Test Exercise 3

Coursera/Erasmus U., Econometric Methods and Applications

Anthony Nguyen

Training Exercise 3.5

Notes:

- See website for how to submit your answers and how feedback is organized.
- This exercise uses the datafile `TestExer3` and requires a computer.

Goals and skills being used:

- Experience the process of model selection.
- Apply methods to compare models.
- Apply tests to evaluate a model.

Questions

This test exercise is of an applied nature and uses data that are available in the data file `TestExer3`. We consider the so-called Taylor rule for setting the (nominal) interest rate. This model describes the level of the nominal interest rate that the central bank sets as a function of equilibrium real interest rate and inflation, and considers the current level of inflation and production. Tayor (1993)[1] considers the model:

$$i_t = r^* + \pi_t + 0.5(\pi_t - \pi^*) + 0.5g_t,$$

with i_t the Federal funds target interest rate at time t, r^* the equilibrium real federal funds rate, π_t a measure of inflation, π^* the target inflation rate and g_t the output gap (how much actual output deviates from potential output). We simplify the Taylor rule in tow manners. First we avoid determining r^* and π^* and simply add an intercept to the model to capture these two variables (and any other deviation in the means). Second, we consider production y_t rather than the output gap. In this form the Taylor rule is:

$$i_t = \beta_1 + \beta_2 \pi_t + \beta_3 y_t + \varepsilon_t. \tag{1}$$

Monthly data are available for the USA over the period 1960 through 2014 for the following variables[2]:

- INTRATE: Federal funds interest rate
- INFL: Inflation
- PROD: Production
- UNEMPL: Unemployment
- COMMPRI: Commodity prices

[1] "Discretion Versus Policy Rules in Practice", Carnegie-Rochester Conference Series on Public Policy 39, pages 1455-1508.

[2] The data are from the St. Louis Federal Reserve Economic Dataset (FRED), with IDs FEDFUNDS, CPIAUCSL, INDPRO, PAYEMS, NAPMPRI, PCE, A229RX0 and HOUST respectively (all percent change from a year ago, except for the Federal funds rate).

- PCE: Personal consumption expenditure
- PERSINC: Personal income
- HOUST: Housing starts

(a) Use general-to-specific to come to a model. Start by regressing the federal funds rate on the other 7 variables and eliminate 1 variable at a time.

(b) Use specific-to-general to come to a model. Start by regressing the federal funds rate on only a constant and add 1 variable at a time. Is the model the same as in (a)?

(c) Compare your model from (a) and the Taylor rule of equation (1). Consider R^2, AIC and BIC. Which of the models do you prefer?

(d) Test the Taylor rule of equation (1) using the RESET test, Chow break and forecast test (with January 1980 as the break date in both tests) and a Jarque-Bera test. What do you conclude?

Answers

(a) Use general-to-specific to come to a model. Start by regressing the federal funds rate on the other 7 variables and eliminate 1 variable at a time.

```r
#regress all variables, remove lowest absolute t-value (or highest p-value) each round until all variab
mod1 <- lm(INTRATE~INFL+PROD+UNEMPL+COMMPRI+PCE+PERSINC+HOUST, data = TestExer3)
mod2 <- lm(INTRATE~INFL+PROD+COMMPRI+PCE+PERSINC+HOUST, data = TestExer3)
mod3 <- lm(INTRATE~INFL+COMMPRI+PCE+PERSINC+HOUST, data = TestExer3)

#print results from general-to-specific regressions
stargazer(mod1, mod2, mod3, title="General-to-specific model specification",  header = FALSE)
```

