STA457 Proj2

John Morales

2022-08-10

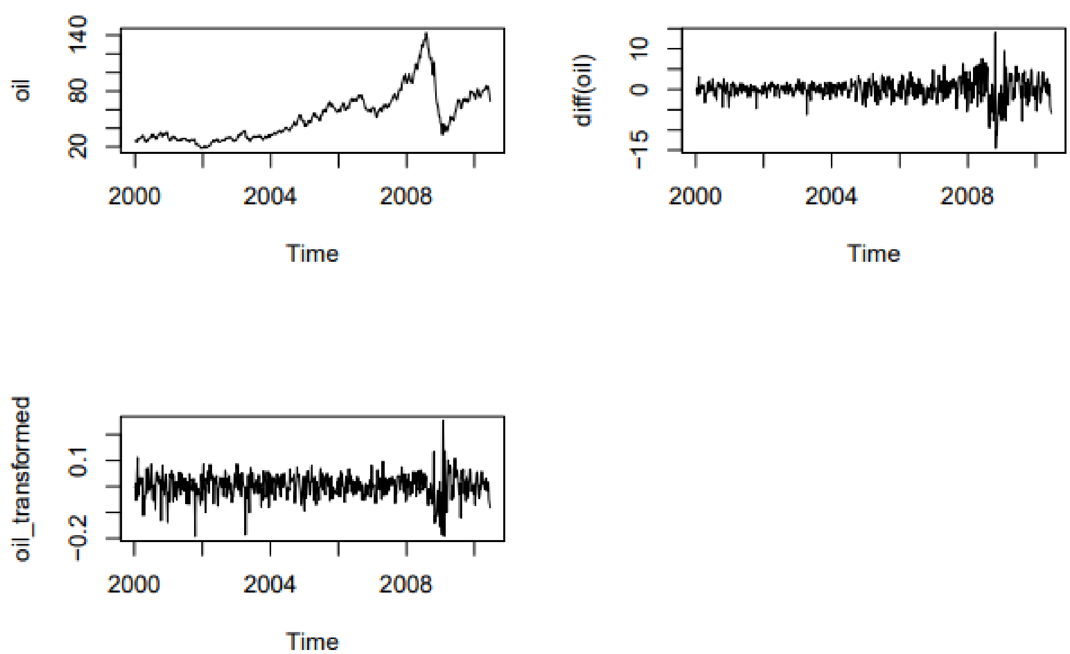
5.6 Weekly crude oil spot prices in dollars per barrel are in `oil`; see Problem Problem 2.10 and Appendix R for more details. Investigate whether the growth rate of the weekly oil price exhibits GARCH behavior. If so, fit an appropriate model to the growth rate.

Question 5.6

Transforming data

- Plotting the series, we see that it is non-stationary. (Top left)
- Transform data by differencing (Top right)
- After differencing, we see that the trend is gone BUT variance changes over time (grows!)
- Logging the logged difference makes it a little better.. Variance seems constant except at around 2009 (Bottom left)

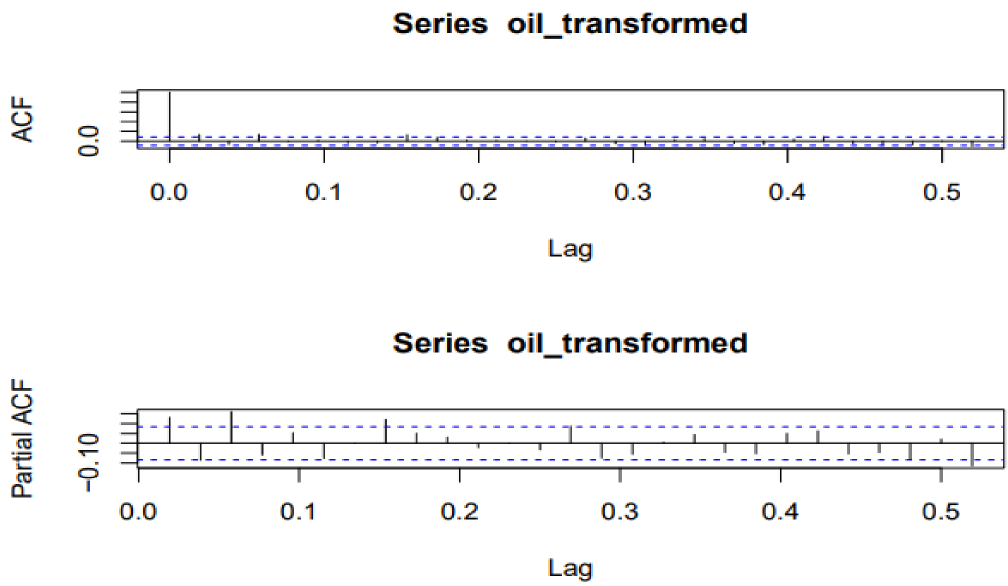
```
par(mfrow = c(2, 2))
plot(oil)
plot(diff(oil))
oil_transformed <- diff(log(oil))
plot(oil_transformed)
```



