



Università degli Studi di Palermo

Facoltà di Economia

Dipartimento di Scienze Statistiche e Matematiche "S. Vianelli"
Dottorato di Ricerca in Statistica e Finanza Quantitativa - XXI Ciclo

Sergio Salvino Guirrieri

Analisi econometrica delle serie storiche con R.

1 luglio 2008

Indice

1	Cenni sull'analisi delle serie storiche univariate e multivariate	3
1.1	Serie storiche univariate	3
1.2	Serie storiche multivariate.	13
2	Processi integrati	19
3	Test per le radici unitarie	25
3.1	Il test di Dickey–Fuller.	27
3.2	Il test di Phillips–Perron.	31
3.3	Il test KPSS.	35
4	Serie storiche cointegrate.	39
4.1	Modelli a correzione d'errore.	41
4.2	VECM con 2 relazioni di cointegrazione.	50
5	Analisi della cointegrazione della struttura dei tassi di interesse.	59
5.1	Analisi della cointegrazione tra 3 tassi d'interesse.	60
5.2	Analisi della cointegrazione tra 4 tassi d'interesse.	66
	Riferimenti Bibliografici.	72

Premessa

Lo scopo di questo articolo è quello di fornire un contributo e gli strumenti basilari per l'analisi econometrica delle serie storiche in ambiente R ([R Development Core Team, 2008](#)). In particolare, per la modellazione delle serie storiche univariate si farà riferimento alle funzioni che si trovano nel pacchetto base di R mentre, per l'analisi econometrica e multivariata delle serie storiche sarà necessario installare i moduli, `vars`, `urca` e `tseries`. Le varie tecniche econometriche che vengono utilizzate hanno lo scopo di dare un'idea di cosa è possibile fare attualmente in ambiente R, quale sono le sue potenzialità e cosa si può ulteriormente fare. Sicuramente, in questo articolo non s'intende affrontare in modo esaustivo ed approfondito tutte le funzioni che sono presenti in R per l'analisi econometrica delle serie storiche, ma l'intenzione è di fornirne una semplice introduzione. Il articolo principalmente è basato sull'*analisi della cointegrazione* e alla sua rappresentazione in forma ECM (*error correction model*).

Nel *paper* non mancheranno cenni sia alla teoria delle serie storiche che alla teoria econometrica, ma lo scopo principale è l'implementazione in R utilizzando delle simulazioni e facendo delle applicazioni su un data set, che è composto dai tassi di interesse a 3 mesi, 1 anno, 7 anni e 20 anni, della Federal Reserve Bank, da Febbraio 1982 fino a Dicembre 2006¹.

¹Il file può essere scaricato dal mio sito internet <http://www.webalice.it/sergio.guirreri>

Capitolo 1

Cenni sull'analisi delle serie storiche univariate e multivariate

1.1 Serie storiche univariate

Sia `dati.txt` il file ascii di dati, collocato nella stessa directory di lavoro dell'ambiente di lavoro R, per cui è possibile procedere alla lettura per mezzo della funzione `read.table`, la quale è utilizzata per caricare dei dati da un file in formato `.txt`. Il data set può essere, successivamente, convertito in un oggetto di formato *time series* con la funzione `ts`, la quale è presente nel pacchetto base di R. È possibile specificare sia l'arco temporale (`start = c(1982, 2)`, `end = c(2006, 12)`) che la frequenza di rilevazione dei dati (`deltat = 1/12` oppure `frequency=12`).

```
> tassi <- read.table("dati articolo.txt", header = TRUE, sep = "",
+ dec = ".")
> tassi.ts <- ts(tassi, start = c(1982, 2), end = c(2006, 12),
+ deltat = 1/12)
```

Come è ben noto dalla teoria sulle serie storiche, è fondamentale verificare la stazionarietà¹ o meno della serie, poichè le tecniche di inferenza statistica su serie stazionarie hanno caratteristiche e proprietà decisamente differenti dalle tecniche inferenziali usate nell'ambito delle serie integrate².

AD esempio, un'analisi grafica del data set può dare un'idea sulla stazionarietà o meno della serie osservata, infatti dal plot riportato in figura (1.1) si evince una non stazionarietà dei dati, che nel caso di serie macroeconomiche è piuttosto frequente. Riguardo la stazionarietà di una serie storica, esistono una serie di test per

¹La stazionarietà di una serie storica suppone che certe proprietà statistiche di una serie risultino invarianti rispetto ad una traslazione nel tempo. Nel contesto si fa riferimento alla stazionarietà in senso debole che si basa sull'invarianza nel tempo dei momenti fino al secondo ordine (media, varianza, covarianza).

²Un riferimento bibliografico molto utile in proposito [Lutkepohl \(2005\)](#)

Serie storica dei tassi di interesse

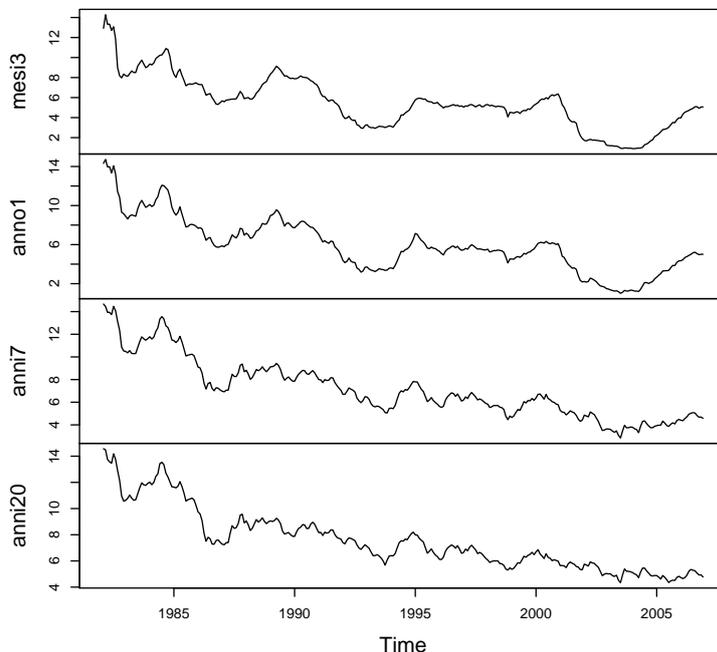


Figura 1.1: Grafico della serie dei tassi.

il controllo della presenza o meno di radici unitarie che sono piú rigorosi rispetto alla semplice analisi grafica della serie – i quali saranno esaminati nel capitolo 2 – tali test vanno sotto il nome di “test per le radici unitarie” (*unit root test*)³.

Uno dei tanti motivi per cui è importante verificare la presenza di radici unitarie in una serie storica, si può evincere dalla formula della funzione di autocovarianza di un processo AR(1). Infatti, una serie storica $\{y_1, y_2, \dots, y_T\}$ è un processo autoregressivo del primo ordine se soddisfa la seguente equazione:

$$y_t = \phi y_{t-1} + \epsilon_t \quad t = 1, \dots, T \quad (1.1)$$

dove $\epsilon_t \sim WN(0, \sigma^2)$ e ϕ parametro incognito da stimare. La funzione di autocovarianza⁴ di un processo AR(1) è data dalla seguente espressione

$$\gamma_0 = \frac{\sigma^2}{1 - \phi^2}. \quad (1.2)$$

Nella (1.2) si può notare, chiaramente, che quando il parametro ϕ del processo AR(1) è pari ad 1 la funzione (1.2) perde di significato, mentre per valori di ϕ prossimi ad 1 la (1.2) tende ad infinito.

³Un processo AR(1) ha una radice unitaria se la radice del suo polinomio caratteristico è uguale ad 1, oppure se il coefficiente ϕ è uguale ad 1

⁴Per una dimostrazione piú rigorosa si veda [Hamilton \(1994\)](#).

Si dimostra che un processo AR(1) è stazionario se la radice dell' *equazione caratteristica* è maggiore di 1. L'equazione caratteristica corrispondente al processo (1.1) è la seguente

$$1 - \phi z = 0, \quad (1.3)$$

dalla quale si ricava che $z = \frac{1}{\phi}$; per cui la stazionarietà è soddisfatta se il valore assoluto della radice dell'equazione (1.3) è maggiore di 1, $|\frac{1}{\phi}| > 1$ oppure $|\phi| < 1$. Altri test per diagnosticare la stazionarietà di una serie nel caso univariato, sono il test di *Box-Pierce* ed il test di *Ljung-Box*. Questi due test prendono in considerazione m autocorrelazioni di ordine d ([Santamaria, 2000](#))

$$\rho_d = \frac{\sum_{t=1}^{T-d} (y_t - \bar{y}_T)(y_{t+d} - \bar{y}_T)}{\sum_{t=1}^T (y_t - \bar{y}_T)^2}, \quad (1.4)$$

dove \bar{y}_T è la media aritmetica della serie storica. Il test di Box-Pierce è basato sulla seguente relazione

$$Q_m = T \sum_{d=1}^m \rho_d^2, \quad (1.5)$$

mentre il test di Ljung-Box è calcolato secondo la seguente relazione

$$Q'_m = T(T+2) \sum_{d=1}^m \frac{\rho_d^2}{T-d}, \quad (1.6)$$

Si dimostra che, sotto l'ipotesi nulla H_0 di stazionarietà, questi due test si distribuiscono asintoticamente secondo una variabile casuale χ^2_ν con " $\nu = m$ gradi di libertà". In R, per calcolare i test (1.5) e (1.6), può essere utilizzata la funzione `Box.test`, che si trova nel pacchetto base.

Nel caso della serie dei tassi di interesse, considerando almeno 5 coefficienti di autocorrelazione, otteniamo i seguenti risultati:

```
> Box.test(tassi.ts[, 1], lag = 5) ## serie dei tassi a 3 mesi

Box-Pierce test

data:  tassi.ts[, 1]
X-squared = 1257.210, df = 5, p-value < 2.2e-16

> Box.test(tassi.ts[, 1], lag = 5, type = "L") ## serie dei tassi a 3 mesi

Box-Ljung test

data:  tassi.ts[, 1]
X-squared = 1277.886, df = 5, p-value < 2.2e-16

> Box.test(tassi.ts[, 2], lag = 5) ## serie dei tassi ad 1 anno

Box-Pierce test

data:  tassi.ts[, 2]
X-squared = 1264.076, df = 5, p-value < 2.2e-16

> Box.test(tassi.ts[, 2], lag = 5, type = "L") ## serie dei tassi ad 1 anno
```

```

Box-Ljung test

data:  tassi.ts[, 2]
X-squared = 1284.899, df = 5, p-value < 2.2e-16

> Box.test(tassi.ts[, 3], lag = 5) ## serie dei tassi a 7 anni

Box-Pierce test

data:  tassi.ts[, 3]
X-squared = 1282.967, df = 5, p-value < 2.2e-16

> Box.test(tassi.ts[, 3], lag = 5, type = "L") ## serie dei tassi a 7 anni

Box-Ljung test

data:  tassi.ts[, 3]
X-squared = 1304.184, df = 5, p-value < 2.2e-16

> Box.test(tassi.ts[, 4], lag = 5) ## serie dei tassi a 20 anni

Box-Pierce test

data:  tassi.ts[, 4]
X-squared = 1277.296, df = 5, p-value < 2.2e-16

> Box.test(tassi.ts[, 4], lag = 5, type = "L") ## serie dei tassi a 20 anni

Box-Ljung test

data:  tassi.ts[, 4]
X-squared = 1298.422, df = 5, p-value < 2.2e-16

```

Dall'analisi univariata della serie storica dei tassi di interesse, si ha evidenza contro l'ipotesi nulla di stazionarietà.

L'approccio [Box and Jenkins \(1970\)](#) per lo studio delle serie storiche univariate, si basa sulla modellazione dei dati attraverso modelli autoregressivi (AR), a media mobile (MA), modelli ARMA oppure, nel caso di serie integrate, dei modelli ARIMA. Sostanzialmente, l'approccio Box-Jenkins si incentra sulle note tre fasi:

1. identificazione del modello: scegliendo il modello più appropriato tra AR(p), MA(q), ARMA(p,q), ARIMA(p,d,q);
2. stima del modello: stima dei parametri, gli errori standard e le statistiche test t ;
3. controllo dell'adeguatezza del modello stimato tramite l'analisi dei residui.

L'identificazione del modello avviene tramite la stima della funzione di autocovarianza (ACF) della serie storica. Infatti, si può facilmente notare che per processi autoregressivi l'ACF corrispondente, ha un andamento che decresce esponenzialmente all'aumentare dei lag, mentre per processi a media mobile l'ACF decade a zero subito dopo il lag di ordine q [Harvey \(1993, cap.2\)](#).

In R con la funzione `arima` è possibile stimare i modelli AR(p), MA(q), ARMA(p,q)

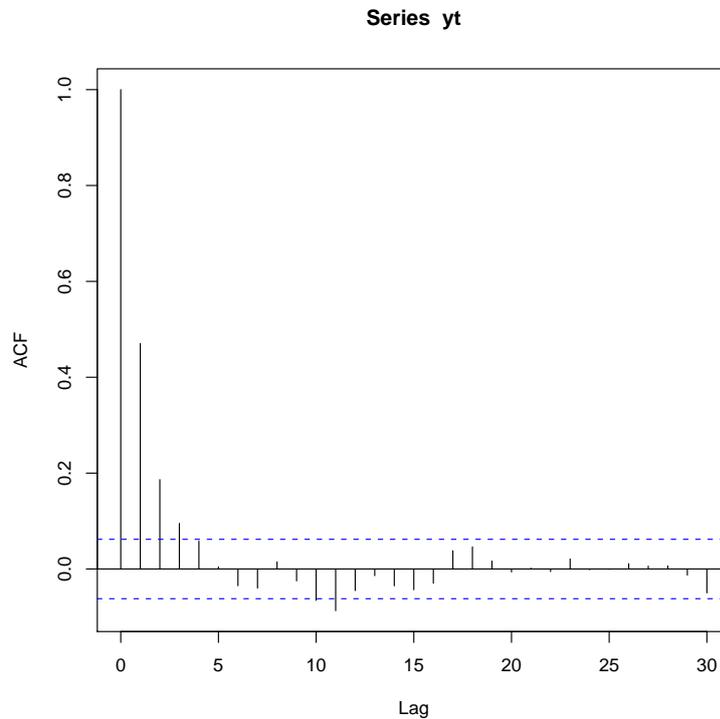


Figura 1.2: Funzione di autocorrelazione del processo yt simulato.

e ARIMA(p,d,q). Inoltre, se si volesse simulare un processo AR(1), è possibile applicare la funzione `arima.sim`, oppure, un buon esercizio, è eseguire le seguenti istruzioni.

```
> yt <- c(0)
> et <- ts(rnorm(1000))
> for (i in 1:1000) {
+   ifelse(i == 1, yt[i] <- et[i], yt[i] <- 0.5 * yt[i - 1] +
+     et[i])
+ }
> acf(yt)
```

La stima del modello simulato è eseguita con i successivi comandi.

```
> arima(yt, order = c(1, 0, 0))
```

Call:

```
arima(x = yt, order = c(1, 0, 0))
```

Coefficients:

	ar1	intercept
	0.4712	0.1052
s.e.	0.0279	0.0613

sigma² estimated as 1.051: log likelihood = -1444.11, aic = 2894.22

A scopo illustrativo di seguito vengono eseguite le rispettive simulazioni, con la funzione `arima.sim`, di un modello ARMA(2,2) ed un modello ARIMA(2,1,2). Le figure (1.3), (1.5) illustrano il trend dei processi simulati mentre, le figure (1.4), (1.6) rappresentano le funzioni di autocorrelazione.

```
> y.ARMA <- arima.sim(n = 1000, list(ar = c(0.8, -0.4), ma = c(0.1,
+ 0.4)))
> y.ARIMA <- arima.sim(list(order = c(2, 1, 2), ar = c(0.8, -0.4),
+ ma = c(0.1, 0.4)), n = 1000)
> plot.ts(y.ARMA)
> acf(y.ARMA)
> plot.ts(y.ARIMA)
> acf(y.ARIMA)
```

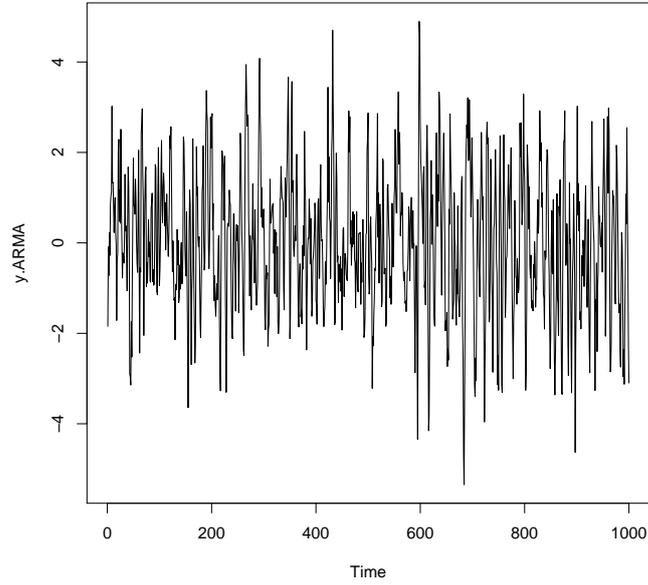


Figura 1.3: Grafico del processo ARMA(2,2) simulato.

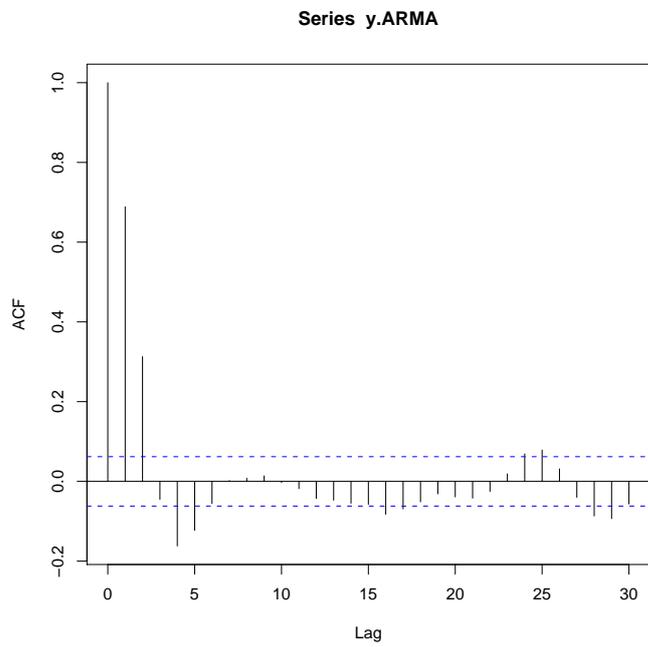


Figura 1.4: Funzione di autocorrelazione del processo ARMA(2,2) simulato.

