Time Series Regression Code

Jude Hays

2/9/2021

```r
library(forecast)
library(fpp2)           # For Accuracy Function
library(ARDL)           # Just Need Denmark Data
library(orcutt)
library(sandwich)
library(xts)
library(magrittr)
library(DataCombine)    #For Slide Function
library(multDM)
```

Time Series Regression

- Estimate some static models. Specifically, estimate a static model with Newey-West standard errors and a model with serially correlated disturbances using FGLS. What do these models suggest about the relationships between your independent variables and dependent variable?

```r
## Ice Cream Data

##Static Model with Newey-West Robust Standard Errors
data(icecream, package = "orcutt")
m <- lm(cons ~ price + income + temp, data = icecream)
summary(m)
```

```
##
## Call:
## lm(formula = cons ~ price + income + temp, data = icecream)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.065302 -0.011873  0.002737  0.015953  0.078986
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.1973151  0.2702162   0.730  0.47179
## price       -1.0444140  0.8343573  -1.252  0.22180
## income       0.0033078  0.0011714   2.824  0.00899 **
## temp         0.0034584  0.0004455   7.762  3.1e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03683 on 26 degrees of freedom
## Multiple R-squared:  0.719,  Adjusted R-squared:  0.6866
## F-statistic: 22.17 on 3 and 26 DF,  p-value: 2.451e-07
```

```r
NW <- NeweyWest(m, prewhite = FALSE)
NW
```

```
##               (Intercept)         price        income          temp
## (Intercept)  0.0761594215 -2.109958e-01 -1.945155e-04 -1.027620e-05
## price       -0.2109958275  6.825935e-01  2.696246e-04 -6.140549e-05
## income      -0.0001945155  2.696246e-04  1.261484e-06  2.257265e-07
## temp        -0.0000102762 -6.140549e-05  2.257265e-07  1.871315e-07
```

```r
##Cochrane-Orcutt Method for FGLS
lm = lm(cons ~ price + income + temp, data=icecream)
coch = cochrane.orcutt(lm)
summary(coch)
```

```
## Call:
## lm(formula = cons ~ price + income + temp, data = icecream)
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  0.15714771  0.28962931   0.543   0.59222
## price       -0.89239565  0.81085041  -1.101   0.28157
## income       0.00320274  0.00154606   2.072   0.04878 *
## temp         0.00355839  0.00055468   6.415 1.022e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0319 on 25 degrees of freedom
## Multiple R-squared:  0.6489 ,  Adjusted R-squared:  0.6068
## F-statistic: 15.4 on 3 and 25 DF,  p-value: < 7.015e-06
##
## Durbin-Watson statistic
## (original):    1.02117 , p-value: 3.024e-04
## (transformed): 1.54884 , p-value: 5.061e-02
```

