Exercises for Chapter 5 of Vinod’s
“HANDS-ON INTERMEDIATE ECONOMETRICS USING R”

H. D. Vinod Professor of Economics,
Fordham University, Bronx, New York 10458

Abstract
These are exercises to accompany the above-mentioned book with the URL: http://www.worldscibooks.com/economics/6895.html. At this time all of the following exercises are suggested by H. D. Vinod (HDV) himself. Vinod invites readers to suggest new and challenging exercises (along with full and detailed answers) dealing with the discussion in Chapter 5 and related discussion from econometric literature by sending e-mail to vinod@fordham.edu. If the answers involve R programs, they must work. Readers may also suggest improvements to the answers and/or hints to existing exercises. If we include exercises and improvements suggested by readers, we promise to give credit to such readers by name. Furthermore, we will attach the initials of readers to individual exercises to identify the reader. Some R outputs are suppressed for brevity.

5 Exercises Mostly Based on Chapter 5 (Multivariate Models) of the text

5.1 Exercise (VAR Models)
1) Show that Vector autoregressive (VAR) models are a special case of VecARIMA models.

2) Write the Irving Fisher model for moneylender variable after including a marginal tax rate variable.
3) What regressor variable appears in that notation of Eq. (5.1.2) of the textbook when the coefficient of a VAR model is $A_{3,23}$?

4) What are H-Q and FPE criteria? [Hint see page 237]

HINTS: See Section 5.1 of the text

5.2 Exercise (VAR Estimation: Sims’ Model)

Use ‘www.forhdem.edu/economics/vinod/errataCD.txt’ to update the snippets of this chapter necessitated by changes at ‘economagic’ website. Update the results and try alternative specifications of the functions ‘VARselect’ and ‘VAR’ and discuss the results.

5.3 Exercise (VAR Estimation: Artificial data)

Recall Section 5.1.2 of the text, eq.(5.1.1) with $K=3$ variables. Using seed 12 and uniform random numbers between 5 and 30 create time series data for 3 endogenous variables $y_1$, $y_2$, and $y_3$ with $T=27$ points for each variable. (1) Estimate VAR(2) model for these data by ordinary least squares.

(2) Write the VAR estimates in the error correction model (ECM) form by using eq.s (5.3.1) and (5.3.2). Explicitly indicate the $\Pi, \Gamma$ matrices.


Answer: Using R our first task is to define the data and get VAR estimates by OLS.

```r
set.seed(12); y1=runif(27, 5,30)
y2=runif(27, 5,30);y3=runif(27, 5,30); #cbind(y1,y2,y3)
```

It is useful to recall some theory from your text. A VAR model consists of a vector of $K = 3$ endogenous variables, $y_t = (y_{1t}, \ldots, y_{kt}, \ldots, y_{Kt})$ for $k = 1, \ldots, K = 3$. One defines VAR($p$)-process for $p = 2$ here as

$$y_t = A_1y_{t-1} + \cdots + A_py_{t-p} + u_t$$

(1)

with $p = 2$ vectors of lagged values and $p$ matrices $A_i$ of dimensions $K \times K$ for $i = 1, \ldots, p$, and where $u_t \sim \text{WN}(0, \Sigma)$, a $K$-dimensional white noise process with $\Sigma$ denoting a time invariant positive definite covariance matrix.
If $K = 2$ we have the bivariate case of Chap. 3. The VAR(2) model in the trivariate $K = 3$ case is an extension of that set up. Where $A_1$, and $A_2$ are $3 \times 3$ matrices with the elements along the first row with additional subscripts $(11, 12, 13)$. The additional subscripts along the second row are $(21, 22, 23)$. The $3 \times 1$ vector $y_t$ is trivariate with elements $y_{1t}$, $y_{2t}$, and $y_{3t}$, and similarly for the $u_t$ vector. The lagged values can be replaced by the $L$ operator, and it is possible to write a matrix polynomial in the lag operator $(I - A_1L - A_2L^2)$ where $I$ is the identity matrix of order 3.

Our numerical task now is to compute the matrices $A_1$ and $A_2$. Let us do this by brute force application following the definitions and compare the results with those given by Pfaff’s package ‘vars.’

```r
y1t=y1[3:27]
y2t=y2[3:27]
y3t=y3[3:27]
y1Lag1=y1[2:26]
y2Lag1=y2[2:26]
y3Lag1=y3[2:26]
y1Lag2=y1[1:25]
y2Lag2=y2[1:25]
y3Lag2=y3[1:25]
reg1=lm(y1t~y1Lag1+y2Lag1+y3Lag1+y1Lag2+y2Lag2+y3Lag2)
reg2=lm(y2t~y1Lag1+y2Lag1+y3Lag1+y1Lag2+y2Lag2+y3Lag2)
reg3=lm(y3t~y1Lag1+y2Lag1+y3Lag1+y1Lag2+y2Lag2+y3Lag2)
library(vars)
V1=VAR(cbind(y1,y2,y3), p=2, type="const")
V2=coef(V1)
myvec=V2$y1[,1]
n=length(myvec)#need to make intercept first coeff
Pfaf=c(myvec[n],myvec[1: n-1])
cb1=cbind(coef(reg1),Pfaf)
colnames(cb1)=c("brute force OLS", "vars package")
```

The following output from R for the first regression of $y_1$ shows that the R package is doing what we intend for it.

```
<table>
<thead>
<tr>
<th>bruteforce OLS</th>
<th>vars package</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>28.816529541</td>
</tr>
<tr>
<td></td>
<td>28.816529541</td>
</tr>
</tbody>
</table>
```

3
The regression coefficients are exactly the same.

The VAR\((p)\) model of (5.1.1) in your text can be written in error correction model (ECM) form of (5.3.1) as:

\[
\Delta X_t = \sum_{i=1}^{p-1} \Gamma_i \Delta X_{t-i} + \Pi X_{t-p} + \Phi D_t + \mu + \varepsilon_t, \quad (t = 1, \ldots, T),
\]

where \(D_t\) are seasonal dummies orthogonal to the constant term. The ECM writes \(\Delta X_t\) as a function of lagged levels \(X_{t-1}\) and lagged \(\Delta X_t\). The name error correction arises because lagged \(\Delta X_t\) can be viewed as past equilibrium errors. In the current case of our numerical example, we have no seasonal dummies so the above equation (5.3.2) is simpler.