Table 1: General-to-specific model specification

	Dependent variable:		
	INTRATE		
	(1)	(2)	(3)
INFL	0.696***	0.693***	0.718***
	(0.062)	(0.062)	(0.057)
PROD	−0.058	−0.025	
	(0.040)	(0.026)	
UNEMPL	0.102		
	(0.097)		
COMMPRI	−0.006*	−0.007**	−0.008***
	(0.003)	(0.003)	(0.003)
PCE	0.344***	0.369***	0.341***
	(0.069)	(0.066)	(0.059)
PERSINC	0.247***	0.252***	0.240***
	(0.061)	(0.060)	(0.059)
HOUST	−0.019***	−0.021***	−0.021***
	(0.005)	(0.004)	(0.004)
Constant	−0.221	−0.291	−0.240
	(0.245)	(0.236)	(0.230)
Observations	660	660	660
R^2	0.639	0.638	0.637
Adjusted R^2	0.635	0.635	0.635
Residual Std. Error	2.188 (df = 652)	2.188 (df = 653)	2.188 (df = 654)
F Statistic	164.532*** (df = 7; 652)	191.731*** (df = 6; 653)	229.889*** (df = 5; 654)
Note:			*p<0.1; **p<0.05; ***p<0.01

(b) Use specific-to-general to come to a model. Start by regressing the federal funds rate on only a constant and add 1 variable at a time. Is the model the same as in (a)?

Here we follow a similar pattern to the specific-to-general method but in reverse, where we regress each variable one at a time, then add the one with the highest absolute t-value (lowest p-value) to our model at the end, and then repeat for as many rounds needed until no significant variables can be added anymore:

```
#Round 1: regress INTRATE by all columns apart from first two
R1 <- lapply(TestExer3[,-c(1,2)], function(x) summary(lm(TestExer3$INTRATE ~ x)))

#display coefficients
lapply(R1, coef)
```

```
## $INFL
##              Estimate Std. Error  t value      Pr(>|t|)
## (Intercept) 1.6420865 0.15863009 10.35167  2.280046e-23
## x           0.9453384 0.03268296 28.92451 2.472635e-119
##
## $PROD
##               Estimate Std. Error    t value      Pr(>|t|)
## (Intercept)  5.39418540 0.16550553 32.5921763 1.921477e-139
## x           -0.01591652 0.02965744 -0.5366789  5.916708e-01
##
## $UNEMPL
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 4.5446184 0.18496025 24.570785 4.276745e-95
## x           0.4524717 0.07018149  6.447165 2.205673e-10
##
## $COMMPRI
##               Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept)  5.40174284 0.141559343 38.158858 6.360031e-169
## x           -0.01152555 0.004191266 -2.749897  6.125172e-03
##
## $PCE
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -0.3361253  0.2816116 -1.193578 2.330734e-01
## x            0.8293834  0.0379894 21.831971 6.741202e-80
##
## $PERSINC
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 5.1245362 0.20843591 24.585668 3.533062e-95
## x           0.1042915 0.07186074  1.451299 1.471731e-01
##
## $HOUST
##               Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept)  5.40363985 0.138710232 38.956318 5.055331e-173
## x           -0.03095007 0.006061692 -5.105846  4.317089e-07
```

