In-Class Lab 10

ECON 4223 (Prof. Tyler Ransom, U of Oklahoma)

March 1, 2022

The purpose of this in-class lab is to use R to practice estimating time series regression models with standard errors corrected for heteroskedasticity and serial correlation (HAC). To get credit, upload your .R script to the appropriate place on Canvas.

For starters

First, install the `pdfetch`, `tsibble`, `sandwich`, and `COVID19` packages. `pdfetch` stands for "Public Data Fetch" and is a slick way of downloading statistics on stock prices, GDP, inflation, unemployment, etc. `tsibble` is a package useful for working with time series data. It is the "tibble" for time series data. `sandwich` is helpful for obtaining HAC standard errors. `COVID19` pulls up-to-date data on COVID-19 cases, deaths, etc.

Open up a new R script (named `ICL10_XYZ.R`, where `XYZ` are your initials) and add the usual "preamble" to the top:

```
# Add names of group members HERE
library(tidyverse)
library(wooldridge)
library(modelsummary)
library(broom)
library(car)
library(magrittr)
library(lmtest)
# new packages installed today:
library(pdfetch)
library(sandwich)
library(tsibble)
library(COVID19)
```

Load the data

We're going to use data on US COVID-19 cases, deaths and other information.

```
df <- covid19(c("US"))
df.ts <- as_tsibble(df, key=id, index=date)
```

Now it will be easy to include lags of various variables into our regression models.

Plot time series data

Let's have a look at the data on **new** daily cases and deaths:

```
df.ts %<>% mutate(new_deaths = difference(deaths),    # data comes in cumulative format
                  new_tests  = difference(tests),     # "difference" converts it to
                  new_cases  = difference(confirmed)) # "new cases" etc.
```

```
# plots
ggplot(df.ts, aes(date, new_cases)) + geom_line()
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```

```
ggplot(df.ts, aes(date, new_deaths)) + geom_line()
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```

```
# plots with 7-day rolling average
ggplot(df.ts, aes(date, new_cases)) + geom_line(aes(y=rollmean(new_cases, 7, na.pad=TRUE)))
```

```
## Warning: Removed 7 row(s) containing missing values (geom_path).
```

```
ggplot(df.ts, aes(date, new_deaths)) + geom_line(aes(y=rollmean(new_deaths, 7, na.pad=TRUE)))
```

```
## Warning: Removed 7 row(s) containing missing values (geom_path).
```

Determinants of US COVID-19 cases

Now let's estimate the following regression model:

$$\log(new\_cases_t) = \beta_0 + \beta_1 gath_t + \beta_2 gath_{t-7} + \beta_3 gath_{t-14} + \beta_4 \log(new\_cases_{t-7}) + u_t$$

where new_cases is the number of new COVID cases, and $gath$ is a variable taking on values 0–4 representing severity of gatherings restricitons.

```
df.ts %<>% mutate(log.new.cases = log(new_cases),
                  log.new.cases = replace(log.new.cases,new_cases==0,NA_real_)) # since log(0) = -Inf
```

```
## Warning in log(new_cases): NaNs produced
```

```
est <- lm(log.new.cases ~ gatherings_restrictions + lag(gatherings_restrictions,7) +
                          lag(gatherings_restrictions,14) + lag(log.new.cases,7), data=df.ts)
```

1. Are any of these variables significant determinants of new COVID cases? If so, which ones?

Correcting for Serial Correlation

Now let's compute HAC (Heteroskedasticity and Autocorrelation Consistent) standard errors. To do so, we'll use the `vcov` option in the `modelsummary()` function along with the `NeweyWest` function from the `sandwich` package.

```
modelsummary(est) # re-display baseline results
modelsummary(est, vcov=sandwich::NeweyWest(est))
```

or, putting them side-by-side:

```
modelsummary(list(est,est),
             vcov=list("iid",sandwich::NeweyWest(est))
            )
```

2. How does your interpretation of the the effect of gathering restrictions change after using the Newey-West standard errors?

Oklahoma COVID cases

```
df.ok <- covid19(c("US"),level=2) %>%
         filter(administrative_area_level_2=="Oklahoma")
```

```
df.ts.ok <- as_tsibble(df.ok, key=id, index=date)
df.ts.ok %<>% mutate(new_deaths = difference(deaths),
                     new_tests  = difference(tests),
                     new_cases  = difference(confirmed))
ggplot(df.ts.ok, aes(date, new_cases)) + geom_line(aes(y=rollmean(new_cases, 7, na.pad=TRUE)))
```

```
## Warning: Removed 12 row(s) containing missing values (geom_path).
```

```
ggplot(df.ts.ok, aes(date, new_deaths)) + geom_line(aes(y=rollmean(new_deaths, 7, na.pad=TRUE)))
```

```
## Warning: Removed 12 row(s) containing missing values (geom_path).
```

```
df.ts.ok %<>% mutate(log.new.cases = log(new_cases),
                     log.new.cases = replace(log.new.cases,new_cases==0,NA_real_)) # since log(0) = -In
est.ok <- lm(log.new.cases ~ gatherings_restrictions + lag(gatherings_restrictions,7) +
                           lag(gatherings_restrictions,14) + lag(log.new.cases,7), data=df.ts.ok)
```

With HAC standard errors:

```
modelsummary(list(est.ok,est.ok),
             vcov=list("iid",sandwich::NeweyWest(est.ok)))
```

Traditional macroeconomic model of interest rates and inflation

Load the data

We can also use data on US macroeconomic indicators. The `wooldridge` data set is called `intdef`.

```
df.ts <- as_tsibble(intdef, key=NULL, index=year)
```

Now it will be easy to include lags of various variables into our regression models.

Plot time series data

Let's have a look at the inflation rate for the US over the period 1948–2003:

```
ggplot(df.ts, aes(year, inf)) + geom_line()
```

Determinants of the interest rate

Now let's estimate the following regression model:

$$i3_t = \beta_0 + \beta_1 inf_t + \beta_2 inf_{t-1} + \beta_3 inf_{t-2} + \beta_4 def_t + u_t$$

where $i3$ is the 3-month Treasury Bill interest rate, inf is the inflation rate (as measured by the CPI), and def is the budget deficit as a percentage of GDP.

```
est <- lm(i3 ~ inf + lag(inf,1) + lag(inf,2) + def, data=df.ts)
modelsummary(list(est,est), vcov=list("iid",sandwich::NeweyWest(est)))
```