Finding ARIMA(p,d,q) by looking at ACF / PACF.

- Seems like after Lag 0, the ACF and PACF are both tailing off... Suggesting ARMA(1,1) since we differenced, we fit an ARIMA(1,1,1) model!

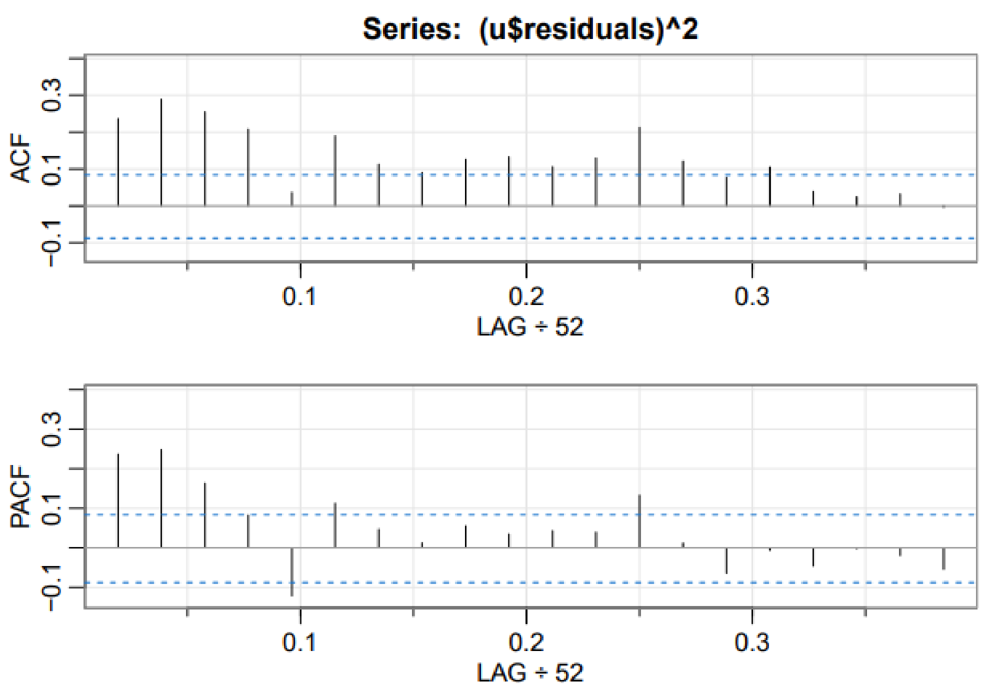
```
par(mfrow = c(2, 1))
# acf2 looked too messy
acf(oil_transformed)
pacf(oil_transformed)
```



Looking for GARCH(p,q) looking at ACF of squared residuals

- Squared residuals seem to tail off as well thus we can use GARCH(1,1)

```
u <- arima(diff(log(oil)), order=c(1,1,1))
acf2((u$residuals)^2,20)
```



```
##       [,1] [,2] [,3] [,4]  [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.24 0.29 0.26 0.21  0.04 0.19 0.11 0.09 0.13  0.13  0.11  0.13  0.21
## PACF 0.24 0.25 0.16 0.08 -0.12 0.11 0.05 0.01 0.06  0.03  0.04  0.04  0.13
##       [,14] [,15] [,16] [,17] [,18] [,19] [,20]
## ACF   0.12  0.08  0.10  0.04  0.02  0.03  0.00
## PACF  0.01 -0.07 -0.01 -0.05  0.00 -0.02 -0.05
```

Final Model Summary

```
oil_fit <- garchFit(~arma(1,1)+garch(1,1), data=oil_transformed)
```