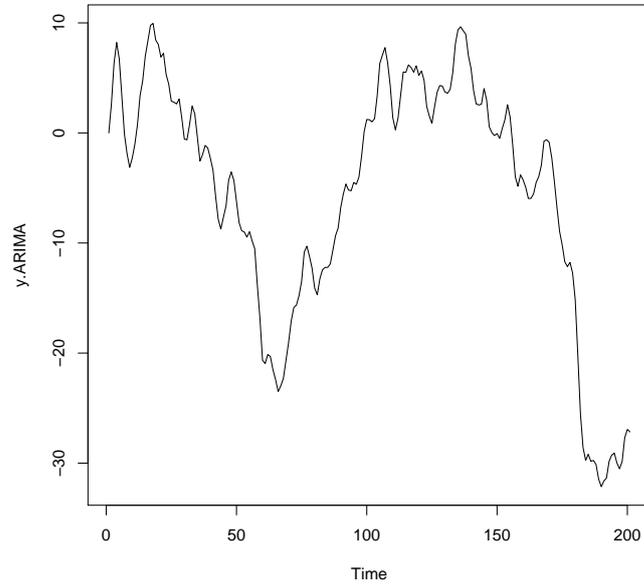


Figura 1.5: Grafico del processo ARIMA(2,1,2) simulato.

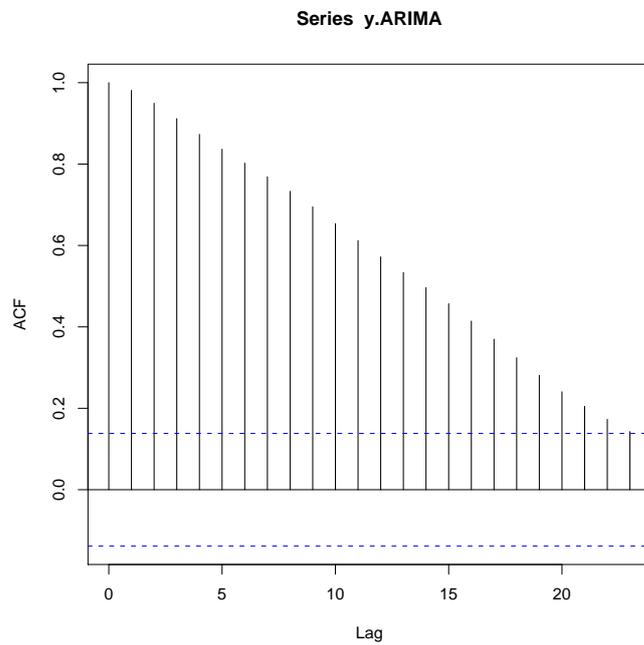


Figura 1.6: Funzione di autocorrelazione del processo ARIMA(2,1,2).

Si noti come i grafici (1.3) e (1.5) sono molto differenti, infatti l'ACF del processo

ARIMA è molto persistente ed in corrispondenza al lag=10 corrisponde un valore del coefficiente di autocorrelazione elevato, mentre il processo ARMA oltre ad essere “mean reverting” ha dei coefficienti di autocorrelazioni significativi fino al 4 lag. Di seguito vengono riportate le stime dei due processi simulati, con le rispettive analisi dei residui eseguite per mezzo della funzione `tsdiag`, come riportato nelle figure (1.7) (1.8).

```
> model.ARMA <- arima(y.ARMA, c(2, 0, 2))
> model.ARMA
```

```
Call:
arima(x = y.ARMA, order = c(2, 0, 2))
```

```
Coefficients:
      ar1      ar2      ma1      ma2  intercept
    0.7616 -0.3596  0.1385  0.3963    0.0757
s.e.  0.0874  0.0642  0.0860  0.0421    0.0844
```

```
sigma^2 estimated as 1.081:  log likelihood = -1458.66,  aic = 2929.33
```

```
> tsdiag(model.ARMA)
```

```
> model.ARIMA <- arima(y.ARIMA, c(2, 1, 2))
> model.ARIMA
```

```
Call:
arima(x = y.ARIMA, order = c(2, 1, 2))
```

```
Coefficients:
      ar1      ar2      ma1      ma2
    0.8186 -0.4113  0.0625  0.3657
s.e.  0.0819  0.0576  0.0805  0.0463
```

```
sigma^2 estimated as 0.9619:  log likelihood = -1400.08,  aic = 2810.16
```

```
> tsdiag(model.ARIMA)
```

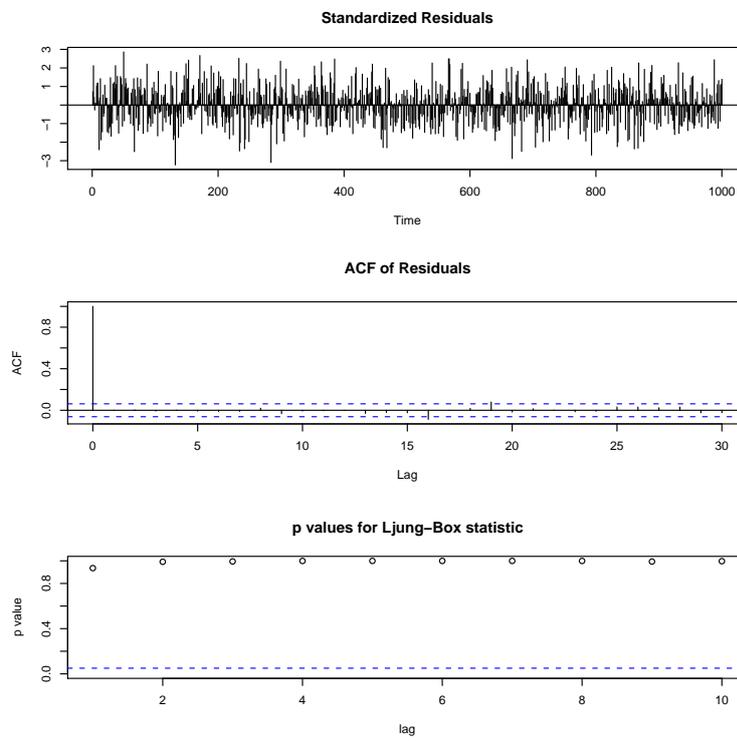


Figura 1.7: Diagnostica del modello ARMA(2,2).

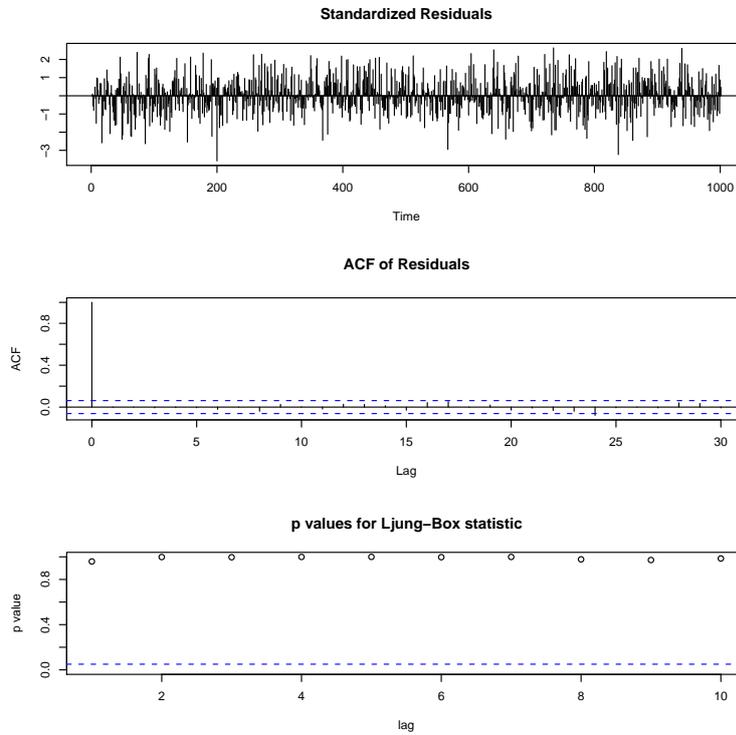


Figura 1.8: Diagnostica del modello ARIMA(2,1,2).

1.2 Serie storiche multivariate.

Il modello autoregressivo vettoriale ⁵ è la tecnica usata per modellare le serie storiche multivariate, in cui si focalizza l'attenzione sul comportamento congiunto degli elementi del vettore $\mathbf{y}_t = (\mathbf{y}_{1t}, \mathbf{y}_{2t}, \dots, \mathbf{y}_{Kt})'$, di dimensione $(K \times 1)$ nell'arco temporale $t = 1, \dots, T$. I modelli VAR sono stati introdotti in uno storico articolo di Sims (1980), il quale propose questo tipo di modelli criticando i modelli di equazioni simultanee, che erano i modelli più utilizzati in macroeconomia.

In generale, un modello VAR di ordine p , VAR(p), è riportato nella seguente forma (Lutkepohl, 2005):

$$\mathbf{y}_t = \boldsymbol{\mu} + \mathbf{A}_1 \mathbf{y}_{t-1} + \dots + \mathbf{A}_p \mathbf{y}_{t-p} + \mathbf{u}_t \quad t=0,1,\dots \quad (1.7)$$

dove \mathbf{A}_i sono matrici di coefficienti di dimensione $(K \times K)$, $\boldsymbol{\mu} = (\mu_1, \dots, \mu_K)'$ e un vettore di intercette di dimensione $(K \times 1)$, $\mathbf{u}_t = (u_{1t}, \dots, u_{Kt})'$ è un processo *white noise*⁶ K dimensionale, in cui la matrice di varianze e covarianze è assunta

⁵Spesso indicato con l'acronimo di VAR *vector autoregressive model*.

⁶Un processo *white noise* \mathbf{u}_t ha le seguenti caratteristiche:

$$E(\mathbf{u}_t) = 0; \quad (1.8)$$

$$E(\mathbf{u}_t \mathbf{u}_t') = \Sigma_u; \quad (1.9)$$

$$E(\mathbf{u}_t \mathbf{u}_s') = 0 \quad \text{con } s \neq t. \quad (1.10)$$

essere non singolare.

Per verificare la stazionarietà di un processo VAR(1) si dimostra che è sufficiente controllare che tutti gli autovalori della matrice \mathbf{A}_1 siano minori di uno, che è equivalente alla condizione che:

$$\det(\mathbf{I}_K - \mathbf{A}_1 \mathbf{z}) \neq 0, |\mathbf{z}| < 1, \quad (1.11)$$

in cui \mathbf{I}_K è la matrice identità di dimensione K mentre, \mathbf{z} è il vettore degli autovalori di dimensione K . Più in generale, un processo VAR(p) soddisfa le condizioni di stazionarietà se tutti gli autovalori della matrice *companion*⁷ cadono dentro il cerchio unitario, ossia sono minori di 1.

In R il pacchetto che permette di effettuare analisi multivariate ed econometriche sulle serie storiche è il pacchetto `vars` (Pfaff, 2008), il quale dipende da una serie di pacchetti, tra i quali, quello che verrà utilizzato per l'analisi della cointegrazione, il pacchetto `urca` (Pfaff, 2006).

Dopo aver caricato il pacchetto `vars`, si procede alla stima del numero di variabili ritardate da introdurre nel modello e, successivamente, alla stima dei parametri e all'analisi della stabilità del VAR.

```
> library(vars)
> VARselect(tassi.ts)

$selection
AIC(n)  HQ(n)  SC(n) FPE(n)
      4      2      2      4

$criteria
          1          2          3          4          5
AIC(n) -1.627183e+01 -1.672942e+01 -1.676069e+01 -1.677135e+01 -1.675222e+01
HQ(n)   -1.617016e+01 -1.654641e+01 -1.649635e+01 -1.642568e+01 -1.632521e+01
SC(n)   -1.601810e+01 -1.627270e+01 -1.610098e+01 -1.590866e+01 -1.568655e+01
FPE(n)  8.575142e-08  5.426773e-08  5.260562e-08  5.206325e-08  5.309431e-08
          6          7          8          9          10
AIC(n) -1.670730e+01 -1.663583e+01 -1.659755e+01 -1.660676e+01 -1.655406e+01
HQ(n)   -1.619896e+01 -1.604615e+01 -1.592653e+01 -1.585441e+01 -1.572037e+01
SC(n)   -1.543864e+01 -1.516419e+01 -1.492292e+01 -1.472915e+01 -1.447346e+01
FPE(n)  5.557291e-08  5.974851e-08  6.216046e-08  6.169203e-08  6.516434e-08

> tassi.var <- VAR(tassi.ts, p = 4)
> roots(tassi.var)
```

⁷La matrice *companion* è una matrice ($Kp \times Kp$) i cui elementi sono le matrici \mathbf{A}_i della (1.7), così composta:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 & \dots & \mathbf{A}_{p-1} & \mathbf{A}_p \\ \mathbf{I}_K & 0 & \dots & 0 & 0 \\ 0 & \mathbf{I}_K & \dots & 0 & 0 \\ \vdots & \dots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & \mathbf{I}_K & 0 \end{bmatrix}$$

```
[1] 0.9769321 0.9543815 0.9543815 0.7620743 0.7620743 0.5571712 0.5571712
[8] 0.5530648 0.5530648 0.5224984 0.5224984 0.4984459 0.3644176 0.3644176
[15] 0.2782984 0.2679477
```

Il numero di lags significativi vengono stimati per mezzo della funzione `VARselect` la quale si basa sui criteri di informazione di: *Akaike* (AIC), *Hannan-Queen* (HQ), *Schwarz* (SC) ed il *final prediction error* (FPE). Il numero di variabili ritardata da introdurre è compreso tra 2 e 4, per cui il modello dovrebbe essere un VAR(2), VAR(3) oppure VAR(4).

La funzione `VAR` implementa la procedura che consente di stimare il vector autoregressive, che si assume di ordine $p=4$, mentre la funzione `roots` stima gli autovalori della matrice companion associata al VAR(4).

Dall'analisi degli autovalori si nota come alcuni di questi siano molto vicini all'unità, per cui è confermata l'ipotesi della presenza di radici unitarie nella serie dei tassi d'interesse.

Sull'oggetto `tassi.var`, a cui è associata la classe `varest`⁸, è possibile applicare diverse funzioni, tra cui:

- `summary`, che effettua un riepilogo molto più dettagliato sul modello stimato, tra cui: le stime dei parametri, gli errori standard, i p-value, la matrice di varianze e covarianze dei residui, la matrice di correlazione dei residui e le radici del polinomio caratteristico;
- `plot`, esegue un'analisi grafica del modello stimato mostrando la serie stimata, la serie osservata, il grafico della funzione di autocorrelazione globale e parziale.

Il controllo della adeguatezza del modello stimato è effettuata tramite la diagnostica dei residui, infatti, se il vector autoregressive è stato specificato correttamente i suoi residui devono essere "puliti", cioè:

- assenza di autocorrelazione, quindi l'acf ed il pacf non devono mostrare dei lag significativi;
- varianza costante, la quale implica che il plot dei residui deve oscillare entro determinate bande di confidenza;
- *mean reverting*;
- normalità.

È possibile procedere alla diagnostica dei residui, quindi controllare le sopra elencate proprietà, applicando all'oggetto `tassi.var` le funzioni: `serial.test`⁹, `arch.test`, `normality.test`¹⁰

Ad esempio, si supponga che si voglia simulare in un vector autoregressive bivariato con un solo ritardo, quindi un VAR(1), ovvero

$$y_{1t} = 0.1 + 0.5 y_{1t-1} + 0.3 y_{2t-1} + \epsilon_{1t} \quad (1.12)$$

$$y_{2t} = -2.5 + 0.2 y_{1t-1} + 0.1 y_{2t-1} + \epsilon_{2t} \quad (1.13)$$

⁸Per ottenere una lista dettagliata degli oggetti che è possibile estrarre dall'oggetto `tassi.var` si può eseguire `str(tassi.var)`.

⁹Esegue il test *Portmanteau* ed il test di Breusch-Godfrey (Breusch, 1978; Godfrey, 1978).

¹⁰Esegue il test di Jarque and Bera (1987) ed il test sulla curtosi e sull'asimmetria.

Dalla (1.7) si ha che, se il processo è stazionario in varianza, la media è data da

$$\boldsymbol{\nu} = (\mathbf{I} - \mathbf{A}_1 - \dots - \mathbf{A}_p)^{-1} \boldsymbol{\mu}.$$

Dalle (1.12) e (1.13) ricaviamo le seguenti quantità:

$$\mathbf{A}_1 = \begin{pmatrix} 0.5 & 0.3 \\ 0.2 & 0.1 \end{pmatrix} \quad \boldsymbol{\mu} = \begin{pmatrix} 0.1 \\ -2.5 \end{pmatrix} \quad \boldsymbol{\nu} = \begin{pmatrix} -1.69 \\ -3.15 \end{pmatrix} \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Per effettuare questo tipo di simulazione ci si può avvalere del pacchetto `mAr`, oppure eseguire i seguenti codici.

```
> library(MASS)
> mu <- c(0.1, -2.5)
> sigma.var <- diag(1, 2)
> A1 = matrix(c(0.5, 0.3, 0.2, 0.1), 2, 2, byrow = TRUE)
> E <- mvrnorm(1000, mu, sigma.var)
> Y <- matrix(0, 1000, 2)
> colnames(Y) <- c("y1", "y2")
> for (i in 2:1000) {
+   Y[i, ] <- A1 %*% Y[i - 1, ] + E[i, ]
+ }
```

Come verifica si procede alla stima dei parametri del modello simulato.

```
> est.sim.var <- VAR(Y, type = "const", lag = 1)
> coef(est.sim.var)
```

```
$y1
      Estimate Std. Error   t value    Pr(>|t|)
y1.11 0.4889677 0.02601510 18.7955333 1.086581e-67
y2.11 0.2999398 0.03129606  9.5839492 7.181436e-21
const 0.0972169 0.10575774  0.9192415 3.581918e-01
```

```
$y2
      Estimate Std. Error   t value    Pr(>|t|)
y1.11 0.1909070 0.02569189  7.430633 2.322202e-13
y2.11 0.1165236 0.03090724  3.770107 1.727762e-04
const -2.4640824 0.10444383 -23.592418 4.072496e-98
```

Dato che gli autovalori della matrice \mathbf{A}_1 sono minori di 1, il vector autoregressive simulato è stazionario inoltre, la stima del VAR sulla serie dei dati simulati, restituisce i coefficienti del sistema (1.12–1.13).

Adesso, si supponga di voler simulare da un processo autoregressivo bivariato con due variabili ritardate come il seguente:

$$y_{1t} = 0.1 + 0.5 y_{1t-1} + 0.3 y_{2t-1} + 0.2 y_{1t-2} - 0.4 y_{2t-2} + \epsilon_{1t} \quad (1.14)$$

$$y_{2t} = -2.5 + 0.2 y_{1t-1} + 0.1 y_{2t-1} - 0.1 y_{1t-2} + 0.1 y_{2t-2} + \epsilon_{2t} \quad (1.15)$$

La funzione `mAr.sim` consente di simulare un processo autoregressivo multivariato, dove gli argomenti da passare alla funzione sono:

- il vettore di intercette;

- le matrici dei coefficienti del processo AR;
- la matrice di varianze e covarianze delle ϵ_t ;
- il numero di osservazioni della serie storica.