It is interesting that under certain conditions one need not actually estimate (5.3.1) but simply re-use the lag coefficient matrices of \(A_1\) to \(A_p\) in (5.1.1) and recover the \(\Pi\) and \(\Gamma\) matrices as in eq. (5.3.2) of the text:

\[
\Pi = -I_K + A_1 + A_2 + \cdots + A_p, \\
\Gamma_i = -(A_{i+1} + \cdots + A_p).
\]

These are true by definition of the matrices and definitional relations between differenced and lagged variables. It is interesting to compute them for our numerical example for a hands-on experience. The cointegration test for our example is implemented as follows.

\[
Ac1=Acceof(V1) \\
library(urca) \\
ca1=ca.jo(cbind(y1,y2,y3),ecdet="const") \\
\text{summary(ca1)}
\]

Here is the output where \(r\) denotes the cointegrating rank. Since no eigenvalue is close to zero, we do not have cointegration in this artificial example. We shall consider a real world example in Section 5.10 in the sequel where the cointegrating rank is not full as it is here.
> summary(ca1)

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

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

Eigenvalues (lambda):
[1] 0.5738645 0.5054685 0.2172522

Values of test statistic and critical values of test:

<table>
<thead>
<tr>
<th></th>
<th>test</th>
<th>10pct</th>
<th>5pct</th>
<th>1pct</th>
</tr>
</thead>
<tbody>
<tr>
<td>r &lt;= 2</td>
<td>6.12</td>
<td>6.50</td>
<td>8.18</td>
<td>11.65</td>
</tr>
<tr>
<td>r &lt;= 1</td>
<td>17.60</td>
<td>12.91</td>
<td>14.90</td>
<td>19.19</td>
</tr>
<tr>
<td>r = 0</td>
<td>21.32</td>
<td>18.90</td>
<td>21.07</td>
<td>25.75</td>
</tr>
</tbody>
</table>

Eigenvectors, normalised to first column:
(These are the cointegration relations)

\[
\begin{align*}
y_{1,t} & = 1.00000000 \quad y_{2,t} = 0.39760396 \quad y_{3,t} = 0.07751068 \\
y_{1,t} & = 1.00000000 \quad y_{2,t} = -0.01950001 \quad y_{3,t} = -84.243092 \\
y_{1,t} & = 1.00000000 \quad y_{2,t} = -0.01950001 \quad y_{3,t} = -84.243092 \\
\end{align*}
\]

Weights W:
(This is the loading matrix)

\[
\begin{align*}
y_{1,t} & = -1.4026339 \quad y_{2,t} = -0.2018451 \quad y_{3,t} = -0.2086819 \\
y_{1,t} & = -0.002501758 \quad y_{2,t} = -0.005791933 \quad y_{3,t} = 0.051153253 \\
y_{1,t} & = -0.003335412 \quad y_{2,t} = 0.008058862 \quad y_{3,t} = 0.004380114 \\
\end{align*}
\]

The Π, Γ matrices of equation \( [3] \) for our example are given from the R object of the class ca.jo accessed by the @ symbol. This is somewhat different from the usual $ symbol method.
The weight matrix $W$ from the above output is actually the $\Pi$ matrix of equation (3). We also illustrate the $\Gamma$ matrix below.

\begin{verbatim}
> ca1@PI

y1.l2 y2.l2 y3.l2
y1.d -1.4084711 -0.2766586 -0.1656420
y2.d -0.1995782 -0.7590449 0.3571507
y3.d -0.1531485 -0.4529646 -0.8928381

> ca1@GAMMA

constant y1.dl1 y2.dl1 y3.dl1
y1.d 28.81653 -0.9962188 0.04585655 -0.2601613
y2.d 10.09901 -0.1794351 -1.12091553 0.2379185
y3.d 27.66544 -0.2024416 -0.18659968 -1.0289003
\end{verbatim}

The cointegration hypothesis is a restriction on the $\Pi$ matrix. A decomposition of the $\Pi$ matrix can be a hypothesis of the cointegration:

$$\Pi = \alpha \beta',$$  

(4)

where $\beta$ contains the cointegrating vectors and $\alpha$ contains the adjustment coefficients, both of which are $p \times r$ matrices ($p > r$). Johansen explains that the matrices of $\alpha$ and $\beta$ cannot be uniquely known, that is, they are not identified. In the above output, we report the $\beta$ as ‘Eigenvectors, normalised to first column’.

The space spanned by $\beta$ is the row space and the space spanned by $\alpha$ is the column space of $\Pi$.

The hypothesis of cointegration is the hypothesis of reduced rank of $\Pi$ and a direct way of checking it is to see if any of its singular values are close to zero. In the above output ‘Eigenvalues’ are squares of singular values and no eigenvalue is close to zero since the last one is 0.2172522.

The reader is encouraged to read Pfaff’s vignette accompanying the R package ‘vars’ available at the URL: [http://cran.case.edu/web/packages/vars/vignettes/vars.pdf](http://cran.case.edu/web/packages/vars/vignettes/vars.pdf), which contains useful theoretical discussion of structural VARs, (SVARs). For example, Blanchard and Quah’s 1989 American Economic Review (p. 655) article on SVAR. The package vars has useful functions for estimating the contemporaneous impact and long run impact matrices as follows for our artificial example. The function ‘BQ’ does the work.
BQ(V1)#assume V1 from VAR is in memory

SVAR Estimation Results:
========================
Estimated contemporaneous impact matrix:
y1 y2 y3
y1 6.448 2.180 0.8117
y2 -4.223 5.728 -1.7502
y3 -2.978 3.763 4.3754

Estimated identified long run impact matrix:
y1 y2 y3
y1 6.1133 0.0000 0.000
y2 -7.4544 7.6928 0.000
y3 -0.6025 0.3119 4.901

We shall explain additional features available in R with additional examples in following exercises.