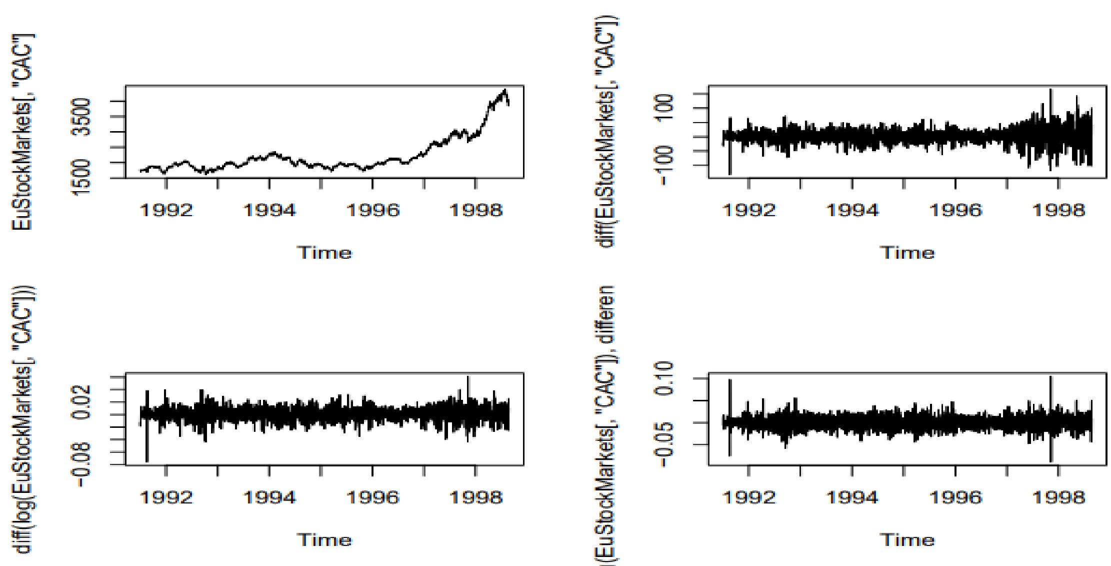
```
round(oil_fit@fit$matcoef,3)
```

```
##         Estimate  Std. Error  t value Pr(>|t|)
## mu         0.004       0.003    1.335    0.182
## ar1       -0.458       0.108   -4.250    0.000
## ma1        0.652       0.090    7.236    0.000
## omega      0.000       0.000    2.222    0.026
## alpha1     0.062       0.017    3.640    0.000
## beta1      0.881       0.035   25.337    0.000
```

5.7 The stats package of R contains the daily closing prices of four major European stock indices; type help(EuStockMarkets) for details. Fit a GARCH model to the returns of one of these series and discuss your findings. (Note: The data set contains actual values, and not returns. Hence, the data must be transformed prior to the model fitting.)

- Note I will only be dealing with CAC because I don't know how to deal with 3 different graphs
- Plotting the graph of series, we see that they are all non-stationary because trending (Top left)
- Need to transform this data. . .
- Differencing shows that variance grows over time, we can deal with this by logging. (top right)
- Difference logged time series looks better. (Bottom left)
- 2nd difference because I noticed the ACF/PACF was easier to read (choose p and q for)

```
par(mfrow = c(2, 2))
plot(EuStockMarkets[,"CAC"])
# Differencing shows that variance grows as time increases... We can deal with this by logging
plot(diff(EuStockMarkets[,"CAC"]))
# Logging the differenced series
plot(diff(log(EuStockMarkets[,"CAC"])))
# Logging the 2nd differenced series, because the ACF / PACF of the first diff was hard to use
plot(diff(log(EuStockMarkets[,"CAC"]), differences = 2))
```

