Midterm Take-Home Component

Economic Forecasting - Summer 2017

25 / 35 points

By signing on the line below you acknowledge that your work is your own and that you neither gave help nor did you seek or receive help from fellow students or professors. If anyone is caught cheating, they will receive a failing grade on this component of the exam. You may use any resource besides the aforementioned. Failure to sign will result in a failing grade on this component of the exam.

Signature: ____

Print Name: ____

This sheet needs to be turned in on Thursday, June 29th at the beginning of class.

Instructions: In this component of the exam you are going to analyze some data by checking for serial correlation and then attempt to remove it by using a two-step Cochrane-Orcutt (C-O) methodology. You will then perform another test for serial correlation to see if the C-O methodology was successful in removing the correlation. Your final submission, via the dropbox on Springboard in the Midterm folder, will be a .txt file named: midterm_[yourlastname].txt For example, my submission would be midterm_hoff.txt This portion of the Midterm is due by **5:00 PM on Thursday, June 25th**. Answer all questions that are printed in **bold** to the best of your ability. I will not answer questions about this part of the exam other than if you cannot find the beginning data set. **IMPORTANT:** Do not use a package to perform either the C-O method or an LM test. That is, you are to code both of these procedures yourself.

Useful R Codes

You may want to apply an `acf()` function to an object that has "NA"s as one or more of its elements. To do this, all you need to do is add the `na.action=na.pass` option to the `acf()` function. For example, if the object x has missing values, the code would be `acf(x, na.action=na.pass)`.

There are two major ways to approach the following scenario

1. not using a `data.frame()` object, and
2. using a `data.frame()` object

I would strongly recommend using method 2 (i.e. `data.frame()` objects). However, I will provide helpful R code for both methods.

Finally, remember that the Internet and Google searches are your friends.

Not using a `data.frame()` object:

Dropping Variables

In this assignment you may nd it necessary to drop observations from a vector. One way to do this is by the following method. Suppose you have a vector `z` that has 10 elements - the integers 11 through 20.

If you want to drop the first element, i.e. the number eleven, and retain the vector `z`, you would write:

- `z <- z[-1]`

`z` would then contain 9 elements, the numbers 12 through 20. If instead you wanted to drop the last element in `z`, i.e. the number 20, you would write:

- `z <- z[-10]`

Now `z` would contain 9 elements, the numbers 11 through 19. So, in general, to rewrite a vector `z` while dropping the ith variable, you would write

- `z <- z[-i]`

Checking Lags and Viewing Data

A good habit to get into when doing any kind of data analysis is to check the data directly. This is especially useful for this course since you would like to make sure you're lags are done correctly. Two functions that are extremely useful for this are the `View()` function, which you used in Problem Set 1 and the `cbind()` function. `cbind()` puts multiple vectors into a single object. For example, suppose that you have a vector `a`, which has 100 elements, and you would like to lag `a` so that you can create a new object `b` which is the difference of `a` and the lag of `a`. The following code would be appropriate:

- `a <- ts(a)`
- `a.lag <- lag(a, k=-1)`
- `a.vec <- cbind(a, a.lag)`
- `View(a.vec)`

Then once you see that R is doing the correct operation you can write, with confidence,

- `b <- a - a.lag`

Now let's suppose that you want to regress `a` on `a.lag` and another variable `x` and you try the following code:

- `lm(a ~ a.lag + x)`

You would see that this model fits perfectly (*i.e.* `a.lag` predicts `a` perfectly) and is incorrect. To get this model to do what you actually want, you should first turn `x` into a time series, drop the first observations of `a` and `x` and then drop the last observation of `a.lag`. To do this, use the following:

- `a <- a[-1]`
- `a.lag <- a.lag[-100]`
- `x <- x[-1]`

You can now safely regress `a` on `a.lag` and `x` and not get a perfect fit. Notice that you will lose a degree of freedom, but this is clearly a small price to pay. One final and important note is to *only drop the first observation of `a` after you have lagged it.*

Using a `data.frame()` object:

Alternatively, you can leave the imported data as a `data.frame()` object and use the `shift()` function found on Springboard.