I codici per effettuare la simulazione sono riportati di seguito.

```
> library(mAr)
> A1 = matrix(c(0.5, 0.3, 0.2, 0.1), 2, 2, byrow = TRUE)
> A2 = matrix(c(0.2, -0.4, -0.1, 0.1), 2, 2, byrow = T)
> A <- cbind(A1, A2)
> mu <- c(0.1, -2.5)
> inv <- solve(diag(2) - A1 - A2)
> nu <- inv %*% mu
> sigma.var <- diag(1, 2)
> varmod.simulate <- mAr.sim(mu, A, sigma.var, 1000)
> est.var <- VAR(varmod.simulate, p = 2, type = "const")
> summary(est.var)
```

```
VAR Estimation Results:
=====
Endogenous variables: X1, X2
Deterministic variables: const
Sample size: 998
Log Likelihood: -2790.128
Roots of the characteristic polynomial:
0.7539 0.6167 0.2899 0.2899
Call:
VAR(y = varmod.simulate, p = 2, type = "const")
```

```
Estimation results for equation X1:
=====
X1 = X1.11 + X2.11 + X1.12 + X2.12 + const
```

	Estimate	Std. Error	t value	Pr(> t)
X1.11	0.47908	0.02848	16.823	< 2e-16 ***
X2.11	0.30654	0.03141	9.759	< 2e-16 ***
X1.12	0.22297	0.02813	7.928	5.98e-15 ***
X2.12	-0.40544	0.03192	-12.702	< 2e-16 ***
const	0.06530	0.14038	0.465	0.642

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
Residual standard error: 0.9819 on 993 degrees of freedom
Multiple R-Squared: 0.7065, Adjusted R-squared: 0.705
F-statistic: 478 on 5 and 993 DF, p-value: < 2.2e-16
```

```
Estimation results for equation X2:
```

```

=====
X2 = X1.l1 + X2.l1 + X1.l2 + X2.l2 + const

      Estimate Std. Error t value Pr(>|t|)
X1.l1  0.23381    0.02849   8.208 6.96e-16 ***
X2.l1  0.11610    0.03142   3.695 0.000232 ***
X1.l2 -0.13985    0.02814  -4.971 7.85e-07 ***
X2.l2  0.07906    0.03193   2.476 0.013450 *
const -2.50402    0.14042 -17.832 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.9822 on 993 degrees of freedom
Multiple R-Squared: 0.9029, Adjusted R-squared: 0.9025
F-statistic: 1848 on 5 and 993 DF, p-value: < 2.2e-16

```

Covariance matrix of residuals:

```

      X1      X2
X1  0.96408 -0.04046
X2 -0.04046  0.96471

```

Correlation matrix of residuals:

```

      X1      X2
X1  1.00000 -0.04196
X2 -0.04196  1.00000

```

> coef(est.var)

\$X1

```

      Estimate Std. Error t value Pr(>|t|)
X1.l1  0.4790782 0.02847803 16.8227292 4.556137e-56
X2.l1  0.3065443 0.03141171  9.7589178 1.513814e-21
X1.l2  0.2229734 0.02812609  7.9276355 5.979956e-15
X2.l2 -0.4054364 0.03191966 -12.7017763 2.320173e-34
const  0.0652994 0.14037612  0.4651745 6.419085e-01

```

\$X2

```

      Estimate Std. Error t value Pr(>|t|)
X1.l1  0.23381032 0.02848723  8.207548 6.959813e-16
X2.l1  0.11610322 0.03142186  3.694983 2.319116e-04
X1.l2 -0.13985337 0.02813518 -4.970765 7.851924e-07
X2.l2  0.07906057 0.03192998  2.476061 1.344987e-02
const -2.50401683 0.14042149 -17.832148 6.420212e-62

```

Se si volesse controllare la stazionarietà del processo VAR(2) stimato, si può fare ricorso alla funzione `roots`, per controllare gli autovalori della matrice companion, oppure basta effettuare l'analisi grafica dei residui con il comando `plot(est.var)`.

Capitolo 2

Processi integrati

Si consideri un processo AR(1) definito dalla relazione (1.1) in cui ϕ rappresenta il parametro da stimare. Come è ben noto in letteratura, la stima dei coefficienti è effettuata tramite il metodo dei minimi quadrati ordinari (*ordinary least square*), assumendo che le $\epsilon_t \sim \text{WN}(0, \sigma^2)$, oppure con il metodo della massima verosimiglianza quando le ϵ_t sono distribuite normalmente. Se $|\phi| < 1$, allora è noto il seguente risultato asintotico per lo stimatore OLS $\hat{\phi}$:

$$\sqrt{T}(\hat{\phi}_T - \phi) \xrightarrow{L} N\left(0, \sigma^2 \left[\frac{1 - \phi^2}{\sigma^2}\right]\right) = N(0, (1 - \phi^2)). \quad (2.1)$$

Si noti che la relazione (2.1) non è più valida quando $\phi = 1$, altrimenti la varianza dello stimatore $\hat{\phi}$ si assumerebbe nulla. Inoltre, nel caso di radice unitaria il processo (1.1) coincide con un processo *random walk*¹ che ha come caratteristica la non stazionarietà. Infatti, nel caso di $\phi = 1$ si ha:

$$y_t = y_{t-1} + \epsilon_t = y_{t-2} + \epsilon_{t-1} + \epsilon_t = \dots = y_0 + \sum_{i=1}^t \epsilon_i, \quad (2.2)$$

dalla quale si ottiene

$$\begin{aligned} E(y_t) &= y_0 \\ \text{var}(y_t) &= \text{var}\left(y_0 + \sum_{i=1}^t \epsilon_i\right) = t \text{var}(\epsilon_i) = t \sigma^2. \end{aligned}$$

Quindi la varianza di un processo random walk non è stazionaria ma dipende dal tempo. Mentre, se nel processo (2.2) introduciamo una costante deterministica diversa da zero, otteniamo un processo *random walk con drift*:

$$y_t = \nu + y_{t-1} + \epsilon_t = \nu + (\nu + y_{t-2} + \epsilon_{t-1}) + \epsilon_t = \dots = y_0 + t \nu + \sum_{i=1}^t \epsilon_i, \quad (2.3)$$

che ha come caratteristica che

$$E(y_t) = y_0 + \nu t$$

¹Un processo *random walk* è caratterizzato dalla seguente relazione: $y_t = y_{t-1} + \epsilon_t$ con $y_0 = \alpha$ e le $\epsilon_t \sim \text{WN}(0, \sigma^2)$.

cioè la costante deterministica nel processo comporta la presenza di un trend lineare nella media, quindi viene a mancare anche la stazionarietà in media.

In R si può simulare un processo random walk, come in fig.(2.1), utilizzando la funzione `cumsum`.

```
> eps <- rnorm(1000, 0, 1)
> r.w.ts <- cumsum(eps)
> plot.ts(r.w.ts)
```

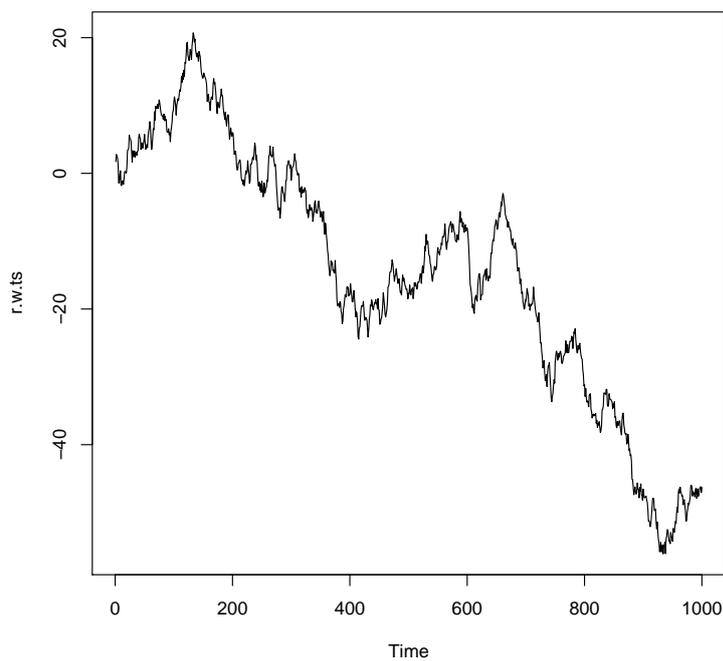


Figura 2.1: Processo random walk.

Mentre per simulare un random walk con drift, fig.(2.2), basta generare da una normale con media diversa da zero oppure aggiungere una costante alle `eps` precedenti.

```
> r.w.drift.ts <- cumsum(eps + 0.3)
> plot.ts(r.w.drift.ts)
```

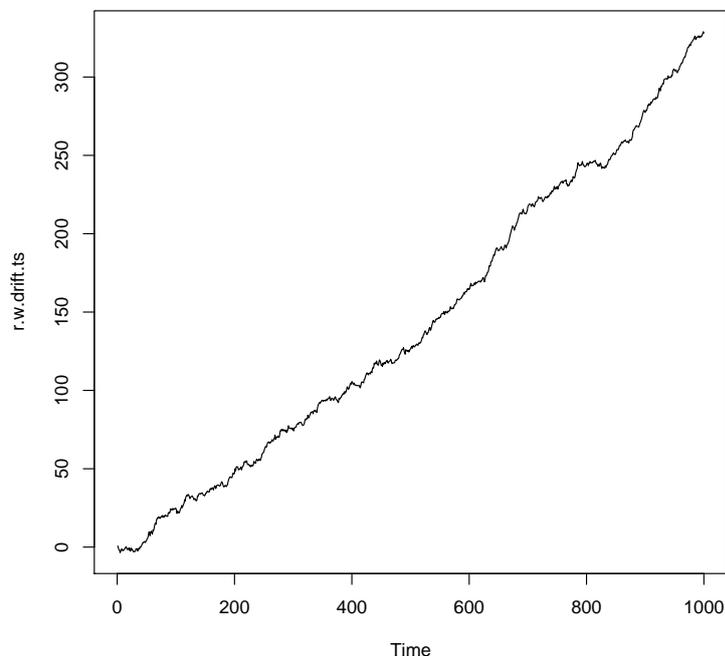


Figura 2.2: Random walk con drift.

Se si volesse stimare il processo autoregressivo associato ai processi simulati `r.w.ts` e `r.w.drift.ts`, si otterrebbe un processo AR(1) con coefficiente prossimo ad 1 ed uno standard error vicino a 0, inoltre, per lo stimatore $\hat{\phi}$ non vale piú la (2.1) in quanto la sua distribuzione è funzione di un *moto browniano* ed i suoi valori critici sono calcolati tramite simulazione²

```
> r.w.ar <- arima(r.w.ts, order = c(1, 0, 0))
> r.w.ar
```

Call:

```
arima(x = r.w.ts, order = c(1, 0, 0))
```

Coefficients:

	ar1	intercept
	0.9990	-19.1510
s.e.	0.0011	17.7982

sigma² estimated as 0.969: log likelihood = -1406.27, aic = 2818.54

In questo caso diremo che i processi (2.2) e (2.3) sono dei *processi integrati di ordine 1*, che si indicano brevemente con $I(1)$, in quanto basta differenziarli una volta per ottenere un processo stazionario. Piú in generale, diremo che un *processo è integrato*

²Per ulteriori chiarimenti ed approfondimenti si veda [Hamilton \(1994\)](#) e [Fuller \(1976\)](#).

di ordine d , lo si indica con $I(d)$, se è necessario differenziarlo d volte per renderlo stazionario.

Nel caso della simulazione del random walk, con la funzione `diff` è possibile differenziare la serie.

```
> d.rw <- diff(r.w.ts)
> d.rw.ar <- arima(d.rw, order = c(1, 0, 0))
> d.rw.ar
```

```
Call:
arima(x = d.rw, order = c(1, 0, 0))
```

```
Coefficients:
      ar1  intercept
      0.0655  -0.0479
s.e.  0.0316   0.0332
```

```
sigma^2 estimated as 0.9629:  log likelihood = -1398.63,  aic = 2803.25
```

```
> acf(resid(d.rw.ar))
> pacf(resid(d.rw.ar))
```

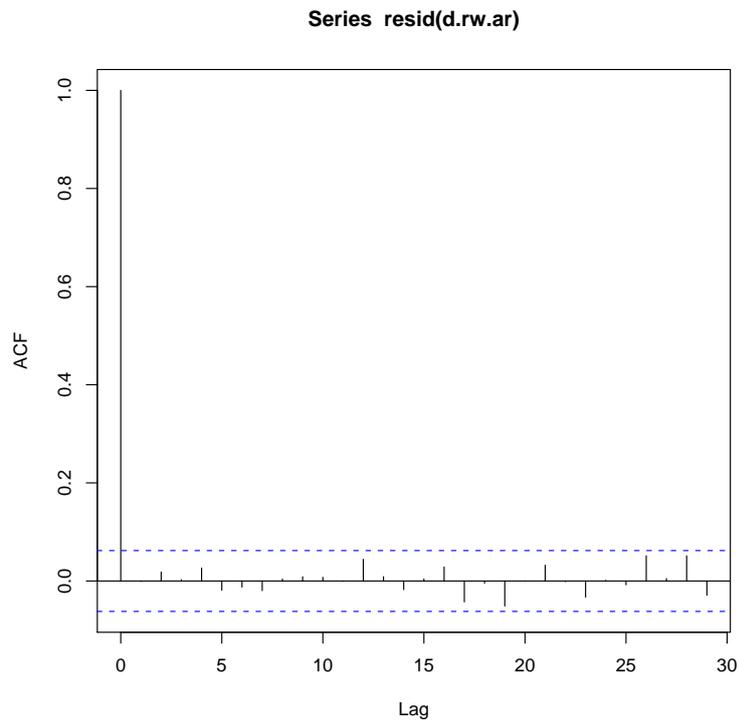


Figura 2.3: Funzione di autocorrelazione dei residui del processo random walk differenziato.

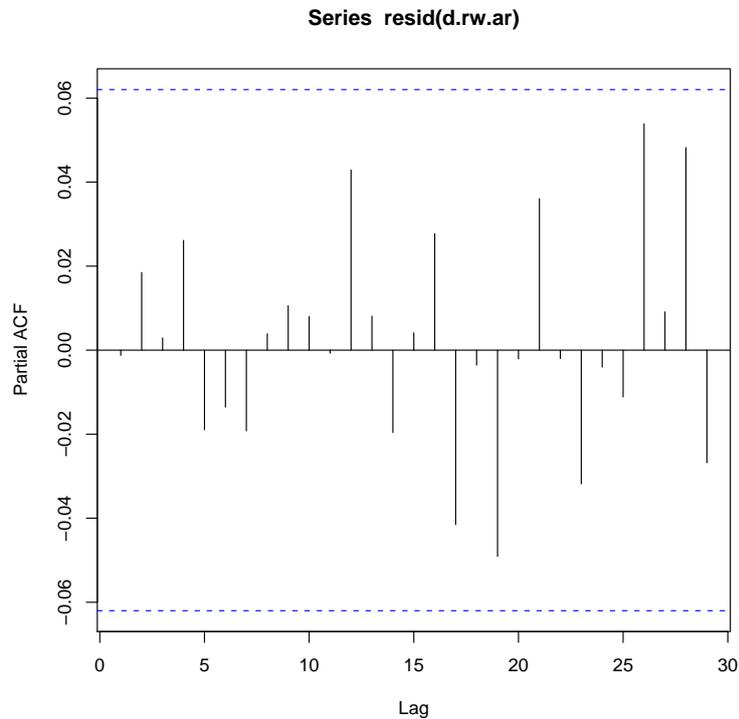


Figura 2.4: Funzione di autocorrelazione parziale dei residui del processo random walk differenziato.

Come si può notare dalla serie nelle differenze si ottiene una stima del coefficiente minore di 1, che indica la stazionarietà del processo, confermata dai grafici (2.3) e (2.4) che mostrano l'assenza di autocorrelazione nei residui del modello. Il random walk con drift è stazionario nelle differenze, come si può notare dal parametro stimato per mezzo della funzione `arima` (in quanto processo $I(1)$), ma è *mean reverting* sulla costante 0.3, come si evince dalla figura (2.5), che è il drift introdotto nel processo.

```
> d.rw.drift <- diff(r.w.drift.ts)
> d.rw.drift.ar <- arima(d.rw.drift, order = c(1, 0, 0))
> d.rw.drift.ar
```

```
Call:
arima(x = d.rw.drift, order = c(1, 0, 0))
```

```
Coefficients:
      ar1  intercept
-0.0433    0.2831
s.e.    0.0317    0.0303
```

```
sigma^2 estimated as 0.9991: log likelihood = -1417.08, aic = 2840.16
```

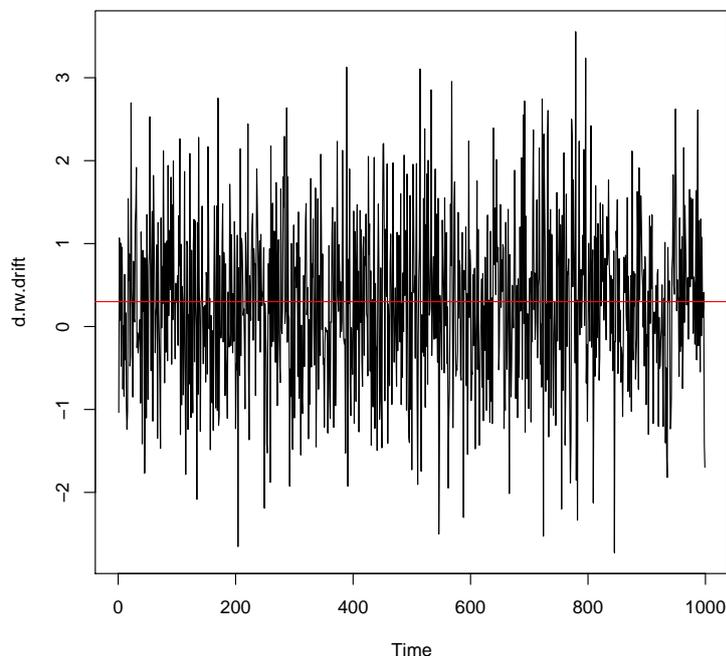


Figura 2.5: Serie differenziata del processo random walk con drift.