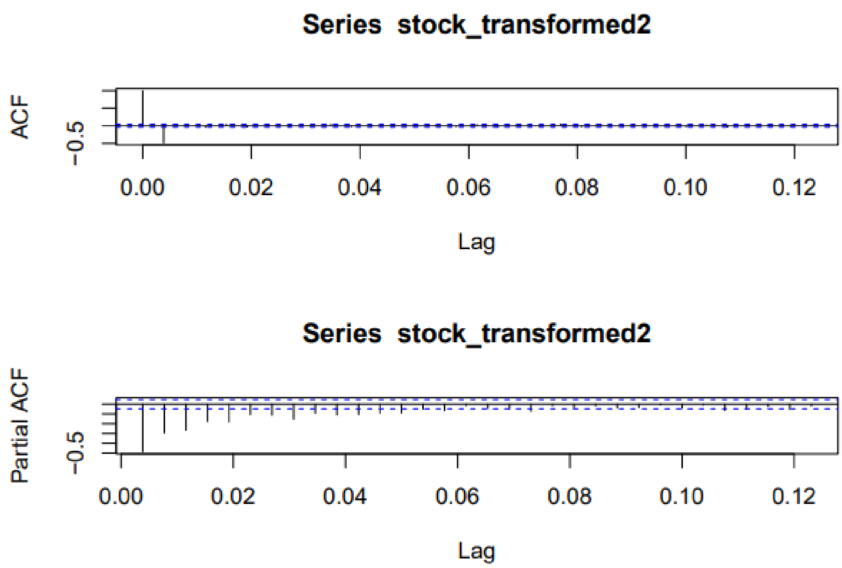



```
stock_transformed <- diff(log(EuStockMarkets[,"CAC"]))
stock_transformed2 <- diff(log(EuStockMarkets[,"CAC"]), differences = 2)
```

Plotting ACF/PACF of 2nd difference to find p and q for ARMA(p,q)

- Plotting the ACF/PACF of the differenced series can see a spike at lag 1 of ACF
- PACF seems to tail off
- Spike at lag 2 of ACF, PACF seems to tail off… Thus we can use MA(2)

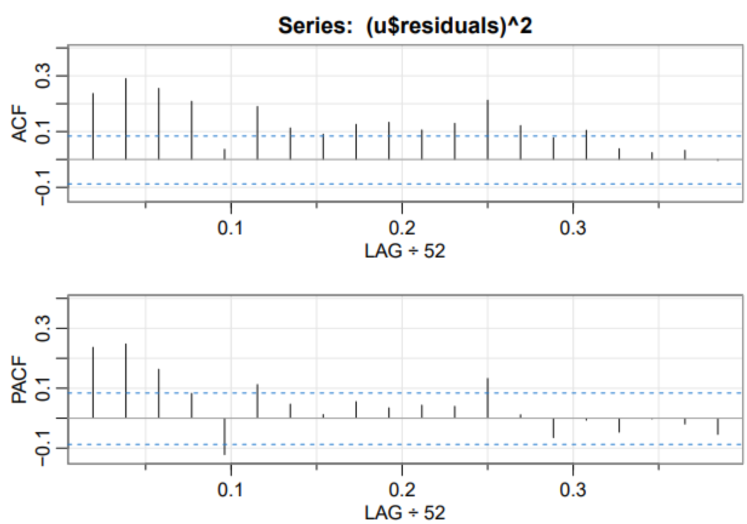
```
par(mfrow = c(2, 1))
# acf2 looked too messy
acf(stock_transformed2)
pacf(stock_transformed2)
```



Looking for GARCH(p,q) looking at ACF of squared residuals

- Squared residuals seem to tail off as well thus we can use GARCH(1,1)

```
v <- arima(stock_transformed2, order=c(2,2,0))
acf2((u$residuals)^2,20)
```




```
##      [,1] [,2] [,3] [,4]  [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.24 0.29 0.26 0.21  0.04 0.19 0.11 0.09 0.13  0.13  0.11  0.13  0.21
## PACF 0.24 0.25 0.16 0.08 -0.12 0.11 0.05 0.01 0.06  0.03  0.04  0.04  0.13
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20]
## ACF   0.12  0.08  0.10  0.04  0.02  0.03  0.00
## PACF  0.01 -0.07 -0.01 -0.05  0.00 -0.02 -0.05
```

Final Model Summary

```
stock_fit <- garchFit(~arma(0,2)+garch(1,1), data=stock_transformed2)
```

```
round(stock_fit@fit$matcoef,3)
```

