Analysis of Integrated and Cointegrated Time Series

Dr. Bernhard Pfaff
bernhard_pfaff@fra.invesco.com

Invesco Asset Management Deutschland GmbH, Frankfurt am Main

The 1st International R/Rmetrics User and Developer Workshop
8–12 July 2007, Meielisalp, Lake Thune, Switzerland

Contents

Contents

Contents

Contents

Contents

Univariate Time Series
Overview

- Definitions
- Representations / Models
- Nonstationary processes
- Statistical tests

Definitions

Stochastic Process

Time Series

A discrete *time series* is defined as an ordered sequence of random numbers with respect to time. More formally, such a *stochastic process* can be written as:

$$\{y(s,t), s \in \mathfrak{S}, t \in \mathfrak{T}\} \ , \qquad (1)$$

where for each $t \in \mathfrak{T}$, $y(\cdot, t)$ is a random variable on the sample space $\mathfrak{S}$ and a realization of this stochastic process is given by $y(s, \cdot)$ for each $s \in \mathfrak{S}$ with regard to a point in time $t \in \mathfrak{T}$.

Definitions

Stochastic Process – Examples

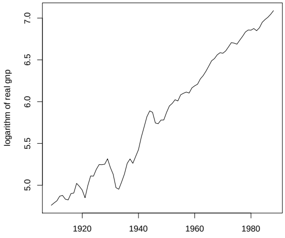


Figure: U.S. GNP

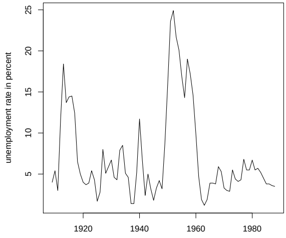


Figure: U.S. unemployment rate

```
> library(urca)
> data(npext)
> y <- ts(na.omit(npext$realgnp), start = 1909, end = 1988, frequency = 1)
> z <- ts(exp(na.omit(npext$unemploy)), start = 1909, end = 1988, frequency = 1)
> plot(y, ylab = "logarithm of real gnp")
> plot(z, ylab = "unemployment rate in percent")
```

Definitions
Stationarity

Weak Stationarity

The ameliorated form of a stationary process is termed *weakly stationary* and is defined as:

$$E[y_t] = \mu < \infty, \forall t \in \mathfrak{T} \ , \quad (2a)$$

$$E[(y_t - \mu)(y_{t-j} - \mu)] = \gamma_j, \forall t, j \in \mathfrak{T} \ . \quad (2b)$$

Because only the first two theoretical moments of the stochastic process have to be defined and being constant, finite over time, this process is also referred to as being *second-order stationary* or *covariance stationary*.

Strict Stationarity

The concept of a *strictly stationary* process is defined as:

$$F\{y_1, y_2, \ldots, y_t, \ldots, y_T\} = F\{y_{1+j}, y_{2+j}, \ldots, y_{t+j}, \ldots, y_{T+j}\} \ , \quad (3)$$

where $F\{\cdot\}$ is the joint distribution function and $\forall t, j \in \mathfrak{T}$.

Note:

Hence, if a process is strictly stationary with finite second moments, then it must be covariance stationary as well. Although a stochastic processes can be set up to be covariance stationary, it need not be a strictly stationary process. It would be the case, for example, if the mean and autocovariances would not be functions of time but of higher moments instead.

Definitions
White Noise

Definition
A *white noise* process is defined as:

$$E(\varepsilon_t) = 0 \ , \tag{4a}$$

$$E(\varepsilon_t^2) = \sigma^2 \ , \tag{4b}$$

$$E(\varepsilon_t \varepsilon_\tau) = 0 \quad \text{for} \quad t \neq \tau \ . \tag{4c}$$

When necessary, ε_t is assumed to be normally distributed: $\varepsilon_t \backsim \mathcal{N}(0, \sigma^2)$. If Equations 4a–4c are amended by this assumption, then the process is said to be a *normal-* or *Gaussian white noise* process. Furthermore, sometimes Equation 4c is replaced with the stronger assumption of independence. If this is the case, then the process is said to be an *independent white noise* process. Please note that for normally distributed random variables, uncorrelatedness and independence are equivalent. Otherwise, independency is sufficient for uncorrelatedness but not *vice versa*.

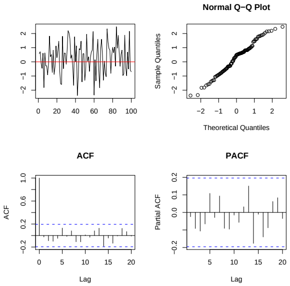
Definitions

White Noise – Example

R code

```
> set.seed(12345)
> gwn <- rnorm(100)
> layout(matrix(1:4, ncol = 2, nrow = 2))
> plot.ts(gwn, xlab = "", ylab = "")
> abline(h = 0, col = "red")
> acf(gwn, main = "ACF")
> qqnorm(gwn)
> pacf(gwn, main = "PACF")
```

R Output



Definitions
Ergodicity

Definition

Ergodicity refers to one type of asymptotic independence. More formally, asymptotic independence can be defined as

$$|F(y_1, \ldots, y_T, y_{j+1}, \ldots, y_{j+T}) - F(y_1, \ldots, y_T)F(y_{j+1}, \ldots, y_{j+T})| \to 0 \; , \tag{5}$$

with $j \to \infty$. The joint distribution of two subsequences of a stochastic process $\{y_t\}$ is equal to the product of the marginal distribution functions the more distant the two subsequences are from each other. A stationary stochastic process is ergodic if

$$\lim_{T \to \infty} \left\{ \frac{1}{T} \sum_{j=1}^{T} E[y_t - \mu][y_{t+j} - \mu] \right\} = 0 \; , \tag{6}$$

holds. This equation would be satisfied if the autocovariances tend to zero with increasing j.

In prose:

Asymptotic independence means that two realizations of a time series become ever closer to independence, the further they are apart with respect to time.

Wold Decomposition

Theorem

Any covariance stationary time series $\{y_t\}$ can be represented in the form:

$$y_t = \mu + \sum_{j=0}^{\infty} \psi_j \varepsilon_{t-j} \ , \ \varepsilon_t \sim WN(0, \sigma^2) \tag{7a}$$

$$\psi_0 = 1 \textit{ and } \sum_{j=0}^{\infty} \psi_j^2 < \infty \tag{7b}$$

Characteristics

- Fixed mean: $E[y_t] = \mu$:
- Finite variance: $\gamma_0 = \sigma^2 \sum_{j=0}^{\infty} \psi_j^2 < \infty$.

Box-Jenkins

- Autoregressive moving average models (ARMA)
- Approximate Wold form of a stationary time series by a parsimonious parametric model
- ARMA(p,q) model:

$$\begin{aligned} y_t - \mu &= \phi_1(y_{t-1} - \mu) + \ldots + \phi_p(y_{t-p} - \mu) \\ &\quad + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \ldots + \theta_q \varepsilon_{t-q} \\ \varepsilon_t &\sim WN(0, \sigma^2) \end{aligned} \tag{8}$$

- Extension for integrated time series: ARIMA(p,d,q) model class.

Box-Jenkins

Procedure

1. If necessary, transform data, such that covariance stationarity is achieved.
2. Inspect, ACF and PACF for initial guesses of p and q.
3. Estimate proposed model.
4. Check residuals (diagnostic tests) and stationarity of process.
5. If item 4 fails, go to item 2 and repeat. If in doubt, choose the more parsimonious model specification.

Box-Jenkins

R Resources

- Package dse1: ARMA
- Package fSeries: ArmaModelling
- Package forecast: arima
- Package mAr: mAr.eig, mAr.est, mAr.pca
- Package stats: ar, arima, acf, pacf, ARMAacf, ARMAtoMA
- Package tseries: arma

Box-Jenkins

Example

R code

```
> set.seed(12345)
> y.ex <- arima.sim(n = 500,
+      list(ar = c(0.9, -0.4)))
> layout(matrix(1:3, nrow = 3, ncol = 1))
> plot(y.ex, xlab = "",
+      main = "Time series plot")
> abline(h = 0, col = "red")
> acf(y.ex, main = "ACF of y.ex")
> pacf(y.ex, main = "PACF of y.ex")
> arma20 <- arima(y.ex, order = c(2, 0, 0),
+      include.mean = FALSE)
> result <- matrix(cbind(arma20$coef,
+      sqrt(diag(arma20$var.coef))),
+      nrow = 2)
> rownames(result) <- c("ar1", "ar2")
> colnames(result) <- c("estimate", "s.e.")
```

R Output

	estimate	s.e.
ar1	0.90	0.04
ar2	−0.39	0.04

Table: ARMA(2, 0) Estimates

R Output

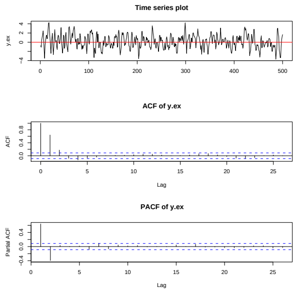


Figure: ARMA(2, 0) – simulated

Nonstationary Processes

General Remarks

- Many economic/financial time series exhibit trending behaviour.
- Task: determine most appropriate form of this trend.
- Stationary time series: time invariants moments
- In distinction: nonstationary processes have time dependent moments (mostly mean and/or variance).

Nonstationary Processes

Time Series decomposition

Trend-Cycle Decomposition

Consider,

$$
\begin{aligned}
y_t &= TD_t + Z_t \\
TD_t &= \beta_1 + \beta_2 \cdot t \\
\phi(L)Z_t &= \theta(L)\varepsilon_t \text{ with } \varepsilon_t \sim WN(0, \sigma^2) \text{ , with} \\
\phi(L) &= 1 - \phi_1 L - \ldots - \phi_p L^p \text{ and} \\
\theta(L) &= 1 + \theta_1 L + \ldots + \theta_q L^q
\end{aligned} \tag{9}
$$

Assumptions:

- $\phi(z) = 0$ has at most one root on the complex unit circle.
- $\theta(z) = 0$ has all roots outside the unit circle.

Nonstationary Processes

Trend Stationary Time Series

Definition

The series y_t is trend stationary if the roots of $\phi(z) = 0$ are outside the unit circle.