```
> plot.ts(d.rw.drift)
> abline(a = 0.3, b = 0, col = c(2), lty = 1)
```

Per il caso multivariato è possibile estendere quanto detto sul caso univariato, piú esattamente un processo multivariato è integrato se almeno una serie è integrata. In questa casi, per stimare il VAR si procede a differenziare tutte le serie e quindi lavorare sulle differenze³. Se, oltre alla non stazionarietà in media, le serie sono non stazionarie in varianza, si è soliti lavorare sulle differenze dei logaritmi cioè sui tassi.

³Nel caso di serie storiche multivariate integrate, in base a quanto detto, è possibile che su 4 serie solamente una serie sia integrata e le rimanenti 3 stazionarie. Nel procedere con la stima del VAR si devono differenziare le 4 serie, motivo per cui si preferisce procedere con *l'analisi della cointegrazione* (argomento del cap. 4).

Capitolo 3

Test per le radici unitarie

Una caratteristica delle serie economiche e finanziarie è quella di essere non stazionarie, come si è accennato precedentemente, per applicare l'approccio Box-Jenkins è necessario trasformare la serie in modo da renderla stazionaria. Alcune delle procedure più comuni usate per rendere la serie stazionaria è quello di differenziarla, applicare i logaritmi oppure le differenze dei logaritmi.

Nel caso di due o più processi I(1), la teoria economica e finanziaria afferma che tra i processi possa sussistere una relazione di equilibrio di lungo periodo attorno alla quale le serie si muovono. Prima di stimare la relazione di equilibrio occorre verificare che le serie oggetto di studio siano integrate; i test per le radici unitarie consentono tale controllo. Successivamente, tramite l'analisi della cointegrazione, si procede alla stima dell'equilibrio comune alle serie.

Per comprendere meglio i test per le radici unitarie ed i test per la stazionarietà si consideri la seguente decomposizione di una serie storica y_t (Zivot and Wang, 2006)

$$y_t = TD_t + z_t \quad (3.1)$$

$$TD_t = k + \delta t \quad (3.2)$$

$$z_t = \phi z_{t-1} + \epsilon_t \quad (3.3)$$

in cui le $\epsilon_t \sim WN(0, \sigma^2)$, TD_t è un trend (lineare) deterministico, k e δ sono delle costanti e z_t è un processo AR(1) con $|\phi| < 1$, quindi stazionario intorno al trend deterministico TD_t . Quando $\phi = 1$, si ha che il processo z_t è I(1) mentre y_t diventa un processo I(1) con drift. In figura (3.1), è riportato l'andamento del sistema (3.1-3.3) simulato dal quale si evince come il processo I(0) sia "reverting" sul trend deterministico a differenza del processo I(1).

```
> z <- arima.sim(250, model = list(ar = c(0.3)))
> trend.det <- 5 + 0.1 * (1:250)
> rnd.walk <- ts(cumsum(rnorm(250)))
> y.I0 <- trend.det + z
> y.I1 <- trend.det + rnd.walk
> ts.plot(y.I0, y.I1, trend.det, col = c(1, 2, 3), lty = c(1),
+       main = "Simulazione di serie I(0) e serie I(1)
+       con trend deterministico")
> legend(0, 30, c("serie y.I(0)", "serie y.I(1)"),
```

```

+ "trend deterministico"),
+ col = c(1, 2, 3), lty = c(1))

```

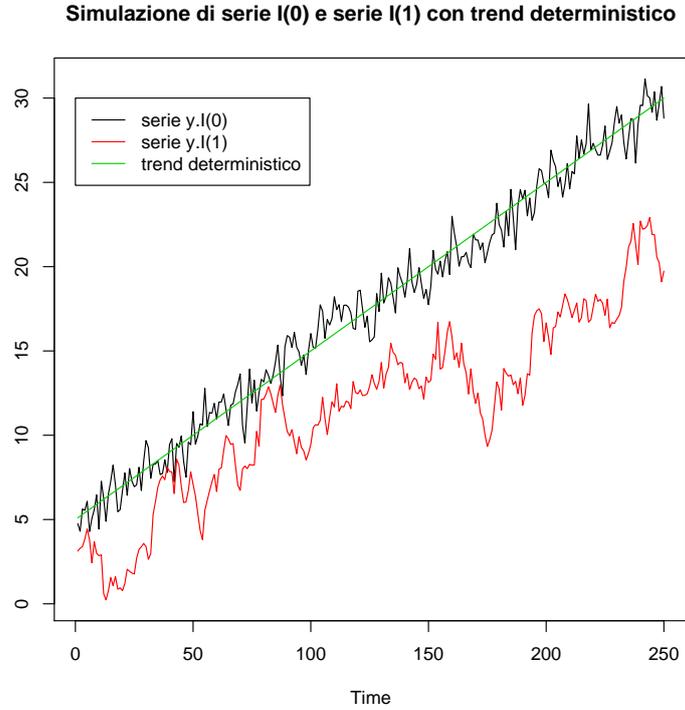


Figura 3.1: Simulazione di serie I(0) e I(1) intorno ad un trend deterministico.

I test per le radici unitarie fanno riferimento ad un sistema d'ipotesi differente rispetto a quello usato dai test per la stazionarietà. In particolare, i test per le radici unitarie considerano il seguente sistema d'ipotesi:

$$\begin{cases} H_0 : \phi = 1 \\ H_1 : |\phi| < 1 \end{cases} \quad (3.4)$$

Quindi sotto l'ipotesi nulla il polinomio caratteristico di un processo AR(1): $\phi(z) = (1 - \phi z) = 0$, ha una radice uguale all'unità. Mentre, i test di stazionarietà fanno riferimento al seguente sistema d'ipotesi:

$$\begin{cases} H_0 : |\phi| < 1 \\ H_1 : \phi = 1 \end{cases} \quad (3.5)$$

Le statistiche test per saggiare i sistemi d'ipotesi (3.4) (3.5), hanno una distribuzione nulla asintotica non standard e non normale, infatti, le loro distribuzioni sono funzioni di moti Browniani standard i cui valori critici vengono solitamente tabulati tramite simulazione.

3.1 Il test di Dickey–Fuller.

Il test ADF (*Augmented Dickey–Fuller*) è un test univariato ed assume che una parte del processo y_t abbia una struttura ARMA, per cui è stimata la seguente regressione:

$$y_t = \beta' \mathbf{D}_t + \phi y_{t-1} + \sum_{j=1}^p \Delta y_{t-j} + \epsilon_t. \quad (3.6)$$

\mathbf{D}_t è un vettore che contiene termini deterministici e β' è un parametro da stimare; le p variabili differenziate ritardate, Δy_{t-j} , descrivono la struttura degli errori del processo, il valore di p è scelto in modo da rendere serialmente incorrelati gli errori ϵ_t , che è anche assunto essere omoschedastico.

La scelta di p troppo piccolo causerà una distorsione del test dovuta alla rimanente correlazione seriale degli errori, mentre un valore elevato di p avrà ripercussioni sulla potenza del test. Un metodo pratico per stimare il valore di p , suggerito da [Schwert \(1989\)](#), è:

$$p_{max} = \left\lceil 12 \left(\frac{T}{100} \right)^{\frac{1}{4}} \right\rceil. \quad (3.7)$$

Per poter eseguire il test ADF in R si può fare ricorso o al pacchetto `vars` oppure al pacchetto `tseries`. Ambedue i pacchetti eseguono il test ADF correttamente, la funzione `ur.df` del pacchetto `urca` fornisce delle informazioni più dettagliate, in quanto è possibile applicare la funzione `summary`. Mentre la funzione `adf.test` del pacchetto `tseries` esegue il test ADF, calcolandosi per default il numero di lags da includere, e come risultato fornisce: il p-value, il valore della statistica test e l'ipotesi alternativa che si è specificata. Di seguito vengono implementati i test ADF, dei due pacchetti, sulla serie dei tassi di interesse.

```
> summary(ur.df(tassi.ts[, 1], lags = 6))

#####
# Augmented Dickey-Fuller Test Unit Root Test #
#####

Test regression none

Call:
lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)

Residuals:
    Min       1Q   Median       3Q      Max
-1.35448 -0.10181  0.02414  0.13324  0.84262

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
z.lag.1         -0.003373   0.002366  -1.426  0.155089
z.diff.lag1      0.625852   0.053030  11.802 < 2e-16 ***
z.diff.lag2     -0.282075   0.060060  -4.697  4.12e-06 ***
z.diff.lag3      0.202377   0.059168   3.420  0.000717 ***
```

```

z.diff.lag4 -0.132636  0.057900  -2.291  0.022705  *
z.diff.lag5  0.247597  0.053954   4.589  6.68e-06  ***
z.diff.lag6 -0.166149  0.046800  -3.550  0.000450  ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.2323 on 285 degrees of freedom
Multiple R-squared: 0.392,      Adjusted R-squared: 0.3771
F-statistic: 26.26 on 7 and 285 DF,  p-value: < 2.2e-16

```

Value of test-statistic is: -1.4256

```

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62

```

```
> summary(ur.df(tassi.ts[, 2], lags = 7))
```

```

#####
# Augmented Dickey-Fuller Test Unit Root Test #
#####

```

Test regression none

```

Call:
lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)

```

```

Residuals:
      Min       1Q   Median       3Q      Max
-1.00643 -0.13120  0.00952  0.16865  0.87561

```

```

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
z.lag.1      -0.003231   0.002397  -1.348  0.17889
z.diff.lag1   0.513571   0.057487   8.934 < 2e-16 ***
z.diff.lag2  -0.134914   0.063046  -2.140  0.03322 *
z.diff.lag3   0.163913   0.059213   2.768  0.00601 **
z.diff.lag4  -0.152944   0.058063  -2.634  0.00890 **
z.diff.lag5   0.152550   0.057523   2.652  0.00845 **
z.diff.lag6  -0.136450   0.056501  -2.415  0.01637 *
z.diff.lag7   0.109508   0.051243   2.137  0.03345 *
---

```

```

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.2515 on 283 degrees of freedom
Multiple R-squared: 0.2699,      Adjusted R-squared: 0.2493
F-statistic: 13.08 on 8 and 283 DF,  p-value: 4.774e-16

```

```

Value of test-statistic is: -1.3475

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62

> summary(ur.df(tassi.ts[, 3], type = "drift", lags = 2))

#####
# Augmented Dickey-Fuller Test Unit Root Test #
#####

Test regression drift

Call:
lm(formula = z.diff ~ z.lag.1 + 1 + z.diff.lag)

Residuals:
      Min       1Q   Median       3Q      Max
-1.17358 -0.16067 -0.02462  0.18508  0.94221

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.087791   0.046944   1.870   0.0625 .
z.lag.1     -0.015610   0.006271  -2.489   0.0134 *
z.diff.lag1  0.420755   0.057199   7.356 1.93e-12 ***
z.diff.lag2 -0.136478   0.057099  -2.390   0.0175 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.272 on 292 degrees of freedom
Multiple R-squared:  0.1763,    Adjusted R-squared:  0.1678
F-statistic: 20.83 on 3 and 292 DF,  p-value: 2.932e-12

```

```

Value of test-statistic is: -2.4894 4.0592

Critical values for test statistics:
      1pct  5pct 10pct
tau2 -3.44 -2.87 -2.57
phi1  6.47  4.61  3.79

> summary(ur.df(tassi.ts[, 4], type = "drift", lags = 2))

#####
# Augmented Dickey-Fuller Test Unit Root Test #
#####

Test regression drift

```

```

Call:
lm(formula = z.diff ~ z.lag.1 + 1 + z.diff.lag)

Residuals:
    Min       1Q   Median       3Q      Max
-0.92013 -0.16069 -0.01365  0.14591  0.86460

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.087908   0.048503   1.812  0.07094 .
z.lag.1      -0.014627   0.006188  -2.364  0.01874 *
z.diff.lag1  0.419749   0.056603   7.416 1.32e-12 ***
z.diff.lag2 -0.158707   0.056516  -2.808  0.00532 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2414 on 292 degrees of freedom
Multiple R-squared:  0.1768,    Adjusted R-squared:  0.1683
F-statistic: 20.91 on 3 and 292 DF,  p-value: 2.689e-12

```

Value of test-statistic is: -2.3639 3.968

```

Critical values for test statistics:
      1pct  5pct 10pct
tau2 -3.44 -2.87 -2.57
phi1  6.47  4.61  3.79

```

Il test ADF conferma l'ipotesi di non stazionarietà per la serie dei tassi di interesse¹, in particolare si può notare come i tassi a 3 mesi ed 1 anno abbiano una struttura di autocorrelazione molto intensa, in quanto è stato necessario introdurre le differenze ritardate fino al 7 ordine. Mentre i tassi a lungo periodo, diversamente da quelli a breve periodo, hanno una struttura di autocorrelazione più debole ma con la presenza del drift.

```

> library(tseries)
> ADFtseries <- apply(tassi.ts, 2, adf.test)
> ADFtseries

```

\$mesi3

Augmented Dickey-Fuller Test

data: newX[, i]

¹L'ipotesi che i tassi di interesse siano dei processi I(1) non è vera in senso stretto, poiché un processo I(1) non è limitato, mentre i tassi di interesse sono limitati inferiormente dallo zero. Nonostante ciò, le serie analizzate hanno un andamento che è più simile ad un processo I(1) piuttosto che I(0).

```
Dickey-Fuller = -2.4321, Lag order = 6, p-value = 0.3941
alternative hypothesis: stationary
```

```
$anno1
```

```
Augmented Dickey-Fuller Test
```

```
data: newX[, i]
Dickey-Fuller = -2.9076, Lag order = 6, p-value = 0.1937
alternative hypothesis: stationary
```

```
$anni7
```

```
Augmented Dickey-Fuller Test
```

```
data: newX[, i]
Dickey-Fuller = -3.8963, Lag order = 6, p-value = 0.01459
alternative hypothesis: stationary
```

```
$anni20
```

```
Augmented Dickey-Fuller Test
```

```
data: newX[, i]
Dickey-Fuller = -4.0475, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

Ambedue i test confermano l'ipotesi di non stazionarietà della serie dei tassi d'interesse.

3.2 Il test di Phillips–Perron.

Il test di [Phillips \(1987\)](#) e [Phillips and Perron \(1988\)](#) per saggiare l'ipotesi nulla di presenza di radici unitarie, si basa sulla seguente relazione:

$$y_t = \beta' \mathbf{D}_t + \phi y_{t-1} + u_t \quad (3.8)$$

in cui u_t è un processo $I(0)$ che può essere eteroschedastico. In pratica, la principale differenza tra il test ADF ed il test PP riguarda l'assunzione di omoschedasticità degli errori. Il test ADF considera una struttura ARMA nella (3.6) per spiegare la dinamica degli errori assumendoli omoschedastici, mentre il test PP nell'equazione (3.8) ignora qualsiasi tipo di correlazione seriale negli errori con possibilità di eteroschedasticità.

Un vantaggio del test di Phillips–Perron rispetto al test ADF è che il test PP risulta essere più robusto nell'ipotesi di eteroschedasticità delle u_t , inoltre non è sensibile alla specificazione del numero di variabili lagged da inserire.

Le statistiche test utilizzate nel test PP sono due, che vengono utilizzate in base

all'ipotesi che sia presente nella (3.8) la parte deterministica o meno. Di seguito è applicato il test di Phillips–Perron alla struttura dei tassi d'interesse. Nell'applicare la funzione `pp.test` è sufficiente controllare il valore del p-value che restituisce la funzione, per decidere se accettare o meno l'ipotesi nulla.

```
> PPTseries <- apply(tassi.ts, 2, pp.test)
> PPTseries
```

```
$mesi3
```

```
Phillips-Perron Unit Root Test
```

```
data: newX[, i]
Dickey-Fuller Z(alpha) = -12.3072, Truncation lag parameter = 5,
p-value = 0.4203
alternative hypothesis: stationary
```

```
$anno1
```

```
Phillips-Perron Unit Root Test
```

```
data: newX[, i]
Dickey-Fuller Z(alpha) = -13.7868, Truncation lag parameter = 5,
p-value = 0.3373
alternative hypothesis: stationary
```

```
$anni7
```

```
Phillips-Perron Unit Root Test
```

```
data: newX[, i]
Dickey-Fuller Z(alpha) = -20.6972, Truncation lag parameter = 5,
p-value = 0.05966
alternative hypothesis: stationary
```

```
$anni20
```

```
Phillips-Perron Unit Root Test
```

```
data: newX[, i]
Dickey-Fuller Z(alpha) = -21.5699, Truncation lag parameter = 5,
p-value = 0.04816
alternative hypothesis: stationary
```

Nel caso della funzione `ur.pp` è preso in considerazione il valore della statistica `test` che restituisce il `test`² oppure si può esaminare la significatività dei coefficienti

²Nel caso in esame sono tutti valori negativi quindi si accetta la nulla.

della variabile ritardata, che restituisce la funzione `summary` applicata all'oggetto di classe `ur.pp` (che nel caso in esame sono tutti significativi e prossimi ad 1).

```
> PPurca <- apply(tassi.ts, 2, ur.pp)
> lapply(PPurca, summary)
```

```
$mesi3
```

```
#####
# Phillips-Perron Unit Root Test #
#####
```

```
Test regression with intercept
```

```
Call:
lm(formula = y ~ y.l1)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-2.69894 -0.10406  0.01776  0.15853  1.54340
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.088817   0.042087    2.11  0.0357 *
y.l1         0.978931   0.006949  140.88 <2e-16 ***
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.313 on 296 degrees of freedom
Multiple R-squared: 0.9853, Adjusted R-squared: 0.9853
F-statistic: 1.985e+04 on 1 and 296 DF, p-value: < 2.2e-16
```

```
Value of test-statistic, type: Z-alpha is: -8.3939
```

```
      aux. Z statistics
Z-tau-mu      2.1567
```

```
$anno1
```

```
#####
# Phillips-Perron Unit Root Test #
#####
```

```
Test regression with intercept
```

```
Call:
```

```
lm(formula = y ~ y.l1)
```

```
Residuals:
```

	Min	1Q	Median	3Q	Max
	-1.616843	-0.139560	-0.009038	0.168793	0.925364

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.099062	0.042661	2.322	0.0209 *
y.l1	0.977929	0.006546	149.397	<2e-16 ***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.3119 on 296 degrees of freedom  
Multiple R-squared: 0.9869, Adjusted R-squared: 0.9869  
F-statistic: 2.232e+04 on 1 and 296 DF, p-value: < 2.2e-16
```

```
Value of test-statistic, type: Z-alpha is: -8.5994
```

	aux. Z statistics
Z-tau-mu	2.2781

```
$anni7
```

```
#####  
# Phillips-Perron Unit Root Test #  
#####
```

```
Test regression with intercept
```

```
Call:
```

```
lm(formula = y ~ y.l1)
```

```
Residuals:
```

	Min	1Q	Median	3Q	Max
	-1.34391	-0.17751	-0.03227	0.20309	0.91998

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.104275	0.049884	2.09	0.0374 *
y.l1	0.980553	0.006598	148.62	<2e-16 ***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.295 on 296 degrees of freedom  
Multiple R-squared: 0.9868, Adjusted R-squared: 0.9867
```

F-statistic: 2.209e+04 on 1 and 296 DF, p-value: < 2.2e-16

Value of test-statistic, type: Z-alpha is: -7.0248

```
      aux. Z statistics
Z-tau-mu      2.0974
```

