



Queen's Economics Department Working Paper No. 1273

# FCVARmodel.m: A Matlab software package for estimation and testing in the fractionally cointegrated VAR model

Morten Åyrregaard Nielsen  
Queen's University and CREATES

Lealand Morin  
Queen's University

Department of Economics  
Queen's University  
94 University Avenue  
Kingston, Ontario, Canada  
K7L 3N6

3-2014

**FCVARmodel.m:**  
A Matlab software package for estimation and testing in  
the fractionally cointegrated VAR model\*  
version 1.7

Morten Ørregaard Nielsen<sup>†</sup>  
Queen's University and CREATES  
Email: mon@econ.queensu.ca

Lealand Morin  
Queen's University  
Email: morinl@econ.queensu.ca

March 10, 2014

**Abstract**

This manual describes the usage of the accompanying freely available software package for estimation and testing in the fractionally cointegrated vector autoregressive (VAR) model.

**JEL Codes:** C22, C32.

**Keywords:** cofractional process, cointegration rank, computer program, fractional autoregressive model, fractional cointegration, fractional unit root, Matlab, VAR model.

---

\*We are grateful to Federico Carlini, Andreas Noack Jensen, Søren Johansen, James MacKinnon, Daniela Osterrieder, and Michal Popiel for comments, and to the Canada Research Chairs program, the Social Sciences and Humanities Research Council of Canada (SSHRC), and the Center for Research in Econometric Analysis of Time Series (CREATES, funded by the Danish National Research Foundation) for financial support.

<sup>†</sup>Corresponding author. If you find any bugs or other problems, please let us know.

# Contents

<b>1</b>	<b>Obtaining and using the software package</b>	<b>3</b>
1.1	Disclaimer . . . . .	3
1.2	Obtaining the software package . . . . .	3
1.3	Citation . . . . .	3
1.4	Using the software package . . . . .	3
1.5	Version history . . . . .	3
<b>2</b>	<b>The fractionally cointegrated VAR model</b>	<b>5</b>
2.1	Maximum likelihood estimation . . . . .	5
2.2	Cointegration rank tests . . . . .	6
2.3	Restricted models . . . . .	7
<b>3</b>	<b>Main program: FCVARmodel.m</b>	<b>8</b>
<b>4</b>	<b>Output: the Danish data</b>	<b>9</b>
<b>5</b>	<b>Options files</b>	<b>14</b>
5.1	Default estimation options: DefaultEstnOptions.m . . . . .	14
5.2	Restricted estimation options: RestrictEstnOptions.m . . . . .	16
<b>6</b>	<b>Other main files</b>	<b>18</b>
6.1	Rank tests: RankTests.m . . . . .	18
6.2	Model estimation: FCVARestn.m . . . . .	19
6.3	Obtain residuals: GetResiduals.m . . . . .	20
<b>7</b>	<b>Auxilliary files</b>	<b>21</b>
7.1	AdjustEstnOptions.m . . . . .	21
7.2	CharPolyRoots.m . . . . .	23
7.3	FCVARhess.m . . . . .	24
7.4	FCVARlike.m . . . . .	25
7.5	FracDiff.m . . . . .	26
7.6	FreeParams.m . . . . .	27
7.7	FullFCVARlike.m . . . . .	28
7.8	GetParams.m . . . . .	29
7.9	GetParamsSwitching.m . . . . .	30
7.10	Lbk.m . . . . .	31
7.11	SEmat2vec.m . . . . .	32
7.12	SEvec2mat.m . . . . .	33
7.13	TransformData.m . . . . .	34
<b>8</b>	<b>List of references</b>	<b>35</b>

# 1 Obtaining and using the software package

## 1.1 Disclaimer

We have done our best to make this program as functional and free from errors as possible, but no warranty is given whatsoever. We cannot guarantee that we have been 100% successful in eliminating bugs, so if you find any please let us know.

## 1.2 Obtaining the software package

The software package can be downloaded from the first author's website at Queen's University:

<http://www.econ.queensu.ca/faculty/mon/software/>

It is freely available for non-commercial, academic use.

## 1.3 Citation

If you use this software, or any program or computer codes based on this software, we ask that you please cite this document. For example, you could add "The results were obtained using the computer software by Nielsen and Morin (2013)" in the main text of your paper and then add the citation

Nielsen, M. Ø. and L. Morin (2014). "FCVARmodel.m: A Matlab software package for estimation and testing in the fractionally cointegrated VAR model, v1.7." QED working paper 1273, Queen's University.

to your list of references.

## 1.4 Using the software package

The use of this software package requires a functioning installation of Matlab. Any recent version should work. Unzip the contents of the zip file into any directory, maintaining the directory structure in the zip file, will place all the main files into that directory and create a subdirectory with the auxiliary files.

The next section describes the fractionally cointegrated VAR model and the restricted models that can be estimated with this software package. Section 3 describes the functioning of the main program, and section 4 shows an example of the output. Section 5 describes the options file and the various settings. With these programs, both the unrestricted and restricted models can be estimated. It is then straightforward to calculate likelihood ratio test statistics and perform the tests. The cointegration rank tests can be calculated automatically using the program in section 6.1 and the model parameters can be estimated using the program in section 6.2. Section 6.3 describes how to obtain residuals for model diagnostics and section 7 describes auxiliary files.

## 1.5 Version history

v1.7 (March 10, 2014): Fixed an issue with estimation when restrictions are imposed on the  $\Gamma$  matrices.

v1.6 (January 2, 2014): Fixed the calculation and plot of the characteristic roots. Also moved the selection of  $N$  (number of initial values to condition upon in the estimation) and the deterministic term to the main program.

v1.5 (June 5, 2013): Corrected the calculation of AIC and BIC.

v1.4 (May 31, 2013): Fixed a problem with the calculation of the roots of the characteristic polynomial when  $r = 0$ .

v1.3 (May 24, 2013): Updated to use the fast fractional difference algorithm of Jensen and Nielsen (2013). The computations are MUCH faster for large sample sizes.

v1.2 (February 1, 2012): Updated the manual related to the restricted constant term.

v1.1 (September 30, 2011): Changed the sign on  $\rho$ .

v1.0 (August 26, 2011): First publicly available version.