5.4 Exercise (VAR Estimation: Macrodata)

Estimate the VAR model for the ‘Macrodat’ data of package ‘Ecdat’ (1) Discuss why only some of the series can be included due to NA’s in the Japanese exchange rate data during the war years. Comment on the criteria used for selection of the number of lags and parsimony issues in the ultimate choice.

ANSWER:

```r
library(Ecdat); library(vars)
data(Macrodat)
Macrodat=as.data.frame(Macrodat)
attach(Macrodat)
options(digits=4, show.signif.stars=FALSE) # easier to read
head(Macrodat); tail(Macrodat)
summary(Macrodat) #look for NAs missing data
#we could have used the na.omit function
#if we wanted to keep Japanese GDP as a variable
VARselect(Macrodat[,1:5], lag.max=6)
```
var.2c=VAR(Macrodat[,1:5], p=2, type="const")
summary(var.2c)
mycol=c(1,2,3,4,7) #select subset of columns
VARselect(Macrodat[,mycol], lag.max=6)
#this fails because of missing data in col. 7
mycol=c(1,2,3,4,6) #revise column selection
VARselect(Macrodat[,mycol], lag.max=6)
#choose lag=2
var.2c=VAR(Macrodat[,mycol], p=2, type="const")
summary(var.2c)

The variable names are: lhur= unemployment rate (average of months in quarter), punew= cPI (Average of Months in Quarter), fyff= federal funds interest rate (last month in quarter), fygm3= 3 month treasury bill interest rate (last month in quarter), fygt1= 1 year treasury bond interest rate (last month in quarter), exruk= dollar / Pound exchange rate (last month in quarter), gdpjp= real GDP for Japan.

The output is large even if we have chosen only two lags. Variable names followed by ‘.l1’ and ‘.l2’ refer to variables with lag 1 and lag 2 respectively. The SC and HQ criteria suggest lag 2 from the ‘selection.’ It is customary to choose a parsimonious model. After all each regression involves some 11 coefficients and with five variables in the model we have 55 coefficients to be estimated. Even if the data series has 168 observations, it is not enough for larger models.

$selection
VARselect(Macrodat[,mycol], lag.max=6)

$selection
AIC(n) HQ(n) SC(n) FPE(n)
6 2 2 6

$criteria
1 2 3 4 5 6
AIC(n) -1.2809 -2.34256 -2.47985 -2.46240 -2.471072 -2.51048
HQ(n) -1.0487 -1.91696 -1.86078 -1.64988 -1.465088 -1.31109
SC(n) -0.7091 -1.29431 -0.95511 -0.46118 0.006629 0.4438
FPE(n) 0.2778 0.09618 0.08403 0.08586 0.085685 0.0832
> var.2c=VAR(Macrodat[,mycol], p=2, type="const")
> summary(var.2c)

VAR Estimation Results:
========================
Endogenous variables: lhur, punew, fyff, fygm3, exruk
Deterministic variables: const
Sample size: 166
Log Likelihood: -933.811
Roots of the characteristic polynomial:
   1 0.968 0.814 0.814 0.753 0.678 0.48 0.363 0.258 0.175
Call:
VAR(y = Macrodat[, mycol], p = 2, type = "const")

Estimation results for equation lhur:
=====================================

lhur = lhur.l1 + punew.l1 + fyff.l1 + fygm3.l1 + exruk.l1 + lhur.l2 + punew.l2 + fyff.l2 + fygm3.l2 + exruk.l2 + const

             Estimate Std. Error  t value  Pr(>|t|)
lhur.l1     1.414830  0.067617   20.92  < 2e-16 ***
punew.l1    0.076654  0.049086    1.56   0.1204
fyff.l1     0.055109  0.035023    1.57   0.1176
fygm3.l1    -0.065886  0.042332   -1.56   0.1216
exruk.l1   -0.000751  0.002513    -0.30   0.7654
lhur.l2    -0.455852  0.069155    -6.59  6.4e-10 ***
punew.l2   -0.076988  0.049404   -1.56   0.1212
fyff.l2     0.097857  0.035751     2.74   0.0069 **
fygm3.l2   -0.092847  0.042685    -2.18   0.0311 *
exruk.l2   0.001250  0.002393     0.52   0.6022
const      0.044642  0.320622     0.14   0.8894

---

Residual standard error: 0.236 on 155 degrees of freedom
Multiple R-Squared: 0.977,   Adjusted R-squared: 0.975
F-statistic: 653 on 10 and 155 DF,  p-value: <2e-16
### Estimation results for equation punew:

\[
\begin{align*}
\text{punew} &= \text{lhur.l1} + \text{punew.l1} + \text{fyff.l1} + \text{fygm3.l1} + \text{exruk.l1} \\
&+ \text{lhur.l2} + \text{punew.l2} + \text{fyff.l2} + \text{fygm3.l2} + \text{exruk.l2} + \text{const}
\end{align*}
\]

|         | Estimate | Std. Error | t value | Pr(>|t|) |
|---------|----------|------------|---------|----------|
| lhur.l1 | -0.01312 | 0.09158    | -0.14   | 0.886    |
| punew.l1| 1.60463  | 0.06648    | 24.14   | <2e-16 ***|
| fyff.l1 | 0.09347  | 0.04743    | 1.97    | 0.051 .  |
| fygm3.l1| 0.05677  | 0.05733    | 0.99    | 0.324    |
| exruk.l1| 0.00148  | 0.00340    | 0.43    | 0.665    |
| lhur.l2 | 0.01477  | 0.09366    | 0.16    | 0.875    |
| punew.l2| -0.60331 | 0.06691    | -9.02   | 7e-16 ***|
| fyff.l2 | -0.04578 | 0.04842    | -0.95   | 0.346    |
| fygm3.l2| -0.06737 | 0.05781    | -1.17   | 0.246    |
| exruk.l2| -0.00224 | 0.00324    | -0.69   | 0.490    |
| const   | 0.13555  | 0.43425    | 0.31    | 0.755    |