- $\phi(L)$ is invertible.
- Z_t has the Wold representation:

$$\begin{aligned} Z_t &= \phi(L)^{-1}\theta(L)\varepsilon_t \\ &= \psi(L)\varepsilon_t \end{aligned} \tag{10}$$

with $\psi(L) = \phi(L)^{-1}\theta(L) = \sum_{j=0}^{\infty} \psi_j L^j$ and $\psi_0 = 1$ and $\psi(1) \neq 0$.

Nonstationary Processes

Trend Stationary Time Series – Example

R code

```
> set.seed(12345)
> y.tsar2 <- 5 + 0.5 * seq(250) +
+   arima.sim(list(ar = c(0.8, -0.2)), n = 250)
> plot(y.tsar2, ylab="", xlab = "")
> abline(a=5, b=0.5, col = "red")
```

R Output

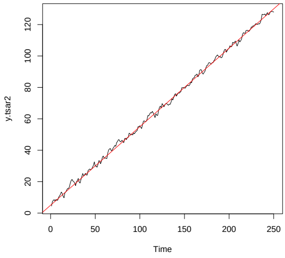


Figure: Trend-stationary series

Nonstationary Processes

Difference Stationary Time Series

Definition

The series y_t is difference stationary if $\phi(z) = 0$ has one root on the unit circle and the others are outside the unit circle.

- $\phi(L)$ can be factored as

$$\phi(L) = (1 - L)\phi^*(L) \text{ whereby} \tag{11}$$

  $\phi^*(z) = 0$ has all $p - 1$ roots outside the unit circle.
- ΔZ_t is stationary and has an ARMA(p-1, q) representation.
- If Z_t is difference stationary, then Z_t is integrated of order one: $Z_t \sim I(1)$.
- Recursive substitution yields: $y_t = y_0 + \sum_{j=1}^{t} u_j$.

Nonstationary Processes

Difference Stationary Time Series – Example

R code

```
> set.seed(12345)
> u.ar2 <- arima.sim(
+    list(ar = c(0.8, -0.2)), n = 250)
> y1 <- cumsum(u.ar2)
> TD <- 5.0 + 0.7 * seq(250)
> y1.d <- y1 + TD
> layout(matrix(1:2, nrow = 2, ncol = 1))
> plot.ts(y1, main = "I(1) process without drift",
+    ylab="", xlab = "")
> plot.ts(y1.d, main = "I(1) process with drift",
+    ylab="", xlab = "")
> abline(a=5, b=0.7, col = "red")
```

R Output

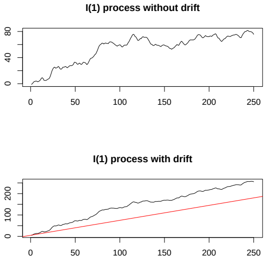


Figure: Difference-stationary series

Note:

If $u_t \sim IWN(0, \sigma^2)$, then y_t is a *random walk*.

Statistical tests

Unit Root vs. Stationarity Tests

General Remarks

Consider, the following trend-cycle decomposition of a time series y_T:

$$y_t = TD_t + Z_t = TD_T + TS_t + C_t \text{ with} \quad (12)$$

TD_t signifies the deterministic trend, TS_t is the stochastic trend and C_t is a stationary component.

- Unit root tests: $H_0 : TS_t \neq 0$ vs. $H_1 : TS_t = 0$, that is $y_t \sim I(1)$ vs. $y_t \sim I(0)$.
- Stationarity tests: $H_0 : TS_t = 0$ vs. $H_1 : TS_t \neq 0$, that is $y_t \sim I(0)$ vs. $y_t \sim I(1)$.

Autoregressive unit root tests

General Remarks

Tests are based on the following framework:

$$y_t = \phi y_{t-1} + u_t \ , u_t \sim I(0) \tag{13}$$

- $H_0 : \phi = 1, \ H_1 : |\phi| < 1$
- Tests: ADF- and PP-test.
- ADF: Serial correlation in u_t is captured by autoregressive parametric structure of test.
- PP: Non-parametric correction based on estimated long-run variance of Δy_t.

Autoregressive unit root tests

Augmented Dickey-Fuller Test, I

Test Regression

$$y_t = \beta' D_t + \phi y_{t-1} + \sum_{j=1}^{p} \psi_j \Delta y_{t-j} + u_t \ , \tag{14}$$

$$\Delta y_t = \beta' D_t + \pi y_{t-1} + \sum_{j=1}^{p} \psi_j \Delta y_{t-j} + u_t \text{ with } \pi = \phi - 1 \tag{15}$$

Test Statistic

$$ADF_t : t_{\phi=1} = \frac{\hat{\phi} - 1}{SE(\phi)} \ , \tag{16}$$

$$ADF_t : t_{\pi=0} = \frac{\hat{\pi}}{SE(\pi)} \ . \tag{17}$$

Autoregressive unit root tests

Augmented Dickey-Fuller Test, II

R Resources

- Function ur.df in package urca.
- Function ADF.test in package uroot.
- Function adf.test in package tseries.
- Function urdfTest in package fSeries.

Literature

- Dickey, D. and W. Fuller, Distribution of the Estimators for Autoregressive Time Series with a Unit Root, *Journal of the American Statistical Society*, 74 (1979), 427–341.
- Dickey, D. and W. Fuller, Likelihood Ratio Statistics for Autoregressive Time Series with a Unit Root, *Econometrica*, 49, 1057–1072.
- Fuller, W., *Introduction to Statistical Time Series*, 2nd Edition, 1996, New York: John Wiley.
- MacKinnon, J., Numerical Distribution Functions for Unit Root and Cointegration Tests, *Journal of Applied Econometrics*, 11 (1996), 601-618.

Autoregressive unit root tests

Augmented Dickey-Fuller Test, III

R code

```
> library(urca)
> y1.adf.nc.2 <- ur.df(y1,
+     type = "none", lags = 2)
> dy1.adf.nc.2 <- ur.df(diff(y1),
+     type = "none", lags = 1)
> plot(y1.adf.nc.2)
```

R Output

	Statistic	1pct	5pct	10pct
y_1	0.85	−2.58	−1.95	−1.62
Δy_1	−8.14	−2.58	−1.95	−1.62

Table: ADF-test results

R Output

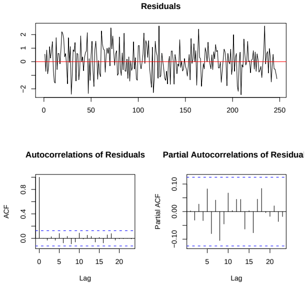


Figure: Residual plot of $y1$ ADF-regression

Note:

Use critical values of Dickey & Fuller, Fuller or MacKinnon.

Autoregressive unit root tests

Phillips & Perron Test, I

Test Regression

$$\Delta y_t = \beta' D_t + \pi y_{t-1} + u_t \ , u_t \sim I(0) \tag{18}$$

Test Statistic

$$Z_t = \left(\frac{\hat{\sigma}^2}{\hat{\lambda}^2}\right)^{1/2} \cdot t_{\pi=0} - \frac{1}{2}\left(\frac{\hat{\lambda}^2 - \hat{\sigma}^2}{\hat{\lambda}^2}\right) \cdot \left(\frac{T \cdot SE(\hat{\pi})}{\hat{\sigma}^2}\right) \ , \tag{19}$$

$$Z_\pi = T\hat{\pi} - \frac{T^2 \cdot SE(\hat{\pi})}{2\hat{\sigma}^2} \cdot (\hat{\lambda}^2 - \hat{\sigma}^2) \ . \tag{20}$$

$\hat{\lambda}$ and $\hat{\sigma}$ signify consistent estimates of the error variance.

Autoregressive unit root tests

Phillips & Perron Test, II

R Resources

- Function ur.pp in package urca.
- Function pp.test in package tseries.
- Function urppTest in package fSeries.
- Function PP.test in package stats.

Literature

- Phillips, P.C.B., Time Series Regression with a Unit Root, *Econometrica*, 55, 227–301.
- Phillips, P.C.B. and P. Perron, Testing for Unit Roots in Time Series Regression, *Biometrika*, 75, 335–346.

Autoregressive unit root tests

Phillips & Perron Test, III

R code

```
> library(urca)
> y1.pp.ts <- ur.pp(y1, type = "Z-tau",
+     model = "trend", lags = "short")
> dy1.pp.ts <- ur.pp(diff(y1), type = "Z-tau",
+     model = "trend", lags = "short")
> plot(y1.pp.ts)
```

R Output

	Statistic	1pct	5pct	10pct
y_1	−2.04	−4.00	−3.43	−3.14
Δy_1	−7.19	−4.00	−3.43	−3.14

Table: PP-test results

R Output

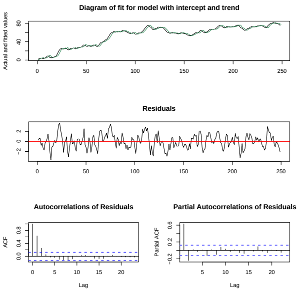


Figure: Residual plot of $y1$ PP-regression

Note:

Same asymptotic distribution as ADF-Tests.

Autoregressive unit root tests

Remarks

- ADF and PP test are asymptotically equivalent.
- PP has better small sample properties than ADF.
- Both have low power against $I(0)$ alternatives that are close to being $I(1)$ processes.
- Power of the tests diminishes as deterministic terms are added to the test regression.

Efficient unit root tests

Elliot, Rothenberg & Stock, I

Model

$$y_t = d_t + u_t \;, \tag{21}$$

$$u_t = a u_{t-1} + v_t \tag{22}$$

Test Statistics

- Point optimal test:

$$P_T = \frac{S(a = \bar{a}) - \bar{a} S(a = 1)}{\hat{\omega}^2} \;, \tag{23}$$

- DF-GLS test:

$$\Delta y_t^d = \alpha_0 y_{t-1}^d + \alpha_1 \Delta y_{t-1}^d + \ldots + \alpha_p \Delta y_{t-p}^d + \varepsilon_t \tag{24}$$

Efficient unit root tests

Elliot, Rothenberg & Stock, II

R Resources

- Function ur.ers in package urca.
- Function urersTest in package fSeries.

Literature

- Elliot, G., T.J. Rothenberg and J.H. Stock, Efficient Tests for an Autoregressive Time Series with a Unit Root, *Econometrica*, 64 (1996), 813–836.

