IV regression

Tse Chun Hei Vincent

2021/3/24

Simulating Data

```
library(MASS)
set.seed(1234)
n=1000
Rho=matrix(c(1, 0.5, 0.5, 1), 2, 2, byrow = TRUE)
sims = mvrnorm(n,c(0,0), Rho)
e = sims[,1]
v = sims[,2]
z = runif(n)
x = 0.5 + 0.8 * z + v
y = -0.3 + x + e
```

Run IV regression

```
OLS1=lm(y~x)
OLS1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Coefficients:
## (Intercept)            x
##     -0.7397       1.4606
```

```
first_stage = lm(x~z)
reduced_form = lm (y~z)
first_stage
```

```
##
## Call:
## lm(formula = x ~ z)
##
## Coefficients:
## (Intercept)            z
##      0.4745       0.8192
```

```
reduced_form
```

```
##
## Call:
## lm(formula = y ~ z)
##
## Coefficients:
## (Intercept)            z
##      0.1615       0.7850
```

```
0.7850/0.8192
```

```
## [1] 0.958252
```

I found the betas of x regress on z and y regress on z. Finally I found the beta of 2SLS.

```
library(AER)
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
## Registered S3 methods overwritten by 'tibble':
##   method     from
##   format.tbl pillar
##   print.tbl  pillar
```

```
## Loading required package: lmtest
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: sandwich
```

```
## Loading required package: survival
```

```
iv_results=ivreg(y~x|z)
summary(iv_results)
```

```
##
## Call:
## ivreg(formula = y ~ x | z)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8096 -0.6405 -0.0326  0.6351  3.0896
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.2932     0.1206  -2.431   0.0152 *
## x             0.9582     0.1310   7.313 5.34e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9918 on 998 degrees of freedom
## Multiple R-Squared: 0.6764,  Adjusted R-squared: 0.6761
## Wald test: 53.49 on 1 and 998 DF,  p-value: 5.34e-13
```

```
confint(iv_results,level = 0.95)
```

```
##                 2.5 %     97.5 %
## (Intercept) -0.529575 -0.0567682
## x            0.701438  1.2150502
```

95%confidence interval (0.7014, 1.215)

Generate variables

```
library("readxl")
df=read_excel('cigarette.xlsx')
df$ravgprs = df$avgprs/df$cpi
df$rtax = df$tax/df$cpi
df$rtaxs = df$taxs/df$cpi
df$rtaxso = df$rtaxs-df$rtax
df$lpackpc = log(df$packpc)
df$lravgprs = log(df$ravgprs)
df$perinc = df$income/(df$pop*df$cpi)
df$lperinc = log(df$perinc)
df
```

```
## Warning: `...` is not empty.
##
## We detected these problematic arguments:
## * `needs_dots`
##
## These dots only exist to allow future extensions and should be empty.
## Did you misspecify an argument?
```

```
## # A tibble: 96 x 17
##    state  year   cpi    pop packpc income   tax avgprs  taxs ravgprs  rtax rtaxs
##    <chr> <dbl> <dbl>  <dbl>  <dbl>  <dbl> <dbl>  <dbl> <dbl>   <dbl> <dbl> <dbl>
##  1 AL     1985  1.08 3.97e6   116. 4.60e7  32.5  102.   33.3    95.0  30.2  31.0
```

```
##  2 AR     1985  1.08 2.33e6   129. 2.62e7  37    101.   37      94.3  34.4  34.4
##  3 AZ     1985  1.08 3.18e6   105. 4.40e7  31    109.   36.2   101.   28.8  33.6
##  4 CA     1985  1.08 2.64e7   100. 4.47e8  26    108.   32.1   100.   24.2  29.8
##  5 CO     1985  1.08 3.21e6   113. 4.95e7  31     94.3  31      87.6  28.8  28.8
##  6 CT     1985  1.08 3.20e6   109. 6.01e7  42    128.   51.5   119.   39.0  47.8
##  7 DE     1985  1.08 6.18e5   144. 9.93e6  30    102.   30      95.3  27.9  27.9
##  8 FL     1985  1.08 1.14e7   122. 1.67e8  37    115.   42.5   107.   34.4  39.5
##  9 GA     1985  1.08 5.96e6   127. 7.84e7  28     97.0  28.8    90.2  26.0  26.8
## 10 IA     1985  1.08 2.83e6   114. 3.79e7  34    102.   37.9    94.6  31.6  35.2
## # ... with 86 more rows, and 5 more variables: rtaxso <dbl>, lpackpc <dbl>,
## #   lravgprs <dbl>, perinc <dbl>, lperinc <dbl>
```