\$anni20

```
#####
# Phillips-Perron Unit Root Test #
#####
```

Test regression with intercept

```
Call:
lm(formula = y ~ y.l1)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-1.0680 -0.1650 -0.0280  0.1729  0.9115
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.113529   0.051709   2.196   0.0289 *
y.l1         0.980630   0.006537 150.004 <2e-16 ***
---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2635 on 296 degrees of freedom
Multiple R-squared: 0.987, Adjusted R-squared: 0.987
F-statistic: 2.25e+04 on 1 and 296 DF, p-value: < 2.2e-16

Value of test-statistic, type: Z-alpha is: -6.8453

```
      aux. Z statistics
Z-tau-mu      2.1792
```

3.3 Il test KPSS.

Il test di Kwiatkowski *et al.* (1992)³ è un test di stazionarietà, per cui il sistema d'ipotesi che s'intende saggiare è del tipo (3.5), in cui la serie si assume essere I(0) contro l'alternativa che sia non stazionaria o che abbia delle radici unitarie. Il test

³Indicato con l'acronimo di test KPSS.

KPSS si basa sul seguente modello:

$$y_t = \beta' \mathbf{D}_t + \mu_t + u_t \quad (3.9)$$

$$\mu_t = \mu_{t-1} + \epsilon_t \quad (3.10)$$

dove $\epsilon_t \sim \text{WN}(0, \sigma_\epsilon^2)$, \mathbf{D}_t indica la parte deterministica (costante o costante piú trend), u_t è un processo I(0) che può essere eteroschedastico, mentre μ_t è un processo random walk e β è un parametro da stimare.

Il sistema d'ipotesi che si saggia con il test KPSS è il seguente:

$$\begin{cases} H_0 : \sigma_\epsilon^2 = 0 \\ H_1 : \sigma_\epsilon^2 > 0 \end{cases} \quad (3.11)$$

Sotto H_0 , si ha che y_t è un processo stazionario in quanto μ_t è una costante. La statistica test del KPSS è il moltiplicatore di Lagrange (LM) utilizzato per saggiare il sistema (3.11).

Nel loro studio gli autori dimostrarono che il test KPSS converge ad una funzione di un moto Browniano standard che dipende dalla forma dei termini deterministici \mathbf{D}_t ma non dai valori dei loro coefficienti β' . Il test KPSS è un test sulla coda destra della sua distribuzione, cosicchè l'ipotesi nulla di stazionarietà è rigettata al livello $100 \alpha \%$ se la statistica test KPSS è maggiore del $100 (1 - \alpha) \%$ quantile della sua distribuzione. In R il test KPSS è eseguito tramite la funzione `ur.kpss` del pacchetto `urca`, oppure tramite la funzione `kpss.test` del pacchetto `tseries`. In ambedue i casi le informazioni che si estrapolano non sono molto differenti.

```
> KPSSseries <- apply(tassi.ts, 2, kpss.test)
> KPSSseries
```

```
$mesi3
```

```
      KPSS Test for Level Stationarity
```

```
data: newX[, i]
KPSS Level = 4.7626, Truncation lag parameter = 3, p-value = 0.01
```

```
$anno1
```

```
      KPSS Test for Level Stationarity
```

```
data: newX[, i]
KPSS Level = 5.0291, Truncation lag parameter = 3, p-value = 0.01
```

```
$anni7
```

```
      KPSS Test for Level Stationarity
```

```
data: newX[, i]
KPSS Level = 6.1357, Truncation lag parameter = 3, p-value = 0.01
```

\$anni20

KPSS Test for Level Stationarity

data: newX[, i]
KPSS Level = 6.1883, Truncation lag parameter = 3, p-value = 0.01

```
> KPSSurca <- apply(tassi.ts, 2, ur.kpss)
> lapply(KPSSurca, summary)
```

\$mesi3

```
#####
# KPSS Unit Root Test #
#####
```

Test is of type: mu with 5 lags.

Value of test-statistic is: 3.2453

Critical value for a significance level of:

	10pct	5pct	2.5pct	1pct
critical values	0.347	0.463	0.574	0.739

\$anno1

```
#####
# KPSS Unit Root Test #
#####
```

Test is of type: mu with 5 lags.

Value of test-statistic is: 3.4233

Critical value for a significance level of:

	10pct	5pct	2.5pct	1pct
critical values	0.347	0.463	0.574	0.739

\$anni7

```
#####
# KPSS Unit Root Test #
#####
```

Test is of type: mu with 5 lags.

Value of test-statistic is: 4.1661

Critical value for a significance level of:
10pct 5pct 2.5pct 1pct
critical values 0.347 0.463 0.574 0.739

\$anni20

KPSS Unit Root Test #
#####

Test is of type: mu with 5 lags.

Value of test-statistic is: 4.2027

Critical value for a significance level of:
10pct 5pct 2.5pct 1pct
critical values 0.347 0.463 0.574 0.739

Il test KPSS rigetta l'ipotesi di stazionarietà per la serie dei tassi di interesse, ad un livello di significatività dell'1%.

Capitolo 4

Serie storiche cointegrate.

Come si è visto nei paragrafi precedenti, la metodologia dei *vector autoregressive models* è applicata a processi di tipo I(0), per cui quando il processo è I(1) occorre effettuare delle trasformazioni della serie originaria per renderla stazionaria. Procedendo alla manipolazione della serie, quale differenze prime oppure differenze dei logaritmi, si possono perdere delle informazioni rilevanti che sono contenute nella serie nei livelli.

Nei processi di tipo I(1) spesso si osserva una relazione di equilibrio di lungo periodo tra le variabili, che nella maggior parte dei casi è confermata dalla teoria economica. Questa relazione di equilibrio di lungo periodo può essere espressa tramite una relazione lineare tra le variabile oggetto di studio. Ad esempio, si supponga che le variabili oggetto di studio siano rappresentate dalla serie $\mathbf{Y}_t = (y_{1t}, \dots, y_{kt})'$. Una relazione di equilibrio di lungo periodo può essere la seguente:

$$\boldsymbol{\beta}'\mathbf{Y}_t = \beta_1 y_{1t} + \dots + \beta_k y_{kt} = 0,$$

dove $\boldsymbol{\beta} = (\beta_1, \dots, \beta_k)'$ può essere conosciuto a priori oppure stimato. Empiricamente, questa relazione può non essere soddisfatta in qualsiasi periodo, per cui è plausibile che vi sia un disequilibrio del tipo $\boldsymbol{\beta}'\mathbf{Y}_t = z_t$. Il processo z_t rappresenta la relazione di disequilibrio tra le variabili della serie \mathbf{Y}_t , quindi se sussiste realmente un equilibrio tra le variabili è attendibile che il processo z_t sia un processo stabile e che le variabili \mathbf{Y}_t si muovano insieme, o meglio ancora che siano guidate da un unico *trend stocastico comune*. Diremo che le variabili \mathbf{Y}_t sono *cointegrate* se esiste una *combinazione lineare che rende il processo z_t stazionario*. Le variabili del processo k-dimensionale \mathbf{Y}_t sono dette *cointegrate*, mentre il vettore $\boldsymbol{\beta}$ è detto *vettore di cointegrazione* (Granger, 1981; Engle and Granger, 1987). Un processo cointegrato bivariato con un vettore di cointegrazione è rappresentato nella figura (4.1), la cui simulazione in R è stata generata dalle seguenti istruzioni.

Simulazione di un sistema cointegrato bivariato

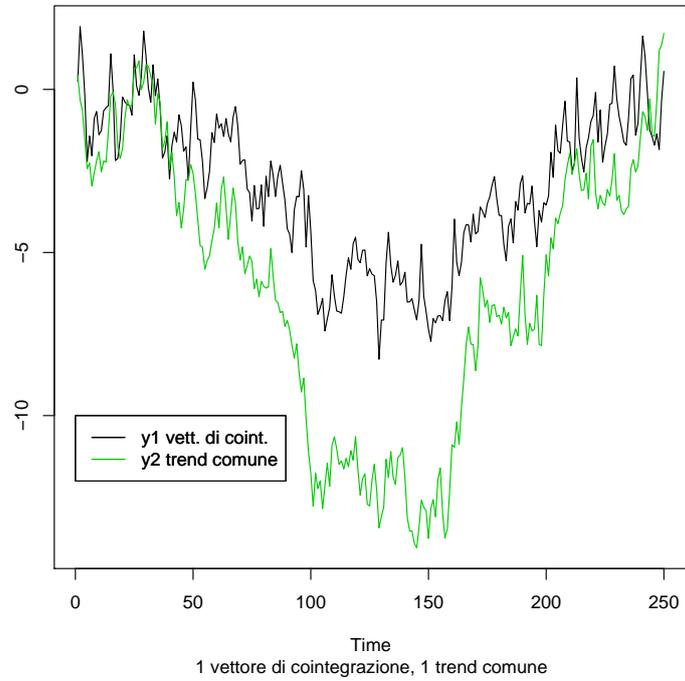
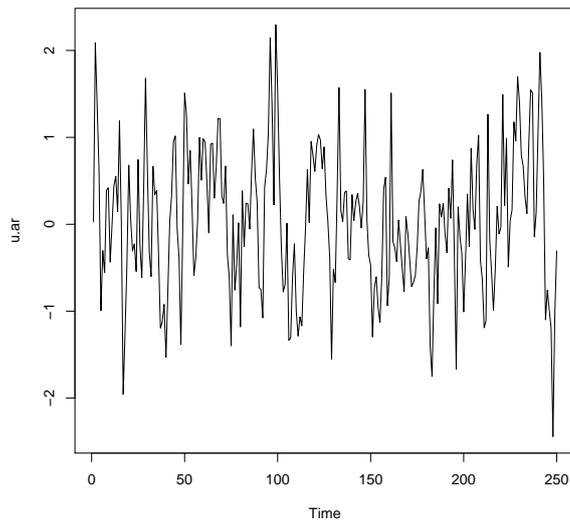


Figura 4.1: Simulazione di un sistema bivariato con un vettore di cointegrazione.

Residui di cointegrazione—relazione di disequilibrio



```
> sigma <- diag(0.5, 2)  
> e <- mvrnorm(250, rep(0, 2), sigma)
```

```

> u.ar <- arima.sim(250, model = list(ar = c(0.5)), innov = e[,
+   1])
> y2 <- cumsum(e[, 2])
> y1 <- 0.5 * y2 + u.ar
> ts.plot(y1, y2, col = c(1, 3), main = "Simulazione di un sistema cointegrato
+ bivariato", sub = "1 vettore di cointegrazione, 1 trend comune")
> legend(0,-10, legend = c("y1 rel. di coint.",
+   "y2 trend comune"), col = c(1, 3), lty = c(1))
> ts.plot(u.ar, main = "Residui di cointegrazione-relazione di disequilibrio")

```

Il vettore di cointegrazione non è unico, in quanto basta moltiplicarlo per una costante diversa da zero per ottenere un ulteriore vettore di cointegrazione, proporzionale al primo. Per determinare in maniera univoca il vettore di cointegrazione si procede alla sua *normalizzazione*, per cui se la relazione di cointegrazione è la seguente:

$$\beta_1 y_{1t} = \beta_2 y_{2t} + \beta_3 y_{3t},$$

è normalizzata rispetto alla variabile y_{1t} , ottenendo la seguente relazione

$$y_{1t} = \beta_2^* y_{2t} + \beta_3^* y_{3t},$$

in cui $\beta_2^* = \frac{\beta_2}{\beta_1}, \beta_3^* = \frac{\beta_3}{\beta_1}$ in modo tale che

$$y_{1t} - \beta_2^* y_{2t} - \beta_3^* y_{3t} \sim I(0).$$

Questa prassi di normalizzare il vettore di cointegrazione è effettuata da tutti i software econometrici ed anche da R.

Supponiamo che valga il VAR(p) (1.7) il cui polinomio caratteristico ha una radice sul cerchio unitario, quindi almeno una variabile è I(1). Si dice che il processo è *cointegrato di rango r* se la matrice:

$$\mathbf{\Pi} = -(I_k - A_1 - \dots - A_p) \quad (4.1)$$

ha rango $r < k$. In R è implementata la procedura sviluppata da [Johansen \(1988, 1995\)](#) per determinare il rango di cointegrazione ed anche la metodologia implementata da [Phillips and Ouliaris \(1990\)](#). Prima di approfondire le funzioni utilizzate in R per condurre l'analisi della cointegrazione, è necessario introdurre i *modelli a correzione d'errore (error correction models)*.

4.1 Modelli a correzione d'errore.

Consideriamo un processo trivariato integrato di ordine 1 del tipo $\mathbf{Y}_t = (y_{1t}, y_{2t}, y_{3t})$ e supponiamo che sia cointegrato con vettore di cointegrazione $\boldsymbol{\beta} = (1, -\beta_2, -\beta_3)'$, in modo tale che $\boldsymbol{\beta}'\mathbf{Y}_t = y_{1t} - \beta_2 y_{2t} - \beta_3 y_{3t} \sim I(0)$. Si dimostra, tramite il famoso **teorema di Rappresentazione di Granger** ([Engle and Granger, 1987](#)), che *se le serie sono cointegrate allora esiste una rappresentazione in forma ECM, e se esiste una forma ECM e le serie sono integrate allora sono cointegrate*. Questo famoso teorema, dalla dimostrazione abbastanza complessa, definisce un legame tra le serie cointegrate e la loro rappresentazione in ECM.

La rappresentazione del sistema trivariato cointegrato, $\beta' \mathbf{Y}_t$, in forma ECM è la seguente:

$$\Delta y_{1t} = c_1 + \alpha_1(y_{1t-1} - \beta_2 y_{2t-1} - \beta_3 y_{3t-1}) + \gamma_{11} \Delta y_{1t-1} + \gamma_{12} \Delta y_{2t-1} + \gamma_{13} \Delta y_{3t-1} + u_{1t} \quad (4.2)$$

$$\Delta y_{2t} = c_2 + \alpha_2(y_{1t-1} - \beta_2 y_{2t-1} - \beta_3 y_{3t-1}) + \gamma_{21} \Delta y_{1t-1} + \gamma_{22} \Delta y_{2t-1} + \gamma_{23} \Delta y_{3t-1} + u_{2t} \quad (4.3)$$

$$\Delta y_{3t} = c_3 + \alpha_3(y_{1t-1} - \beta_2 y_{2t-1} - \beta_3 y_{3t-1}) + \gamma_{31} \Delta y_{1t-1} + \gamma_{32} \Delta y_{2t-1} + \gamma_{33} \Delta y_{3t-1} + u_{3t} \quad (4.4)$$

Il sistema (4.2–4.4) è rappresentato in forma matriciale nel modo seguente:

$$\Delta \mathbf{Y}_t = \mathbf{\Pi} \mathbf{Y}_{t-1} + \mathbf{\Gamma}_1 \Delta \mathbf{Y}_{t-1} + \mathbf{u}_t \quad (4.5)$$

La (4.5) è la rappresentazione del VAR(2) trivariato in forma *vector error correction model*, che è la versione multivariata del test di Dickey–Fuller (3.6).

Se il sistema è cointegrato allora la matrice $\mathbf{\Pi}$ può essere scritta come prodotto tra $\alpha \beta'$, in cui: α di dimensione $(k \times r)$, è chiamata *matrice dei pesi (matrix loading)*, mentre β di dimensione $(k \times r)$, è la matrice di cointegrazione. Il numero di relazioni di cointegrazione nella serie \mathbf{Y}_t è determinato tramite il rango della matrice $\mathbf{\Pi}$ che è indicato con r . In particolare, si può avere che:

- se $\text{rango}(\mathbf{\Pi}) = r = 0$ allora $\Delta \mathbf{Y}_t$ è un VAR(p-1) stazionario;
- se $\text{rango}(\mathbf{\Pi}) = r = k$ allora \mathbf{Y}_t non ha radici unitarie, quindi è un VAR(p) stazionario;
- se $0 < \text{rango}(\mathbf{\Pi}) < k$ allora la serie \mathbf{Y}_t ha r relazioni di cointegrazione e $k - r$ trend comuni. La matrice $\mathbf{\Pi}$, di dimensione $(k \times k)$, può essere decomposta come prodotto di $\alpha \beta'$.

In R l'analisi della cointegrazione è implementata nel pacchetto `urca`, in particolare, la funzione `ca.jo` esegue la procedura di Johansen che consente di determinare il rango di cointegrazione. Questa metodologia utilizza due test: *il test traccia (trace test)* oppure *il test del massimo autovalore (maximum eigenvalue statistic)*, i quali sono entrambi un rapporto di verosimiglianza con distribuzione diversa dalla χ^2 . La distribuzione di questi test statistici dipende dal numero delle relazioni di cointegrazione specificate sotto l'ipotesi nulla H_0 .

Il test traccia $\lambda_{LR}(r_0, k)$, fa riferimento al seguente sistema d'ipotesi:

$$\begin{cases} H_0 : \text{rango}(\mathbf{\Pi}) = r_0 \\ H_1 : r_0 < \text{rango}(\mathbf{\Pi}) < k. \end{cases} \quad (4.6)$$

Il test del massimo autovalore $\lambda_{LR}(r_0, r_0 + 1)$, saggia il seguente sistema d'ipotesi:

$$\begin{cases} H_0 : \text{rango}(\mathbf{\Pi}) = r_0 \\ H_1 : \text{rango}(\mathbf{\Pi}) = r_0 + 1. \end{cases} \quad (4.7)$$

Nel VECM (4.5) può essere aggiunta anche la parte deterministica, che può essere una costante oppure un trend lineare o entrambi¹ La stima del VECM è effettuata

¹Per maggiori dettagli sull'inclusione della parte deterministica nel VECM si rimanda [Lutkepohl \(2005\)](#).

tramite gli OLS oppure, se il processo \mathbf{Y}_t è gaussiano², tramite il metodo della massima verosimiglianza.

Si supponga che il DGP del sistema cointegrato, rappresentato in figura (4.2) i cui codici in R sono riportati di seguito, sia il seguente:

$$y_{1t} = 0.2y_{2t} + 0.8y_{3t} + u_t \quad (4.8)$$

$$y_{2t} \sim I(1); \quad y_{3t} \sim I(1) \quad (4.9)$$