Efficient unit root tests

Elliot, Rothenberg & Stock, III

R code

```
> library(urca)
> set.seed(12345)
> u.ar1 <- arima.sim(
+    list(ar = 0.99), n = 250)
> TD <- 5.0 + 0.7 * seq(250)
> y1.ni <- cumsum(u.ar1) + TD
> y1.ers <- ur.ers(y1.ni, type = "P-test",
+    model = "trend", lag = 1)
> y1.adf <- ur.df(y1.ni, type = "trend")
```

R Output

	Statistic	1pct	5pct	10pct
ERS	33.80	3.96	5.62	6.89
ADF	−1.40	−3.99	−3.43	−3.13

Table: ERS / ADF-tests

R Output

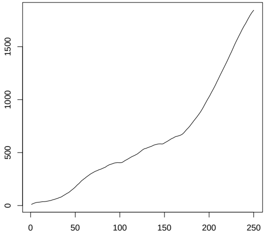


Figure: Near $I(1)$ process

Unit Root Tests, other

Schmidt & Phillips, I

- Problem of DF-type tests: nuisance parameters, *i.e.*, the coefficients of the deterministic regressors, are either not defined or have a different interpretation under the alternative hypothesis of stationarity.
- Solution: LM-type test, that has the same set of nuisance parameters under both the null and alternative hypothesis.
- Higher polynomials than a linear trend are allowed.

Unit Root Tests, other

Schmidt & Phillips, II

Model

$$y_t = \alpha + Z_t\delta + x_t \qquad \text{with} \qquad x_t = \pi x_{t-1} + \varepsilon_t \tag{25}$$

Test Regression

$$\Delta y_t = \Delta Z_t \gamma + \phi \tilde{S}_{t-1} + v_t \tag{26}$$

Test Statistics

$$Z(\rho) = \frac{\tilde{\rho}}{\hat{\omega}^2} = \frac{T\tilde{\phi}}{\hat{\omega}^2} \tag{27}$$

$$Z(\tau)_{\phi=0} = \frac{\tilde{\tau}}{\hat{\omega}^2} \tag{28}$$

Unit Root Tests, other
Schmidt & Phillips, III

R Resources

- Function ur.sp in package urca.
- Function urspTest in package fSeries.

Literature

- Schmidt, P. and P.C.B. Phillips, LM Test for a Unit Root in the Presence of Deterministic Trends, *Oxford Bulletin of Economics and Statistics*, 54(3) (1992), 257-287.

Unit Root Tests, other

Schmidt & Phillips, IV

R code

```
> set.seed(12345)
> y1 <- cumsum(rnorm(250))
> TD <- 5.0 + 0.7 * seq(250) + 0.1 * seq(250)^2
> y1.d <- y1 + TD
> plot.ts(y1.d, xlab = "", ylab = "")
> y1.d.sp <- ur.sp(y1.d, type = "tau",
+     pol.deg = 2, signif = 0.05)
```

R Output

	Statistic	1pct	5pct	10pct
$Z(\tau)$	−2.53	−4.08	−3.55	−3.28
$Z(\rho)$	−12.70	−32.40	−24.80	−21.00

Table: S & P tests

R Output

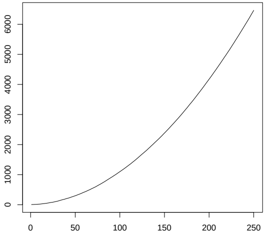


Figure: I(1)-process with polynomial trend

Unit Root Tests, other

Zivot & Andrews, I

- Problem: Difficult to statistically distinguish between an $I(1)$–series from a stable $I(0)$ that is contaminated by a structrual shift.
- If break point is known: Perron and Perron & Vogelsang tests.
- But risk of data mining if break point is exogenously determined.
- Solution: Endogenously determine potential break point: Zivot & Andrews test.

Unit Root Tests, other
Zivot & Andrews, II

Test Statistic

$$t_{\hat{\alpha}^i}[\hat{\lambda}^i_{\text{inf}}] = \inf_{\lambda \in \Delta} t_{\hat{\alpha}^i}(\lambda) \quad \text{for} \quad i = A, B, C \;, \tag{29}$$

A, B, C refer to models that allow for unknown breaks in the intercept and/or trend. The test statistic is the Student t ratio $t_{\hat{\alpha}^i}(\lambda)$ for $i = A, B, C$.

Unit Root Tests, other
Zivot & Andrews, III

R Resources

- Function ur.za in package urca.
- Function urzaTest in package fSeries.

Literature

- Zivot, E. and D.W.K. Andrews, Further Evidence on the Great Crash, the Oil-Price Shock, and the Unit-Root Hypothesis, *Journal of Business & Economic Statistics*, 10(3) (1992), 251-270.
- Perron, P., The Great Crash, the Oil Price Shock, and the Unit Root Hypothesis, *Econometrica*, 57(6) (1989), 1361–1401.
- Perron, P., Testing for a Unit Root in a Time Series With a Changing Mean, *Journal of Business & Economic Statistics*, 8(2) (1990), 153–162.
- Perron, P. and T.J. Vogelsang, Testing for a unit root in a time series with a changing mean: corrections and extensions, *Journal of Business & Economic Statistics*, 10 (1992), 467–470.
- Perron, P., Erratum: The Great Crash, the Oil Price Shock and the Unit Root Hypothesis, *Econometrica*, 61(1) (1993), 248–249.

Unit Root Tests, other

Zivot & Andrews, IV

R code

```
> set.seed(12345)
> u.ar2 <- arima.sim(list(ar = c(0.8, -0.2)),
+    n = 250)
> TD1 <- 5 + 0.3 * seq(100)
> TD2 <- 35 + 0.8 * seq(150)
> TD <- c(TD1, TD2)
> y1.break <- u.ar2 + TD
> plot.ts(y1.break, xlab = "", ylab = "")
> y1.break.za <- ur.za(y1.break,
+    model = "trend", lag = 2)
> plot(y1.break.za)
> y1.break.df <- ur.df(y1.break,
+    type = "trend", lags = 2)
```

R Output

	Statistic	1pct	5pct	10pct
ZA	−7.72	−4.93	−4.42	−4.11
ADF	−1.80	−3.99	−3.43	−3.13

Table: Z & A and ADF tests

R Output

Figure: Plot of Statistic

Stationarity Tests

KPSS, I

Model

$$y_t = \beta' D_t + \mu_t + u_t \ , \ u_t \sim I(0) \tag{30}$$

$$\mu_t = \mu_{t-1} + \varepsilon_t \ , \ \varepsilon_t \sim WN(0, \sigma^2) \tag{31}$$

Hypothesis

$$H_0 : \ \sigma_\varepsilon^2 = 0 \quad \text{and} \quad H_1 : \ \sigma_\varepsilon^2 > 0 \tag{32}$$

Test Statistic

$$LM = \frac{T^{-2} \sum_{t=1}^{T} S_t^2}{\hat{\lambda}^2} \tag{33}$$

Stationarity Tests
KPSS, II

R Resources

- Function ur.kpss in package urca.
- Function urkpssTest in package fSeries.
- Function kpss.test in package tseries.
- Function KPSS.test and KPSS.rectest in package uroot.

Literature

- Kwiatkowski, D., P.C.B. Phillips, P. Schmidt and Y. Shin, Testing the Null Hypothesis of Stationarity Against the Alternative of a Unit Root, *Journal of Econometrics*, 54 (1992), 159–178.

Stationarity Tests

KPSS, III

R code

```
> set.seed(12345)
> u.ar2 <- arima.sim(list(ar = c(0.8, -0.2)),
+     n = 250)
> TD1 <- 5 + 0.3 * seq(250)
> TD2 <- rep(3, 250)
> y1.td1 <-  u.ar2 + TD1
> y1.td2 <-  u.ar2 + TD2
> y2.rw <- cumsum(rnorm(250))
> y1td1.kpss <- ur.kpss(y1.td1, type = "tau")
> y1td2.kpss <- ur.kpss(y1.td2, type = "mu")
> y2rw.kpss <- ur.kpss(y2.rw, type = "mu")
```

R Output

	Statistic	1pct	5pct	10pct
I(0) trd.	0.05	0.12	0.15	0.22
I(0) const	0.30	0.35	0.46	0.74
I(1)	3.21	0.35	0.46	0.74

Table: KPSS tests

R Output

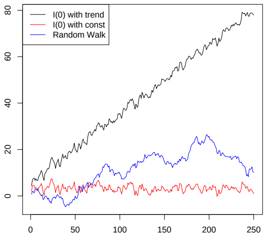


Figure: Generated Series

Multivariate Time Series

Overview

- Stationary VAR(p)-models
- SVAR models
- Cointegration: Concept, models and methods
- SVEC models

VAR
Definition

A VAR(p)-process is defined as:

$$\mathbf{y}_t = A_1\mathbf{y}_{t-1} + \ldots + A_p\mathbf{y}_{t-p} + CD_t + \mathbf{u}_t \quad , \tag{34}$$

- A_i: coefficient matrices for $i = 1, \ldots, p$
- $\mathbf{u}_t$: K-dimensional white noise process with time invariant positive definite covariance matrix $E(\mathbf{u}_t\mathbf{u}_t') = \Sigma_{\mathbf{u}}$.
- C: coefficient matrix of potentially deterministic regressors.
- D_t: column vector holding the appropriate deterministic regressors.

VAR

Companion Form

A VAR(p)-process as VAR(1):

$$\xi_t = A\xi_{t-1} + \mathbf{v}_t \text{ , with} \tag{35}$$

$$\xi_t = \begin{bmatrix} \mathbf{y}_t \\ \vdots \\ \mathbf{y}_{t-p+1} \end{bmatrix} \text{ , } A = \begin{bmatrix} A_1 & A_2 & \cdots & A_{p-1} & A_p \\ I & 0 & \cdots & 0 & 0 \\ 0 & I & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & I & 0 \end{bmatrix} \text{ , } \mathbf{v}_t = \begin{bmatrix} \mathbf{u}_t \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

If the moduli of the *eigenvalues* of A are less than one, then the VAR(p)-process is stable.