First stage regression:

```
df2 = subset(df, year==1995)
lravphat=lm(lravgprs~rtaxso, data=df2)
summary(lravphat)
```

```
##
## Call:
## lm(formula = lravgprs ~ rtaxso, data = df2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.221027 -0.044324  0.000111  0.063730  0.210717
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.616546   0.029108   158.6  < 2e-16 ***
## rtaxso      0.030729   0.004802     6.4 7.27e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09394 on 46 degrees of freedom
## Multiple R-squared:  0.471,  Adjusted R-squared:  0.4595
## F-statistic: 40.96 on 1 and 46 DF,  p-value: 7.271e-08
```

Second stage regression:

```
df2$lravphat=predict(lravphat)
OLS3=lm(lpackpc~lravphat, data= df2, year==1995)
summary(OLS3)
```

```
##
## Call:
## lm(formula = lpackpc ~ lravphat, data = df2, subset = year ==
##     1995)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.63180 -0.15802  0.00524  0.13574  0.61434
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.7199     1.8012   5.396  2.3e-06 ***
## lravphat     -1.0836     0.3766  -2.877  0.00607 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2264 on 46 degrees of freedom
## Multiple R-squared:  0.1525, Adjusted R-squared:  0.1341
## F-statistic: 8.277 on 1 and 46 DF,  p-value: 0.006069
```

IV regression function

```
iv2=ivreg(lpackpc~lravgprs|rtaxso, data=df2)
summary(iv2)
```

```
##
## Call:
## ivreg(formula = lpackpc ~ lravgprs | rtaxso, data = df2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64619 -0.07732  0.02981  0.11283  0.41904
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.7199     1.5141   6.420 6.79e-08 ***
## lravgprs     -1.0836     0.3166  -3.422  0.00131 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1904 on 46 degrees of freedom
## Multiple R-Squared: 0.4011,  Adjusted R-squared: 0.3881
## Wald test: 11.71 on 1 and 46 DF,  p-value: 0.001313
```

AJR wants to find out the fudamental causes of the large differences in income per capita acroos countries. AJR's key theory is that they think colonial institutions are different for diferent purposes. The colonialism after hundred years can still affect the current economic performance. Therefore, they want to find out that whether the european settlement would give positive effect or negative effect to the countries.

Regress loggdp on risk:

```
df3 = read.csv('ajr.csv')
ols = lm(loggdp~risk, data=df3)
summary(ols)
```

```
##
## Call:
## lm(formula = loggdp ~ risk, data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8351 -0.4449  0.1804  0.4834  1.2072
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.73119    0.41465  11.410  < 2e-16 ***
## risk         0.50511    0.06232   8.105 3.24e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7225 on 60 degrees of freedom
## Multiple R-squared:  0.5227, Adjusted R-squared:  0.5147
## F-statistic:  65.7 on 1 and 60 DF,  p-value: 3.241e-11
```

```
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
stargazer(ols, type = "text")
```

```
##
## ===============================================
##                         Dependent variable:
##                     ---------------------------
##                                loggdp
## -----------------------------------------------
## risk                          0.505***
##                                (0.062)
##
## Constant                      4.731***
##                                (0.415)
##
## -----------------------------------------------
## Observations                    62
## R2                              0.523
## Adjusted R2                     0.515
## Residual Std. Error       0.722 (df = 60)
## F Statistic           65.696*** (df = 1; 60)
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

We cannot interpret the results of "ols" causally, because we need to consider the mortality.

Estimate the rst-stage regression of risk on logmort0:

```
first_stage2 = lm(risk~logmort0, data = df3)
summary(first_stage2)
```

```
##
## Call:
## lm(formula = risk ~ logmort0, data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6425 -0.9952  0.0388  0.8577  3.4002
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.3882     0.6333  14.825  < 2e-16 ***
## logmort0     -0.6196     0.1308  -4.736 1.37e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.277 on 60 degrees of freedom
## Multiple R-squared:  0.2721, Adjusted R-squared:   0.26
## F-statistic: 22.43 on 1 and 60 DF,  p-value: 1.374e-05
```

The risk factor has negtive effect towards the log mortality. The log mortality will decrease 43% for every increase in risk.

Estimate the reduced-formregression:

```
reduced_form2 = lm(loggdp~logmort0, data = df3)
summary(reduced_form2)
```

```
##
## Call:
## lm(formula = loggdp ~ logmort0, data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6962 -0.5022  0.1022  0.4829  1.4268
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.63361    0.38221  27.822  < 2e-16 ***
## logmort0    -0.56088    0.07895  -7.105 1.66e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7706 on 60 degrees of freedom
## Multiple R-squared:  0.4569, Adjusted R-squared:  0.4478
## F-statistic: 50.48 on 1 and 60 DF,  p-value: 1.658e-09
```

The log mortality has negtive effect towards the log gdp. The log gdp decreases by 0.56% for every 1% increase of log mortality.

IV regression:

```
iv=ivreg(loggdp~risk|logmort0, data=df3)
summary(iv)
```

```
##
## Call:
## ivreg(formula = loggdp ~ risk | logmort0, data = df3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.35130 -0.54193  0.05887  0.67539  1.63873
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1345     1.0139   2.105   0.0395 *
## risk          0.9053     0.1552   5.834 2.32e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9384 on 60 degrees of freedom
## Multiple R-Squared: 0.1946,  Adjusted R-squared: 0.1812
## Wald test: 34.04 on 1 and 60 DF,  p-value: 2.321e-07
```

The r-squared is much lower than the 'ols'. That means the actual variation explained is that big.

```
df3$first_stage2 = predict(first_stage2)
OLS4=lm(loggdp~first_stage2, data= df3)
summary(OLS4)
```

```
##
## Call:
## lm(formula = loggdp ~ first_stage2, data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6962 -0.5022  0.1022  0.4829  1.4268
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.1345     0.8326   2.564   0.0129 *
## first_stage2   0.9053     0.1274   7.105 1.66e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7706 on 60 degrees of freedom
## Multiple R-squared:  0.4569, Adjusted R-squared:  0.4478
## F-statistic: 50.48 on 1 and 60 DF,  p-value: 1.658e-09
```

Including malaria as an additional regressor:

```
ols8 = lm(loggdp~risk+malaria, data=df3)
summary(ols8)
```

```
##
## Call:
## lm(formula = loggdp ~ risk + malaria, data = df3)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4626 -0.3124  0.1124  0.3511  1.0995
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.29587    0.40873  15.404  < 2e-16 ***
## risk         0.33889    0.05541   6.116 8.30e-08 ***
## malaria     -1.14546    0.18256  -6.274 4.51e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5642 on 59 degrees of freedom
## Multiple R-squared:  0.7137, Adjusted R-squared:  0.704
## F-statistic: 73.54 on 2 and 59 DF,  p-value: < 2.2e-16
```

```
OLS7=lm(risk~logmort0 + malaria, data= df3)
summary(OLS7)
```

```
##
## Call:
## lm(formula = risk ~ logmort0 + malaria, data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2758 -0.9160  0.0290  0.7708  3.3267
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.8342     0.7540  11.717   <2e-16 ***
## logmort0     -0.4380     0.1882  -2.328   0.0234 *
## malaria      -0.6962     0.5220  -1.334   0.1874
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.269 on 59 degrees of freedom
## Multiple R-squared:  0.2934, Adjusted R-squared:  0.2695
## F-statistic: 12.25 on 2 and 59 DF,  p-value: 3.547e-05
```

```
iv9=ivreg(loggdp~risk|+malaria, data=df3)
summary(iv9)
```

```
##
## Call:
## ivreg(formula = loggdp ~ risk | +malaria, data = df3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.63783 -0.63604 -0.08633  0.86148  2.08249
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   1.0908     1.3045   0.836    0.406
## risk          1.0661     0.1999   5.334 1.54e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.108 on 60 degrees of freedom
## Multiple R-Squared: -0.1221, Adjusted R-squared: -0.1408
## Wald test: 28.46 on 1 and 60 DF,  p-value: 1.536e-06
```