```
> sigma <- diag(0.5, 3)
> e <- mvrnorm(500, rep(0, 3), sigma)
> u.ar <- arima.sim(500, model = list(ar = c(0.2)), innov = e[,
+   1])
> y2 <- cumsum(e[, 2])
> y3 <- cumsum(e[, 3])
> y1 <- 0.2 * y2 + 0.8 * y3 + u.ar
> ts.plot(y1, y2, y3, col = c(2, 3, 4), main = "Simulazione
di un sistema cointegrato trivariato", sub = "1 equilibrio
+   e 2 trend comuni")
> legend(0, -8, legend = c("y1- equilibrio", "y2- trend comune",
+   "y3 - trend comune"), col = c(2, 3, 4), lty = c(1))
> plot(u.ar, main = "Residui cointegrati di y1")
```

²Oppure, analogamente, se $\mathbf{u}_t \sim N(0, \Sigma_u)$.

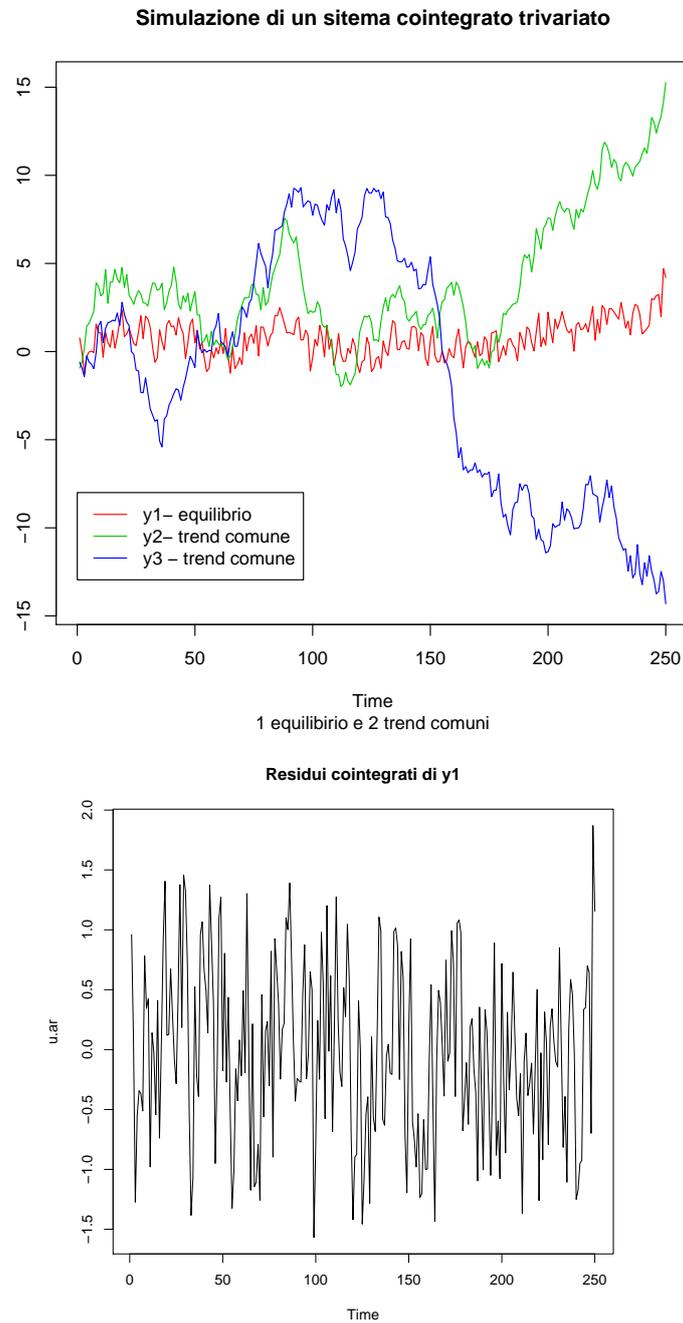


Figura 4.2: Sistema cointegrato trivariato con una relazione di cointegrazione

In questo caso l'innovazione u_t della (4.8) è stata generata come un processo autoregressivo stazionario con coefficiente $\phi = 0.2$, volendo sottolineare che la stima

del VECM dipender  anche dalla caratteristica del processo u_t .

Applicando la funzione `ca.jo` ottengo la stima del rango di cointegrazione del sistema.

```
> coint.sim <- ca.jo(cbind(y1, y2, y3), type = "eigen", ecdet = "none",
+   K = 2, spec = "transitory")
> summary(coint.sim)
```

```
#####
# Johansen-Procedure #
#####
```

Test type: maximal eigenvalue statistic (lambda max) , with linear trend

Eigenvalues (lambda):

```
[1] 0.3084836098 0.0103671758 0.0000305967
```

Values of teststatistic and critical values of test:

	test	10pct	5pct	1pct
r <= 2	0.02	6.50	8.18	11.65
r <= 1	5.19	12.91	14.90	19.19
r = 0	183.70	18.90	21.07	25.75

Eigenvectors, normalised to first column:

(These are the cointegration relations)

	y1.l1	y2.l1	y3.l1
y1.l1	1.0000000	1.0000000	1.0000000
y2.l1	-0.2007047	-0.1076897	-4.736765
y3.l1	-0.7921848	-9.7999852	-1.223227

Weights W:

(This is the loading matrix)

	y1.l1	y2.l1	y3.l1
y1.d	-0.958114045	0.0005172315	9.170458e-05
y2.d	-0.006363464	-0.0012471395	8.921492e-05
y3.d	-0.127082391	0.0012585487	1.011657e-04

All'interno della funzione `ca.jo`   stato specificato:

- la serie storica su cui effettuare l'analisi, `cbind(y1,y2,y3)`. In questo caso   effettuata la normalizzazione rispetto alla prima variabili inserita, quindi `y1`;
- il tipo di test da eseguire, `eigen` oppure `trace`;
- la parte deterministica, `none` oppure `const` o `trend`;
- la lunghezza del VAR nei livelli $K=2$, a cui corrisponde un VECM di lunghezza "k-1";

- L'opzione `transitory` oppure `longrun` è utilizzata per stimare due tipi VECM. Utilizzando `longrun` si stima il seguente VECM:

$$\Delta \mathbf{Y}_t = \Gamma_1 \Delta \mathbf{Y}_{t-1} + \dots + \Gamma_{k-1} \Delta \mathbf{Y}_{t-k+1} + \Pi \mathbf{Y}_{t-k} + \boldsymbol{\mu} + \boldsymbol{\Phi} \mathbf{D}_t + \boldsymbol{\varepsilon}_t; \quad (4.10)$$

dove le $\Gamma_i = -(I - A_1 - \dots - A_i)$, (con A_i matrici del VAR ed $i = 1, \dots, k-1$) sono le matrici che misurano l'impatto di lungo periodo sulla variabile $\Delta \mathbf{Y}_t$ ad una variazione di $\Delta \mathbf{Y}_{t-i}$. Con l'opzione `transitory` si ha:

$$\Delta \mathbf{Y}_t = \Gamma_1 \Delta \mathbf{Y}_{t-1} + \dots + \Gamma_{k-1} \Delta \mathbf{Y}_{t-k+1} + \Pi \mathbf{Y}_{t-1} + \boldsymbol{\mu} + \boldsymbol{\Phi} \mathbf{D}_t + \boldsymbol{\varepsilon}_t; \quad (4.11)$$

in questo caso le matrici $\Gamma_i = -(A_{i+1} + \dots + A_k)$, con ($i = 1, \dots, k-1$), misurano l'impatto immediato di breve periodo sulla variabile $\Delta \mathbf{Y}_t$, e la matrice Π si riferisce alle variabili ritardate al tempo $t - 1$ piuttosto che al tempo $t - k$.

Il valore della statistica test è pari a 5.19, che porta all'accettazione dell'ipotesi nulla che il rango di cointegrazione sia pari a 1. Applicando la funzione `summary` all'oggetto di classe `ca.jo` si ottengono delle informazioni più dettagliate, come ad esempio: il valore della statistica test, il rango di cointegrazione, le relazioni di cointegrazione e la matrice dei pesi o coefficienti di aggiustamento. Tutte queste informazioni possono essere estratte guardando la struttura della classe `ca.jo-class`. La verifica d'ipotesi sul rango di cointegrazione è effettuata tramite la statistica test del massimo autovalore, il cui valore si estrapola con il comando `coint.sim@teststat`, mentre i valori critici si ottengono per mezzo dell'istruzione `coint.sim@cval` da cui si può saggiare il sistema d'ipotesi (4.7).

Con il comando `coint.sim@V` si ottiene la matrice di cointegrazione, normalizzata rispetto ad `y1`, dove nella prima colonna sono riportati i coefficienti stimati del sistema cointegrato da cui si è simulato.

```
> coint.sim@V
           y1.l1      y2.l1      y3.l1
y1.l1  1.0000000  1.0000000  1.0000000
y2.l1 -0.2007047 -0.1076897 -4.736765
y3.l1 -0.7921848 -9.7999852 -1.223227
```

Con il comando `coint.sim@W` si ottiene la matrice dei pesi.

```
> coint.sim@W
           y1.l1      y2.l1      y3.l1
y1.d -0.958114045  0.0005172315  9.170458e-05
y2.d -0.006363464 -0.0012471395  8.921492e-05
y3.d -0.127082391  0.0012585487  1.011657e-04
```

Con il comando `coint.sim@PI` si ottiene la matrice Π della (4.11).

```
> coint.sim@PI
           y1.l1      y2.l1      y3.l1
y1.d -0.957505109  0.191807913  0.75382235
y2.d -0.007521388  0.000988891  0.01715386
y3.d -0.125722677  0.024891303  0.08821523
```

```
> coint.sim@W %*% t(coint.sim@V)

          y1.l1      y2.l1      y3.l1
y1.d -0.957505109 0.191807913 0.75382235
y2.d -0.007521388 0.000988891 0.01715386
y3.d -0.125722677 0.024891303 0.08821523
```

Per controllare che la relazione di cointegrazione sia effettivamente valida si dovrebbe ottenere che la combinazione lineare $\hat{\beta}'\mathbf{Y}_{t-1}$ sia un processo stazionario. Per effettuare tale controllo si procede nel seguente modo.

```
> beta <- coint.sim@V[, 1]
> Yl1 <- coint.sim@ZK
> beta.Yl1 <- t(beta %*% t(Yl1))
> summary(ur.df(beta.Yl1))
```

```
#####
# Augmented Dickey-Fuller Test Unit Root Test #
#####
```

Test regression none

```
Call:
lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)
```

```
Residuals:
      Min       1Q   Median       3Q      Max
-2.0075867 -0.4441779 -0.0002759  0.5445737  2.2274859
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
z.lag.1    -0.838313    0.058418  -14.350  <2e-16 ***
z.diff.lag -0.002163    0.045098   -0.048   0.962
---

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
Residual standard error: 0.7189 on 494 degrees of freedom
Multiple R-squared: 0.4191, Adjusted R-squared: 0.4167
F-statistic: 178.2 on 2 and 494 DF, p-value: < 2.2e-16
```

Value of test-statistic is: -14.3501

Critical values for test statistics:

```
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62
```

```
> summary(ur.kpss(beta.Yl1))
```

```
#####
# KPSS Unit Root Test #
```

```
#####

Test is of type: mu with 5 lags.

Value of test-statistic is: 0.0512

Critical value for a significance level of:
          10pct  5pct  2.5pct  1pct
critical values 0.347 0.463  0.574 0.739

> summary(ur.pp(beta.Y11))

#####
# Phillips-Perron Unit Root Test #
#####

Test regression with intercept

Call:
lm(formula = y ~ y.l1)

Residuals:
    Min       1Q   Median       3Q      Max
-2.04157 -0.47230 -0.03452  0.51108  2.19346

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.03234    0.03223   1.003 0.316265
y.l1         0.15729    0.04448   3.536 0.000444 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7178 on 495 degrees of freedom
Multiple R-squared:  0.02464,    Adjusted R-squared:  0.02267
F-statistic: 12.5 on 1 and 495 DF,  p-value: 0.0004441

Value of test-statistic, type: Z-alpha is: -403.6288

      aux. Z statistics
Z-tau-mu          0.999
```

I due test sulle radici unitarie ed il test di stazionarietà confermano che il processo generato, dalla combinazione lineare tra le relazione di cointegrazione stimate e la serie ritardata è un processo stazionario. Per procedere con la stima del VECM (4.11) si utilizza la funzione `cajools`, la quale esegue una stima dei parametri del VECM con il metodo dei minimi quadrati ordinari.

```

> ecm.stima <- cajools(coint.sim)
> summary(ecm.stima)

Response y1.d :

Call:
lm(formula = substitute(y1.d), data = data.mat)

Residuals:
      Min       1Q   Median       3Q      Max
-3.083980 -0.593154 -0.005293  0.647565  2.862536

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
constant    0.06360    0.04711   1.350   0.178
y1.dl1      0.06289    0.05989   1.050   0.294
y2.dl1      0.07468    0.06364   1.173   0.241
y3.dl1     -0.11569    0.07771  -1.489   0.137
y1.l1     -0.95751    0.07776 -12.314 <2e-16 ***
y2.l1      0.19181    0.01708  11.230 <2e-16 ***
y3.l1      0.75382    0.06247  12.066 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9514 on 491 degrees of freedom
Multiple R-squared:  0.3264,    Adjusted R-squared:  0.3168
F-statistic: 33.98 on 7 and 491 DF,  p-value: < 2.2e-16

Response y2.d :

Call:
lm(formula = substitute(y2.d), data = data.mat)

Residuals:
      Min       1Q   Median       3Q      Max
-1.95339 -0.47213 -0.00954  0.44230  2.45594

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
constant    0.0081261    0.0338571   0.240   0.8104
y1.dl1      0.0411599    0.0430412   0.956   0.3394
y2.dl1      0.0373023    0.0457336   0.816   0.4151
y3.dl1     -0.1435433    0.0558405  -2.571   0.0104 *
y1.l1     -0.0075214    0.0558772  -0.135   0.8930
y2.l1      0.0009889    0.0122745   0.081   0.9358
y3.l1      0.0171539    0.0448940   0.382   0.7026
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Residual standard error: 0.6837 on 491 degrees of freedom
 Multiple R-squared: 0.02441, Adjusted R-squared: 0.0105
 F-statistic: 1.755 on 7 and 491 DF, p-value: 0.09434

Response y3.d :

Call:
 lm(formula = substitute(y3.d), data = data.mat)

Residuals:

	Min	1Q	Median	3Q	Max
	-2.145192	-0.493700	0.006235	0.508887	2.168437

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
constant	0.04540	0.03661	1.240	0.2155
y1.dl1	0.06187	0.04654	1.329	0.1844
y2.dl1	-0.07652	0.04945	-1.547	0.1225
y3.dl1	-0.05575	0.06038	-0.923	0.3563
y1.l1	-0.12572	0.06042	-2.081	0.0380 *
y2.l1	0.02489	0.01327	1.875	0.0613 .
y3.l1	0.08822	0.04855	1.817	0.0698 .

 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7393 on 491 degrees of freedom
 Multiple R-squared: 0.01912, Adjusted R-squared: 0.005134
 F-statistic: 1.367 on 7 and 491 DF, p-value: 0.2171

Tramite la funzione `summary` applicato all'oggetto `ecm.stima`, di classe `lm`, si può notare come le relazioni di cointegrazione significative sono solamente quelle presenti nell'equazione della variabile Δy_1 .

4.2 VECM con 2 relazioni di cointegrazione.

Prima di procedere all'analisi della serie dei tassi di interesse, si analizza il caso in cui vi siano due relazioni di cointegrazione. La simulazione di un sistema cointegrato con rango di cointegrazione pari a 2 è mostrato in figura (4.3).

```
> sigma <- diag(0.5, 4)
> e <- mvrnorm(500, rep(0, 4), sigma)
> u.ar <- arima.sim(500, model = list(ar = c(0.2)), innov = e[,
+   1])
> u.ar2 <- arima.sim(500, model = list(ar = c(0.4)), innov = e[,
+   2])
> y2 <- cumsum(e[, 3])
> y3 <- cumsum(e[, 4])
> y1 <- 0.2 * y2 + u.ar
> y4 <- 0.8 * y3 + u.ar2
```

```

> ts.plot(y1, y2, y3, y4, col = c(2, 3, 4, 5), main = "Simulazione di un sistema
+ cointegrato trivariato", sub = "2 equilibri e 2 trend comuni")
> legend(max(y1, y2, y3, y4), max(y1, y2, y3, y4), legend = c("y1- equ. 1",
+ "y2- trend equ.1 ", "y3- trend equ.2", "y4- equ.2"), col = c(2,
+ 3, 4, 5), lty = c(1))
> layout(matrix(1:2, 2, 1))
> plot(u.ar, main = "Residui cointegrati di y1")
> plot(u.ar2, main = "Residui cointegrati di y4")

```

La stima delle relazioni di cointegrazione è leggermente diversa da interpretare da quella vista nel par(4.1), in quanto si devono estrapolare le prime due colonne della matrice di cointegrazione e normalizzarle rispetto alle variabili di interesse.

```

> coint.sim2 <- ca.jo(cbind(y1, y2, y3, y4), type = "eigen", ecdet = "none",
+ K = 2, spec = "transitory")
> summary(coint.sim2)

```

```

#####
# Johansen-Procedure #
#####

```

Test type: maximal eigenvalue statistic (lambda max) , with linear trend

Eigenvalues (lambda):

```
[1] 0.265764149 0.238753586 0.020157056 0.002377317
```

Values of teststatistic and critical values of test:

	test	10pct	5pct	1pct
r <= 3		1.19	6.50	8.18 11.65
r <= 2		10.14	12.91	14.90 19.19
r <= 1		135.85	18.90	21.07 25.75
r = 0		153.84	24.78	27.14 32.14

Eigenvectors, normalised to first column:

(These are the cointegration relations)

	y1.l1	y2.l1	y3.l1	y4.l1
y1.l1	1.0000000	1.0000000	1.0000000	1.0000000
y2.l1	-0.2091475	-0.1935250	4.574329	7.244332
y3.l1	0.6318859	-1.0614094	3.395324	-7.170379
y4.l1	-0.7910182	1.3208501	0.423693	-1.916357

Weights W:

(This is the loading matrix)

	y1.l1	y2.l1	y3.l1	y4.l1
y1.d	-0.42785765	-0.21529126	-0.001331199	-0.0000837573
y2.d	0.06725300	0.03038004	-0.002395068	-0.0003270030
y3.d	0.04794871	0.01613869	-0.003235761	0.0002433711
y4.d	0.34773724	-0.30678135	-0.002258646	0.0001967150

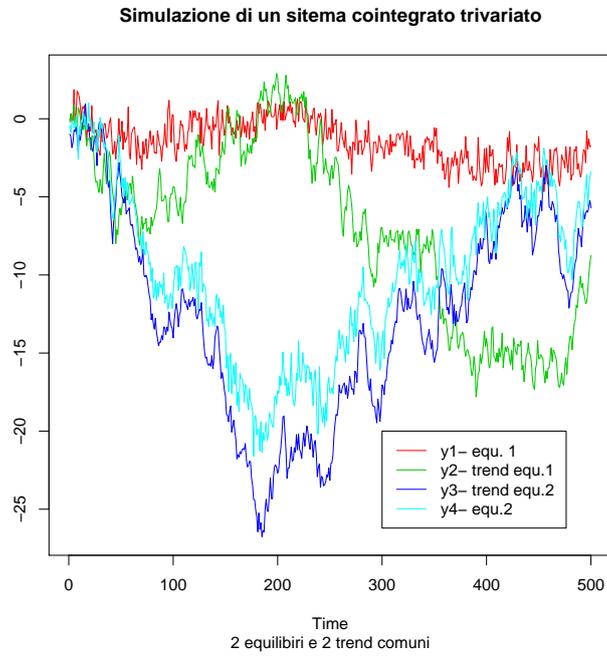


Figura 4.3: Sistema cointegrato con 2 relazioni di cointegrazione

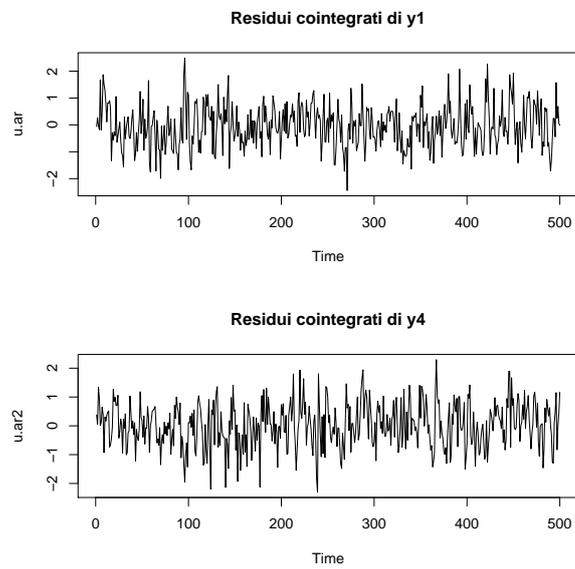


Figura 4.4: Residui del sistema cointegrato

Dalla lettura del `summary` si evince che il rango di cointegrazione è pari a 2, mentre le relazioni di cointegrazione sono da correggere poichè la seconda colonna è normalizzata sempre rispetto alla prima variabile.