VAR
Wold Decomposition

$$\mathbf{y}_t = \Phi_0 \mathbf{u}_t + \Phi_1 \mathbf{u}_{t-1} + \Phi_2 \mathbf{u}_{t-2} + \ldots \quad , \tag{36}$$

with $\Phi_0 = I_K$ and the Φ_s matrices can be computed recursively according to:

$$\Phi_s = \sum_{j=1}^{s} \Phi_{s-j} A_j \quad \text{for} \quad s = 1, 2, \ldots \quad , \tag{37}$$

whereby $\Phi_0 = I_K$ and $A_j = 0$ for $j > p$.

VAR
Empirical Lag Order Selection

$$\mathrm{AIC}(p) = \log\det(\tilde{\Sigma}_u(p)) + \frac{2}{T}\, pK^2 \quad , \tag{38a}$$

$$\mathrm{HQ}(p) = \log\det(\tilde{\Sigma}_u(p)) + \frac{2\log(\log(T))}{T}\, pK^2 \quad , \tag{38b}$$

$$\mathrm{SC}(p) = \log\det(\tilde{\Sigma}_u(p)) + \frac{\log(T)}{T}\, pK^2 \quad \text{or,} \tag{38c}$$

$$\mathrm{FPE}(p) = \left(\frac{T+p^*}{T-p^*}\right)^K \det(\tilde{\Sigma}_u(p)) \quad , \tag{38d}$$

with $\tilde{\Sigma}_u(p) = T^{-1}\sum_{t=1}^{T} \hat{\mathbf{u}}_t \hat{\mathbf{u}}_t'$ and p^* is the total number of the parameters in each equation and p assigns the lag order.

VAR

Simulation / Estimation, I

Example of simulated VAR(2):

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_t = \begin{bmatrix} 0.5 & 0.2 \\ -0.2 & -0.5 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_{t-1} + \begin{bmatrix} -0.3 & -0.7 \\ -0.1 & 0.3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_{t-2} + \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}_t$$

- Simulation of VAR-processes with packages dse1 and mAr
- Estimation of VAR-processes with packages dse1, mAr and vars.

VAR

Simulation / Estimation, II

R code

```
> library(dse1)
> library(vars)
> Apoly   <- array(c(1.0, -0.5, 0.3, 0,
+    0.2, 0.1, 0, -0.2, 0.7, 1, 0.5, -0.3) ,
+    c(3, 2, 2))
> B <- diag(2)
> var2  <- ARMA(A = Apoly, B = B)
> varsim <- simulate(var2, sampleT = 500,
+    noise = list(w = matrix(rnorm(1000),
+ nrow = 500, ncol = 2)),
+    rng = list(seed = c(123456)))
> vardat <- matrix(varsim$output,
+    nrow = 500, ncol = 2)
> colnames(vardat) <- c("y1", "y2")
> infocrit <- VARselect(vardat, lag.max = 3,
+     type = "const")
> varsimest <- VAR(vardat, p = 2,
+     type = "none")
> roots <- roots(varsimest)
```

R Output

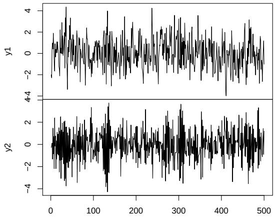


Figure: Generated VAR(2)

VAR

Simulation / Estimation, II

	Estimate	Std. Error	t value	Pr(>|t|)
y1.l1	0.4954	0.0366	13.55	0.0000
y2.l1	0.1466	0.0404	3.63	0.0003
y1.l2	−0.2788	0.0364	−7.66	0.0000
y2.l2	−0.7570	0.0455	−16.64	0.0000

Table: VAR result for y_1

	Estimate	Std. Error	t value	Pr(>|t|)
y1.l1	−0.2076	0.0375	−5.54	0.0000
y2.l1	−0.4899	0.0414	−11.83	0.0000
y1.l2	−0.1144	0.0373	−3.07	0.0023
y2.l2	0.3375	0.0467	7.23	0.0000

Table: VAR result for y_2

VAR

Simulation / Estimation, III

	1	2	3
AIC(n)	0.60	0.01	0.01
HQ(n)	0.62	0.04	0.05
SC(n)	0.64	0.08	0.11
FPE(n)	1.84	1.02	1.02

Table: Empirical Lag Selection

	1	2	3	4
Eigen values	0.84	0.66	0.57	0.57

Table: Stability

VAR

Diagnostic testing, I

Statistical Tests

- Serial correlation: Portmanteau Test, Breusch & Godfrey
- Heteroskedasticity: ARCH
- Normality: Jarque & Bera, Skewness, Kurtosis
- Structural Stability: EFP, CUSUM, CUSUM-of-Squares, Fluctuation Test *etc.*

R Resources

- Functions serial, arch, normality and stability in package vars.
- Function checkResiduals in package dse1.

VAR

Diagnostic testing, II

R code

```
> var2c.serial <- serial(varsimest)
> var2c.arch <- arch(varsimest)
> var2c.norm <- normality(varsimest)
> plot(var2c.serial)
```

R Output

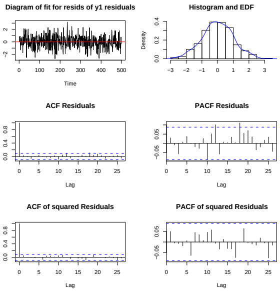


Figure: Residuals of y1

R Output

	Statistic	p-value
PT y1	52.673	0.602
PT y2	53.632	0.565
LMh	18.953	0.525
LMFh	0.938	0.538
ARCH y1	9.298	0.901
ARCH y2	7.480	0.963
ARCH VAR	45.005	0.472
JB y1	0.018	0.991
JB y2	1.354	0.508
JB VAR	1.369	0.850
Kurtosis	0.029	0.986
Skewness	1.340	0.512

Table: Diagnostic tests of VAR(2)

VAR

Diagnostic testing, III

R code

```
> reccusum <- stability(varsimest,
+     type = "Rec-CUSUM")
> fluctuation <- stability(varsimest,
+     type = "fluctuation")
```

R Output

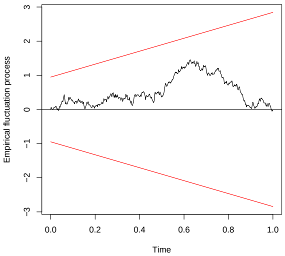


Figure: CUSUM Test y1

R Output

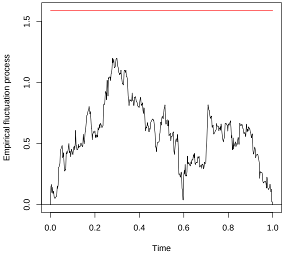


Figure: Fluctuation Test y2

VAR
Causality, I

Granger-causality

$$\begin{bmatrix} \mathbf{y}_{1t} \\ \mathbf{y}_{2t} \end{bmatrix} = \sum_{i=1}^{p} \begin{bmatrix} \alpha_{11,i} & \alpha_{12,i} \\ \alpha_{21,i} & \alpha_{22,i} \end{bmatrix} \begin{bmatrix} \mathbf{y}_{1,t-i} \\ \mathbf{y}_{2,t-i} \end{bmatrix} + CD_t + \begin{bmatrix} \mathbf{u}_{1t} \\ \mathbf{u}_{2t} \end{bmatrix} , \qquad (39)$$

- Null hypothesis: subvector $\mathbf{y}_{1t}$ does not Granger-cause $\mathbf{y}_{2t}$, is defined as $\alpha_{21,i} = 0$ for $i = 1, 2, \ldots, p$
- Alternative hypothesis is: $\exists\, \alpha_{21,i} \neq 0$ for $i = 1, 2, \ldots, p$.
- Statistic: $F(pK_1K_2, KT - n^*)$, with n^* equal to the total number of parameters in the above VAR(p)-process, including deterministic regressors.

VAR

Causality, II

Instantaneous-causality

The null hypothesis for non-instantaneous causality is defined as: $H_0 : C\sigma = 0$, where C is a $(N \times K(K+1)/2)$ matrix of rank N selecting the relevant co-variances of $\mathbf{u}_{1t}$ and $\mathbf{u}_{2t}$; $\tilde{\sigma} = vech(\tilde{\Sigma}_u)$. The Wald statistic is defined as:

$$\lambda_W = T\tilde{\sigma}' C' [2CD_K^{+}(\tilde{\Sigma}_u \otimes \tilde{\Sigma}_u)D_K^{+'} C']^{-1} C\tilde{\sigma} \ , \qquad (40)$$

hereby assigning the Moore-Penrose inverse of the duplication matrix D_K with D_K^{+} and $\tilde{\Sigma}_u = \frac{1}{T}\Sigma_{t=1}^{T}\hat{\mathbf{u}}_t\hat{\mathbf{u}}_t'$. The test statistic λ_W is asymptotically distributed as $\chi^2(N)$.

VAR
Causality, III

R Resources

- Function causality in package vars.

R Code

```
> var.causal <- causality(varsimest, cause = "y2")
```

R Output

	Statistic	p-value
Granger	254.53	0.00
Instant	0.00	0.96

Table: Causality tests

VAR
Prediction, I

- Recursive predictions according to:

$$\mathbf{y}_{T+1|T} = A_1 \mathbf{y}_T + \ldots + A_p \mathbf{y}_{T+1-p} + C D_{T+1} \qquad (41)$$

- Forecast error covariance matrix:

$$Cov\left(\begin{bmatrix} \mathbf{y}_{T+1} - \mathbf{y}_{T+1|T} \\ \vdots \\ \mathbf{y}_{T+h} - \mathbf{y}_{T+h|T} \end{bmatrix}\right) = \begin{bmatrix} I & 0 & \cdots & 0 \\ \Phi_1 & I & & 0 \\ \vdots & & \ddots & 0 \\ \Phi_{h-1} & \Phi_{h-2} & \ldots & I \end{bmatrix} (\Sigma_{\mathbf{u}} \otimes I_h) \begin{bmatrix} I & 0 & \cdots & 0 \\ \Phi_1 & I & & 0 \\ \vdots & & \ddots & 0 \\ \Phi_{h-1} & \Phi_{h-2} & \ldots & I \end{bmatrix}'$$

and the matrices Φ_i are the coefficient matrices of the Wold moving average representation of a stable VAR(p)-process.

