Heteroskedasticity and GLS

Ryuya Ko

11/27/2019

概要

- 分散均一の仮定が成り立たない場合の分析手法として、GLS(Generalized Least Square) を復習する
  - autocorrelation については除外して考える
- 分散不均一の場合に関する R を用いたデータ分析の方法を確認する
  - GLS の実装
  - robust covariance matrix を用いた検定

Heteroskedasticity

今までは簡単化のため分散均一の過程を置いてきた。すなわち、全ての i について以下が成り立つ:

$$Var(\varepsilon_i|x_i) = \sigma^2.$$

しかし、実際のデータではこのような仮定は成り立たないことが多い。例えば、都道府県ごとの一人あたり GDP を考える。都道府県 j の居住者 i について, 一人あたり GDP を y_{ij}, 年齢や職種などの説明変数の (K, 1) ベクトルを x_{ij}, 撹乱項を ε_{ij} として表し、

$$y_i = x_i'\beta + \varepsilon_i.$$

という関係式が成り立つと仮定する。その他ランダムサンプリングなどの仮定も (分散均一性を含めて) 成り立つことを仮定する。

一方で手に入るデータは都道府県平均の一人あたり GDP である。 $\bar{y}_j, \bar{x}_j, \bar{\varepsilon}_j$ をそれぞれ平均値として表すと

$$\bar{y}_j = \bar{x}_j'\beta + \bar{\varepsilon}_j,$$

という関係が成り立つことがわかる。

ここで各都道府県 j の人口数を n_j とすると、$Var(\bar{\varepsilon}_j|\bar{x}_j) = \frac{\sigma^2}{n_j^2}$ となり、我々が入手できるデータについては分散均一性が成り立たないことがわかる。

分散不均一なデータの確認

実際のデータを見ていく。使用するのは米国の GPA に関するデータ。まずはワーキングディレクトリを設定して必要なライブラリを読み込んでいく。 `lmtest` ライブラリを使うため、インストールしていない場合は `install.package` で R に落としておくこと。

ライブラリを読み込んだら `help` でデータを確認する。 `spring` の値で学期が別れているので、後々のために数値型から因子型に変換する。 `head` でざっと中身だけ確認しよう。

```
setwd('~/econ/subsemi/introductory_econometrics')
library(dplyr);library(ggplot2);library(wooldridge)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(lmtest);library(car);library(estimatr)
```

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

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

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

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

```
##
## Attaching package: 'car'
```

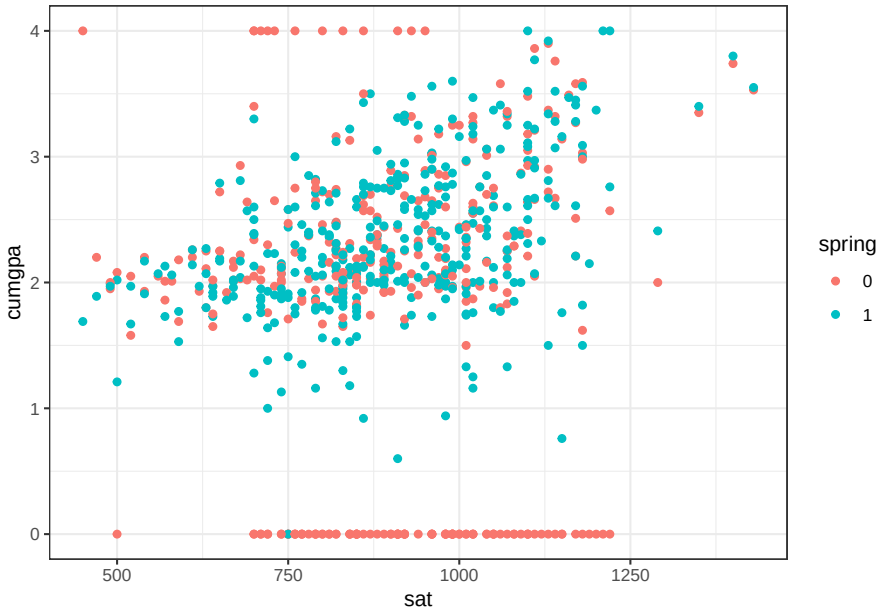
```
## The following object is masked from 'package:dplyr':
```

```
##
##     recode
```

```
data("gpa3");help(gpa3)
gpa3[,'spring'] <- gpa3[, 'spring'] %>% as.factor()
head(gpa3)
```

ggplot を用いてプロットする。今回はもっとも関係のありそうな SAT のスコアと cumgpa との間で散布図を描いてみる。

```
ggplot(data=gpa3, aes(x=sat, y=cumgpa, color=spring))+
  geom_point()+
  theme_bw()
```