## 2 The fractionally cointegrated VAR model

The fractionally cointegrated vector autoregressive (FC-VAR) model was proposed in Johansen (2008) and analyzed by Johansen and Nielsen (2010, 2012), henceforth JN. For a time series  $X_t$  of dimension  $p$ , the fractionally cointegrated VAR model is given in error correction form as

$$\Delta^d X_t = \Delta^{d-b} L_b \alpha \beta' X_t + \sum_{i=1}^k \Gamma_i \Delta^d L_b^i X_t + \varepsilon_t, \quad t = 1, \dots, T, \quad (1)$$

where  $\varepsilon_t$  is  $p$ -dimensional *i.i.d.*( $0, \Omega$ ),  $d \geq b > 0$ ,  $\Delta^d$  is the fractional difference operator, and  $L_b = 1 - \Delta^b$  is the fractional lag operator.<sup>1</sup> For the model with  $d = b$  it is also possible to include a constant term to obtain

$$\Delta^d X_t = L_d \alpha (\beta' X_t + \rho') + \sum_{i=1}^k \Gamma_i \Delta^d L_d^i X_t + \varepsilon_t, \quad t = 1, \dots, T, \quad (2)$$

The models (1) and (2) include the Johansen (1996) cointegrated VAR model as the special case  $d = b = 1$ , and the interpretation of the model parameters is similar. Thus, the observed data  $X_t$  are integrated of order  $d$ , and  $b$  is the strength of the cointegrating relations in the sense that higher  $b$  implies less persistence in the cointegrating relations. Writing  $\Pi = \alpha \beta'$ , where the  $p \times r$  matrices  $\alpha$  and  $\beta$  with  $r \leq p$  are assumed to have full column rank  $r$ , the columns of  $\beta$  are the  $r$  cointegrating (cofractional) relations that determine the long-run equilibria, and  $\alpha$  holds the coefficients that determine speed of adjustment towards the equilibria. The parameters  $\Gamma = (\Gamma_1, \dots, \Gamma_k)$  govern the short-run dynamics. The parameter  $\rho$  is the so-called restricted constant term (since the constant term in the model is restricted to be of the form  $\mu = \alpha \rho'$ ), which is interpreted as giving the mean level of the long-run equilibria, i.e.  $E \beta' X_t + \rho' = 0$ . The rank  $r$  is termed the cointegrating, or cofractional, rank. This model thus has the same main structure as in the standard C-VAR in that it allows for modeling of both cointegration and adjustment towards equilibrium, but is more general since it accommodates fractional integration and cointegration.

In the next three subsections we briefly describe estimation and testing within the model (1). For details we refer to JN (2012).

### 2.1 Maximum likelihood estimation

The model (1) is estimated by conditional maximum likelihood (given initial values) by maximizing the function

$$\log L_T(\lambda) = -\frac{T}{2} \log \det(T^{-1} \sum_{t=1}^T \varepsilon_t(\lambda) \varepsilon_t(\lambda)'), \quad (3)$$

where

$$\varepsilon_t(\lambda) = \Delta^d X_t - \Delta^{d-b} L_b \alpha \beta' X_t - \sum_{i=1}^k \Gamma_i \Delta^d L_b^i X_t, \quad \lambda = (d, b, \alpha, \beta, \Gamma), \quad (4)$$

---

<sup>1</sup>Both are defined in terms of their binomial expansion in the lag operator  $L$ . Note that the expansion of  $L_b$  has no term in  $L^0$  and thus only lagged disequilibrium errors appear in (1).

for model (1) and

$$\varepsilon_t(\lambda) = \Delta^d X_t - L_d \alpha (\beta' X_t + \rho') - \sum_{i=1}^k \Gamma_i \Delta^d L_d^i X_t, \quad \lambda = (d, \alpha, \beta, \rho, \Gamma), \quad (5)$$

for model (2). It is shown in JN (2012) how, for fixed  $(d, b)$  the estimation reduces to regression and reduced rank regression as in Johansen (1996). In this way the parameters  $(\alpha, \beta, \rho, \Gamma)$  can be concentrated out of the likelihood function, and numerical optimization is only needed to optimize the profile likelihood function over the two fractional parameters,  $d$  and  $b$ .

JN (2012) shows that asymptotic theory is standard when  $b < 0.5$ , and for the case  $b > 0.5$  asymptotic theory is non-standard and involves fractional Brownian motion of type II. Specifically, when  $b > 0.5$ , JN (2012) shows that under i.i.d. errors with suitable moment conditions, the conditional maximum likelihood parameter estimates  $(\hat{d}, \hat{b}, \hat{\alpha}, \hat{\Gamma}_1, \dots, \hat{\Gamma}_k)$  are asymptotically Gaussian, while  $(\hat{\beta}, \hat{\rho})$  are locally asymptotically mixed normal. These results allow asymptotically standard (chi-squared) inference on all parameters of the model, including the cointegrating relations and orders of fractionality, using quasi-likelihood ratio tests.

## 2.2 Cointegration rank tests

The likelihood ratio (LR) test statistic of the hypothesis  $\mathcal{H}_r : \text{rank}(\Pi) = r$  against  $\mathcal{H}_p : \text{rank}(\Pi) = p$  is of particular interest because it deals with an important empirical question. Let  $L(d, b, r)$  be the profile likelihood function given rank  $r$ , where  $(\alpha, \beta, \Gamma)$  have been concentrated out by regression and reduced rank regression; see JN (2012) for details. In the case of model (2) the test is  $\mathcal{H}_r : \text{rank}(\Pi, \mu) = r$  against  $\mathcal{H}_p : \text{rank}(\Pi, \mu) = p$  and  $L(d, r)$  denotes the profile likelihood function for rank  $r$ , where  $(\alpha, \beta, \rho, \Gamma)$  have been concentrated out.

The profile likelihood function is maximized both under the hypothesis  $\mathcal{H}_r$  and under  $\mathcal{H}_p$  and the LR test statistic is then  $\text{LR}_T(q) = 2 \log(L(\hat{d}_p, \hat{b}_p, p) / L(\hat{d}_r, \hat{b}_r, r))$ , where  $q = p - r$ ,

$$L(\hat{d}_p, \hat{b}_p, p) = \max_{d, b} L(d, b, p), \quad L(\hat{d}_r, \hat{b}_r, r) = \max_{d, b} L(d, b, r),$$

and with obvious modification for model (2). This problem is qualitatively different from that in Johansen (1996) since the asymptotic distribution of  $\text{LR}_T(q)$  depends qualitatively (and quantitatively) on the parameter  $b$ . In the case with  $0 < b < 1/2$  (sometimes known as “weak cointegration”),  $\text{LR}_T(q)$  has a standard asymptotic distribution, namely  $\text{LR}_T(q) \xrightarrow{D} \chi^2(q^2)$ . On the other hand, when  $1/2 < b \leq d$  (“strong cointegration”), asymptotic theory is nonstandard and

$$\text{LR}_T(q) \xrightarrow{D} \text{Tr} \left\{ \int_0^1 dW(s) F'(s) \left( \int_0^1 F(s) F'(s) ds \right)^{-1} \int_0^1 F(s) dW'(s) \right\}, \quad (6)$$

where the vector process  $dW$  is the increment of ordinary vector Brownian motion of dimension  $q = p - r$ . The vector process  $F$  depends on the deterministics similar to the cointegrated VAR model in Johansen (1996). When no deterministic term is in the model,  $F(u) = W_b(u)$ ,

and when the restricted constant term is included in model (2),  $F(u) = (W'_d(u), 1)'$ , where  $W_b(u) = \Gamma(b)^{-1} \int_0^u (u-s)^{b-1} dW(s)$  is vector fractional Brownian motion of type II.