Il comando `coint.sim2@Vorg` restituisce i valori non normalizzati delle relazioni di cointegrazione, per cui bisogna manipolare questa matrice in modo da ottenere le relazioni corrette. Un indizio sulle vere relazioni di cointegrazione è ottenuto per mezzo della stima del VECM, che indica quali solo le relazioni significative.

```
> stima.coint2 <- cajools(coint.sim2)
> summary(stima.coint2)
```

Response y1.d :

```
Call:
lm(formula = substitute(y1.d), data = data.mat)
```

```
Residuals:
      Min       1Q   Median       3Q      Max
-2.2443004 -0.5263807  0.0007891  0.4640506  2.1326386
```

```
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
constant -0.09860    0.11320  -0.871    0.384
y1.dl1   -0.03084    0.04616  -0.668    0.504
y2.dl1    0.04551    0.04674   0.974    0.331
y3.dl1    0.04433    0.06107   0.726    0.468
y4.dl1   -0.02489    0.04673  -0.533    0.594
y1.l1   -0.64456    0.05352 -12.044 <2e-16 ***
y2.l1    0.12445    0.01275   9.758 <2e-16 ***
y3.l1   -0.04576    0.04397  -1.041    0.298
y4.l1    0.05367    0.05445   0.986    0.325
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.7572 on 489 degrees of freedom
Multiple R-squared: 0.3236, Adjusted R-squared: 0.3111
F-statistic: 25.99 on 9 and 489 DF, p-value: < 2.2e-16
```

Response y2.d :

```
Call:
lm(formula = substitute(y2.d), data = data.mat)
```

```
Residuals:
      Min       1Q   Median       3Q      Max
-2.22016 -0.48866 -0.02273  0.54870  1.98228
```

```
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
```

```

constant -0.194361  0.110079 -1.766  0.07808 .
y1.dl1  -0.074807  0.044884 -1.667  0.09622 .
y2.dl1   0.018306  0.045452  0.403  0.68731
y3.dl1  -0.102842  0.059389 -1.732  0.08396 .
y4.dl1   0.009751  0.045440  0.215  0.83018
y1.l1    0.094911  0.052042  1.824  0.06880 .
y2.l1   -0.033270  0.012402 -2.683  0.00755 **
y3.l1    0.004463  0.042756  0.104  0.91690
y4.l1   -0.013459  0.052952 -0.254  0.79947

```

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.7363 on 489 degrees of freedom
Multiple R-squared: 0.02521, Adjusted R-squared: 0.007273
F-statistic: 1.405 on 9 and 489 DF, p-value: 0.1827

```

Response y3.d :

```

Call:
lm(formula = substitute(y3.d), data = data.mat)

```

```

Residuals:
    Min       1Q   Median       3Q      Max
-2.25344 -0.48689  0.01134  0.50913  2.32023

```

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
constant -0.2850632  0.1099712  -2.592  0.00982 **
y1.dl1   -0.0622639  0.0448401  -1.389  0.16559
y2.dl1   -0.0135828  0.0454080  -0.299  0.76497
y3.dl1    0.0760748  0.0593310   1.282  0.20038
y4.dl1   -0.0784171  0.0453958  -1.727  0.08473 .
y1.l1     0.0610950  0.0519912   1.175  0.24053
y2.l1    -0.0261900  0.0123896  -2.114  0.03503 *
y3.l1     0.0004368  0.0427146   0.010  0.99184
y4.l1    -0.0184489  0.0529000  -0.349  0.72743

```

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.7356 on 489 degrees of freedom
Multiple R-squared: 0.03076, Adjusted R-squared: 0.01292
F-statistic: 1.725 on 9 and 489 DF, p-value: 0.0808

```

Response y4.d :

```

Call:
lm(formula = substitute(y4.d), data = data.mat)

```

Residuals:

Min	1Q	Median	3Q	Max
-2.95213	-0.63666	0.03125	0.62683	2.73268

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
constant	-0.20380	0.13734	-1.484	0.138
y1.d11	-0.04537	0.05600	-0.810	0.418
y2.d11	-0.04304	0.05671	-0.759	0.448
y3.d11	0.01712	0.07410	0.231	0.817
y4.d11	-0.08159	0.05670	-1.439	0.151
y1.l1	0.03889	0.06493	0.599	0.549
y2.l1	-0.02227	0.01547	-1.439	0.151
y3.l1	0.53627	0.05335	10.053	<2e-16 ***
y4.l1	-0.68161	0.06607	-10.317	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9187 on 489 degrees of freedom

Multiple R-squared: 0.2839, Adjusted R-squared: 0.2707

F-statistic: 21.54 on 9 and 489 DF, p-value: < 2.2e-16

Nella prima equazione i coefficienti significativi sono dati dalla variabili y1.l1 e y2.l1, che sono le vere relazioni di cointegrazione. I parametri stimati sono da dividere per il corrispondente valore del coefficiente di aggiustamento α . L'ultima equazione restituisce come parametri significativi quelli corrispondenti alle variabili y3.l1 e y4.l1.

Dalla matrice di cointegrazione non normalizzata possiamo ottenere le vere relazioni normalizzando la prima colonna rispetto alla prima variabile e la seconda colonna rispetto alla variabile y4.

```
> X <- coint.sim2@Vorg
> beta.y1 <- X[, 1]/X[1, 1]
> beta.y4 <- X[, 2]/X[4, 2]
> beta.y1

      y1.l1      y2.l1      y3.l1      y4.l1
1.0000000 -0.2091475  0.6318859 -0.7910182

> beta.y4

      y1.l1      y2.l1      y3.l1      y4.l1
0.7570882 -0.1465155 -0.8035805  1.0000000
```

A questo punto può essere interessante effettuare delle restrizioni su alcuni valori dei vettori $\hat{\beta}_1$ e $\hat{\beta}_2$ tramite le funzioni `bh5lrtest`. Così facendo è possibile controllare se alcuni valori possono essere considerati nulli.

```
> H1 <- c(1, -0.2, 0, 0)
> summary(bh5lrtest(coint.sim2, H1, r = 2))
```

```
#####
# Johansen-Procedure #
#####
```

Estimation and testing under partly known beta

The VECM has been estimated subject to:
 beta=H*phi and/or alpha=A*psi

```
      [,1]
[1,]  1.0
[2,] -0.2
[3,]  0.0
[4,]  0.0
```

Eigenvalues of restricted VAR (lambda):
 [1] 0.2493 0.0204 0.0024

The value of the likelihood ratio test statistic:
 0.12 distributed as chi square with 2 df.
 The p-value of the test statistic is: 0.94

Eigenvectors, normalised to first column
 of the restricted VAR:

```
      [,1]      [,2]
[1,]  1.0      1.0000
[2,] -0.2      5.0000
[3,]  0.0 -541.6381
[4,]  0.0  675.7080
```

Weights W of the restricted VAR:

```
      [,1]      [,2]
y1.d -0.6447  1e-04
y2.d  0.0949  0e+00
y3.d  0.0606  0e+00
y4.d  0.0395 -1e-03
```

Con il vettore H1 si è impostata l'ipotesi da verificare, sulle quale è eseguito un test del rapporto di verosimiglianza che è distribuito secondo una variabile χ^2 con $(p-r) \cdot r_1$ gradi di libertà, dove $r_1 < r$ è il numero di relazioni di cointegrazione da verificare. Dalla lettura dei risultati si può constatare che l'ipotesi nulla specificata con il vettore H1 è accettata con un p-value pari a 0.94.

```
> H2 <- c(0, 0, -0.8, 1)
> summary(bh5lrtest(coint.sim2, H2, r = 2))
```

```
#####
# Johansen-Procedure #
#####
```

Estimation and testing under partly known beta

The VECM has been estimated subject to:
beta=H*phi and/or alpha=A*psi

```
      [,1]
[1,]  0.0
[2,]  0.0
[3,] -0.8
[4,]  1.0
```

Eigenvalues of restricted VAR (lambda):
[1] 0.2561 0.0205 0.0040

The value of the likelihood ratio test statistic:
0.98 distributed as chi square with 2 df.
The p-value of the test statistic is: 0.61

Eigenvectors, normalised to first column
of the restricted VAR:

```
      [,1]  [,2]
[1,]  0.0  1.0000
[2,]  0.0 -0.2040
[3,] -0.8 -0.0013
[4,]  1.0 -0.0011
```

Weights W of the restricted VAR:

```
      [,1]  [,2]
y1.d  0.0559 -0.6430
y2.d -0.0067  0.0981
y3.d -0.0148  0.0642
y4.d -0.6757  0.0422
```

Anche in questo caso l'ipotesi specificata con il vettore H2 è accettata con un p-value pari a 0.61.

Infine si controlla se le due restrizioni, congiuntamente, possono essere considerate significative. A tal proposito, si utilizza la funzione `blrtest` nel seguente modo.

```
> H <- cbind(H1, H2)
> summary(blrttest(coint.sim2, H, 2))
```

```
#####
# Johansen-Procedure #
#####
```

Estimation and testing under linear restrictions on beta

The VECM has been estimated subject to:

beta=H*phi and/or alpha=A*psi

	H1	H2
[1,]	1.0	0.0
[2,]	-0.2	0.0
[3,]	0.0	-0.8
[4,]	0.0	1.0

Eigenvalues of restricted VAR (lambda):

[1] 0.2648 0.2380

The value of the likelihood ratio test statistic:

1.14 distributed as chi square with 4 df.

The p-value of the test statistic is: 0.89

Eigenvectors, normalised to first column
of the restricted VAR:

	[,1]	[,2]
[1,]	1.0000	1.0000
[2,]	-0.2000	-0.2000
[3,]	0.5969	-1.1164
[4,]	-0.7461	1.3955

Weights W of the restricted VAR:

	[,1]	[,2]
y1.d	-0.4460	-0.1987
y2.d	0.0648	0.0300
y3.d	0.0463	0.0142
y4.d	0.3406	-0.3020

Le restrizione imposte nella matrice H risultano significative entrambe ad un livello di significatività dell'89%.

Capitolo 5

Analisi della cointegrazione della struttura dei tassi di interesse.

Riguardo la struttura dei tassi d'interesse a termine, sono state condotte diverse analisi e studi con lo scopo di stimare una relazione di equilibrio di lungo periodo tra i tassi a breve periodo con quelli a lungo periodo. [Hall et al. \(1992\)](#) condussero uno studio su dodici tassi a termine: una serie per il tasso ad un mese, un'altra serie per il tasso a 2 mesi fino al tasso a 12 mesi. Nel loro studio gli autori hanno stimato una relazione di equilibrio tra bonds che avevano durata simile. Questo risultato fu anche confermato empiricamente, su un sottoinsieme del data set utilizzato nel loro articolo. Inoltre, gli autori stimarono anche il modello a correzione d'errore corrispondente al sistema cointegrato. Il modello ECM da loro trovato, ha dato evidenza al fatto che i rendimenti a lungo periodo "guidano" la struttura dei tassi, mentre le variazioni che si verificano sui tassi a breve termine sono degli aggiustamenti per i rendimenti a lungo periodo.

Prima di procedere con l'analisi econometrica della serie storica è opportuno verificare se sono soddisfatte certe proprietà, come ad esempio la normalità di ogni singola serie storica. Per condurre questo tipo di analisi applichiamo il test di Jarque-Bera presente nel pacchetto `tseries`.

```
> apply(tassi.ts, 2, jarque.bera.test)
```

```
$mesi3
```

```
Jarque Bera Test
```

```
data: newX[, i]
```

```
X-squared = 12.5516, df = 2, p-value = 0.001881
```

```
$anno1
```

```
Jarque Bera Test
```

```
data: newX[, i]
X-squared = 18.6404, df = 2, p-value = 8.96e-05
```

```
$anni7
```

```
Jarque Bera Test
```

```
data: newX[, i]
X-squared = 35.5266, df = 2, p-value = 1.930e-08
```

```
$anni20
```

```
Jarque Bera Test
```

```
data: newX[, i]
X-squared = 50.1132, df = 2, p-value = 1.312e-11
```

I p-value indicano che l'ipotesi nulla di normalità è ampiamente rigettata, per cui una delle ipotesi sottostanti alla metodologia dei VAR e del VECM non risulta essere soddisfatta. Prima di procedere alla stima del rango di cointegrazione è necessario determinare il numero di variabili *lagged* da inserire nel VAR e nel VECM. Come si è visto nel paragrafo (1.2) il numero di variabili ritardate stimate per mezzo della funzione `VARselect`, che si basa sui criteri di informazione, è compreso tra 2 e 4. Tramite la procedura di Johansen è stimato il rango di cointegrazione, utilizzando sia il test traccia che il test del massimo autovalore. In base alla letteratura esistente (Hall *et al.*, 1992), la relazione di cointegrazione dovrebbe interessare due tassi di interesse. Sul data set attuale l'ipotesi di cointegrazione tra i due tassi di interesse è rifiutata, tranne per il tasso a 3 mesi con quello ad 1 anno. Dai risultati riportati in tabella (5.1), sembra che l'unico equilibrio esistente sia tra i due tassi a breve periodo, mentre i tassi a 7 anni e quello a 20 anni sembrano non condividere nessun equilibrio comune tra di loro e tra gli altri tassi di interesse.

5.1 Analisi della cointegrazione tra 3 tassi d'interesse.

Effettuando l'analisi della cointegrazione fra tre tassi di interesse risultano significative 4 relazioni di cointegrazione differenti, i cui risultati sono riportati nella tabella (5.2).

```
> tassi2 <- cbind(mesi3, anni7, anni20)
> tassi3 <- cbind(mesi3, anno1, anni7)
> tassi4 <- cbind(mesi3, anno1, anni20)
> tassi5 <- cbind(anno1, anni7, anni20)
> tassi2.ecm <- ca.jo(tassi2, type = "eigen", ecdet = "const",
+ K = 3, spec = "transitory")
> tassi3.ecm <- ca.jo(tassi3, type = "eigen", ecdet = "const",
+ K = 4, spec = "transitory")
```

stat-test 3m e 1a		10pct	5pct	1pct
r <= 1	8.94	7.52	9.24	12.97
r = 0	26.07	13.75	15.67	20.20
stat-test 3m e 7a		10pct	5pct	1pct
r <= 1	6.28	7.52	9.24	12.97
r = 0	10.23	13.75	15.67	20.20
stat-test 3m e 20a		10pct	5pct	1pct
r <= 1	4.77	7.52	9.24	12.97
r = 0	10.29	13.75	15.67	20.20
stat-test 1a e 7a		10pct	5pct	1pct
r <= 1	4.87	7.52	9.24	12.97
r = 0	9.86	13.75	15.67	20.20
stat-test 1a e 20a		10pct	5pct	1pct
r <= 1	4.25	7.52	9.24	12.97
r = 0	9.67	13.75	15.67	20.20
stat-test 7a e 20a		10pct	5pct	1pct
r <= 1	5.83	7.52	9.24	12.97
r = 0	8.25	13.75	15.67	20.20

Tabella 5.1: Statistiche test e valori critici per l'analisi della cointegrazione tra due tassi.

```
> tassi4.ecm <- ca.jo(tassi4, type = "eigen", ecdet = "const",
+ K = 4, spec = "transitory")
> tassi5.ecm <- ca.jo(tassi5, type = "eigen", ecdet = "const",
+ K = 4, spec = "transitory")
```

Il test di Dickey-Fuller eseguito sulle relazioni di disequilibrio dei 4 sistemi cointegrati, rigetta l'ipotesi nulla di presenza di radici unitarie.

Dalla stima dei VECM, corrispondenti a ciascuno dei 4 sistemi cointegrati, risulta significativa ad un livello del 5%, solamente la relazione di cointegrazione tra i tassi a 3 mesi, 7 anni e 20 anni. Di seguito si riporta la stima del VECM solamente per questa relazione.