---

Residual standard error: 0.32 on 155 degrees of freedom  
Multiple R-Squared: 1,  Adjusted R-squared: 1  
F-statistic: 3.84e+05 on 10 and 155 DF, p-value: <2e-16

### Estimation results for equation fyff:

\[
\begin{align*}
fyff &= \text{lhur.l1} + \text{punew.l1} + \text{fyff.l1} + \text{fygm3.l1} + \text{exruk.l1} \\
&+ \text{lhur.l2} + \text{punew.l2} + \text{fyff.l2} + \text{fygm3.l2} + \text{exruk.l2} + \text{const}
\end{align*}
\]

|         | Estimate | Std. Error | t value | Pr(>|t|) |
|---------|----------|------------|---------|----------|
| lhur.l1 | -0.9054  | 0.3789     | -2.39   | 0.018 *  |
| punew.l1| 0.2461   | 0.2751     | 0.89    | 0.372    |
| fyff.l1 | 0.3780   | 0.1962     | 1.93    | 0.056 .  |
| fygm3.l1| 0.4776   | 0.2372     | 2.01    | 0.046 *  |
| exruk.l1| 0.0317   | 0.0141     | 2.25    | 0.026 *  |
| lhur.l2 | 0.8674   | 0.3875     | 2.24    | 0.027 *  |
punew.l2 -0.2474 0.2768 -0.89 0.373
fyff.l2 0.2369 0.2003 1.18 0.239
fygm3.l2 -0.0913 0.2392 -0.38 0.703
exruk.l2 -0.0297 0.0134 -2.22 0.028 *
const -0.0367 1.7966 -0.02 0.984
---

Residual standard error: 1.32 on 155 degrees of freedom
Multiple R-Squared: 0.841, Adjusted R-squared: 0.831
F-statistic: 82.2 on 10 and 155 DF, p-value: <2e-16

Estimation results for equation fygm3:
======================================
fygm3 = lhur.l1 + punew.l1 + fyff.l1 + fygm3.l1 + exruk.l1 + lhur.l2 + punew.l2 + fyff.l2 + fygm3.l2 + exruk.l2 + const

| Estimate  | Std. Error | t value | Pr(>|t|) |
|-----------|------------|---------|----------|
| lhur.l1   | -0.63940   | 0.31887 | -2.01    | 0.0467 * |
| punew.l1  | 0.22362    | 0.23148 | 0.97     | 0.3355   |
| fyff.l1   | 0.15582    | 0.16516 | 0.94     | 0.3469   |
| fygm3.l1  | 0.54706    | 0.19963 | 2.74     | 0.0069 **|
| exruk.l1  | 0.01734    | 0.01185 | 1.46     | 0.1454   |
| lhur.l2   | 0.66721    | 0.32612 | 2.05     | 0.0425 * |
| punew.l2  | -0.22452   | 0.23298 | -0.96    | 0.3367   |
| fyff.l2   | 0.00964    | 0.16860 | 0.06     | 0.9545   |
| fygm3.l2  | 0.15566    | 0.20130 | 0.77     | 0.4405   |
| exruk.l2  | -0.01598   | 0.01129 | -1.42    | 0.1589   |
| const     | 0.14763    | 1.51200 | 0.10     | 0.9223   |
---

Residual standard error: 1.11 on 155 degrees of freedom
Multiple R-Squared: 0.831, Adjusted R-squared: 0.82
F-statistic: 76.3 on 10 and 155 DF, p-value: <2e-16

Estimation results for equation exruk:
======================================
exruk = lhur.l1 + punew.l1 + fyff.l1 + fygm3.l1 + exruk.l1
+ lhur.l2 + punew.l2 + fyff.l2 + fygm3.l2 + exruk.l2 + const

|         | Estimate | Std. Error | t value | Pr(>|t|) |
|---------|----------|------------|---------|---------|
| lhur.l1 | -1.2792  | 2.1796     | -0.59   | 0.5581  |
| punew.l1| 1.8984   | 1.5823     | 1.20    | 0.2321  |
| fyff.l1 | -0.6441  | 1.1289     | -0.57   | 0.5691  |
| fygm3.l1| 0.6396   | 1.3645     | 0.47    | 0.6399  |
| exruk.l1| 1.0875   | 0.0810     | 13.43   | < 2e-16 *** |
| lhur.l2 | -0.1567  | 2.2291     | -0.07   | 0.9441  |
| punew.l2| -1.9963  | 1.5925     | -1.25   | 0.2119  |
| fyff.l2 | 1.1890   | 1.1524     | 1.03    | 0.3038  |
| fygm3.l2| -1.9983  | 1.3759     | -1.45   | 0.1484  |
| exruk.l2| -0.2025  | 0.0771     | -2.63   | 0.0095 ** |
| const   | 42.6920  | 10.3349    | 4.13    | 5.9e-05 *** |

---

Residual standard error: 7.61 on 155 degrees of freedom
Multiple R-Squared: 0.979, Adjusted R-squared: 0.977
F-statistic: 706 on 10 and 155 DF, p-value: <2e-16

Covariance matrix of residuals:

<table>
<thead>
<tr>
<th></th>
<th>lhur</th>
<th>punew</th>
<th>fyff</th>
<th>fygm3</th>
<th>exruk</th>
</tr>
</thead>
<tbody>
<tr>
<td>lhur</td>
<td>0.05568</td>
<td>-0.00389</td>
<td>-0.142</td>
<td>-0.1125</td>
<td>0.331</td>
</tr>
<tr>
<td>punew</td>
<td>-0.00389</td>
<td>0.10213</td>
<td>0.104</td>
<td>0.0943</td>
<td>0.201</td>
</tr>
<tr>
<td>fyff</td>
<td>-0.14189</td>
<td>0.10365</td>
<td>1.748</td>
<td>1.3521</td>
<td>-2.247</td>
</tr>
<tr>
<td>fygm3</td>
<td>-0.11253</td>
<td>0.09428</td>
<td>1.352</td>
<td>1.2382</td>
<td>-1.884</td>
</tr>
<tr>
<td>exruk</td>
<td>0.33146</td>
<td>0.20136</td>
<td>-2.247</td>
<td>-1.8845</td>
<td>57.849</td>
</tr>
</tbody>
</table>