Here we can see that the absolute t-value for $INFL$ is the largest among the seven regressors, so we add that one to our model and re-run the remaining 6 regressors individually in a second round and, again, look for the largest t-value:

```r
#Round 1: regress INTRATE~INFL + all columns apart from first three
R2 <- lapply(TestExer3[,-c(1,2,3)], function(x) summary(lm(TestExer3$INTRATE~TestExer3$INFL+ x)))

#display coefficients
lapply(R2, coef)
```

```
## $PROD
##                  Estimate Std. Error   t value      Pr(>|t|)
## (Intercept)    1.24889991 0.17618878  7.088419  3.510578e-12
## TestExer3$INFL 0.97497557 0.03273424 29.784580 5.026241e-124
## x              0.09471968 0.01971315  4.804897  1.920033e-06
##
## $UNEMPL
##                 Estimate Std. Error   t value      Pr(>|t|)
## (Intercept)    1.1229740 0.16879067  6.653057  6.052444e-11
## TestExer3$INFL 0.9257301 0.03159436 29.300485 2.353745e-121
## x              0.3358096 0.04641504  7.234930  1.302459e-12
##
## $COMMPRI
##                    Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept)     1.682028928 0.160251380 10.496190  6.221424e-24
## TestExer3$INFL  0.940701793 0.032760106 28.714858 4.076617e-118
## x              -0.004636848 0.002803477 -1.653963  9.861265e-02
##
## $PCE
##                 Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)    0.1012467 0.23416526  0.432373 6.656122e-01
## TestExer3$INFL 0.7157541 0.04094048 17.482795 1.745166e-56
## x              0.3561606 0.04146252  8.589943 6.311519e-17
##
## $PERSINC
##                 Estimate Std. Error   t value      Pr(>|t|)
## (Intercept)    0.4471996 0.19509908  2.292167  2.221136e-02
## TestExer3$INFL 1.0122447 0.03147935 32.155834 5.304375e-137
## x              0.4359701 0.04599775  9.478074  4.616637e-20
##
## $HOUST
##                    Estimate  Std. Error    t value      Pr(>|t|)
## (Intercept)     1.676665912 0.163124888 10.2784188  4.419373e-23
## TestExer3$INFL  0.938289824 0.033589770 27.9337973 8.735175e-114
## x              -0.003840948 0.004214667 -0.9113289  3.624564e-01
```

After the second round, we can see that $PERSINC$ has the highest t-value, so we can add this to our model and re-run the process.

(Note: from here on out, I will stop displaying the regression results to save space)

After our third round, we find that PCE has the highest t-value, so we add it to our model.

After round four, we find that $HOUST$ has the highest t-value, so we add it to the model.

After round give, we find that $COMMPRI$ has the highest t-value, so we add it to the model.

After round six, the only remaining variables, $PROD$ and $UNEMPL$ have low t-values that are not significant, so we stop here. Looking at part (a), we can see that, in the end, we arrived at the same model using this method.

(c) Compare your model from (a) and the Taylor rule of equation (1). Consider R^2, AIC and BIC. Which of the models do you prefer?

To quickly recap, using the variable names from our dataset, the two models under consideration are:

$$INTRATE = \beta_1 + \beta_2 INFL + \beta_3 COMMPRI + \beta_4 PCE + \beta_5 PERSINC + \beta_6 HOUST + \varepsilon_t. \qquad \text{General-to-specific}$$
$$INTRATE = \beta_1 + \beta_2 INFL + \beta_3 PROD + \varepsilon_t \qquad \text{Taylor rule}$$

After running the regressions for the two models, we can compare the relevant statistics:

```
#run Taylor-model regression
mod_Taylor <- lm(INTRATE~INFL+PROD, data = TestExer3)

#AIC and BIC are not calculated automatically with the lm() function in R
#add AIC and BIC values to mod3
mod3$AIC <- AIC(mod3)
mod3$BIC <- BIC(mod3)

#add AIC and BIC values to mod_Taylor
mod_Taylor$AIC <- AIC(mod_Taylor)
mod_Taylor$BIC <- BIC(mod_Taylor)

#display and compare summary of both models
stargazer(mod3, mod_Taylor, title="Model (a) vs. Taylor model", header = FALSE, float = FALSE)
```

	Dependent variable:	
	INTRATE	
	(1)	(2)
INFL	0.718***	0.975***
	(0.057)	(0.033)
COMMPRI	−0.008***	
	(0.003)	
PCE	0.341***	
	(0.059)	
PERSINC	0.240***	
	(0.059)	
HOUST	−0.021***	
	(0.004)	
PROD		0.095***
		(0.020)
Constant	−0.240	1.249***
	(0.230)	(0.176)
Observations	660	660
R^2	0.637	0.575
Adjusted R^2	0.635	0.573
Akaike Inf. Crit.	2,914.423	3,013.616
Bayesian Inf. Crit.	2,945.869	3,031.585
Residual Std. Error	2.188 (df = 654)	2.364 (df = 657)
F Statistic	229.889*** (df = 5; 654)	443.899*** (df = 2; 657)
Note:		*p<0.1; **p<0.05; ***p<0.01

Looking at the two models, we can see that the AIC and the BIC are lower for the model we derived in part (a). Additionally, the model from (a) has a higher R^2 valued, so for all three statistics, model (a) is preferred over the Taylor rule model.