The asymptotic distribution (6) of the test statistic  $LR_T(q)$  depends on both  $b$  and  $q = p - r$ . The dependence on the unknown parameter  $b$  complicates empirical analysis compared to the C-VAR model. However, computer programs for computing asymptotic critical values and asymptotic  $P$  values for the LR rank tests in the fractional models (1) and (2), based on numerical distribution functions, are made available by MacKinnon and Nielsen (2012).

### 2.3 Restricted models

Note that a reduced rank restriction has already been imposed on models (1) and (2), where the coefficient matrix  $\Pi = \alpha\beta'$  has been restricted to rank  $r \leq p$ . Other restrictions on the model parameters can be considered as in Johansen (1996). The most interesting restrictions from an economic point of view would likely be restrictions on the adjustment parameters  $\alpha$  and cointegration vectors  $\beta$ .

We consider restrictions on  $\alpha$  and  $\beta$  of two forms. The parameters can be partitioned into sets of columns, each of which can be restricted either to known values or to a linear subspace. Partition  $\alpha$  and  $\beta$  into  $\alpha = [\alpha_1, \alpha_2, \dots]$  and  $\beta = [\beta_1, \beta_2, \dots]$  such that each  $\alpha_i$  and  $\beta_i$  are  $p \times r_i$  matrices with  $\sum_i r_i = r$ . Restrictions to known values  $a_i$  and  $b_i$  can be specified for each  $\alpha_i$  and  $\beta_i$ , respectively. Linear restrictions can be imposed by specifying matrices  $A_i$  ( $p \times s_i$ ) and  $H_i$  ( $p \times s_i$ ) so that  $\alpha_i = A_i\psi_i$  and  $\beta_i = H_i\phi_i$  for conforming matrices  $\psi_i$  and  $\phi_i$ , see Johansen (1996, chapters 5 and 7) for details. Any submatrices without such restrictions provided are estimated unrestrictedly.

The estimation is performed using a switching algorithm as in Johansen (1996, chapter 7). The parameters for each pair of submatrices  $\alpha_i$  and  $\beta_i$  are estimated in turn by taking the other submatrices as fixed. The switching algorithm repeats the estimation through the sequence of submatrices until convergence.

When the restricted constant term  $\rho$  is included in the model, restrictions on columns of  $\rho$  must be of the same form as those for corresponding columns of  $\beta$ . This is because the estimation is performed by estimating the two parameters jointly through the matrix  $\beta^* = (\beta', \rho)'$  of dimension  $p1 \times r$ , where  $p1 = p + 1$ . For columns of  $\beta$  specified as known, the corresponding columns of  $\rho$  must also be restricted to known values. This is done in the program by specifying  $\beta^*$  as known. For linear restrictions the relevant hypotheses become  $\beta_i^* = H_i^*\phi_i$ , where  $H_i^*$  is  $p1 \times s_i$ .

For the fractional parameters,  $d$  and  $b$ , any linear restrictions can be imposed. For the  $\Gamma_i$  matrices, only exclusion restrictions are permitted.

The restrictions described above can be implemented individually or simultaneously by the software package. The restrictions are imposed by setting the appropriate options, and the implementation is described in section 5 below.



### 3 Main program: FCVARmodel.m

This program demonstrates the estimation of the fractionally cointegrated VAR model. There are seven main function calls, each of which are described in detail in subsequent sections.

- The function `DefaultEstnOptions(db0, k, r, p, initval, const)` is called to set estimation options, see section 5.
- The function `RankTests(x, k, db0, FCVARoptions)` is called to perform tests to determine cointegrating rank, see section 6.1. Note that the array `FCVARoptions` is the output from the function `DefaultEstnOptions` called above.
- The function `FCVARestn(x, k, r, db0, FCVARoptions)` is called to estimate the model, see section 6.2.
- The function `GetResiduals(x, k, r, dbHat, alphaHat, betaHat, rhoHat, GammaHat, EstnOptions)` is called to compute the residuals, see section 6.3. These could then be used for model diagnostics. The array `EstnOptions` is part of the output from the function `DefaultEstnOptions`.
- The function `RestrictEstnOptions(db0, k, r, p, defaultFCVARoptions)` is called to set estimation options to impose restrictions on parameter estimates, see section 5. This will provide the specifications to estimate a restricted model.
- The functions `FCVARestn` and `GetResiduals` are called again to estimate the restricted model and compute the residuals from the restricted model, see sections 6.2 and 6.3. This second estimation can be used to perform likelihood ratio tests of the imposed parameter restrictions.

## 4 Output: the Danish data

This section shows the output from executing the example file `FCVARmodel.m`. The data set is the “Danish data” from Johansen (1996), and the model specification and restrictions imposed (in the second half of the output) are as in Johansen (1996, section 7.3.1).

-----  
-----  
-----  
Restriction  $d = b$  is imposed.  
-----  
-----

### Likelihood Ratio Tests for Cointegrating Rank

-----  
-----

Dimension of System:	4	Number of observations for estimation:	50
Deterministics:	restricted constant	Initial values:	5
Number of Lags:	1	Number of observations in sample:	55

-----  
-----

Rank	d	b	Log-likelihood	LR statistic
0	1.010	1.010	593.689	71.709
1	0.966	0.966	615.707	27.673
2	0.468	0.468	620.678	17.731
3	0.470	0.470	627.615	3.856
4	0.470	0.470	629.543	----

-----  
-----  
-----  
-----

### Fractionally Cointegrated VAR: Estimation Results

-----  
-----

Dimension of System:	4	Number of observations for estimation:	50
Deterministics:	restricted constant	Number of initial values:	5
Number of Lags:	1	Number of observations in sample:	55

-----

Cointegrating rank:	1	AIC:	-1181.414
Log-likelihood:	615.707	BIC:	-1133.614
log(det(Omega_hat)):	-35.980	Number of free parameters:	25

-----

Fractional Parameters:

-----

Coefficient	Estimate	Standard Error
d	0.966	0.006

-----

Cointegrating Equations (beta and rho):

-----

Variable	CI equation 1
Var1	1.000
Var2	0.821
Var3	7.088
Var4	-0.326
Constant	-17.869

-----

Note: Identifying restriction imposed.