Correlation matrix of residuals:

<table>
<thead>
<tr>
<th></th>
<th>lhur</th>
<th>punew</th>
<th>fyff</th>
<th>fygm3</th>
<th>exruk</th>
</tr>
</thead>
<tbody>
<tr>
<td>lhur</td>
<td>1.0000</td>
<td>-0.0516</td>
<td>-0.455</td>
<td>-0.429</td>
<td>0.1847</td>
</tr>
<tr>
<td>punew</td>
<td>-0.0516</td>
<td>1.0000</td>
<td>0.245</td>
<td>0.265</td>
<td>0.0828</td>
</tr>
<tr>
<td>fyff</td>
<td>-0.4548</td>
<td>0.2453</td>
<td>1.000</td>
<td>0.919</td>
<td>-0.2235</td>
</tr>
<tr>
<td>fygm3</td>
<td>-0.4286</td>
<td>0.2651</td>
<td>0.919</td>
<td>1.000</td>
<td>-0.2227</td>
</tr>
<tr>
<td>exruk</td>
<td>0.1847</td>
<td>0.0828</td>
<td>-0.223</td>
<td>-0.223</td>
<td>1.0000</td>
</tr>
</tbody>
</table>
Figure 1: Five step ahead prediction based on VAR estimation using five ‘Macrodat’ variables from ‘Ecdat’ package. A fanchart shows 90% confidence interval.

5.5 Exercise (VAR Forecasting: Macrodata)

Estimate the VAR model for the ‘Macrodat’ data of package ‘Ecdat’ as in the previous exercise. Now use the ‘predict’ command to forecast 5 time periods and indicate the 90% confidence interval. Use ‘fanchart’ to depict the results.

ANSWER:

```r
#entire run of the previous exercise must be in the memory of R
disregard previous code block
var.f5=predict(var.2c, n.ahead=5, ci=0.90)
fanchart(var.f5)
```
5.6 Exercise (VAR Impulse Response Analysis, ‘Macrodat’)

Estimate the VAR model as above and study the response of a unit impulse in interest rate variables on unemployment, inflation and exchange rate variables.

**ANSWER:**

```
# entire run of the previous 2 exercises must be in the memory of R
irf.1=irf(var.2c, impulse="fygm3", response= c("lhur", "punew", "exruk"), boot=F)
irf.1$irf$fygm3
```

The following 11 by 3 matrix of responses is from an impulse to the interest rate variable ‘fygm3’. As interest increases, unemployment and exchange rate decrease, while inflation increases.

```
lhur  punew  exruk
[1,]  0.00000  0.00000 -0.4246
[2,] -0.02841  0.02413 -0.1828
[3,] -0.08371  0.04172 -0.8796
[4,] -0.11264  0.05907 -0.9639
[5,] -0.12730  0.08127 -0.9821
[6,] -0.12546  0.10495 -0.8423
[7,] -0.11365  0.12969 -0.6780
[8,] -0.09528  0.15359 -0.5102
[9,] -0.07403  0.17580 -0.3733
[10,] -0.05231  0.19580 -0.2756
[11,] -0.03189  0.21321 -0.2204
```

5.7 Exercise (Canonical Correlation, artificial Data)

Use R seed 10, create a long array of 100 uniform random numbers in the range 10 to 60 and make 4 columns (y1, y2, x1, x2) from these numbers with the first 1:25 for y1 and next 25 for y2 and so forth. Compute the canonical correlation coefficient between y1 y2 as a set on the one hand and x1 x2 as a set on the other.

Is it necessary to have same number of variables in the two sets?
Extend the example to first have 3 variables on the left and 2 variables on the right. Extend again to have 2 variables on the left and 3 variables on the right.

Answer:

```r
set.seed(10)
xx = runif(100, min=10, max=60)
yy = matrix(xx, nrow=25)
y1 = yy[,1]; y2 = yy[,2]; x1 = yy[,3]; x2 = yy[,4]
ca = cancor(cbind(y1,y2), cbind(x1,x2))
ca #print output

$cor
[1] 0.3323867 0.2288096

$xcoef
[,1]  [,2]
y1 0.012665095 0.01307055
y2 0.008898103 -0.01010734

$ycoef
[,1]  [,2]
x1 -0.014413392 -0.008339541
x2 -0.007689104  0.013075976

$xcenter
y1  y2
32.24092 33.61918

$ycenter
  x1   x2
30.72755 32.47735
```

There are as many sets of answers as are variables in the smaller of the two sets. From among the answers, we generally pick the first set as the one associated with the largest eigenvalue and hence best fitting. For example, the largest canonical correlation coefficient is given by the first element of ‘$cor’ in the output above. Also it is the first rows of coefficients under
‘$xcoef$ and $ycoef$' that are the relevant regression coefficients of the best fitting relation.

We can extend the problem as follows.

\[
\begin{align*}
yy &= \text{matrix}(xx, nrow=20) \\
y1 &= yy[,1]; y2=yy[,2]; x1=yy[,3]; x2=yy[,4]; x3=yy[,5] \\
ca &= \text{cancor}(\text{cbind}(y1, y2), \text{cbind}(x1, x2, x3)); ca \\
ca &= \text{cancor}(\text{cbind}(y1, y2, x3), \text{cbind}(x1, x2)); ca
\end{align*}
\]

The output of the extension is suppressed for brevity.