(d) Test the Taylor rule of equation (1) using the RESET test, Chow break and forecast test (with January 1980 as the break date in both tests) and a Jarque-Bera test. What do you conclude?

We can calculate the RESET test statistic as follows:

```
# using `resettest()` from `lmtest` package
resettest(mod_Taylor, power = 2, type = "fitted", data = TestExer3)
```

```
##
##  RESET test
##
## data:  mod_Taylor
## RESET = 2.5371, df1 = 1, df2 = 656, p-value = 0.1117
```

We can calculate the Chow break statistic as follows:

```r
# Save SSR for complete model
SSR_C <- sum(residuals(mod_Taylor)^2)

# Split OBS column in TestExer3 into Year and Month

TestExer3 <- TestExer3 %>% mutate(Year = str_split_fixed(TestExer3$OBS, ":", n = 2)[, 1])
TestExer3$Year <- as.numeric(TestExer3$Year)

# Split TestExer3 into two subsets using Jan.1980 as the break date

TestExer3_1 <- TestExer3 %>% filter(OBS < 1980)
TestExer3_2 <- TestExer3 %>% filter(OBS >= 1980)

# Run same regression on subsets
mod_Taylor_1 <- lm(INTRATE~INFL+PROD, data = TestExer3_1)
mod_Taylor_2 <- lm(INTRATE~INFL+PROD, data = TestExer3_2)

# Save SSRs for subsets
SSR_1 <- sum(residuals(mod_Taylor_1)^2)
SSR_2 <- sum(residuals(mod_Taylor_2)^2)

# Calculate F_Chow statistic with:
# \frac{(S_C - (S_1 + S_2))/k}{(S_1 + S_2)/(N_1 +N_2 - 2k)

F_Chow <- ((SSR_C - (SSR_1 + SSR_2))/2) / ((SSR_1 + SSR_2)/(240 + 420 - 2*3))

# Calculate p-value
p_Chow <- pf(F_Chow, 3, 660, lower.tail = F)

# Print results
paste("F_Chow =", round(F_Chow,3))
```

```
## [1] "F_Chow = 43.103"
```

```r
paste("p_Chow =", round(p_Chow,3))
```

```
## [1] "p_Chow = 0"
```

We can calculate the Chow forecast statistic as follows:

```r
# Formula is \frac{(SSR_C - SSR_1)/q}{SSR_1/(N_1 - k)}
F_ChowForecast <- ((SSR_C - SSR_1)/420)/(SSR_1/(240 - 3))
p_ChowForecast <- pf(F_ChowForecast, 420, 237, lower.tail = F)

# Print results
paste("F_ChowForecast =", round(F_ChowForecast,3))
```

```
## [1] "F_ChowForecast = 5.511"
```

```r
paste("p_ChowForecast =", round(p_ChowForecast, 3))
```

```
## [1] "p_ChowForecast = 0"
```

Finally, we can calculate Jarque-Bera statistic as follows:

```r
# using `jarque.bera.test()` from `Tseries` package
jarque.bera.test(residuals(mod_Taylor))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(mod_Taylor)
## X-squared = 12.444, df = 2, p-value = 0.001985
```

Looking over our results from the four tests, we can see that we do not reject the null hypothesis for the RESET test, but we do reject the null for the Chow break, Chow forecast and Jarque-Bera (JB) tests. This implies that our Taylor rule model is *not* a polynomial, and that it *has* a structural break and is *not* normally distributed.