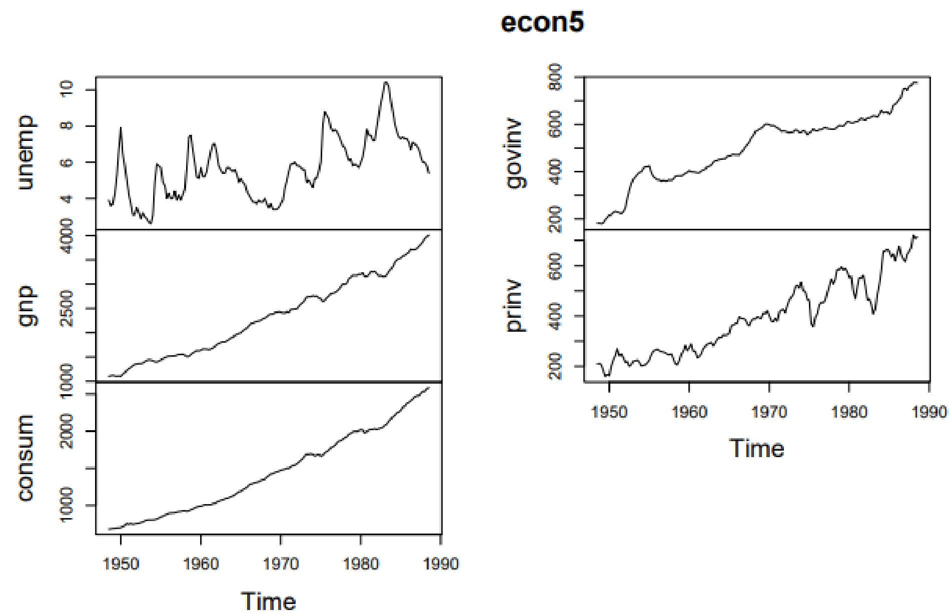
```
##         Estimate  Std. Error  t value Pr(>|t|)
## mu         0.000       0.000    1.112    0.266
## ma1       -0.952       0.024  -39.464    0.000
## ma2       -0.039       0.024   -1.621    0.105
## omega      0.000       0.000    2.361    0.018
## alpha1     0.056       0.014    4.068    0.000
## beta1      0.888       0.034   26.484    0.000
```

5.12 Consider the data set `econ5` containing quarterly U.S. unemployment, GNP, consumption, and government and private investment from 1948-III to 1988-II. The seasonal component has been removed from the data. Concentrating on unemployment (U_t), GNP (G_t), and consumption (C_t), fit a vector ARMA model to the data after first logging each series, and then removing the linear trend. That is, fit a vector ARMA model to $x_t = (x_{1t}, x_{2t}, x_{3t})'$, where, for example, $x_{1t} = \log(U_t) - \hat{\beta}_0 - \hat{\beta}_1 t$, where $\hat{\beta}_0$ and $\hat{\beta}_1$ are the least squares estimates for the regression of $\log(U_t)$ on time, t. Run a complete set of diagnostics on the residuals.

Question 5.12

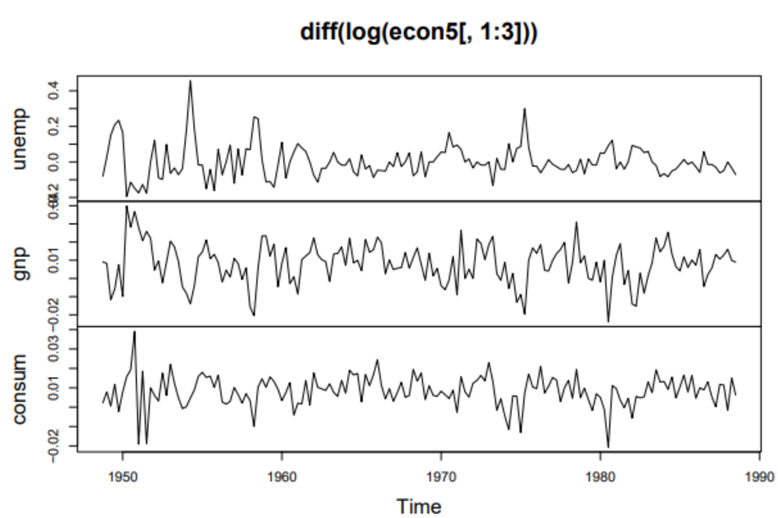
- Plotting data to know what it looks like
- Question only wants unemp, gnp, and consum graphs… so we clean it via vector [,1:3] on dataframe
- Question said to log series AND remove trend.

```
plot(econ5)
```



- Logging did not remove trend, so we also differenced.

```
plot(diff(log(econ5[,1:3])))
```