5.8 Exercise (Canonical Correlation Estimation: California data)

Estimate the ‘cancor’ model for the ‘Caschool’ data of package ‘Ecdat’. Use two dependent variables: readscr = average reading score, mathscr = average math score. Use following regressors: compstu= computer per student, expnstu= expenditure per student, str= student teacher ratio, avginc= district average income, and elpct= percent of English learners. We are interested in the effect of student teacher ratio ‘str’ on math and reading scores. Use bootstrap confidence intervals to assess these effects.

ANSWER:

\[
\begin{align*}
\texttt{rm(list=ls())} & \quad \# \text{clean out memory of R new problem} \\
\texttt{library(Ecdat)} & \\
\texttt{data(Caschool)} & \\
\texttt{attach(Caschool)} & \\
\texttt{expstu=expnstu/1000} & \quad \# \text{rescale for comparability} \\
\texttt{cc=cancor(cbind(compstu, expstu, str, avginc, elpct), cbind(readscr, mathscr))} & \quad \# \text{Inputs first then outputs} \\
\texttt{options(digits=4)} & \quad \# \\
\texttt{cc}
\end{align*}
\]

The following output is produced by R.

\[
\begin{align*}
\texttt{cor} & \\
[1] & 0.8601 0.3299
\end{align*}
\]
alph = \text{cc}$xcoef[,1]
bet = \text{cc}$ycoef[,1]
if (bet[1]<0) {bet=-bet; alph=-alph}
rho = \sqrt{\text{cc}$cor[1]}\)
print("The fitted relation is:")
cout = c(bet, rho*alph)
round(cout, 5) # round answers for printing

It appears that the sign of the response is different for reading and math scores. Is it related to the order in which outputs are entered? Is it due to collinearity between expenditures on students and student teacher ratio? We try the following version.
cor(expstu,str) #correlation exceeds 0.6
c2=cancor(cbind(compstu, str, avginc, elpct),
cbind(mathscr, reads)) #output order reversed
alph=c2$xcoef[,1]
bet=c2$ycoef[,1]
if (bet[1]<0) {bet=-bet; alph=-alph}
rho=sqrt(c2$cor[1])
print("The fitted relation is:")
cout2=c(bet, rho*alph)
round(cout2,5) #round answers for

mathscr readscompstu str avginc elpct
0.00018 -0.00259 -0.03621 0.00046 -0.00380 0.00150

It appears that the sign of the response remains distinct for reading and math scores even after re-ordering as they should. Omission of a correlated variable has changed the sign of ‘str’ coefficient. In the sequel we will keep out the expenditure variable for brevity.

What about effect of input variables on output variables? In particular we are interested in the effect of student teacher ratio on math and reading scores.

Recall the definitional relation from the text:

\[ F = Y\beta - \rho X\alpha. \]  

(5)

One estimates the effect of input \( x_1 \) on the first output \( y_1 \) by using the implicit function theorem on \( F \) as follows: (See (eq. 5.2.2) of the text):

\[ \frac{dy_1}{dx_1} = -\frac{\text{Num}}{\text{Den}}, \text{ where } \text{Num} = \frac{\partial F}{\partial x_1}, \text{ Den} = \frac{\partial F}{\partial y_1}. \]  

(6)

outF=c(bet,-rho*alph) #sign of second term changed
round(outF,5) #

Partials of \( F \) with respect to various variables are given quite simply as:

mathscr readscompstu str avginc elpct
0.00018 -0.00259 0.03621 -0.00046 0.00380 -0.00150

18
Substituting these into equation (6) above we have \( \frac{d\text{mathscr}}{d\text{str}} = -\frac{-0.00046}{0.00018} = 2.556 \).

\( \frac{d\text{readscr}}{d\text{str}} = -\frac{-0.00046}{-0.00259} = 0.1776 \).

This suggests that crowded class rooms increase the math score but decrease the reading score. Our next task it to construct confidence intervals around these estimates. This will involve using the R function ‘sample’ for resampling from the data.

```r
n999=999 #size of resample
dmaths=matrix(rep(NA, n999), nrow=n999, ncol=1)
dreads=matrix(rep(NA, n999), nrow=n999, ncol=1)
#above are places to store bootstrap estimates
set.seed(345)
for (i in 1:n999){
  mathscr2=sample(mathscr,replace=T)
  readscr2=sample(readscr,replace=T)
  compstu2=sample(compstu ,replace=T)
  str2 =sample(str ,replace=T)
  avginc2 =sample(avginc ,replace=T)
  elpct2 =sample(elpct ,replace=T)
  cc=cancor(cbind(compstu2, str2, avginc2, elpct2),
  cbind(mathscr2,readscr2))
  alph=cc$xcoef[,1] #x-side coefficients
  bet=cc$ycoef[,1]
  rho=sqrt(cc$cor[1])
  #print("The fitted relation is:")
  #if sign of weight on sr is negative, change all signs
  #if (bet[1]<0) {bet=-bet; alph=-alph}
  outF=c(bet,-rho*alph)
  #outF this is where we store bootstrap results
  dmath=-outF[4]/outF[1]
  dread=-outF[4]/outF[2]
  dmaths[i,]=dmath
  dreads[i,]=dread
} #end of i loop for 999 iterations
partial.maths=quantile(dmaths, c(0.025, 0.975))
partial.reading=quantile(dreads, c(0.025, 0.975))
```
\begin{verbatim}
reg1=lm(mathscr^compstu+ str+ avginc+ elpct)  
reg2=lm(readscr^compstu+ str+ avginc+ elpct)  
OLS.maths=confint(reg1)[3,]  
OLS.reading=confint(reg2)[3,]  
rbind(partial.maths,OLS.maths, partial.reading, OLS.reading)

Paritals with respect to student teacher ratio have following confidence limits.  
We include comparable OLS limits when only one output is used. The results  
agree for both. Joint estimation gives much wider intervals here, suggesting  
a greater uncertainty than what is revealed by OLS.

\begin{tabular}{lrr}
 & 2.5\% & 97.5\% \\
\text{partial.maths} & -18.0527 & 15.2177 \\
\text{OLS.maths} & -0.3741 & 0.8745 \\
\text{partial.reading} & -18.8731 & 20.3324 \\
\text{OLS.reading} & -0.7074 & 0.4155 \\
\end{tabular}

Since zero is inside all these intervals, we conclude that student teacher ratio  
does \textit{not} have a statistically significant effect on math or reading scores.  
Since we are talking about the same student, the math and reading scores  
are \\
‘jointly’ produced and our canonical correlations approach has intuitive  
merit. However, the confidence intervals based on traditional iid bootstrap  
seem too wide. Hence, we may need to apply maximum entropy bootstrap  
(package=‘meboot’) \textsuperscript{(5)} designed for time series (discussed in Chapter 9).  
See \textsuperscript{(3)}.