-----

Adjustment Matrix (alpha):

-----

Variable	CI equation 1
Var 1	-0.305
SE 1	( 0.043 )
Var 2	-0.091
SE 2	( 0.031 )
Var 3	0.004
SE 3	( 0.012 )
Var 4	0.009
SE 4	( 0.008 )

-----

Note: Standard Errors in parenthesis.

-----

Long Run Matrix (Pi):

-----

Variable	Var 1	Var 2	Var 3	Var 4
Var 1	-0.305	-0.250	-2.159	0.099
Var 2	-0.091	-0.074	-0.643	0.030
Var 3	0.004	0.004	0.031	-0.001
Var 4	0.009	0.008	0.067	-0.003

-----

-----  
-----  
Lag Matrix 1 (Gamma\_1):

Variable	Var 1	Var 2	Var 3	Var 4
Var 1	-0.431	0.179	0.303	-0.105
SE 1	( 0.129 )	( 0.153 )	( 0.426 )	( 0.539 )
Var 2	0.169	-0.086	0.454	-0.545
SE 2	( 0.122 )	( 0.143 )	( 0.372 )	( 0.498 )
Var 3	-0.006	0.108	0.394	-0.010
SE 3	( 0.048 )	( 0.056 )	( 0.144 )	( 0.193 )
Var 4	0.027	0.022	0.384	0.075
SE 4	( 0.032 )	( 0.038 )	( 0.097 )	( 0.131 )

Note: Standard Errors in parenthesis.  
-----

-----  
Inverse roots of the characteristic polynomial (eigenvalues of companion form)

Number	Real Part	Imaginary Part	Modulus
1	1.000	0.000	1.000
2	1.000	0.000	1.000
3	1.000	0.000	1.000
4	0.559	0.233	0.606
5	0.559	-0.233	0.606
6	-0.593	0.000	0.593
7	0.251	0.000	0.251
8	-0.174	0.000	0.174

-----  
-----  
Restrictions Imposed

Linear restrictions on beta[1]:

ans =

1	0	0
-1	0	0
0	1	0
0	-1	0
0	0	1

Linear restrictions on alpha[1]:

ans =

1	0
---	---

0 1  
0 0  
0 0

-----  
Warning: Number of free parameters reported do not reflect the parameters in alpha, beta, and rho.  
Calculate the number of free parameters for those manually and add to the number reported below.  
-----

Warning: Estimates are on based on linear restrictions so no standard errors are provided.

-----  
-----  
Fractionally Cointegrated VAR: Estimation Results  
-----  
-----

Dimension of System:	4	Number of observations for estimation:	50
Deterministics: restricted constant		Number of initial values:	5
Number of Lags:	1	Number of observations in sample:	55

-----  
Cointegrating rank: 1 AIC: -1178.724  
Log-likelihood: 606.362 BIC: -1146.220  
log(det(Omega\_hat)): -35.606 Number of free parameters: 17  
-----

Fractional Parameters:

Coefficient	Estimate	Standard Error
d	1.005	

No standard errors are provided, see warning above.  
-----

Cointegrating Equations (beta and rho):

Variable	CI equation 1
Var1	-0.161
Var2	0.161
Var3	-0.964
Var4	0.964
Constant	1.000

-----  
Adjustment Matrix (alpha):

---

Variable	CI equation 1
Var 1	1.635
Var 2	-0.210
Var 3	0.000
Var 4	0.000

---

No standard errors are provided, see warning above.

---

Long Run Matrix (Pi):

---

Variable	Var 1	Var 2	Var 3	Var 4
Var 1	-0.263	0.263	-1.577	1.577
Var 2	0.034	-0.034	0.203	-0.203
Var 3	0.000	0.000	0.000	0.000
Var 4	0.000	0.000	0.000	0.000

---

Lag Matrix 1 (Gamma\_1):

---

Variable	Var 1	Var 2	Var 3	Var 4
Var 1	-0.204	0.047	0.299	-1.534
Var 2	0.253	-0.105	-0.165	-0.819
Var 3	0.005	0.117	0.379	0.019
Var 4	0.022	0.026	0.420	0.098

---

No standard errors are provided, see warning above.

---

Inverse roots of the characteristic polynomial (eigenvalues of companion form)

---

Number	Real Part	Imaginary Part	Modulus
1	1.000	0.000	1.000
2	1.000	0.000	1.000
3	1.000	-0.000	1.000
4	0.837	0.000	0.837
5	0.348	0.326	0.477
6	0.348	-0.326	0.477
7	-0.331	0.091	0.344
8	-0.331	-0.091	0.344

---



---



---

## 5 Options files

### 5.1 Default estimation options: `DefaultEstnOptions.m`

The function `DefaultEstnOptions(db0, k, r, p, initval, const)` in this file returns a set of preliminary settings corresponding to the default set of options specified in the file. These options are set by the user.

The output from this file is later processed automatically by the function `AdjustEstnOptions(db0, k, r, p, initialFCVARoptions)`, see section 7, which completes the specification of the options array. Note that the latter function does not need any action from the user. The options specified in `DefaultEstnOptions.m` govern all options with the exception of those for restrictions on parameter estimates, which are specified in the file `RestrictEstnOptions.m` discussed below. Note that the latter function does not need any action from the user, but it could also be called by the user to change any of the options (e.g. start with unrestricted estimation and add restrictions later).

#### Inputs

- **db0**: vector or scalar of starting values (in the numerical optimization) for the fractional differencing parameters,  $d$  and  $b$ . The size of **db0** indicates whether or not restriction  $d = b$  is imposed (if **db0** is scalar the restriction is imposed).
- **k**: scalar denoting the number of lags.
- **r**: scalar denoting cointegrating rank.
- **p**: scalar denoting the number of variables in the system.
- **initval**: scalar denoting the number of initial values to condition upon in the estimation.
- **const**: scalar to select the deterministic term in the model - set equal to 1 to include a restricted constant term, else 0 for no deterministic term.

#### Output

The output is the cell array `FCVARoptions = { dbEstnOptions, EstnOptions }`, an array of 2 arrays of options for use in many other routines.

