In-Class Lab 4

ECON 4223

September 5, 2023

The purpose of this in-class lab is to further practice your regression skills. To get credit, upload your .R script to the appropriate place on Canvas.

For starters

Open up a new R script (named `ICL4_XYZ.R`, where `XYZ` are your initials) and add the usual "preamble" to the top:

```
# Add names of group members HERE
library(tidyverse)
library(broom)
library(wooldridge)
library(modelsummary)
```

For this lab, let's use data on house prices. This is located in the `hprice1` data set in the `wooldridge` package. Each observation is a house.

```
df <- as_tibble(hprice1)
```

Check out what's in `df` by typing

```
glimpse(df)
```

Or for some summary statistics:

```
datasummary_skim(df,histogram=FALSE)
```

Multiple Regression

Let's estimate the following regression model:

$$price = \beta_0 + \beta_1 sqrft + \beta_2 bdrms + u$$

where *price* is the house price in thousands of dollars.

The code to do so is:

```
est1 <- lm(price ~ sqrft + bdrms, data=df)
modelsummary(est1)
```

You should get a coefficient of `0.128` on `sqrft` and `15.2` on `bdrms`. Interpret these coefficients. (You can type the interpretation as a comment in your .R script.) Do these numbers seem reasonable?

You should get $R^2 = 0.632$. Based on that number, do you think this is a good model of house prices?

Check that the average of the residuals is zero:

```
mean(est1$residuals)
```

Adding in non-linearities

The previous regression model had an estimated intercept of `-19.3`, meaning that a home with no bedrooms and no square footage would be expected to have a sale price of -$19,300.

To ensure that our model always predicts a positive sale price, let's instead use $log(price)$ as the dependent variable. Let's also add quadratic terms for `sqrft` and `bdrms` to allow those to exhibit diminishing marginal returns.

First, let's use `mutate()` to add these new variables:

```
df <- df %>% mutate(logprice = log(price), sqrftSq = sqrft^2, bdrmSq = bdrms^2)
```

Now run the new model:

```
est2 <- lm(logprice ~ sqrft + sqrftSq + bdrms + bdrmSq, data=df)
modelsummary(est2)
# or, for more decimals:
modelsummary(est2, fmt = 10)
```

The new coefficients have much smaller magnitudes. Explain why that might be.

The new $R^2 = 0.595$ which is less than 0.632 from before. Does that mean this model is worse?

Using the Frisch-Waugh Theorem to obtain partial effects

Let's experiment with the Frisch-Waugh Theorem, which says:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{N} \hat{r}_{i1} y_i}{\sum_{i=1}^{N} \hat{r}_{i1}^2}$$

where $\hat{r}_{i1}$ is the residual from a regression of x_1 on $x_2, \ldots, x_k$

Let's do this for the model we just ran. First, regress `sqrft` on the other X's and store the residuals as a new column in `df`.

```
est <- lm(sqrft ~ sqrftSq + bdrms + bdrmSq, data=df)
df <- df %>% mutate(sqrft.resid = est$residuals)
```

Now, if we run a simple regression of `logprice` on `sqrft.resid` we should get the same coefficient as that of `sqrft` in the original regression (=`3.74e-4`).

```r
est <- lm(logprice ~ sqrft.resid, data=df)
modelsummary(est)
```

Frisch-Waugh by hand

We can also compute the Frisch-Waugh formula by hand:

```r
beta1 <- sum(df$sqrft.resid*df$logprice)/sum(df$sqrft.resid^2)
print(beta1)
```

Which indeed gives us what we expected.