- Next, estimate several dynamic time series regressions–the autoregressive distributed lag (ADL), finite distributed lag (FDL), and partial adjustment (PA) models. Use diagnostics to justify your specification choices. What are the differences implied by these dynamic models? Which model fits the data best? Which model makes the most sense theoretically?

```r
data(denmark)
denmark<-as.data.frame(denmark)

# TS Regression Forecasting Work Around, Using Denmark Data from ARDL package

LRM <- ts(denmark$LRM, start=1974.75, frequency=4)
LRY <- ts(denmark$LPY, start=1974.75, frequency=4)
IBO <- ts(denmark$IBO, start=1974.75, frequency=4)
IDE <- ts(denmark$IDE, start=1974.75, frequency=4)

# Manually Calculate Lags of y and X

LRM_LAG <- cbind(
  LRMLag0 = stats::lag(LRM,0),
  LRMLag1 = stats::lag(LRM, -1),
  LRMLag2 = stats::lag(LRM, -2),
  LRMLag3 = stats::lag(LRM, -3))
LRM_LAG <- LRM_LAG[,-1]
```

```r
OTH_LAG <- cbind(
  LRYLag0 = stats::lag(LRY,0),
  LRYLag1 = stats::lag(LRY,-1),
  IBOLag0 = stats::lag(IBO,0),
  IBOLag1 = stats::lag(IBO,-1),
  IBOLag2 = stats::lag(IBO,-2),
  IBOLag3 = stats::lag(IBO,-3),
  IDELag0 = stats::lag(IDE,0),
  IDELag1 = stats::lag(IDE,-1),
  IDELag2 = stats::lag(IDE,-2),
  IDELag3 = stats::lag(IDE,-3)
)

#Define Training and Testing Datasets for All Variables

LRM_tr <-LRM[4:40]
LRM_tst <-LRM[38:55]

LRM_LAG_tr <-LRM_LAG[4:40,1:3]
LRM_LAG_tst <- LRM_LAG[38:55,1:3]

OTH_LAG_tr <-OTH_LAG[4:40,1:3]
OTH_LAG_tst <- OTH_LAG[38:55,1:3]

#Do One-Step Ahead Forecasts for two or more models. First: (3,2,2,2)

fit<-Arima(LRM_tr, xreg=cbind(LRM_LAG_tr,OTH_LAG_tr),order=c(0,0,0))
fit2<-Arima(LRM_tst, xreg=cbind(LRM_LAG_tst,OTH_LAG_tst), model=fit)

onestep <- fitted(fit2)
e <- LRM_tst - onestep
rmsqe <- mean(sqrt(e^2))

#One-Step Ahead Forecasts for two or more models. Second: (3,0,0,0)
fit_ar3<-Arima(LRM_tr, xreg=LRM_LAG_tr ,order=c(0,0,0))
fit2_ar3<-Arima(LRM_tst, xreg=LRM_LAG_tst, model=fit_ar3)

onestep_ar3 <- fitted(fit2_ar3)
e_ar3 <- LRM_tst - onestep_ar3
rmsqe_ar3 <- mean(sqrt(e_ar3^2))

#One-Step Ahead Forecasts for two or more models. Second: (1,0,0,0)
AR1<-LRM_LAG[,-c(2,3)]
AR1_tr<-AR1[2:38]
AR1_tst<-AR1[38:55]

fit_ar1<-Arima(LRM_tr, xreg=AR1_tr ,order=c(0,0,0))
fit2_ar1<-Arima(LRM_tst, xreg=AR1_tst, model=fit_ar1)

onestep_ar1 <- fitted(fit2_ar1)
e_ar1 <- LRM_tst - onestep_ar1
rmsqe_ar1 <- mean(sqrt(e_ar1^2))
```

```r
#DM Tests
test1 <-DM.test(e, e_ar3, LRM_tst)
test2 <-DM.test(e, e_ar1, LRM_tst)
test3 <-DM.test(e_ar3, e_ar1, LRM_tst)

#In-Sample Fit Diagnostics
compare <- data.frame(AIC = c(AIC(fit), AIC(fit_ar3), AIC(fit_ar1)),
BIC = c(BIC(fit), BIC(fit_ar3), BIC(fit_ar1)))
compare
```

```
##         AIC       BIC
## 1 -172.2860 -159.3986
## 2 -149.8155 -141.7609
## 3 -104.2020  -99.3692
```

```r
checkresiduals(fit)
```

Assignment2_Rcode_files/figure-latex/unnamed-chunk-4-1.pdf

```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0) errors
## Q* = 28.733, df = 3, p-value = 2.548e-06
##
## Model df: 7.   Total lags used: 10
```

```r
checkresiduals(fit_ar3)
```

Assignment2_Rcode_files/figure-latex/unnamed-chunk-4-2.pdf

```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0) errors
## Q* = 6.1309, df = 3, p-value = 0.1054
##
## Model df: 4.   Total lags used: 7
```

```r
checkresiduals(fit_ar1)
```

Assignment2_Rcode_files/figure-latex/unnamed-chunk-4-3.pdf

```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0) errors
## Q* = 27.501, df = 5, p-value = 4.555e-05
##
## Model df: 2.   Total lags used: 7
```

```r
Box.test(fit$residuals, lag=20, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  fit$residuals
## X-squared = 48.33, df = 20, p-value = 0.0003823
```

```r
Box.test(fit_ar3$residuals, lag=20, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  fit_ar3$residuals
## X-squared = 18.736, df = 20, p-value = 0.5391
```

```r
Box.test(fit_ar1$residuals, lag=20, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  fit_ar1$residuals
## X-squared = 47.835, df = 20, p-value = 0.0004487
```