5.9 Exercise (Canonical Correlation using ‘meboot’)

Since traditional iid bootstrap confidence intervals seem too wide, apply max-  
imum entropy bootstrap (package=‘meboot’) \textsuperscript{(5)} designed for dependent data  
(time series or spatial data \textsuperscript{(2)}). See \textsuperscript{(3)}, \textsuperscript{(4)} and \textsuperscript{(6)}. Show how to use me-  
boot for this problem.

Hints: Generate 999 replicates of each of the series and estimate 999  
canonical correlations.

\begin{verbatim}
library(meboot)
n999=999 #size of resample
dmaths=matrix(rep(NA, n999), nrow=n999, ncol=1)
\end{verbatim}
\end{verbatim}
dreads = matrix(rep(NA, n999), nrow=n999, ncol=1)
# above are places to store bootstrap estimates
mathscr2 = meboot(x=mathscr, reps=999)$ensemble
readscr2 = meboot(x=readscr, reps=999)$ensemble
compstu2 = meboot(x=compstu, reps=999)$ensemble
str2 = meboot(x=str, reps=999)$ensemble
avginc2 = meboot(x=avginc, reps=999)$ensemble
elpct2 = meboot(x=elpct, reps=999)$ensemble
for (i in 1:n999){
    cc = cancor(cbind(compstu2[,i], str2[,i], avginc2[,i], elpct2[,i]),
                cbind(mathscr2[,i], readscr2[,i]))
    alph = cc$xcoef[,1]  # x-side coefficients
    bet = cc$ycoef[,1]
    rho = sqrt(cc$cor[1])
    # print("The fitted relation is:")
    # if sign of weight on sr is negative, change all signs
    # if (bet[1]<0) {bet=-bet; alph=-alph}
    outF = c(bet, -rho*alph)
    # outF this is where we store bootstrap results
    dmath = -outF[4]/outF[1]
    dread = -outF[4]/outF[2]
    dmaths[i,] = dmath
    dreads[i,] = dread
}  # end of i loop for 999 iterations
partial.maths = quantile(dmaths, c(0.025, 0.975))
partial.reading = quantile(dreads, c(0.025, 0.975))
reg1 = lm(mathscr ~ compstu + str + avginc + elpct)
reg2 = lm(readscr ~ compstu + str + avginc + elpct)
OLS.maths = confint(reg1)[3,]
OLS.reading = confint(reg2)[3,]
rbind(partial.maths, OLS.maths, partial.reading, OLS.reading)

Using meboot seems to make one change in the confidence intervals for ‘partial.reading’ staying only in the negative region.

<table>
<thead>
<tr>
<th></th>
<th>2.5%</th>
<th>97.5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>partial.maths</td>
<td>-15.4093</td>
<td>20.69276</td>
</tr>
<tr>
<td>OLS.maths</td>
<td>-0.3741</td>
<td>0.87447</td>
</tr>
</tbody>
</table>
The confidence interval is a bit wider for math scores using canonical correlations and meboot, but is much shorter and meaningful for reading scores (compared to iid boot). This shows that meboot does not shorten the confidence intervals, willy nilly. These meboot results support the claim that crowded class rooms have a significantly negative impact on reading scores of pupils.

5.10 Exercise (Canonical Correlation Testing: Macrodata)

Consider the VAR model for the ‘Macrodat’ data of package ‘Ecdat’ using variables ‘lhur’, ‘punew’, ‘fyff’, ‘fygm3’, and ‘exruk’. Use Johansen procedure (which uses a trace statistic and canonical correleions) to determine the number \( r \) of cointegrating vectors in these data.

```r
rm(list=ls()) #clean out memory of R new problem
library(Ecdat); library(urca)
data(Macrodat)
Macrodat=as.data.frame(Macrodat)
attach(Macrodat)
options(digits=4, show.signif.stars=FALSE)
mycol=c(1,2,3,4,6)
summary(ca.jo(Macrodat[,mycol],K=2, type="trace",ecdet="const"))
```

```
Eigenvalues (lambda):
[1] 3.233e-01 2.149e-01 1.441e-01 9.573e-02 1.998e-02 -1.316e-16

Values of test statistic and critical values of test:
```

---

partial.reading  -0.2705  -0.03297
OLS.reading      -0.7074   0.41553
| r <= 4 | 3.35 | 7.52 | 9.24 | 12.97 |
| r <= 3 | 20.05 | 17.85 | 19.96 | 24.60 |
| r <= 2 | 45.89 | 32.00 | 34.91 | 41.07 |
| r <= 1 | 86.06 | 49.65 | 53.12 | 60.16 |
| r = 0 | 150.89 | 71.86 | 76.07 | 84.45 |

Eigenvectors, normalized to first column:
(These are the cointegration relations)

| lhur.l2  punew.l2  fyff.l2  fygm3.l2  exruk.l2  constant |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| lhur.l2         | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 |
| punew.l2        | 0.040673 | 0.05282  | -0.08943  | 0.04328  | 0.01210  | -0.11608 |
| fyff.l2         | -4.991919 | 7.61919  | 2.78274   | 0.13974  | 0.70550  | 0.35830  |
| fygm3.l2        | 6.447836 | -8.31223 | -5.99222  | -0.26559 | -0.11799 | -0.35058 |
| exruk.l2        | -0.007286 | 0.01650  | -0.10606  | 0.06121  | -0.02113 | -0.05075 |