VAR

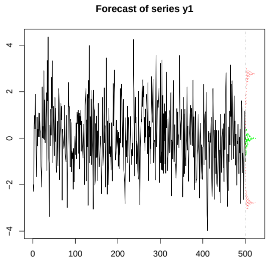
Prediction, II

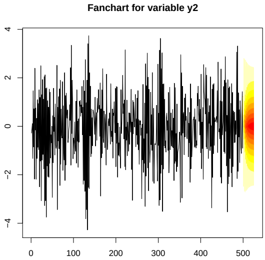
R Resources

- Method predict in package vars for objects of class varest.

R Code

```
> predictions <- predict(varsimest, n.ahead = 25)
> plot(predictions)
> fanchart(predictions)
```





VAR

Impulse Response Function, I

- Based on Wold decomposition of a stable VAR(p).
- Investigate the dynamic interactions between the endogenous variables.
- The $(i, j)th$ coefficients of the matrices Φ_s are thereby interpreted as the expected response of variable $y_{i,t+s}$ to a unit change in variable y_{jt}.
- Can be cumulated through time $s = 1, 2, \ldots$: cumulated impact of a unit change in variable j to the variable i at time s.
- Orthogonalised impulse reponses: underlying shocks are less likely to occur in isolation (derived from Choleski Decomposition).

VAR

Impulse Response Function, II

- Orthogonalised impulse responses: $\Sigma_{\mathbf{u}} = PP'$ with P being a lower triangular.
- Transformed moving average representation:

$$\mathbf{y}_t = \Psi_0 \varepsilon_t + \Psi_1 \varepsilon_{t-1} + \dots \; , \tag{42}$$

  with $\varepsilon_t = P^{-1}\mathbf{u}_t$ and $\Psi_i = \Phi_i P$ for $i = 0, 1, 2, \dots$ and $\Psi_0 = P$.
- Confidence bands by bootstrapping.

R Resources

- Methods irf, Phi and Psi in package vars.

VAR

Impulse Response Function, III

R Code

```
> irf.y1 <- irf(varsimest, impulse = "y1", response = "y2", n.ahead = 10, ortho = FALSE,
+     cumulative = FALSE, boot = TRUE, seed = 12345)
> irf.y2 <- irf(varsimest, impulse = "y2", response = "y1", n.ahead = 10, ortho = FALSE,
+     cumulative = FALSE, boot = TRUE, seed = 12345)
> plot(irf.y1)
> plot(irf.y2)
```

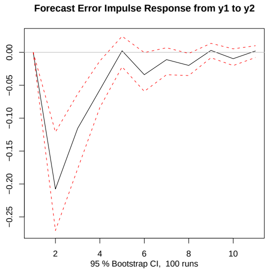


Figure: IRF of y1

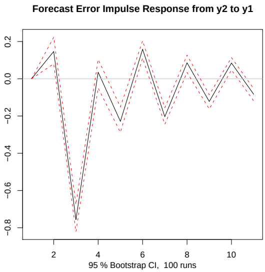


Figure: IRF of y2

VAR

Forecast Error Variance Decomposition, I

- FEVD: based on orthogonalised impulse response coefficient matrices Ψ_n
- Analyse the contribution of variable j to the h-step forecast error variance of variable k.
- Elementwise squared orthogonalised impulse reponses are divided by the variance of the forecast error variance, $\sigma_k^2(h)$:

$$\omega_{kj}(h) = (\psi_{kj,0}^2 + \ldots + \psi_{kj,h-1}^2)/\sigma_k^2(h) \, . \tag{43}$$

R Resources

- Method fevd in package vars.

VAR

Forecast Error Variance Decomposition, II

R Code

```
> fevd.var2 <- fevd(varsimest, n.ahead = 10)
> plot(fevd.var2)
```

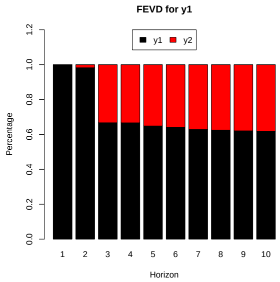


Figure: FEVD of y1

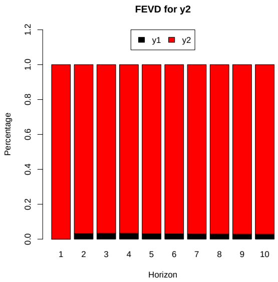


Figure: IRF of y2

SVAR
Models, I

- VAR can be viewed as a reduced form model.
- SVAR is its structural form and is defined as:

$$A\mathbf{y}_t = A_1^*\mathbf{y}_{t-1} + \ldots + A_p^*\mathbf{y}_{t-p} + B\varepsilon_t \; . \tag{44}$$

- Structural errors: ε_t are white noise.
- Coefficient matrices: A_i^* for $i = 1, \ldots, p$, are structural coefficients that might differ from their reduced form counterparts.
- Use of SVAR: identify shocks and trace these out by IRF and/or FEVD through imposing restrictions on the matrices A and/or B.

SVAR

Models, II

- Reduced form residuals can be retrieved from a SVAR-model by $\mathbf{u}_t = A^{-1}B\varepsilon_t$ and its variance-covariance matrix by $\Sigma_{\mathbf{u}} = A^{-1}BB'A^{-1'}$.
- A model: B ist set to I_K (minimum number of restrictions for identification is $K(K-1)/2$).
- B model: A ist set to I_K (minimum number of restrictions for identification is $K(K-1)/2$).
- AB model: restrictions can be placed on both matrices (minimum number of restrictions for identification is $K^2 + K(K-1)/2$).

SVAR
Estimation

- Directly, by minimising the negative of the Log-Likelihood:

$$\ln L_c(A, B) = -\frac{KT}{2}\ln(2\pi) + \frac{T}{2}\ln|A|^2 - \frac{T}{2}\ln|B|^2 - \frac{T}{2} tr(A'B'^{-1}B^{-1}A\tilde{\Sigma}_u) , \qquad (45)$$

- Scoring algorithm proposed by Amisano and Giannini (1997).
- Overidentification test:

$$LR = T(\log det(\tilde{\Sigma}^r_{\mathbf{u}}) - \log det(\tilde{\Sigma}_{\mathbf{u}})) \qquad (46)$$

with $\tilde{\Sigma}_{\mathbf{u}}$: reduced form variance-covariance matrix and $\tilde{\Sigma}^r_{\mathbf{u}}$: restricted structural form estimation.

R Resources

- Functions BQ, SVAR and SVAR2 in package vars.

SVAR

A-Model, I

The Model

$$\begin{bmatrix} 1.0 & 0.7 \\ -0.8 & 1.0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_t = \begin{bmatrix} 0.5 & 0.2 \\ -0.2 & -0.5 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_{t-1} + \begin{bmatrix} -0.3 & -0.7 \\ -0.1 & 0.3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_{t-2} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \end{bmatrix}_t$$

Restrictions

Restrictions for A matrix in explicit form:

$$\text{vec}\,(A) = R_a \gamma_a + r_a$$

$$\begin{bmatrix} 1 \\ \alpha_{21} \\ \alpha_{12} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

SVAR

A-Model, II

R Code

```
> Apoly   <- array(
+    c(1.0, -0.5, 0.3, 0.8,
+    0.2, 0.1, -0.7, -0.2,
+    0.7, 1, 0.5, -0.3) ,
+    c(3, 2, 2))
> B <- diag(2)
> svarA  <- ARMA(A = Apoly, B = B)
> svarsim <- simulate(svarA,
+    sampleT = 500, rng = list(seed = c(123)))
> svardat <- matrix(svarsim$output,
+    nrow = 500, ncol = 2)
> colnames(svardat) <- c("y1", "y2")
> Ra <- matrix(c(0, 1, 0, 0, 0, 0, 1, 0),
+    nrow = 4, ncol = 2)
> ra <- c(diag(2))
> varest <- VAR(svardat, p = 2, type = "none")
> svara <- SVAR2(varest, Ra = Ra, ra = ra)
```

R Output

	y1	y2
y1	1.00	0.75
y2	−0.80	1.00

Table: A matrix

	y1	y2
y1	0.00	0.05
y2	0.06	0.00

Table: S.E. of A

SVAR
B-Model, I

The Model

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_t = \begin{bmatrix} 0.5 & 0.2 \\ -0.2 & -0.5 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_{t-1} + \begin{bmatrix} -0.3 & -0.7 \\ -0.1 & 0.3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_{t-2} + \begin{bmatrix} 1.0 & 0.0 \\ -0.8 & 1.0 \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \end{bmatrix}_t$$

Restrictions

Restrictions for B matrix in explicit form:

$$\begin{aligned} \text{vec}\,(B) &= R_b \gamma_b + r_b \\ \begin{bmatrix} 1 \\ \beta_{21} \\ 0 \\ 1 \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} \gamma_1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned}$$

SVAR

B-Model, II

R Code

```
> Apoly   <- array(
+    c(1.0, -0.5, 0.3, 0,
+    0.2, 0.1, 0.0, -0.2,
+    0.7, 1.0, 0.5, -0.3) ,
+    c(3, 2, 2))
> B <- diag(2)
> B[2, 1] <- -0.8
> svarB  <- ARMA(A = Apoly, B = B)
> svarsim <- simulate(svarB, sampleT = 500,
+    rng = list(seed = c(123456)))
> svardat <- matrix(svarsim$output,
+    nrow = 500, ncol = 2)
> colnames(svardat) <- c("y1", "y2")
> Rb <- matrix(c(0, 1, 0, 0),
+    nrow = 4, ncol = 1)
> rb <- c(diag(2))
> varest <- VAR(svardat, p = 2, type = "none")
> svarb <- SVAR2(varest, Rb = Rb, rb = rb)
```

R Output

	y1	y2
y1	1.00	0.00
y2	−0.84	1.00

Table: B matrix

	y1	y2
y1	0.00	0.00
y2	0.04	0.00

Table: S.E. of B

SVAR

Impulse Response Analysis, I

- Impulse response coefficients for SVAR:

$$\Theta_i = \Phi_i A^{-1} B \text{ for } i = 1, \ldots, n. \tag{47}$$

- Orthogonalisation not meaningingful, hence not implemented

R Resources

- Method irf in package vars.