`dbEstnOptions = { constrained, C_db, c_db, R_db, r_db, dbFminOptions }` is an array of options for the estimation of the fractional differencing parameters  $d$  and  $b$ .

- **constrained**: scalar indicating if the restrictions  $dbMax \geq d \geq b \geq dbMin$  are imposed, given user inputs **dbMax** and **dbMin**.
- **C\_db** and **c\_db** are matrices for imposing restrictions of the form  $C\_db * (d, b)' \leq c\_db$ . These are produced automatically by the function `AdjustEstnOptions` from the user-defined bounds **dbMax** and **dbMin**.
  - **dbMax**: scalar denoting the upper bound for the estimation of  $d$  and  $b$ .

- `dbMin`: scalar denoting the lower bound for the estimation of  $d$  and  $b$ .
- `R_db` and `r_db`: optional matrices for imposing restrictions of the form  $R\_db * (d, b)' = r\_db$ . These are left blank in `DefaultEstnOptions.m` and only used in `RestrictEstnOptions.m`.
- `dbFminOptions`: array of options for built-in Matlab (numerical optimization) functions for the optimization over  $d$  and  $b$ .

`EstnOptions = { N, M, deterministic, R_Gamma, r_Gamma, [], [], [], nCols, A, H, a, b, [], {}, {}, {}, {}, [], [], [], print2screen, printGammas, printRoots, restrictFminOptions }` is a cell array of optional arguments for the estimation of the remaining parameters.

- `N`: scalar denoting the number of initial values of  $X$  that are to be conditioned upon in the estimation of the model.
- `M`: scalar parameter, which is unused at the moment, but may be used in later versions of the program.
- `deterministic`: string to determine the deterministic included in the model. The options are `'restricted constant'` and `' none '`. Note the spaces around the word `'none'`, for which a sample is commented out in `DefaultEstnOptions.m`. Anything other than `'restricted constant'` is interpreted as `' none '`.
- `R_Gamma` and `r_Gamma` are optional matrices for imposing restrictions of the form  $R\_Gamma * \Gamma = r\_Gamma$ . These are left blank in `DefaultEstnOptions.m` and only used in `RestrictEstnOptions.m`.
- `nCols`, `A`, `H`, `a`, `b`, `h` are optional matrices for use when estimating restricted models with Johansen's switching algorithm. These are also left blank in `DefaultEstnOptions.m` and only used in `RestrictEstnOptions.m`.
- `print2screen`: scalar indicator to print results (from `RankTests.m` and `FCVARestn.m`) to screen and to generate graphs of characteristic roots.
- `printGammas`: scalar indicator to print coefficients of lagged differences as well.
- `printRoots`: scalar indicator to print and plot roots of the characteristic polynomial.
- `restrictFminOptions`: array of options for the switching algorithm to optimize over all the other parameters besides  $d$  and  $b$ . This array has a format similar to that in `dbFminOptions`.



## 5.2 Restricted estimation options: RestrictEstnOptions.m

The function `RestrictEstnOptions(db0, k, r, p, defaultFCVARoptions)` returns a set of preliminary settings corresponding to the restrictions on parameters. These restrictions are set by the user. It is to be run after `DefaultEstnOptions(db0, k, r, p)` has already been called to set other options not directly related to restrictions on parameter estimates.

### Inputs

- `db0`: vector or scalar of starting values (in the numerical optimization) for the fractional differencing parameters,  $d$  and  $b$ . The size of `db0` indicates whether or not restriction  $d = b$  is imposed (if `db0` is scalar the restriction is imposed).
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `p`: scalar denoting the number of variables in the system.
- `defaultFCVARoptions`: array of options output from `DefaultEstnOptions`.

### Output

The output is the cell array `restrictFCVARoptions = { dbEstnOptions, EstnOptions }`, an array of 2 arrays of options for use in many other routines.

`dbEstnOptions = { constrained, C_db, c_db, R_db, r_db, dbFminOptions }` is an array of options for the estimation of the fractional differencing parameters  $d$  and  $b$ .

- The options `constrained` (including `dbMax` and `dbMin`) and `dbFminOptions` are unchanged from those in `defaultFCVARoptions.m`.
- `C_db` and `c_db`: optional matrices for imposing restrictions of the form  $C\_db * (d, b)' = c\_db$ . These are generated to reflect the options `constrained`, `dbMax` and `dbMin` through the function `AdjustEstnOptions`, first called in `DefaultEstnOptions.m`. These matrices can be modified by the user to impose additional linear inequality restrictions on  $d$  and  $b$ . Note that, if `C_db` is a matrix, `AdjustEstnOptions` does not modify `C_db` or `c_db`, so that the only linear inequality restrictions imposed are those given here (i.e. the constraint indicated by `constrained` would need to be imposed manually).
- `R_db` and `r_db`: optional matrices for imposing restrictions of the form  $R\_db * (d, b)' = r\_db$ .

`EstnOptions = { N, M, deterministic, R_Gamma, r_Gamma, [], [], [], nCols, A, H, a, b, [], {}, {}, {}, {}, [], [], [], print2screen, printGammas, printRoots, restrictFminOptions }` is a cell array of optional arguments for the estimation of the remaining parameters.

- The options `N`, `M`, `deterministic`, `print2screen`, `printGammas`, and `printRoots` are unchanged from those in `defaultFCVARoptions.m`.

- `R_Gamma` and `r_Gamma` are optional matrices for imposing restrictions of the form  $\mathbf{R\_Gamma} * \mathbf{\Gamma} = \mathbf{r\_Gamma}$ . Note: This is intended for exclusion restrictions only and does not work with other types of restrictions!
- `nCols`, `A`, `H`, `a`, `b` are optional matrices for use when estimating restricted models with Johansen's switching algorithm.
  - `nCols`: vector of dimensions  $r_i$  of the partitioned matrices of  $\alpha$  and  $\beta^*$ , where  $\sum_i r_i = r$  and  $\beta^* = \beta$  if no constant is present and  $\beta^* = (\beta', \rho)'$  if the constant is included. In the latter case,  $\beta^*$  is  $p1 \times r$  with  $p1 = p + 1$ .
  - `A`: array of linear restrictions on each alpha such that  $\alpha_i = A_i \psi_i$ , where  $A_i$  is  $p \times s_i$  and  $\psi_i$  is  $s_i \times r_i$ .
  - `H`: array of linear restrictions on each beta such that  $\beta_i^* = H_i^* \phi_i$ , where  $H_i^*$  is  $p1 \times s_i$  and  $\phi_i$  is  $s_i \times r_i$ .
  - `a`: array of submatrices of  $\alpha$  considered known.
  - `b`: array of submatrices of  $\beta$  considered known.