Weights W:
(This is the loading matrix)

| lhur.d  punew.d  fyff.d  fygm3.d  exruk.d  constant |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| lhur.d          | -0.025977 | 0.006898 | -0.010458 | -0.01404 | 0.002556 | 1.156e-16 |
| punew.d         | 0.020517  | 0.023273 | -0.007004 | -0.03017 | -0.004965 | -1.597e-16 |
| fyff.d          | 0.006938  | -0.044054 | 0.003272 | 0.03694  | -0.041136 | -5.928e-16 |
| fygm3.d         | -0.016564 | 0.004167 | 0.024470 | 0.04979  | -0.034058 | -4.240e-16 |
| exruk.d         | -0.192377 | -0.125578 | 0.255766 | -1.41191 | 0.038191 | 2.652e-15  |

The test statistic for the hypothesis that number of cointegrating vectors $r \leq 3$ is 20.05. Since the 5% critical value is smaller (=19.96), we reject the hypothesis. Similarly we reject the hypothesis $r \leq 2$, (the statistic= 45.89 is less than the critical value=34.91). We conclude that there are 2 or 3 cointegrating vectors in these US Macroeconomic time series.
5.11 Exercise (VAR Estimation: Money data)
Estimate the VAR model and impulse response function for the ‘Money’ data of package ‘Ecdat’. (Hint: Use the earlier exercises for Macrodata as a template)

5.12 Exercise (VAR Estimation: artificial data)
Use seed=369, sample the integers from 1 to 100 and place them into a 20 by 5 matrix. Name the variables as a to e. Now estimate the VAR model and impulse response function for these artificial data. Plot your results.

```r
library(vars)
set.seed(369)
x=matrix(sample(1:100),20,5)
colnames(x)=letters[1:5]
attach(as.data.frame(x))
library(vars)
set.seed(369)
x=matrix(sample(1:100,20,5)
colnames(x)=letters[1:5]
attach(as.data.frame(x))
V1=VAR(x)
iV1=irf(V1); iV1
plot(iV1)
```

5.13 Exercise (VAR model statement, estimation forecasting and Causality)
1) Write out equation formulas for (5.1.1) and (5.1.2) using Latex for VAR(3) when K=3 variables are involved. Collect Internet US data from 3 quarterly macro series: Money supply, average wage and real GDP. (Get up to date and long time series).
2) Indicate using ‘VARselect’ function of ‘urca’ package, how many lags (say $p$) are most appropriate for your data. Then estimate the VAR($p$) model.
3) Does money supply Granger cause real GDP?
4) Use the estimated VAR model to draw fan charts for 3 variables assuming 5-step ahead forecasting. Draw the effect of an impulse in money supply on the other two variables.

HINT: Here is some LATEX code:

\begin{equation}
y_{t} = A_{1}y_{t - 1} + \cdots + A_{p}y_{t - p} + u_{t}
\end{equation}

\begin{align}
y_{1t} &= A_{1,11} y_{1,t -1} + A_{1,12} y_{2,t -1} + A_{2,11} y_{1,t -2} + A_{2,12} y_{2,t -2} + u_{1t}, \\
y_{2t} &= A_{1,21} y_{1,t -1} + A_{1,22} y_{2,t -1} + A_{2,21} y_{1,t -2} + A_{2,22} y_{2,t -2} + u_{2t},
\end{align}

5.14 Exercise (Joint production functions)

1) In a joint production function with 3 outputs and 3 inputs write the formula for the production function and for the marginal productivity and marinal elasticity of the first output $y_{1}$ w.r.t. the second input $x_{2}$.

2) Use the data of the urca package for UK ppp and uip mentioned in the snippet R5.3.1. (Hint: data set is called UKpppuip). Use two endogenous variables as the output variables and two regressors as inputs and estimate a joint production function using ‘cancor’.

5.15 Exercise (Cointegration testing UK data)

1) Use the data of the urca package for UK ppp and uip mentioned in the snippet R5.3.1. Estimate whether the variables are cointegrated by using Johansen procedure, Phillips-Ouliaris test and the Likelihood ration test using the ‘blrtest’ function.

2) How would you write the alpha and beta appearing in eq (5.3.3) for the model used above?
5.16 Exercise (VARMA estimation and forecasting)

Use Macroeconomic data from [http://research.stlouisfed.org/fred2/](http://research.stlouisfed.org/fred2/) coded as GDPC96, GDPDEF and TB3MS. Next define gdp, inf as 400 times the difference of the log of the original series (i.e., annualized log growth rates). Define tbi or treasury bill interest rate as the quarterly average of the monthly data. Now estimate VARMA(1,1) and VARMA(2,1) models for these data from 1947:Q2 to 2014:Q2. Now forecast the these data for h=2 extra steps for 2014:Q3 to 2014:Q4. Time series literature claims that ARMA(p,p-1) models are parsimonious and hence superior to VAR(4), VAR(2) models. Compare the out-of-sample forecasts of all these competing models for at least two steps ahead using the actual values to define forecast errors.

Hints:

```r
rm(list=ls()) #clean out memory of R new problem
library(bvarsv) #install this package first
data(usmacro)#this has 1947:Q2 to 2014:Q2 data for gdp, inf and tbi.
#you will have to get latest data from \`FRED\' website.
plot(usmacro)
summary(usmacro)
library(MTS)#install this package first
V4=VAR(da=usmacro, p=4)
#V4 is short for VAR4
Vp4=VARpred(model=V4, h=4)#four step ahead forecast
VM1=VARMA(da=usmacro, p=1,q=1)
#V1 is short for VARMA11
Vp1=VARMApred(model=VM1, h=4)#four step ahead forecast
VM21=VARMA(da=usmacro,p=2,q=1)
Vp21=VARMApred(model=VM21, h=4)#four step ahead forecast
#compare to true values for two steps and plot the results
```

Now we need to attach forecasts to the original data, define forecast errors and compare the mean squared errors of forecasts.

References