```
> summary(cajools(tassi2.ecm))
```

Response mesi3.d :

Call:

```
lm(formula = substitute(mesi3.d), data = data.mat)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.092271	-0.096582	0.008718	0.133639	0.781821

Coefficients:

stat-test 3m, 1a e 20a		10pct	5pct	1pct	
r <= 2		7.05	7.52	9.24	12.97
r <= 1		9.58	13.75	15.67	20.20
r = 0		33.70	19.77	22.00	26.81
stat-test 3m, 1a e 7a		10pct	5pct	1pct	
r <= 2		8.97	7.52	9.24	12.97
r <= 1		10.16	13.75	15.67	20.20
r = 0		30.44	19.77	22.00	26.81
stat-test 3m, 7a e 20a		10pct	5pct	1pct	
r <= 2		4.20	7.52	9.24	12.97
r <= 1		8.79	13.75	15.67	20.20
r = 0		29.99	19.77	22.00	26.81
stat-test 1a, 7a e 20a		10pct	5pct	1pct	
r <= 2		5.66	7.52	9.24	12.97
r <= 1		11.14	13.75	15.67	20.20
r = 0		25.02	19.77	22.00	26.81

Tabella 5.2: Statistiche test e valori critici per l'analisi della cointegrazione tra tre tassi.

```

                Estimate Std. Error t value Pr(>|t|)
mesi3.dl1      0.476438   0.066026   7.216  4.8e-12 ***
anni7.dl1     -0.263693   0.195231  -1.351  0.1779
anni20.dl1    0.290456   0.206819   1.404  0.1613
mesi3.dl2     0.007699   0.064435   0.119  0.9050
anni7.dl2     0.097834   0.195714   0.500  0.6175
anni20.dl2   -0.188276   0.208900  -0.901  0.3682
mesi3.l1     -0.055427   0.024985  -2.218  0.0273 *
anni7.l1      0.131586   0.095349   1.380  0.1686
anni20.l1    -0.105510   0.085230  -1.238  0.2167
constant      0.149097   0.095499   1.561  0.1196
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.2656 on 287 degrees of freedom
Multiple R-squared: 0.2592, Adjusted R-squared: 0.2334
F-statistic: 10.04 on 10 and 287 DF, p-value: 1.867e-14

```

Response anni7.d :

```

Call:
lm(formula = substitute(anni7.d), data = data.mat)

```

Residuals:

Min	1Q	Median	3Q	Max
-0.72164	-0.16777	-0.01917	0.15051	0.80344

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
mesi3.dl1	-0.00521	0.06550	-0.080	0.93666
anni7.dl1	0.03479	0.19369	0.180	0.85757
anni20.dl1	0.46185	0.20518	2.251	0.02515 *
mesi3.dl2	0.16705	0.06393	2.613	0.00944 **
anni7.dl2	0.09548	0.19417	0.492	0.62328
anni20.dl2	-0.34305	0.20725	-1.655	0.09896 .
mesi3.l1	0.04976	0.02479	2.008	0.04562 *
anni7.l1	-0.21902	0.09459	-2.315	0.02130 *
anni20.l1	0.17878	0.08456	2.114	0.03535 *
constant	-0.08632	0.09474	-0.911	0.36301

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2635 on 287 degrees of freedom
 Multiple R-squared: 0.2487, Adjusted R-squared: 0.2226
 F-statistic: 9.502 on 10 and 287 DF, p-value: 1.183e-13

Response anni20.d :

Call:

lm(formula = substitute(anni20.d), data = data.mat)

Residuals:

Min	1Q	Median	3Q	Max
-0.792227	-0.159172	-0.008322	0.129788	0.826598

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
mesi3.dl1	-0.009523	0.059058	-0.161	0.872015
anni7.dl1	-0.182984	0.174628	-1.048	0.295589
anni20.dl1	0.619105	0.184993	3.347	0.000927 ***
mesi3.dl2	0.146278	0.057636	2.538	0.011678 *
anni7.dl2	0.119912	0.175060	0.685	0.493913
anni20.dl2	-0.346396	0.186854	-1.854	0.064790 .
mesi3.l1	0.030445	0.022348	1.362	0.174168
anni7.l1	-0.116831	0.085287	-1.370	0.171801
anni20.l1	0.086079	0.076236	1.129	0.259794
constant	-0.007042	0.085421	-0.082	0.934359

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2376 on 287 degrees of freedom
 Multiple R-squared: 0.2266, Adjusted R-squared: 0.1997

F-statistic: 8.41 on 10 and 287 DF, p-value: 5.308e-12

La relazione di cointegrazione tra i 3 tassi di interesse è la seguente:

$$y_{7(t-1)} = -0.78 + 0.26y_{3(t-1)} + 0.86y_{20(t-1)}.$$

La funzione `plotres` applicata all'oggetto di classe `ca.jo` restituisce il grafico che consente di diagnosticare i residui graficamente, come mostrano le figure (5.1), (5.2) e (5.3).

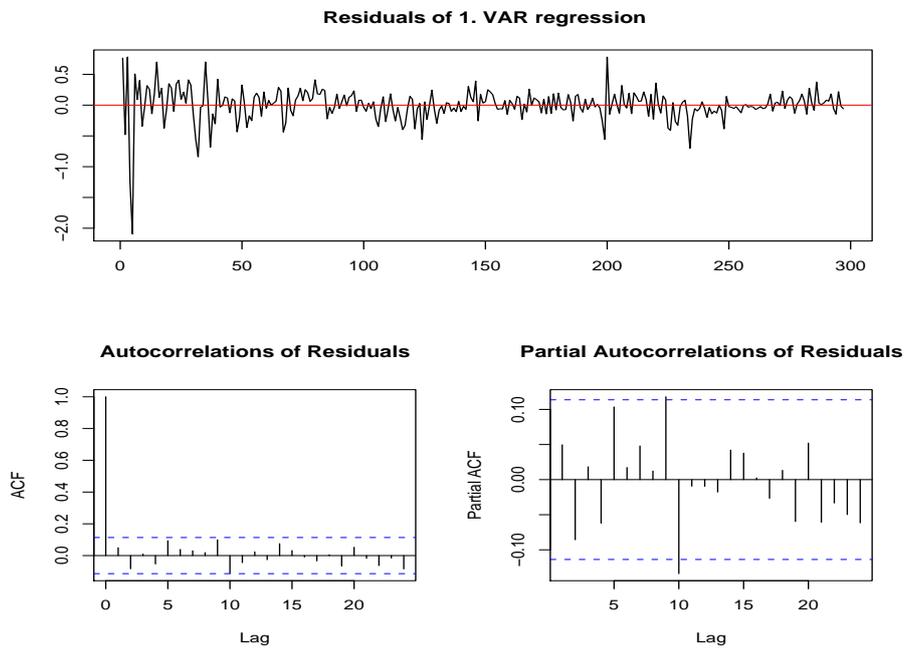


Figura 5.1: Analisi dei residui del tasso a 3 mesi.

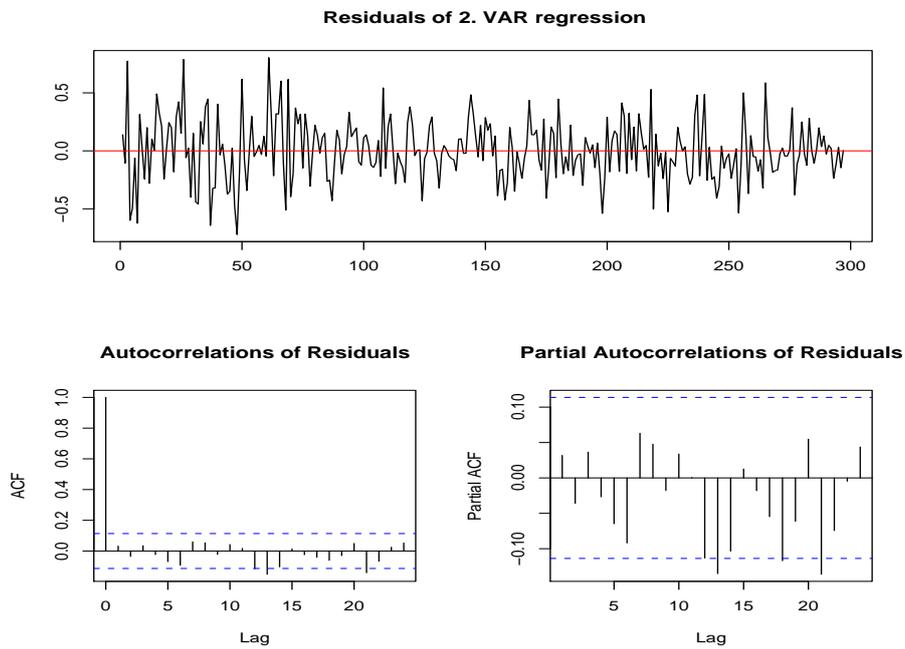


Figura 5.2: Analisi dei residui del tasso a 7 anni.

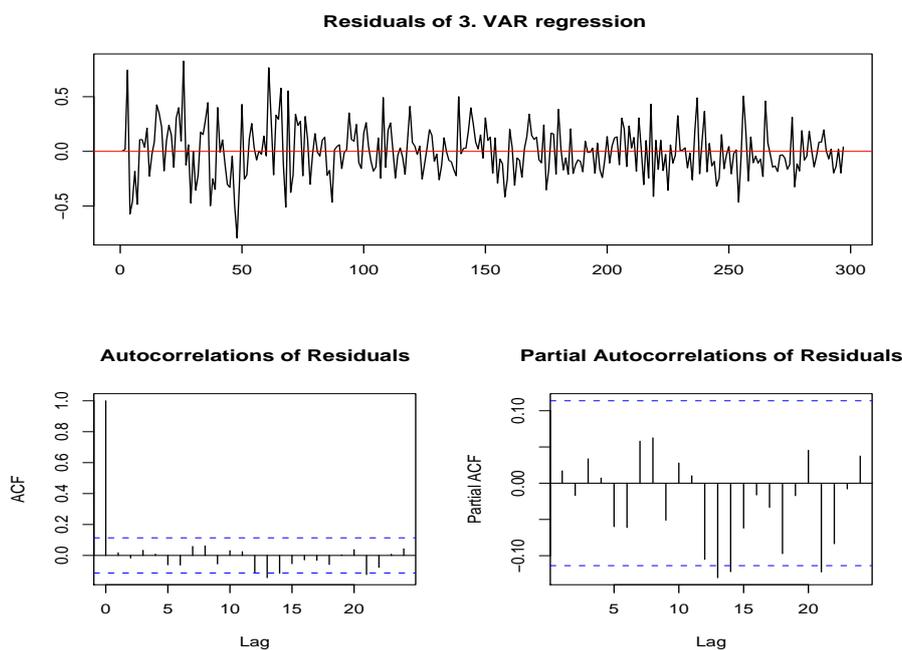


Figura 5.3: Analisi dei residui del tasso a 20 anni.

5.2 Analisi della cointegrazione tra 4 tassi d'interesse.

Volendo considerare l'intera struttura dei tassi di interesse, quindi implementare l'analisi della cointegrazione tra i 4 tassi d'interesse a termine, il risultato che si ottiene è che i 4 tassi, congiuntamente, condividono un solo trend comune. Infatti, eseguendo i seguenti codici si ottengono i risultati riportati nella tabella (5.3) (5.4).

```
> tassi.vecm.eigen <- ca.jo(tassi.ts, type = "eigen", ecdet = "none",
+   K = 4, spec = "transitory")
> tassi.vecm.trace <- ca.jo(tassi.ts, type = "trace", ecdet = "none",
+   K = 4, spec = "transitory")
```

	eigen test	10pct	5pct	1pct
r ≤ 3	4.90	6.50	8.18	11.65
r ≤ 2	10.60	12.91	14.90	19.19
r ≤ 1	13.87	18.90	21.07	25.75
r = 0	43.30	24.78	27.14	32.14

Tabella 5.3: Test del massimo autovalore per la stima del rango di cointegrazione.

	trace test	10pct	5pct	1pct
r <= 3	4.90	6.50	8.18	11.65
r <= 2	15.51	15.66	17.95	23.52
r <= 1	29.38	28.71	31.52	37.22
r = 0	72.68	45.23	48.28	55.43

Tabella 5.4: Test traccia per la stima del rango di cointegrazione.

Sia il test del massimo autovalore che il test traccia accettano l'ipotesi di una relazione di cointegrazione. Il rango di cointegrazione stimato è pari ad 1 (sussiste una sola relazione di equilibrio di lungo periodo), per cui il vettore di cointegrazione è riportato nella prima colonna della matrice di cointegrazione, sulla quale vengono eseguiti i seguenti controlli.

```
> beta.tassi <- tassi.vecm.eigen@V
> beta.tassi[, 1]

    mesi3.l1  anno1.l1  anni7.l1  anni20.l1
1.0000000 -0.7344852 -1.2477279  1.2024913

> Y.tassi.vecm <- tassi.vecm.eigen@ZK ### variabili ritardate
> coint.tassi <- t(beta.tassi[, 1] %*% t(Y.tassi.vecm)) ### relazione di
### cointegrazione come combinazione lineare tra
### il vettore stimato e le variabili ritardate

> mean(coint.tassi) ### media dei residui della relazione di cointegrazione

[1] 1.345142

> summary(ur.df(coint.tassi, lags = 2)) ### verifica della presenza
### di radici unitarie nella relazione di disequilibrio

#####
# Augmented Dickey-Fuller Test Unit Root Test #
#####

Test regression none

Call:
lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)

Residuals:
    Min       1Q   Median       3Q      Max
-1.20170 -0.08925  0.02043  0.11620  1.09108

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
z.lag.1         -0.011386   0.008084  -1.408   0.16
```

```

z.diff.lag1 0.371645 0.056718 6.553 2.59e-10 ***
z.diff.lag2 -0.242732 0.056940 -4.263 2.73e-05 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1913 on 289 degrees of freedom
Multiple R-squared: 0.1503, Adjusted R-squared: 0.1415
F-statistic: 17.04 on 3 and 289 DF, p-value: 3.22e-10

```

Value of test-statistic is: -1.4084

```

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62

```

Il test di Dickey-Fuller rigetta l'ipotesi di stazionarietà ad un livello del 10%, inoltre la media calcolata sulla relazione di disequilibrio suggerisce la presenza di una costante deterministica. Per cui è necessario introdurre la presenza di una costante nella relazione di cointegrazione da stimare.

```

> tassi.vecm.const <- ca.jo(tassi.ts, type = "eigen", ecdet = "const",
+   K = 4, spec = "transitory")
> beta.tassi.const <- tassi.vecm.const@V
> Y.tassi.vecm.const <- tassi.vecm.const@ZK
> coint.tassi.const <- t(beta.tassi.const[, 1] %*% t(Y.tassi.vecm.const))
> mean(coint.tassi.const)

```

```
[1] 0.01889576
```

```
> summary(ur.df(coint.tassi.const, lags = 3))
```

```

#####
# Augmented Dickey-Fuller Test Unit Root Test #
#####

```

Test regression none

```

Call:
lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)

```

```

Residuals:
      Min       1Q   Median       3Q      Max
-1.182707 -0.090301  0.002013  0.109859  0.942069

```

```

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
z.lag.1      -0.19490    0.03247  -6.003 5.85e-09 ***
z.diff.lag1  0.44641    0.05662   7.884 6.67e-14 ***
z.diff.lag2 -0.15978    0.05901  -2.708 0.00718 **

```

```

z.diff.lag3 0.12431 0.05777 2.152 0.03225 *
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1787 on 287 degrees of freedom
Multiple R-squared: 0.2383, Adjusted R-squared: 0.2277
F-statistic: 22.44 on 4 and 287 DF, p-value: 3.844e-16

Value of test-statistic is: -6.0025

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62

> summary(ur.kpss(coint.tassi.const))

#####
# KPSS Unit Root Test #
#####

Test is of type: mu with 5 lags.

Value of test-statistic is: 0.1023

Critical value for a significance level of:
      10pct  5pct 2.5pct  1pct
critical values 0.347 0.463 0.574 0.739

> summary(ur.pp(coint.tassi.const))

#####
# Phillips-Perron Unit Root Test #
#####

Test regression with intercept

Call:
lm(formula = y ~ y.l1)

Residuals:
      Min       1Q   Median       3Q      Max
-1.421050 -0.095703  0.007127  0.100720  1.032422

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.0009257  0.0115801   0.08  0.936
y.l1         0.8501104  0.0302734  28.08 <2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Residual standard error: 0.1983 on 292 degrees of freedom
Multiple R-squared: 0.7298, Adjusted R-squared: 0.7288
F-statistic: 788.6 on 1 and 292 DF, p-value: < 2.2e-16

Value of test-statistic, type: Z-alpha is: -55.9546

aux. Z statistics	
Z-tau-mu	0.1214

Con l'introduzione della costante deterministica si ottiene la relazione di cointegrazione desiderata, che è confermata dal test di Dickey–Fuller (ed anche dagli altri due tests) il quale rigetta l'ipotesi nulla che il processo sia $I(1)$ ad un livello di significatività dell' 1%.

Dalla stima del VECM, corrispondente al sistema cointegrato, non si evince nessuna relazione di cointegrazione significativa, per cui l'equilibrio di lungo periodo più plausibile, per la struttura dei tassi d'interesse considerata, è quella stimata nel paragrafo (5.1).

Bibliografia

- Box, G. E. P. and Jenkins, G. M. (1970). *Time Series Analysis : Forecasting and Control*. Holden Day: San Francisco.
- Breusch, T. S. (1978). Testing for autocorrelation in dynamic linear models. *Australian Economic Papers*, **17**, 334–335.
- Engle, R. F. and Granger, C. W. J. (1987). Co-integration and error correction: Representation, estimation and testing. *Econometrica*, **55**, 251–276.
- Fuller, W. A. (1976). *Introduction to statistical time series*. New York: Wiley.
- Godfrey, L. G. (1978). Testing for higher order serial correlation in regression equations when the regressors include lagged dependent variables. *Econometrica*, **46**, 1303–1313.
- Granger, C. W. J. (1981). Some properties of time series data and their use in econometric model specification. *Journal of Econometrics*, **16**, 121–130.
- Hall, A. D., Anderson, H. M., and Granger, C. W. J. (1992). A cointegration analysis of treasury bill yields. *The Review of Economics and Statistics*, **74**(1), 116–126.
- Hamilton, J. D. (1994). *Time Series Analysis*. Princeton University Press.
- Harvey, A. C. (1993). *Time series models*. 2 edition. Pearson Education Limited.
- Jarque, C. M. and Bera, A. K. (1987). A test for normality of observations and regression residuals. *International Statistical Review*, **55**, 163–172.
- Johansen, S. (1988). Statistical analysis of cointegration vectors. *Journal of Economic Dynamics and Control*, **12**, 231–254.
- Johansen, S. (1995). *Likelihood-based inference in cointegrated vector autoregressive models*. Oxford University Press, Oxford.
- Kwiatkowski, D., Phillips, P. C. B., Schmidt, P., and Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*, **54**, 159–178.
- Lutkepohl, H. (2005). *New introduction to multiple time series analysis*. Springer-Verlag.

- Pfaff, B. (2006). *Analysis of Integrated and Cointegrated Time Series with R*. Springer: New York, first edition. ISBN 0-387-27960-1.
- Pfaff, B. (2008). *vars: VAR Modelling*. R package version 1.3-7.
- Phillips, P. and Ouliaris, S. (1990). Asymptotic properties of residual based tests for cointegration. *Econometrica*, **58**(1), 165–193.
- Phillips, P. C. B. (1987). Time series regression with a unit root. *Econometrica*, **55**, 277–301.
- Phillips, P. C. B. and Perron, P. (1988). Testing for a unit root in time series regression. *Biometrika*, **75**, 335–346.
- R Development Core Team (2008). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.
- Santamaria, L. (2000). *Analisi delle serie storiche economiche*. Vita e Pensiero.
- Schwert, W. (1989). Test for unit roots: A monte carlo investigation. *Journal of Business and Economic Statistics*, **7**, 147–159.
- Sims, C. A. (1980). Macroeconomics and reality. *Econometrica*, **48**(1), 1–48.
- Zivot, E. and Wang, J. (2006). *Modeling financial time series with S-PLUS*. Springer Science+Business Media, Inc., second edition.