## 6 Other main files

### 6.1 Rank tests: RankTests.m

The function `RankTests(x, k, db0, RankTestOptions)` in this file performs a sequence of likelihood ratio tests for cointegrating rank.

Also prints the results to screen depending on the indicator `print2screen` set in the file `DefaultEstnOptions.m`, please see section 5.

#### Inputs

- `x`: vector or matrix of data.
- `k`: scalar denoting the number of lags.
- `db0`: vector or scalar specifying starting values for the estimation of the fractional differencing parameters  $d$  and  $b$ , constrained to be equal if `db0` is a scalar.
- `RankTestOptions = { dbEstnOptions, EstnOptions }`: an array of 2 arrays of options for use in the estimation routines, see sections 5 and 7.1 for details.

#### Output

`rankTestStats`: vector of cointegrating rank test statistics.

#### Dependencies

Calls the function `FCVARlike`, see section 7.

## 6.2 Model estimation: FCVARestn.m

The function `FCVARestn(x, k, r, db0, FCVARoptions)` in this file estimates the fractional cointegration model under desired options.

Also prints the results to screen and creates a graph of the characteristic roots, depending on the indicator `print2screen`. Depending on the additional indicator `printGammas`, the estimates `GammaHat` are also printed. The indicators are set in `DefaultEstnOptions.m` and `RestrictEstnOptions.m`, please see section 5.

### Inputs

- `x`: vector or matrix of data.
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `db0`: vector or scalar specifying starting values for the estimation of the fractional differencing parameters  $d$  and  $b$ , constrained to be equal if `db0` is a scalar.
- `FCVARoptions = { dbEstnOptions, EstnOptions }`: an array of 2 arrays of options for use in estimation routines, see sections 5 and 7.1 for details.

### Output

`estimates = { dbHat, alphaHat, betaHat, rhoHat, PiHat, OmegaHat, GammaHat }`: matrices of parameter estimates. Note that `GammaHat` is reported as the  $p \times kp$  matrix [ `GammaHat1, ..., GammaHatk` ].

### Dependencies

Calls the functions `FCVARlike` and `GetParams` for estimation, the functions `FCVARhess`, `SEmat2vec`, and `SEvec2mat` for calculation of standard errors, and the functions `DegreesOfFreedom` and `CharPolyRoots` for calculation of additional estimation results.

### 6.3 Obtain residuals: `GetResiduals.m`

The function `GetResiduals(x, k, r, db, alpha, beta, rho, Gamma, EstnOptions)` in this file obtains the residuals from the fractionally cointegrated VAR model.

#### Inputs

- `x`: vector or matrix of data.
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `db`: scalar or vector with parameter value(s) for  $d$  and  $b$ , at which to obtain residuals. If a scalar is given, then the restriction  $d = b$  is imposed.
- `alpha`, `beta`, `rho`, `Gamma`: matrices with parameter values for  $\alpha$ ,  $\beta$ ,  $\rho$ , and  $\Gamma$ , at which to obtain residuals.
- `EstnOptions`: cell array with options, see sections 5 and 7.1 for details.

#### Output

`epsilon`: matrix of residuals from fractionally cointegrated VAR model evaluated at the parameter values in the inputs.

#### Dependencies

Calls the function `TransformData`, see section 7.

## 7 Auxilliary files

These are files that are used by the main programs, but which do not require any action or modification by the user. They are stored in the folder `FCVARtools` and added to the path of functions available to Matlab using a command in the file `DefaultEstnOptions.m`.

### 7.1 `AdjustEstnOptions.m`

The function `AdjustEstnOptions(db0, k, r, p, initialFCVARoptions)` completes the specification of the options array, initialized by `DefaultEstnOptions(db0, k, r, p)` and `RestrictEstnOptions(db0, k, r, p, defaultFCVARoptions)`.

#### Inputs

- `db0`: vector or scalar of starting values (in the numerical optimization) for the fractional differencing parameters,  $d$  and  $b$ . The size of `db0` indicates whether or not restriction  $d = b$  is imposed (if `db0` is scalar the restriction is imposed), see section 5.
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `p`: scalar denoting the number of variables in the system.
- `initialFCVARoptions = { dbEstnOptions, EstnOptions }`: These are the arrays `dbEstnOptions` and `EstnOptions` from the function `DefaultEstnOptions` or `RestrictEstnOptions`, see section 5.

#### Output

The output is the cell array `FCVARoptions = { dbEstnOptions, EstnOptions }`, an array of 2 arrays of options with the same form as `initialFCVARoptions`. Both arrays are transformations of the corresponding arrays in the input array `initialFCVARoptions`.

`dbEstnOptions = { constrained, C_db, c_db, R_db, r_db, dbFminOptions }`: array of options for the estimation of the fractional differencing parameters  $d$  and  $b$ .

- `C_db` and `c_db` are optional matrices for imposing restrictions of the form  $C\_db*(d, b)' \leq c\_db$ . Note that `C_db` and `c_db` are automatically generated to impose  $dbMax \geq d \geq b \geq dbMin$  given the appropriate indicator `constrained`. However, the user can modify this to impose other or further restrictions on  $d$  and/or  $b$  within `RestrictEstnOptions.m`.
- The rest are as defined in section 5.

`EstnOptions = { N, M, deterministic, R_Gamma, r_Gamma, [], [], [], nCols, A, H, a, b, [], Abar, Aperp, replaceCols, keepCols, [], [], [], print2screen, printGammas, printRoots, restrictFminOptions }`: array of options for estimating the remaining parameters.

- `A`, `H`, `a`, and `b`: defined as in section 5 but `A` and `H` are replaced with identities where no restrictions are specified.

- **Abar**: an array of matrices  $A_{bar\_i}$  such that  $A_{bar\_i}' * A\_i = I$ .
- **Aperp**: an array of matrices  $A_{perp\_i}$  such that  $A_{perp\_i}' * A\_i = 0$ .

Note that when the columns in the  $i$ 'th partition of  $\alpha$  are assumed known,  $A_{bar\_i}$  and  $A_{perp\_i}$  are defined in terms of lower case  $a\_i$  instead of  $A\_i$  (following Johansen, 1996).

- **replaceCols** and **keepCols**: arrays of vectors specifying the column numbers that are replaced or assumed known in each stage of the switching algorithm.
- The rest are as defined in section 5.

## Dependencies

None.

