B4 Wage Help Documentation

Lukas Arnroth

4 november 2017

Instrumental Variables in R

Data

This help documentation will only treat the subject of two stage least squares (2SLS) in R. As for the first three tasks of this assignment you should have a working knowledge at this point. In this example I will use the dataset *CigarettesSw* from the AER (standing for Applied Econometrics with R) package. The sandwich and lmtest packages will later be used for robust standard errors.

```
library(AER)
library(sandwich)
library(lmtest)
data("CigarettesSW")
str(CigarettesSW)
```

```
## 'data.frame':    96 obs. of  9 variables:
##  $ state     : Factor w/ 48 levels "AL","AR","AZ",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ year      : Factor w/ 2 levels "1985","1995": 1 1 1 1 1 1 1 1 1 1 ...
##  $ cpi       : num  1.08 1.08 1.08 1.08 1.08 ...
##  $ population: num  3973000 2327000 3184000 26444000 3209000 ...
##  $ packs     : num  116 129 105 100 113 ...
##  $ income    : num  4.60e+07 2.62e+07 4.40e+07 4.47e+08 4.95e+07 ...
##  $ tax       : num  32.5 37 31 26 31 ...
##  $ price     : num  102.2 101.5 108.6 107.8 94.3 ...
##  $ taxs      : num  33.3 37 36.2 32.1 31 ...
```

For a brief description of this dataset you can consult the AER package documentation, ?*CigarettesSW*. Lets do some brief data management, starting with standardizing the price with respect to cpi. You can do this two ways using either the dollar sign operator or the with() function. I think using with() is quite neat and gives more readable and shorter code. But in the end, these things comes down to taste. Below you can compare the two methods.

```
CigarettesSW$rprice <- with(CigarettesSW, price/cpi)
CigarettesSW$rprice <- CigarettesSW$price / CigarettesSW$cpi
```

Using with() we create 3 more variables, income per capita adjusted for cpi and difference in avarage tax and local tax adjusted for cpi.

```
CigarettesSW$rincome <- with(CigarettesSW, income/population/cpi)
CigarettesSW$tdiff <- with(CigarettesSW, (taxs - tax)/cpi)
CigarettesSW$rtax <- with(CigarettesSW, tax/cpi)
```

First Stage

Lets assume we want to fit a linear model such as

$$\log(packs_i) = \beta_0 + \beta_1 * \log(rprice_i) + \beta_2 * \log(rincome_i)$$

Here we could suffer from endogenity bias as the price is probably endogenous. The first stage would be to fit the first stage with price as dependent variable of its instruments. These fitted values would then be used in the second stage. For this example I will use $tdiff$ and $rtax$ as candidate instruments for the logarithm of price. I will not consider any theoretical arguments as to why, this is up to you. But generally the type of questions you want to be asking yourself is whether the instruments can truly be seen as exogenous. Note that I will only use data for year being 1995. Note that 1995 is given as a character in the subset argument of lm(), this is due to the data only being sampled for two years so the variable year has been entered as a factor rather than an integer.

```
fs_lm <- lm(log(rincome) ~ tdiff + rtax, data = CigarettesSW,
            subset = year == "1995")
```

A valid instrument has to be exogenous, which is something that you can never test but only theoretically motivate. But you do have the assumption of relevance of instrument, meaning that it has to have a relationship with the variable it instruments for. Relevance of $tdiff$ and $rtax$ comes down to γ_1 and γ_2 being significantly different from 0 below.

$$\log(rprice) = \gamma_0 + \gamma_1 tdiff + \gamma_2 tax/cpi + v$$

```
summary(fs_lm)
```

```
##
## Call:
## lm(formula = log(rincome) ~ tdiff + rtax, data = CigarettesSW,
##     subset = year == "1995")
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.211961 -0.093315 -0.003678  0.084649  0.280133
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.452419   0.062912  38.982  < 2e-16 ***
## tdiff       -0.006572   0.007471  -0.880 0.383679
## rtax         0.007485   0.002072   3.613 0.000759 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.121 on 45 degrees of freedom
## Multiple R-squared:  0.2496, Adjusted R-squared:  0.2163
## F-statistic: 7.486 on 2 and 45 DF,  p-value: 0.001561
```

This doesn't hold for tdiff, so the I will only use rtax as an instrument for this example.

```
firstStage <- lm(log(rincome) ~  rtax, data = CigarettesSW,
            subset = year == "1995")
# save the fitted values in the data frame
CigarettesSW$instrumented_income <- firstStage$fitted.values
```

Second Stage

Now we use the fitted values from the first stage replace $\log(rincome)$.

```
secondStage <- lm(log(packs) ~ log(rprice) + instrumented_income,
                    data = CigarettesSW, subset = year == "1995")
```

```r
summary(secondStage)
```

```
##
## Call:
## lm(formula = log(packs) ~ log(rprice) + instrumented_income,
##     data = CigarettesSW, subset = year == "1995")
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.63439 -0.09395  0.02546  0.11947  0.38296
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           9.9845     1.1202   8.913 1.68e-11 ***
## log(rprice)          -1.7476     0.6695  -2.610   0.0122 *
## instrumented_income   1.0856     1.2862   0.844   0.4031
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1902 on 45 degrees of freedom
## Multiple R-squared:  0.415,  Adjusted R-squared:  0.389
## F-statistic: 15.96 on 2 and 45 DF,  p-value: 5.765e-06
```

The summary output gives you the results from the 2SLS estimates for this example.

Robust Standard Errors

If you want to use robust standard errors you can use the sandwich and lmtest packages as follows.

```r
# vcovHC gives the HAC-estimates
coeftest(secondStage, vcov. = vcovHC)
```

```
##
## t test of coefficients:
##
##                     Estimate Std. Error t value  Pr(>|t|)
## (Intercept)          9.98446    1.03570  9.6403 1.618e-12 ***
## log(rprice)         -1.74764    0.72275 -2.4180   0.01972 *
## instrumented_income  1.08565    1.36435  0.7957   0.43037
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```