SVAR

Impulse Response Analysis, II

R Code

```
> irf.y1 <- irf(svara, impulse = "y1", response = "y2", n.ahead = 10,
+     cumulative = FALSE, boot = FALSE, seed = 12345)
> irf.y2 <- irf(svara, impulse = "y2", response = "y1", n.ahead = 10,
+     cumulative = FALSE, boot = FALSE, seed = 12345)
> plot(irf.y1)
> plot(irf.y2)
```

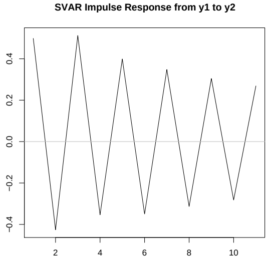


Figure: IRF of y1

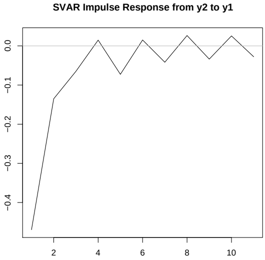


Figure: IRF of y2

SVAR

Forecast Error Variance Decomposition, I

- Forecast errors of $y_{T+h|T}$ are derived from the impulse responses of SVAR and the derivation to the forecast error variance decomposition is similar to the one outlined for VARs.

R Resources

- Method fevd in package vars.

SVAR

Forecast Error Variance Decomposition, II

R Code

```
> fevd.svarb <- fevd(svarb, n.ahead = 10)
> plot(fevd.svarb)
```

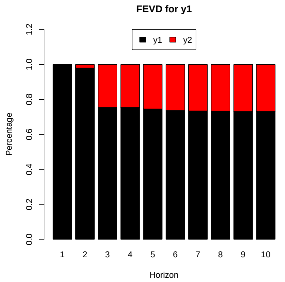


Figure: FEVD of y1

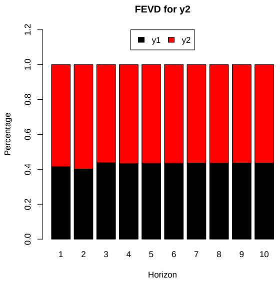


Figure: IRF of y2

Cointegration
Spurious Regression, I

Problem

- I(1) variables that are not cointegrated are regressed on each other.
- Slope coefficients do not converge in probability to zero.
- t-statistics diverge to $\pm\infty$ as $T \to \infty$.
- R^2 tends to unity with $T \to \infty$.
- Rule-of-thumb: Be cautious when R^2 is greater than DW statistic.

Literature

- Phillips, P.C.B., Understanding Spurious Regression in Econometrics, *Journal of Econometrics*, 33 (1986), 311–340.

Cointegration

Spurious Regression, II

R Code

```
> library(lmtest)
> set.seed(54321)
> e1 <- rnorm(500)
> e2 <- rnorm(500)
> y1 <- cumsum(e1)
> y2 <- cumsum(e2)
> sr.reg1 <- lm(y1 ~ y2)
> sr.dw <- dwtest(sr.reg1)
> sr.reg2 <- lm(diff(y1) ~ diff(y2))
```

R Output

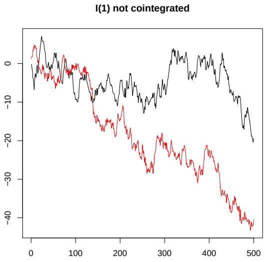


Figure: Spurious relation

Cointegration
Spurious Regression, III

R Output

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	−1.9532	0.3696	−5.28	0.0000
y2	0.1427	0.0165	8.63	0.0000

Table: Level regression

For the level regresion the R^2 is 0.13 and the DW statistic is 0.051.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	−0.0434	0.0456	−0.95	0.3413
diff(y2)	−0.0588	0.0453	−1.30	0.1942

Table: Difference regression

Cointegration

Definition, I

Definition

The components of the vector $\mathbf{y}_t$ are said to be cointegrated of order d, b, denoted $\mathbf{y}_t \sim CI(d, b)$, if (a) all components of $\mathbf{y}_t$ are $I(d)$; and (b) a vector $\beta(\neq 0)$ exists so that $z_t = \beta' \mathbf{y}_t \sim I(d - b)$, $b > 0$. The vector β is called the cointegrating vector.

Common Trends

If the $(n \times 1)$ vector $\mathbf{y}_t$ is cointegrated with $0 < r < n$ cointegrating vectors, then there are $n - r$ common $I(1)$ stochastic trends.

Literature

- Engle, R.F. and C.W.J. Granger, Co-Integration and Error Correction: Representation, Estimation and Testing, *Econometrica*, 55 (1987), 251–276.

Cointegration

Definition, II

R Code

```
> set.seed(12345)
> e1 <- rnorm(250, mean = 0, sd = 0.5)
> e2 <- rnorm(250, mean = 0, sd = 0.5)
> u.ar3 <- arima.sim(model =
+    list(ar = c(0.6, -0.2, 0.1)), n = 250,
+    innov = e1)
> y2 <- cumsum(e2)
> y1 <- u.ar3 + 0.5*y2
> ymax <- max(c(y1, y2))
> ymin <- min(c(y1, y2))
> layout(matrix(1:2, nrow = 2, ncol = 1))
> plot(y1, xlab = "", ylab = "", ylim =
+    c(ymin, ymax), main =
+    "Cointegrated System")
> lines(y2, col = "green")
> plot(u.ar3, ylab = "", xlab = "", main =
+    "Cointegrating Residuals")
> abline(h = 0, col = "red")
```

R Output

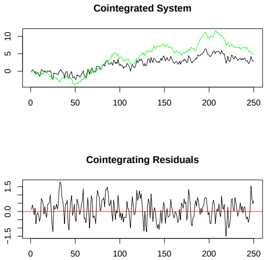


Figure: Bivariate Cointegration

Cointegration

Error Correction Model

Definition

Bivariate $I(1)$ vector $\mathbf{y}_t = (y_{1t}, y_{2t})'$ with cointegrating vector $\beta = (1, -\beta_2)'$, hence $\beta'\mathbf{y}_t = y_{1t} - \beta_2 y_{2t} \sim I(0)$, then an ECM exists in the form of:

$$\Delta y_{1,t} = \alpha_1 + \gamma_1(y_{1,t-1} - \beta_2 y_{2,t-1}) + \sum_{i=1}^{K} \psi_{1,i} \Delta y_{1,t-i} + \sum_{i=1}^{L} \psi_{2,i} \Delta y_{2,t-i} + \varepsilon_{1,t} \ ,$$

$$\Delta y_{2,t} = \alpha_2 + \gamma_2(y_{1,t-1} - \beta_2 y_{2,t-1})_{t-1} + \sum_{i=1}^{K} \xi_{1,i} \Delta y_{1,t-i} + \sum_{i=1}^{L} \xi_{2,i} \Delta y_{2,t-i} + \varepsilon_{2,t} \ .$$

Cointegration

Engle & Granger Two-Step Procedure, I

1. Estimate long-run relationship, *i.e.*, regression in levels and test residuals for $I(0)$.
2. Take residuals from first step and use it in ECM regression.

- Warschau: If ADF-test is used, you need CV provided in Engle & Yoo.
- OLS-estimator is super consistent, convergence T.
- However, OLS can be biased in small samples!

Literature

- Engle, R. and B. Yoo, Forecasting and Testing in Co-Integrated Systems, *Journal of Econometrics*, 35 (1987), 143–159.

Cointegration

Engle & Granger Two-Step Procedure, II

R Code

```
> library(dynlm)
> lr <- lm(y1 ~ y2)
> ect <- resid(lr)[1:249]
> dy1 <- diff(y1)
> dy2 <- diff(y2)
> ecmdat <- cbind(dy1, dy2, ect)
> ecm <- dynlm(dy1 ~ L(ect, 1) + L(dy1, 1)
+      + L(dy2, 1) , data = ecmdat)
```

R Output

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0064	0.0376	0.17	0.8646
L(ect, 1)	−0.6216	0.0725	−8.58	0.0000
L(dy1, 1)	−0.4235	0.0703	−6.03	0.0000
L(dy2, 1)	0.3171	0.0911	3.48	0.0006

Table: Results for ECM

Cointegration
Phillips & Ouliaris, I

- Residual-based tests: Variance Ratio Test & Trace Statistic.
- Based on regression:

$$\mathbf{z}_t = \Pi \mathbf{z}_{t-1} + \xi_t \ , \tag{48}$$

where $\mathbf{z}_t$ is partioned as $\mathbf{z}_t = (y_t, \mathbf{x}_t')$ with a dimension of $\mathbf{x}_t$ equal to $(m = n + 1)$.

- Null hypothesis: Not cointegrated.

Cointegration
Phillips & Ouliaris, II

R Resources

- Function ca.po in package urca.
- Function po.test in package tseries.

Literature

- Phillips, P.C.B. and S. Ouliaris, S., Asymptotic Properties of Residual Based Tests for Cointegration, *Econometrica*, 58 (1) (1990), 165-193.

Cointegration

Phillips & Ouliaris, III

R Code

```
> z <- cbind(y1, y2)
> po.Pu <- ca.po(z, demean = "none", type = "Pu")
> po.Pz <- ca.po(z, demean = "none", type = "Pz")
```

R Output

	Statistic	10pct	5pct	1pct
Pu	167.44	20.39	25.97	38.34
Pz	176.09	33.93	40.82	55.19

Table: Test Statistics

VECM
Definition

- VAR:

$$\mathbf{y}_t = A_1 \mathbf{y}_{t-1} + \ldots + A_p \mathbf{y}_{t-p} + CD_t + \mathbf{u}_t \quad ,$$

- Transitory form of VECM:

$$\begin{aligned} \Delta \mathbf{y}_t &= \Gamma_1 \Delta \mathbf{y}_{t-1} + \ldots + \Gamma_{K-1} \Delta \mathbf{y}_{t-p+1} + \Pi \mathbf{y}_{t-1} + CD_t + \varepsilon_t \ , \\ \Gamma_i &= -(A_{i+1} + \ldots + A_p) \ , \text{ for } i = 1, \ldots, p-1 \ , \\ \Pi &= -(I - A_1 - \cdots - A_p) \ . \end{aligned}$$

- Long-run form of VECM:

$$\begin{aligned} \Delta \mathbf{y}_t &= \Gamma_1 \Delta \mathbf{y}_{t-1} + \ldots + \Gamma_{p-1} \Delta \mathbf{y}_{t-p+1} + \Pi \mathbf{y}_{t-p} + CD_t + \varepsilon_t \ , \\ \Gamma_i &= -(I - A_1 - \ldots - A_i) \ , \text{ for } i = 1, \ldots, p-1 \ , \\ \Pi &= -(I - A_1 - \cdots - A_p) \end{aligned}$$

VECM

The Π matrix

1. $rk(\Pi) = n$, all n combinations must be stationary for balancing: $\mathbf{y}_t$ must be stationary around deterministic components; standard VAR-model in levels.
2. $rk(\Pi) = 0$, no linear combination exists, such that $\Pi\mathbf{y}_{t-1}$ is stationary, except the trivial solution; VAR-model in first differences.
3. $0 < rk(\Pi) = 0 < r < n$, interesting case: $\Pi = \alpha\beta'$ with dimensions $(n \times r)$ and $\beta'\mathbf{y}_{t-1}$ is stationary. Each column of β represents one long-run relationship.