AIC, Hanan-Quinn, FPE selects order 2. BIC selects order = 1 model.

```r
x <- diff(log(econ5[,1:3]))
VARselect(x, lag.max=10, type="both") # Copying from example 5.11 page 275-277
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      2      2      1      2
##
## $criteria
##                    1             2             3             4             5
## AIC(n) -2.489945e+01 -2.499656e+01 -2.494475e+01 -2.492156e+01 -2.487441e+01
## HQ(n)  -2.477714e+01 -2.480086e+01 -2.467566e+01 -2.457908e+01 -2.445855e+01
## SC(n)  -2.459839e+01 -2.451486e+01 -2.428241e+01 -2.407858e+01 -2.385079e+01
## FPE(n)  1.535809e-11  1.394007e-11  1.468848e-11  1.504582e-11  1.579267e-11
##                    6             7             8             9            10
## AIC(n) -2.485355e+01 -2.482964e+01 -2.480870e+01 -2.477915e+01 -2.471488e+01
## HQ(n)  -2.436430e+01 -2.426700e+01 -2.417267e+01 -2.406974e+01 -2.393208e+01
## SC(n)  -2.364929e+01 -2.344475e+01 -2.324317e+01 -2.303298e+01 -2.278807e+01
## FPE(n)  1.615560e-11  1.658824e-11  1.699519e-11  1.757812e-11  1.884240e-11
```

Fitting a vector ARMA model to $x_t = (x_{1t}, x_{2t}, x_{3t})$'

```r
summary(fit <- VAR(x, p=2, type="both")) # From example 5.11, page 275-277
```

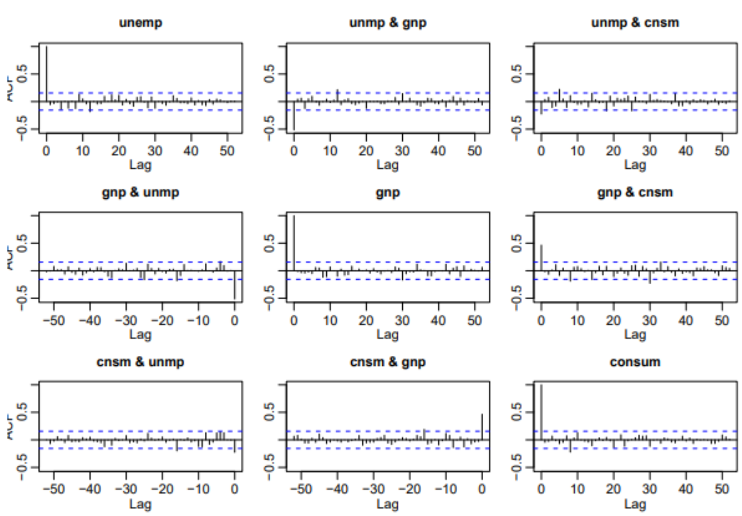
```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: unemp, gnp, consum
## Deterministic variables: both
## Sample size: 158
## Log Likelihood: 1295.967
## Roots of the characteristic polynomial:
## 0.6204 0.6204 0.5117 0.371 0.371 0.2314
## Call:
## VAR(y = x, p = 2, type = "both")
##
##
## Estimation results for equation unemp:
## ======================================
## unemp = unemp.l1 + gnp.l1 + consum.l1 + unemp.l2 + gnp.l2 + consum.l2 + const + trend
##
##             Estimate Std. Error t value Pr(>|t|)
## unemp.l1   0.0136614  0.0912304   0.150  0.88117
## gnp.l1    -3.0858661  0.7384455  -4.179 4.96e-05 ***
## consum.l1 -2.1649240  0.8150524  -2.656  0.00876 **
## unemp.l2  -0.1837207  0.0801581  -2.292  0.02330 *
## gnp.l2    -3.2344516  0.7718302  -4.191 4.73e-05 ***
## consum.l2 -0.1307268  0.8354722  -0.156  0.87587
## const      0.0932145  0.0165610   5.629 8.68e-08 ***
## trend     -0.0002494  0.0001262  -1.976  0.04997 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.07067 