores the following in `r()`:

Scalars

`r(unitmod)`	number of unit moduli imposed on the companion matrix

Matrices

`r(Re)`	real part of the eigenvalues of **A**
`r(Im)`	imaginary part of the eigenvalues of **A**
`r(Modulus)`	moduli of the eigenvalues of **A**

where **A** is the companion matrix of the VAR that corresponds to the VECM.

Methods and formulas

`vecstable` uses the formulas given in *Methods and formulas* of [TS] **irf create** to obtain estimates of the parameters in the corresponding VAR from the `vec` estimates. With these estimates, the calculations are identical to those discussed in [TS] **varstable**. In particular, the derivation of the companion matrix, **A**, from the VAR point estimates is given in [TS] **varstable**.

References

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

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

Lütkepohl, H. 2005. *New Introduction to Multiple Time Series Analysis*. New York: Springer.

Also see

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

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