VECM

Example

R Code

```
> set.seed(12345)
> e1 <- rnorm(250, 0, 0.5)
> e2 <- rnorm(250, 0, 0.5)
> e3 <- rnorm(250, 0, 0.5)
> u1.ar1 <- arima.sim(model = list(ar=0.75),
+    innov = e1, n = 250)
> u2.ar1 <- arima.sim(model = list(ar=0.3),
+    innov = e2, n = 250)
> y3 <- cumsum(e3)
> y1 <- 0.8 * y3 + u1.ar1
> y2 <- -0.3 * y3 + u2.ar1
> ymax <- max(c(y1, y2, y3))
> ymin <- min(c(y1, y2, y3))
> plot(y1, ylab = "", xlab = "",
+    ylim = c(ymin, ymax))
> lines(y2, col = "red")
> lines(y3, col = "blue")
```

R Output

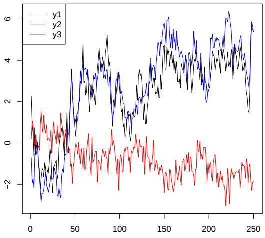


Figure: Simulated VECM

VECM
Inference

- Based on canonical correlations between $\mathbf{y}_t$ and $\Delta\mathbf{y}_t$ with lagged differences.
- Correlations:

$$S_{00} = \frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{u}}_t\hat{\mathbf{u}}_t' \; , \; S_{01} = S_{10} = \sum_{t=1}^{T}\hat{\mathbf{u}}_t\hat{\mathbf{v}}_t' \; , \; S_{11} = \frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{v}}_t\hat{\mathbf{v}}_t'$$

- Eigenvalues:

$$|\lambda S_{11} - S_{10}S_{00}^{-1}S_{01}| = 0$$

- LR-tests: Eigen- and Trace-test.
- Nested Hypothesis: $H(0) \subset \cdots \subset H(r) \subset \cdots \subset H(n)$.

VECM
Resources

R Resources

- Functions ca.jo, cajorls, cajools, cajolst in package urca.
- Hypothesis Testing: alrtest, ablrtest, blrtest, bh5lrtest, bh6lrtest and lttest in package urca.
- Function vec2var in package vars.

Literature

- Johansen, S., Statistical Analysis of Cointegration Vectors, *Journal of Economic Dynamics and Control*, 12 (1988), 231-254.
- Johansen, S. and K. Juselius, Maximum Likelihood Estimation and Inference on Cointegration - with Applications to the Demand for Money, *Oxford Bulletin of Economics and Statistics*, 52(2) (1990), 169-210.
- Johansen, S., Estimation and Hypothesis Testing of Cointegration Vectors in Gaussian Vector Autoregressive Models, *Econometrica*, 59(6) (1991), 1551-1580.

VECM

Estimation, I

R Code

```
> y.mat <- data.frame(y1, y2, y3)
> vecm1 <- ca.jo(y.mat, type = "eigen", spec = "transitory")
> vecm2 <- ca.jo(y.mat, type = "trace", spec = "transitory")
> vecm.r2 <- cajorls(vecm1, r = 2)
```

R Output

	Statistic	10pct	5pct	1pct
r <= 2 |	4.72	2.82	3.96	6.94
r <= 1 |	41.69	12.10	14.04	17.94
r = 0 |	78.17	18.70	20.78	25.52

Table: Maximal Eigenvalue Test

	Statistic	10pct	5pct	1pct
r <= 2 |	4.72	2.82	3.96	6.94
r <= 1 |	46.41	13.34	15.20	19.31
r = 0 |	124.58	26.79	29.51	35.40

Table: Trace Test

VECM
Estimation, II

R Output

	y1.d	y2.d	y3.d
ect1	−0.33	0.06	0.01
ect2	0.09	−0.71	−0.01
constant	0.17	−0.03	0.03
y1.dl1	0.10	−0.04	0.06
y2.dl1	0.05	−0.01	0.05
y3.dl1	−0.15	−0.03	−0.06

Table: VECM with $r = 2$

	ect1	ect2
y1.l1	1.00	0.00
y2.l1	0.00	1.00
y3.l1	−0.73	0.30

Table: Normalised CI-relations

VECM

Prediction, IRF, FEVD, I

- Convert restricted VECM to level-VAR.
- Prediction, IRF, FEVD and diagnostic checking applies likewise to stationary VAR(p)-models as shown in previous slides.

R Resources

- Function vec2var in package vars.

VECM

Prediction, IRF, FEVD, II

R Code

```
> vecm.level <- vec2var(vecm1, r = 2)
> vecm.pred <- predict(vecm.level,
+     n.ahead = 10)
> fanchart(vecm.pred)
> vecm.irf <- irf(vecm.level, impulse = 'y3',
+     response = 'y1', boot = FALSE)
> vecm.fevd <- fevd(vecm.level)
> vecm.norm <- normality(vecm.level)
> vecm.arch <- arch(vecm.level)
> vecm.serial <- serial(vecm.level)
```

R Output

	constant
y1	0.17
y2	−0.03
y3	0.03

Table: Implied Constant

R Output

	y1.l1	y2.l1	y3.l1
y1	0.77	0.14	0.12
y2	0.03	0.28	−0.29
y3	0.07	0.04	0.92

Table: Implied A_1

	y1.l2	y2.l2	y3.l2
y1	−0.10	−0.05	0.15
y2	0.04	0.01	0.03
y3	−0.06	−0.05	0.06

Table: Implied A_2

VECM

Prediction, IRF, FEVD, III

R Output

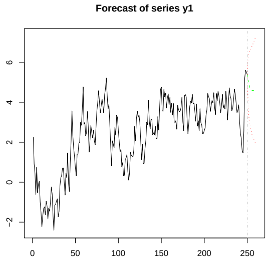


Figure: Prediction of y_1

R Output

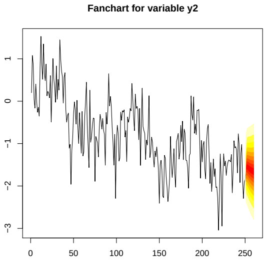


Figure: Fanchart of y_2

VECM

Prediction, IRF, FEVD, IV

R Output

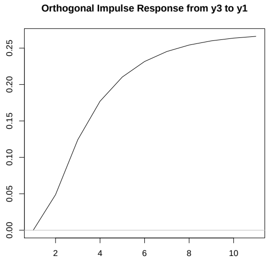


Figure: IRF of y_3 to y_1

R Output

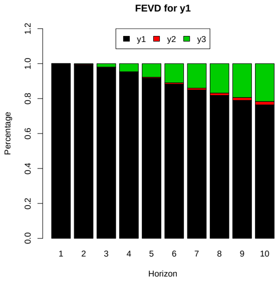


Figure: FEVD of VECM

VECM

Linear Trend Test, I

- Test if linear trend in VAR is existent.
- This corresponds to the inclusion of a constant in the error correction term.
- Statistic is distributed as χ^2 square with $(K - r)$ degrees of freedom.

R Resources

- Function lttest in package urca.

VECM

Linear Trend Test, II

R Code

```
> data(denmark)
> sjd <- as.matrix(denmark[,
+     c("LRM", "LRY", "IBO", "IDE")])
> sjd.vecm <- ca.jo(sjd, constant = TRUE,
+     type = "eigen", K = 2, spec="longrun",
+      season=4)
> lttest.1 <- lttest(sjd.vecm, r=1)
> data(finland)
> sjf <- as.matrix(finland)
> sjf.vecm <- ca.jo(sjf, constant = FALSE,
+       type = "eigen", K=2, spec="longrun",
+       season=4)
> lttest.2 <- lttest(sjf.vecm, r=3)
```

R Output

	Statistic	p-value
Denmark	1.98	0.58
Finland	4.78	0.03

Table: Linear Trend Test

VECM

Restrictions on Loadings, I

- Testing exogenity, *i.e.*, certain variables do not enter into the cointegration relation(s).
- Likelihood ratio test for the hypothesis:

$$\mathcal{H}_4 : \alpha = A\Psi \ , \tag{49}$$

with $(r(K-m))$ degrees of freedom.

R Resources

- Function alrtest in package urca.

VECM

Restrictions on Loadings, II

R Code

```
> data(UKpppuip)
> attach(UKpppuip)
> dat1 <- cbind(p1, p2, e12, i1, i2)
> dat2 <- cbind(doilp0, doilp1)
> H1 <- ca.jo(dat1, K = 2, season = 4,
+     dumvar=dat2)
> A1 <- matrix(c(1,0,0,0,0,
+     0,0,1,0,0,
+     0,0,0,1,0,
+     0,0,0,0,1), nrow=5, ncol=4)
> A2 <- matrix(c(1,0,0,0,0,
+     0,1,0,0,0,
+     0,0,1,0,0,
+     0,0,0,1,0), nrow=5, ncol=4)
> H41 <- summary(alrtest(z = H1,
+     A = A1, r = 2))
> H42 <- summary(alrtest(z = H1,
+     A = A2, r = 2))
```

R Output

	Statistic	p-value
Exog. p2	0.66	0.72
Exog. i2	4.38	0.11

Table: Testing Exogenity

VECM

Restrictions on CI-Relations, I

- Tests do not depend on normalization of β.
- Tests are Likelihood ratio tests, similar for testing restrictions on α.