on 150 degrees of freedom
## Multiple R-Squared: 0.4458,  Adjusted R-squared: 0.4199
## F-statistic: 17.24 on 7 and 150 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation gnp:
## ====================================
## gnp = unemp.l1 + gnp.l1 + consum.l1 + unemp.l2 + gnp.l2 + consum.l2 + const + trend
##
##             Estimate Std. Error t value Pr(>|t|)
## unemp.l1  -6.722e-04  1.287e-02  -0.052  0.95843
## gnp.l1     2.105e-01  1.042e-01   2.020  0.04516 *
## consum.l1  2.780e-01  1.150e-01   2.417  0.01684 *
## unemp.l2   3.124e-02  1.131e-02   2.762  0.00647 **
## gnp.l2     2.504e-01  1.089e-01   2.299  0.02286 *
## consum.l2  1.208e-01  1.179e-01   1.025  0.30713
## const      1.440e-03  2.337e-03   0.616  0.53879
## trend     -6.372e-06  1.781e-05  -0.358  0.72098
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
##
## Residual standard error: 0.009972 on 150 degrees of freedom
## Multiple R-Squared: 0.2354,  Adjusted R-squared: 0.1997
## F-statistic: 6.596 on 7 and 150 DF,  p-value: 8.439e-07
##
##
## Estimation results for equation consum:
## =======================================
## consum = unemp.l1 + gnp.l1 + consum.l1 + unemp.l2 + gnp.l2 + consum.l2 + const + trend
##
##            Estimate Std. Error t value Pr(>|t|)
## unemp.l1   9.557e-03  1.013e-02   0.944  0.34678
## gnp.l1     1.485e-01  8.196e-02   1.812  0.07193 .
## consum.l1 -4.099e-02  9.046e-02  -0.453  0.65109
## unemp.l2   2.011e-02  8.897e-03   2.261  0.02522 *
## gnp.l2     8.143e-02  8.567e-02   0.951  0.34339
## consum.l2  2.180e-01  9.273e-02   2.351  0.02001 *
## const      4.891e-03  1.838e-03   2.661  0.00864 **
## trend      1.088e-06  1.401e-05   0.078  0.93817
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.007844 on 150 degrees of freedom
## Multiple R-Squared: 0.108,   Adjusted R-squared: 0.06637
## F-statistic: 2.594 on 7 and 150 DF,  p-value: 0.01486
##
##
##
## Covariance matrix of residuals:
##             unemp        gnp     consum
## unemp   0.0049945 -3.623e-04 -1.252e-04
## gnp    -0.0003623  9.943e-05  3.638e-05
## consum -0.0001252  3.638e-05  6.153e-05
##
## Correlation matrix of residuals:
##          unemp     gnp  consum
## unemp   1.0000 -0.5140 -0.2259
## gnp    -0.5140  1.0000  0.4651
## consum -0.2259  0.4651  1.0000
```