## 7.2 CharPolyRoots.m

The function `CharPolyRoots(b, alpha, beta, Gamma, printRoots)` calculates the inverse roots of the characteristic polynomial for the fractionally cointegrated VAR model and plots them with the transformed unit circle, see Johansen (2008). The inverse roots are calculated from the companion form of the VAR, where the inverse roots are given as the eigenvalues of the coefficient matrix. The plot also shows the transformed unit circle  $\mathbb{C}_b$  of Johansen (2008), further transformed under the mapping  $1/z$ , so that the inverse roots (eigenvalues of companion form) should be inside the plotted “unit circle”.

### Inputs

- `b`: scalar fractional differencing parameter.
- `alpha`, `beta`, `Gamma`: matrices of parameters. Note: The matrix `Gamma` is a  $p \times pk$  matrix of coefficients in the lag polynomial.
- `printRoots`: scalar indicator to determine if the roots are printed and plotted.

### Output

`charPolyRoots`: complex vector with the inverse roots of the characteristic polynomial.

### Dependencies

None.



### 7.3 FCVARhess.m

The function `FCVARhess(x, k, r, db, alpha, beta, rho, Gamma, EstnOptions)` is the Hessian matrix for the fractionally cointegrated VAR model evaluated at `[ db, alpha, beta, rho, Gamma ]`.

#### Inputs

- `x`: vector or matrix of data.
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `db`: scalar or vector with parameter value(s) for  $d$  and  $b$ , at which to calculate the log-likelihood. If a scalar is given, then the restriction  $d = b$  is imposed.
- `alpha, beta, rho, Gamma`: matrices of parameter values at which to calculate the log-likelihood.
- `EstnOptions`: cell array of options described in detail in the comments to the functions `DefaultEstnOptions` and `RestrictEstnOptions` in section 5 and `AdjustEstnOptions` in section 7.1.

#### Output

`hessian`: the Hessian matrix for the fractionally cointegrated VAR model evaluated at the parameter values in the inputs.

#### Dependencies

Calls the function `FullFCVARlike` to get the likelihood and also calls the functions `SEmat2vec` and `SEvec2mat` to convert parameter vector to matrices and vice versa.

## 7.4 FCVARlike.m

The function `FCVARlike(x, k, r, db, EstnOptions)` in this file is the log-likelihood function for the fractionally cointegrated VAR model.

### Inputs

- `x`: vector or matrix of data.
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `db`: vector or scalar of fractional differencing parameters,  $d$  and  $b$ . The size of `db` indicates whether or not restriction  $d = b$  is imposed (if `db` is scalar the restriction is imposed).
- `EstnOptions`: an array of options for estimating the remaining parameters. The elements of `EstnOptions` are described in detail in the comments to the functions `DefaultEstnOptions` and `RestrictEstnOptions` in section 5 and `AdjustEstnOptions` in section 7.1.

### Output

`like`: scalar value of the profile (concentrated) log-likelihood function evaluated at parameter value `db`.

### Dependencies

Calls the function `GetParams`.

## 7.5 FracDiff.m

The function `FracDiff(x,d)` calculates the fractional difference of  $\mathbf{x}$  using the fast fractional difference algorithm in Jensen and Nielsen (2013).

### Inputs

- $\mathbf{x}$ : vector or matrix of data.
- $d$ : scalar denoting the value of  $d$  at which to calculate the fractional difference.

### Output

$\mathbf{dx}$ : vector or matrix of same dimensions as  $\mathbf{x}$  with  $t$ 'th row given by  $(1 - L)^d X_t$ .

### Dependencies

None.

## 7.6 FreeParams.m

The function `FreeParams(db, k, r, p, FCVARoptions)` calculates the number of free parameters for the fractionally cointegrated VAR model.

When no restrictions are specified on  $\alpha$  or  $\beta$ , the estimation imposes the identifying restriction that the top rows of  $\beta$  are equal the identity matrix. This restriction is included in the calculation of the number of free parameters. For restricted estimation, it is assumed that the user imposes enough restrictions such that the parameters are identified.

Whenever there are restrictions on both  $\alpha$  and  $\beta$ , the warning `fpWarning` is given to indicate that the output `fp` does not include the number of free parameters in  $\alpha$ ,  $\beta$ , and  $\rho$ . This happens when there are restrictions placed on both  $\alpha$  and  $\beta$  together, in which case it is possible that some restrictions are redundant.

### Inputs

- `db`: scalar or vector of fractional differencing parameters  $d$  and  $b$  ( $d = b$  is imposed if `db` is scalar).
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `p`: scalar denoting the number of variables in the system.
- `FCVARoptions = { dbEstnOptions, EstnOptions }`: an array of 2 arrays of options for use in estimation routines, see sections 5 and 7.1 for details.

### Output

- `fp`: scalar with the number of free parameters for the fractionally cointegrated VAR model.
- `fpWarning`: binary variable taking the value 1 if the value of `fp` does not include the number of free parameters in  $\alpha$ ,  $\beta$ , and  $\rho$ .

### Dependencies

None.

## 7.7 FullFCVARlike.m

The function `FullFCVARlike(x, k, r, db, alpha, beta, rho, Gamma, EstnOptions)` in this file is the unconcentrated log-likelihood function for the fractionally cointegrated VAR model.

### Inputs

- **x**: vector or matrix of data.
- **k**: scalar denoting the number of lags.
- **r**: scalar denoting cointegrating rank.
- **db**: scalar or vector with parameter value(s) for  $d$  and  $b$ , at which to calculate the log-likelihood. If a scalar is given, then the restriction  $d = b$  is imposed.
- **alpha, beta, rho, Gamma**: matrices of parameter values at which to calculate the log-likelihood.
- **EstnOptions**: an array of options for estimating the remaining parameters. The elements of `EstnOptions` are described in detail in the comments to the functions `DefaultEstnOptions` and `RestrictEstnOptions` in section 5 and `AdjustEstnOptions` in section 7.1.

### Output

**like**: scalar value of the log-likelihood function evaluated at the parameter values in the inputs.

### Dependencies

Calls the function `GetResiduals`.

## 7.8 GetParams.m

The function `GetParams(x, k, r, db, EstnOptions)` returns the parameter estimates of the fractionally cointegrated VAR model using regression and reduced rank regression.

### Inputs

The inputs `x`, `k`, `r`, `db`, and `EstnOptions` are as defined for the function `FCVARlike`.

### Output

`estimates = { db, alphaHat, betaHat, rhoHat, PiHat, OmegaHat, GammaHat }`: matrices of parameter estimates. Note that `db` is the same as in the `db` in the input, and `GammaHat` is reported as the  $p \times kp$  matrix `[ GammaHat1, ..., GammaHatk ]`.