1. Testing restrictions for all cointegration relations.
2. r_1 cointegrating relations are assumed to be known and r_2 cointegarting relations have to be estimated, $r = r_1 + r_2$.
3. r_1 cointegrating relations are estimated with restrictions and r_2 cointegrating relations are estimated without constraints, $r = r_1 + r_2$.

VECM

Restrictions on CI-Relations, II

- Following previous example: Test purchasing power parity and interest rate differential contained in all CI relations.
- Hypothesis: $\mathcal{H}_3 : \beta = H_3\varphi$ with $H_3(K \times s)$, $\varphi(s \times r)$ and $r \leq s \leq K$: $sp(\beta) \subset sp(H_3)$.
- Functions blrtest and ablrtest in package urca.

Literature

- Johansen, S. and K. Juselius, Testing structural hypothesis in a multivariate cointegration analysis of the PPP and the UIP for UK, *Journal of Econometrics*, 53 (1992), 211–244.
- Johansen, S., Statistical Analysis of Cointegration Vectors, *Journal of Economic Dynamics and Control*, 12 (1988), 231-254.
- Johansen, S. and K. Juselius, Maximum Likelihood Estimation and Inference on Cointegration - with Applications to the Demand for Money, *Oxford Bulletin of Economics and Statistics*, 52(2) (1990), 169-210.
- Johansen, S., Estimation and Hypothesis Testing of Cointegration Vectors in Gaussian Vector Autoregressive Models, *Econometrica*, 59(6) (1991), 1551-1580.

VECM

Restrictions on CI-Relations, III

R Code

```
> H.31 <- matrix(c(1,-1,-1,0,0,
+      0,0,0,1,0,
+      0,0,0,0,1), c(5,3))
> H.32 <- matrix(c(1,0,0,0,0,
+      0,1,0,0,0,
+      0,0,1,0,0,
+      0,0,0,1,-1), c(5,4))
> H31 <- blrtest(z = H1, H = H.31, r = 2)
> H32 <- blrtest(z = H1, H = H.32, r = 2)
```

R Output

	Statistic	p-value
All CI: PPP	2.76	0.60
All CI: ID	13.71	0.00

Table: $\mathcal{H}_3$ - Tests

- PPP in all CI relations: Cannot be rejected.
- ID in all CI relations: Must be rejected.

VECM

Restrictions on CI-Relations, IV

- Following previous example: Test purchasing power parity and interest rate differential directly, *i.e.* $(1, -1, -1, 0, 0)$ and $(0, 0, 0, 1, -1)$.
- In contrast to previous hypothesis $\mathcal{H}_3$, which tested: $(a_i, -a_i, -a_i, *, *)$ and $(*, *, *, b_i, -b_i)$ for $i = 1, \ldots, r$.
- Hypothesis: $\mathcal{H}_5 : \beta = (H_5, \Psi)$ with $H_5(K \times r_1)$, $\Psi(K \times r_2)$, $r = r_1 + r_2$: $sp(H_5) \subset sp(\beta)$.
- Function bh5lrtest in package urca.

VECM

Restrictions on CI-Relations, V

R Code

```
> H.51 <- c(1, -1, -1, 0, 0)
> H.52 <- c(0, 0, 0, 1, -1)
> H51 <- bh5lrtest(z = H1, H = H.51, r = 2)
> H52 <- bh5lrtest(z = H1, H = H.52, r = 2)
```

R Output

	Statistic	p-value
Exact PPP	14.52	0.00
Exact ID	1.89	0.59

Table: $\mathcal{H}_5$ - Tests

- Reject stationarity of PPP.
- Cannot reject stationarity for ID.

VECM

Restrictions on CI-Relations, VI

- Following previous example: Strict PPP not stationary; now test if general CI-relation $(a, b, c, 0, 0)$ exist.
- In contrast to previous hypothesis $\mathcal{H}_5$, which tested: $(1, -1, -1, 0, 0)$.
- $\mathcal{H}_6 : \beta = (H_6\varphi, \Psi)$ with $H_6(K \times s)$, $\varphi(s \times r_1)$, $\Psi(K \times r_2)$, $r_1 \leq s \leq K$, $r = r_1 + r_2$: $\dim(sp(\beta) \cap sp(H_6)) \geq r_1$.
- Function bh6lrtest in package urca.

VECM

Restrictions on CI-Relations, VII

R Code

```
> H.6 <- matrix(rbind(diag(3),
+     c(0, 0, 0),
+     c(0, 0, 0)), nrow=5, ncol=3)
> H6 <- bh6lrtest(z = H1, H = H.6,
+     r = 2, r1 = 1)
```

R Output

	Statistic	p-value
General PPP	4.93	0.03

Table: $\mathcal{H}_6$ - Tests

- Statistic insignificant at 1pct level.

SVEC
Definition, I

- Variables are at most $I(1)$ and DGP is a VECM:

$$\Delta y_t = \alpha\beta' y_{t-1} + \Gamma_1 \Delta y_{t-1} + \cdots + \Gamma_{p-1}\Delta y_{t-p+1} + u_t \quad (50)$$

  for $t = 1, \ldots, T$.

- SVECM is a B-model with $u_t = B\varepsilon_t$ and $\Sigma_u = BB'$.
- For unique identification of B, $\frac{1}{2}K(K-1)$ at least restrictions are required.
- Granger's representation theorem:

$$y_t = \Xi \sum_{i=1}^{t} u_i + \sum_{j=0}^{\infty} \Xi_j^* u_{t-j} + y_0^* \quad (51)$$

SVEC
Definition, II

- $\Xi \sum_{i=1}^{t} u_i$ are the common trends; rank of Ξ is $K - r$.
- Matrix Ξ has the form:

$$\Xi = \beta_\perp \left[\alpha'_\perp \left(I_K - \sum_{i=1}^{p-1} \Gamma_i \right) \beta_\perp \right]^{-1} \alpha'_\perp \qquad (52)$$

- Substitution yields: $\Xi \sum_{i=1}^{t} u_i = \Xi B \sum_{i=1}^{t} \varepsilon_t$.
- Hence, long-run effects of structural innovations are given by ΞB.
- At most r innovations can have transitory effects and at least $K - r$ have permanent effects.

SVEC

Resources

R Resources

- Function SVEC in package vars.
- Methods irf and fevd in package vars.
- Method plot for irf and fevd in package vars.

Literature

- King, R., C. Plosser, J. Stock and M. Watson, Stochastic Trends and economic fluctuations, American Economic Review 81 (1991), 819-840.
- Lütkepohl, H. and M. Krätzig, *Applied Time Series Econometrics*, 2004, Cambridge.

SVEC

Example Canada

R Code

```
> library(vars)
> data(Canada)
> vec.can <- ca.jo(Canada, K = 2,
+     spec = "transitory", season = 4)
> LR <- matrix(0, nrow = 4, ncol = 4)
> LR[, c(1, 2)] <- NA
> SR <- matrix(NA, nrow = 4, ncol = 4)
> SR[3, 4] <- 0
> SR[4, 2] <- 0
> svecm <- SVEC(vec.can, r = 2, LR = LR,
+      SR = SR, max.iter = 200,
+      lrtest = TRUE, boot = FALSE)
> svecm.irf <- irf(svecm, impulse = "e",
+       response = "rw", boot = FALSE,
+       cumulative = FALSE, runs = 100)
> svecm.fevd <- fevd(svecm)
```

R Output

	e	prod	rw	U
e	0.05	−0.22	0.06	−0.26
prod	−0.52	0.19	−0.12	−0.23
rw	−0.08	0.37	0.56	0.00
U	−0.13	0.00	0.04	0.22

Table: Impact Matrix B

	e	prod	rw	U
e	−0.41	−0.47	0.00	0.00
prod	−0.51	0.63	0.00	0.00
rw	−0.67	−0.66	0.00	0.00
U	0.09	0.05	0.00	0.00

Table: Long-run Matrix ΞB

SVEC
IRF and FEVD

R Output

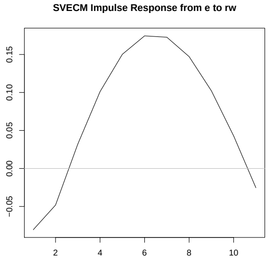


Figure: IRF of e to rw

R Output

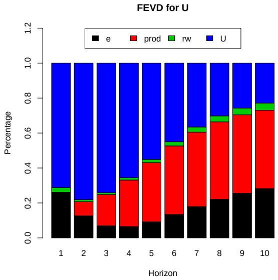


Figure: FEVD of U

Topics left out

- Near-integrated processes (see packages: longmemo, fracdiff and fSeries).
- Seasonal unit roots (see package uroot).
- Bayesian VAR models (see package MSBVAR).

Selected Monographies

G. Amisano and C. Giannini
Topics in Structural Var Econometrics.
Spinger, 1997.

A. Banerjee, J.J. Dolado, J.W. Galbraith and D.F. Hendry
Co-Integration, Error-Correction, and the Econometric Analysis of Non-Stationary Data.
Oxford University Press, 1993.

J. Beran
Statistics for Long-Memory Processes
Chapman & Hall, 1994

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

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

H. Lütkepohl.
New Introduction to Multiple Time Series Analysis.
Springer, 2006.

Cited R packages

Name	Title	Version
dse1	Dynamic Systems Estimation (time series package)	2007.5-2
dynlm	Dynamic Linear Regression	0.1-2
fBasics	Rmetrics - Markets and Basic Statistics	240.10068.1
fracdiff	Fractionally differenced ARIMA aka ARFIMA(p,d,q) models	1.3-1
fSeries	Rmetrics - The Dynamical Process Behind Markets	240.10068
lmtest	Testing Linear Regression Models	0.9-19
longmemo	Statistics for Long-Memory Processes (Jan Beran) – Data and Functions	0.9-5
mAr	Multivariate AutoRegressive analysis	1.1-1
MSBVAR	Bayesian Vector Autoregression Models	0.2.2
tseries	Time series analysis and computational finance	0.10-11
vars	VAR Modelling	0.7-9
urca	Unit root and cointegration tests for time series data	1.1-5
uroot	Unit Root Tests and Graphics for Seasonal Time Series	1.4

Table: Overview of cited R packages