Running diagnostic on residuals

- Plotting cross-correlations of the residuals and examining the multivariate version of the Q-test
- Seems good

```
acf(resid(fit), 52)
```




```
serial.test(fit, lags.pt=12, type="PT.adjusted")
```

```
##
##  Portmanteau Test (adjusted)
##
## data:  Residuals of VAR object fit
## Chi-squared = 111.33, df = 90, p-value = 0.06325
```

Predictions from a VAR(2) fit to the unemployment - consumption data. Prediction interval

```
(fit.pr = predict(fit, n.ahead = 24, ci = 0.95)) # 4 weeks ahead
```

```
## $unemp
##                fcst      lower     upper        CI
##  [1,] -0.0168949590 -0.1554089 0.1216190 0.1385140
##  [2,] -0.0060119127 -0.1670955 0.1550716 0.1610836
##  [3,]  0.0011005217 -0.1787907 0.1809917 0.1798912
##  [4,]  0.0017354584 -0.1824143 0.1858852 0.1841498
##  [5,] -0.0003194392 -0.1854751 0.1848362 0.1851557
##  [6,] -0.0028030833 -0.1882110 0.1826049 0.1854079
##  [7,] -0.0043354017 -0.1899361 0.1812653 0.1856007
##  [8,] -0.0052419739 -0.1909530 0.1804690 0.1857110
##  [9,] -0.0054520359 -0.1912077 0.1803037 0.1857557
## [10,] -0.0054361845 -0.1912021 0.1803297 0.1857659
## [11,] -0.0053241911 -0.1910923 0.1804439 0.1857681
## [12,] -0.0052624252 -0.1910317 0.1805069 0.1857693
## [13,] -0.0052602756 -0.1910304 0.1805099 0.1857702
## [14,] -0.0053143711 -0.1910849 0.1804562 0.1857706
## [15,] -0.0053966038 -0.1911673 0.1803741 0.1857707
## [16,] -0.0054898667 -0.1912605 0.1802808 0.1857707
## [17,] -0.0055817946 -0.1913525 0.1801889 0.1857707
## [18,] -0.0056690893 -0.1914398 0.1801016 0.1857707
## [19,] -0.0057516008 -0.1915223 0.1800191 0.1857707
## [20,] -0.0058312183 -0.1916019 0.1799395 0.1857707
## [21,] -0.0059095544 -0.1916803 0.1798611 0.1857707
## [22,] -0.0059877492 -0.1917585 0.1797830 0.1857707
## [23,] -0.0060662563 -0.1918370 0.1797044 0.1857707
## [24,] -0.0061451583 -0.1919159 0.1796255 0.1857707
##
## $gnp
##              fcst       lower      upper         CI
##  [1,] 0.007324620 -0.01221957 0.02686881 0.01954419
##  [2,] 0.005197791 -0.01563715 0.02603274 0.02083495
##  [3,] 0.005623353 -0.01640247 0.02764918 0.02202582
##  [4,] 0.005595316 -0.01656419 0.02775482 0.02215950
##  [5,] 0.005960759 -0.01624956 0.02817108 0.02221032
##  [6,] 0.006133992 -0.01610163 0.02836962 0.02223562
##  [7,] 0.006234897 -0.01601920 0.02848900 0.02225410
##  [8,] 0.006237399 -0.01602415 0.02849895 0.02226155
##  [9,] 0.006207242 -0.01605613 0.02847061 0.02226337
## [10,] 0.006159161 -0.01610460 0.02842292 0.02226376
## [11,] 0.006118246 -0.01614569 0.02838218 0.02226394
## [12,] 0.006084034 -0.01618004 0.02834810 0.02226407
## [13,] 0.006058129 -0.01620601 0.02832226 0.02226413
## [14,] 0.006036080 -0.01622807 0.02830023 0.02226415
## [15,] 0.006015890 -0.01624827 0.02828005 0.02226416
## [16,] 0.005995602 -0.01626856 0.02825976 0.02226416
## [17,] 0.005974765 -0.01628939 0.02823892 0.02226416
## [18,] 0.005953263 -0.01631090 0.02821742 0.02226416
## [19,] 0.005931354 -0.01633281 0.02819551 0.02226416
## [20,] 0.005909247 -0.01635491 0.02817341 0.02226416
## [21,] 0.005887108 -0.01637705 0.02815127 0.02226416
## [22,] 0.005865005 -0.01639916 0.02812917 0.02226416
## [23,] 0.005842955 -0.01642121 0.02810712 0.02226416
## [24,] 0.005820944 -0.01644322 0.02808510 0.02226416
##
## $consum
##              fcst        lower      upper         CI
##  [1,] 0.008880139 -0.006493587 0.02425387 0.01537373
##  [2,] 0.006278021 -0.009267168 0.02182321 0.01554519
##  [3,] 0.007718926 -0.008449971 0.02388782 0.01616890
##  [4,] 0.007270486 -0.008922060 0.02346303 0.01619255
##  [5,] 0.007783722 -0.008445421 0.02401286 0.01622914
##  [6,] 0.007711128 -0.008539683 0.02396194 0.01625081
##  [7,] 0.007817519 -0.008439414 0.02407445 0.01625693
##  [8,] 0.007762917 -0.008496876 0.02402271 0.01625979
##  [9,] 0.007758547 -0.008501502 0.02401860 0.01626005
## [10,] 0.007723393 -0.008536843 0.02398363 0.01626024
## [11,] 0.007711299 -0.008549064 0.02397166 0.01626036
## [12,] 0.007696615 -0.008563804 0.02395703 0.01626042
## [13,] 0.007690098 -0.008570344 0.02395054 0.01626044
## [14,] 0.007682881 -0.008577564 0.02394333 0.01626045
## [15,] 0.007676986 -0.008583461 0.02393743 0.01626045
## [16,] 0.007670075 -0.008590373 0.02393052 0.01626045
## [17,] 0.007662958 -0.008597490 0.02392341 0.01626045
## [18,] 0.007655330 -0.008605118 0.02391578 0.01626045
## [19,] 0.007647606 -0.008612842 0.02390805 0.01626045
## [20,] 0.007639799 -0.008620650 0.02390025 0.01626045
## [21,] 0.007632035 -0.008628414 0.02389248 0.01626045
## [22,] 0.007624301 -0.008636148 0.02388475 0.01626045
## [23,] 0.007616605 -0.008643844 0.02387705 0.01626045
## [24,] 0.007608924 -0.008651524 0.02386937 0.01626045
```

Predictions from a VAR(2) fit to the unemployment - consumption data (Graph)

```
fanchart(fit.pr) # plot prediction error
```