### Dependencies

Calls the function `TransformData` and, if any restrictions are imposed on the estimation of  $\alpha$  and  $\beta$ , also the function `GetParamsSwitching`.

## 7.9 GetParamsSwitching.m

The function `GetParamsSwitching(beta0, S00, S01, S11, T, p, r, EstnOptions)` returns the parameter estimates of the fractionally cointegrated VAR model using regression and reduced rank regression, by way of Johansen's switching algorithm, see Johansen (1996, chapters 5 and 7).

### Inputs

- **beta0**: matrix of starting values for  $\beta^* = (\beta', \rho)'$  in the switching algorithm. The unrestricted estimate of  $\beta^*$  is used and is then reordered within this function as the matrix closest to satisfying any linear restrictions on  $\beta^*$  as in Johansen (1996).
- **S00, S01 and S11**: matrices of product moments,

$$S_{ij} = T^{-1} \sum_{t=1}^T R_{it} R'_{jt},$$

where  $R_{it}$  are residuals from regressions involving the variables  $Z_{it}$  defined in `TransformData.m`, see also Johansen (1996, section 6.1) for the non-fractional case.

- **T**: number of observations used for estimation.
- **p**: scalar denoting the number of variables in the system.
- **r**: scalar denoting cointegrating rank.
- **EstnOptions**: array of options described in detail in the comments to the functions `DefaultEstnOptions` and `RestrictEstnOptions` in section 5 and `AdjustEstnOptions` in section 7.1.

### Output

**betaStar**, **alphaHat**, and **OmegaHat**: matrices of parameter estimates.

### Dependencies

None.

## 7.10 Lbk.m

The function `Lbk(x, b, k)` in this file calculates a lag polynomial in the fractional lag operator,  $L_b$ .

### Inputs

- **x**: vector or matrix of data.
- **b**: scalar denoting the value of  $b$  at which to calculate the lag polynomial.
- **k**: scalar denoting the number of lags.

### Output

**Lbkx**: matrix with same number of rows but **k** times as many columns as **x** and  $t$ 'th row given by  $[L_b^1 X_t, L_b^2 X_t, \dots, L_b^k X_t]$ , where  $L_b = 1 - (1 - L)^b$ .

### Dependencies

Calls the function `FracDiff`.



## 7.11 SEmat2vec.m

The function `SEmat2vec( db, alpha, Gamma, k, r, p )` transforms the vector `db` and the matrices `alpha` and `Gamma` into either a  $(1 + pr + p^2k)$  or a  $(2 + pr + p^2k)$  parameter vector, depending on the size of `db`. Note that `SEmat2vec( db, alpha, Gamma, k, r, p )` and `SEvec2mat( param, k, r, p )` are inverse functions of each other.

### Inputs

- `db`: scalar or vector of  $d$  and  $b$ .
- `alpha`: matrix containing  $\alpha$ .
- `Gamma`: matrix containing  $\Gamma$ .
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `p`: scalar denoting the number of variables in the system.

### Output

`param`: either a  $(1 + pr + p^2k)$  or a  $(2 + pr + p^2k)$  parameter vector, depending on the size of `db`.

### Dependencies

None.

## 7.12 SEvec2mat.m

The function `SEvec2mat( param, k, r, p )` transforms either a  $(2 + pr + p^2k)$  or a  $(2 + pr + p^2k)$  parameter vector into the vector `db` and the matrices `alpha` and `Gamma`. Note that `SEmat2vec( db, alpha, Gamma, k, r, p )` and `SEvec2mat( param, k, r, p )` are inverse functions of each other.

### Inputs

- `param`: either a  $(1 + pr + p^2k)$  or a  $(2 + pr + p^2k)$  parameter vector, depending on the size of `db`.
- `k`: scalar denoting the number of lags.
- `r`: scalar denoting cointegrating rank.
- `p`: scalar denoting the number of variables in the system.

### Output

- `db`: scalar or vector of  $d$  and  $b$ .
- `alpha`: matrix containing  $\alpha$ .
- `Gamma`: matrix containing  $\Gamma$ .

### Dependencies

None.

### 7.13 TransformData.m

The function `TransformData(x, k, db, EstnOptions)` returns the transformed data required for regression and reduced rank regression.

#### Inputs

- `x`: vector or matrix of data.
- `k`: scalar denoting the number of lags.
- `db`: scalar or vector of fractional differencing parameters  $d$  and  $b$ . If this is a scalar, then the restriction  $d = b$  is imposed.
- `EstnOptions`: array of options described in detail in the comments to the functions `DefaultEstnOptions` and `RestrictEstnOptions` in section 5 and `AdjustEstnOptions` in section 7.1.

#### Output

`Z0`, `Z1`, `Z2`: matrices of transformed data,

$$Z_{0t} = \Delta^d X_t, Z_{1t} = \Delta^{d-b} L_b(X_t, 1), Z_{2t} = (\Delta^d L_b X_t, \dots, \Delta^d L_b^k X_t), t = 1, \dots, T,$$

where the constant term 1 is omitted from  $Z_{1t}$  if there is no constant term in the model. See Johansen (1996, section 6.1) for the non-fractional case.

Note that the input matrix has  $T + N$  rows, where  $N$  is the number of initial values to be conditioned upon in the estimation, whereas the output matrices have  $T$  rows, which are the  $T$  observations that are directly included in the likelihood function.

#### Dependencies

Calls the functions `FracDiff` and `Lbk`.

## 8 List of references

1. Jensen, A. N. and Nielsen, M. Ø. (2013). A fast fractional difference algorithm. QED working paper 1307, Queen's University.
2. Johansen, S. (1996). *Likelihood-Based Inference in Cointegrated Vector Autoregressive Models*, 2nd edition, Oxford University Press, Oxford.
3. Johansen, S. (2008). A representation theory for a class of vector autoregressive models for fractional processes. *Econometric Theory* **24**, 651–676.
4. Johansen, S. and Nielsen, M. Ø. (2010). Likelihood inference for a nonstationary fractional autoregressive model. *Journal of Econometrics* **158**, 51–66.
5. Johansen, S. and Nielsen, M. Ø. (2012). Likelihood inference for a fractionally cointegrated vector autoregressive model. *Econometrica* **80**, 2667–2732.
6. MacKinnon, J. G. and Nielsen, M. Ø. (2012). Numerical distribution functions of fractional unit root and cointegration tests. Forthcoming in *Journal of Applied Econometrics*.