Adding a vector of shorter lengths to a `data.frame()` object

During this portion of the exam you may want to add an object of a shorter length to a data frame object. To do this, you will need to tell R exactly where to put the new vector in the data frame. For example suppose that you have an object `w` that has 100 observations and is part of a data frame named `df`. To add a new vector/object which has 99 observations to the last 99 rows of df, let's call this object `v`, we would use the following code:

- `df$v[2:100] <- v`

This tells R to start with row 2 and add the 99 elements of `v` to the data frame. Thus, the first element of `df$v` will be a "`NA`" - which is what you want.

Getting the Correct Data

Import the data from Springboard labeled "midterm_take_home". To check that you have imported the data correctly, you should have two variables x and y that have the following attributes:

Variable	N	Mean	Standard Deviation
y	200	0.59	2.50
x	200	0.03	1.01

Running the Initial Regression

$$y_t = \alpha + \beta x_t + \epsilon_t \tag{1}$$

- Estimate Equation (1) by OLS
- **Interpret α and β. Are these estimates statistically significant at conventional levels?**

Testing for Serial Correlation

- **Use graphical evidence (i.e. various plots) to make an initial guess if serial correlation is a problem for Equation (1).**
- Use the Breusch-Godfrey LM to determine whether or not serial correlation is a problem. DO NOT USE A FUNCTION FROM A PACKAGE YOU DOWNLOADED TO DO THIS. CODE THIS YOURSELF.
  - This is where you'll need to drop observations if you are not using data frames: you'll want to drop the first observations of the residual and x, and you'll want to drop the last observations of the lagged residual and the lagged x. See the "Useful R Codes" section above.
- **Report the LM test statistic and decide whether or not you reject the null hypothesis. Make sure to state what the null and alternative hypotheses are.**
- Although this will not be graded, you may want to also calculate the Durbin-Watson test statistic.

Cochrane-Orchutt

- Use a one-step Cochrane-Orcutt (C-O) procedure in an attempt to remove the autocorrelation from the data. DO NOT USE A FUNCTION FROM A PACKAGE YOU DOWNLOADED TO DO THIS. CODE THIS YOURSELF.
- **Interpret your new estimates of α and β. Compare these to the results from the "Running the Initial Regression" section.**

Re-Testing for Serial Correlation

- **Again, use graphical evidence to make an initial guess about the success of the one-step Cochrane-Orcutt (C-O) method in removing autocorrelation.**
- Calculate the Breusch-Godfrey LM test statistic to see if the C-O method removed the serial correlation.
  - If not using the data frames method: Again, you'll want to drop the first observations of the residuals and x, but now you'll need to drop the last two observations from the lagged residuals - this can be done by dropping the last, saving the object, and then drop the new last observation.
  - If using the data frames method: this is where you'll need to tell R where to put the object in the data frame.
  - Make sure to use the transformed version of x in the LM test.
- **Report the LM test statistic and decide whether or not to reject the null hypothesis.**
- **Was the C-O one-step effective in removing autocorrelation?**
- Again, it may be helpful to calculate the Durbin-Watson test statistic

Putting it All Together

- **Suppose that the truth is $\beta = 1.25$. Does either the initial regression setup or the one-step C-O estimate of β allow for the true β to fall in the 95% confidence interval of the estimated β. Explain.**
- **Suppose that you decide to use an iterative version of the C-O method to improve your estimate of β. How would this x affect the LM test(s) going forward?**

GRAD STUDENTS ONLY - 10 Points

Another Iteration of the C-O Method

- Perform an additional iteration of the C-O method.
- **Compared to your previous estimation of β, does your new estimate of β increase or decrease in magnitude. Compare this result with the previous estimate of β with respect to the knowledge that the true $\beta = 1.25$.**

Another Serial Correlation Test

- Perform another LM test
- **What is the value of the new test statistic and is autocorrelation removed? Compare this value of the test statistic to the previous value of the test statistic. Does this change make sense?**
- **Intuitively explain (i.e. you don't need to actually write the function) how would write a looping function to find the "best" estimate of ρ within a C-O methodology.**