もちろん他の説明変数で条件付けたときの値を見なければはっきりとしたことは言えないが、なんとなく不均一分散っぽい感じがする。

以下、このデータなどを使って R での実装を行う。漸近分散などについては時間が許せば板書で説明したい。

OLS 回帰と検定

GLS・WLS を行う前に、robust-covariance matrix を用いた実装を考える。

lmtest パッケージの関数を用いることで、lm オブジェクトに関して各種検定を robust-covariance を用いて行うことができる

- coeftest 関数を用いることで t-test を行うことができる

```
reg <- lm(cumgpa~sat+hsperc+tothrs+female+black+white,
          data=gpa3, subset=(spring==1))
summary(reg)
```

```
##
## Call:
## lm(formula = cumgpa ~ sat + hsperc + tothrs + female + black +
##     white, data = gpa3, subset = (spring == 1))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.54320 -0.29104 -0.02252  0.28348  1.24872
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.4700648  0.2298031   6.397 4.94e-10 ***
## sat          0.0011407  0.0001786   6.389 5.20e-10 ***
## hsperc      -0.0085664  0.0012404  -6.906 2.27e-11 ***
## tothrs       0.0025040  0.0007310   3.426 0.000685 ***
## female       0.3034333  0.0590203   5.141 4.50e-07 ***
## black       -0.1282837  0.1473701  -0.870 0.384616
## white       -0.0587217  0.1409896  -0.416 0.677295
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4693 on 359 degrees of freedom
## Multiple R-squared:  0.4006, Adjusted R-squared:  0.3905
## F-statistic: 39.98 on 6 and 359 DF,  p-value: < 2.2e-16
```

```
coeftest(reg) #t-test
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  1.47006477  0.22980308  6.3971 4.942e-10 ***
## sat          0.00114073  0.00017856  6.3885 5.197e-10 ***
## hsperc      -0.00856636  0.00124042 -6.9060 2.275e-11 ***
## tothrs       0.00250400  0.00073099  3.4255 0.0006847 ***
```

```
## female       0.30343329  0.05902033  5.1412 4.497e-07 ***
## black       -0.12828368  0.14737012 -0.8705 0.3846164
## white       -0.05872173  0.14098956 -0.4165 0.6772953
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
coeftest(reg, vcov. = hccm) # t-test using robust-cov
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  1.47006477  0.22938036  6.4089 4.611e-10 ***
## sat          0.00114073  0.00019532  5.8402 1.169e-08 ***
## hsperc      -0.00856636  0.00144359 -5.9341 6.963e-09 ***
## tothrs       0.00250400  0.00074930  3.3418   0.00092 ***
## female       0.30343329  0.06003964  5.0539 6.911e-07 ***
## black       -0.12828368  0.12818828 -1.0007   0.31762
## white       -0.05872173  0.12043522 -0.4876   0.62615
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- linearHypothesis を使うことで、指定した帰無仮説に関する F-test を行うことができる。

```
nullhypo <- c("white = 0", "black = 0")
linearHypothesis(reg, nullhypo) # homoskedasticity-based F-test
```

```
## Linear hypothesis test
##
## Hypothesis:
## white = 0
## black = 0
##
## Model 1: restricted model
## Model 2: cumgpa ~ sat + hsperc + tothrs + female + black + white
##
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    361 79.362
## 2    359 79.062  2   0.29934 0.6796 0.5075
```

```
linearHypothesis(reg, nullhypo, vcov.=hccm) # F test using White's robust covariance mat
```

```
## Linear hypothesis test
##
## Hypothesis:
## white = 0
## black = 0
##
## Model 1: restricted model
## Model 2: cumgpa ~ sat + hsperc + tothrs + female + black + white
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F Pr(>F)
## 1    361
## 2    359  2 0.6725 0.5111
```

また、estimatr パッケージの lm_robust 関数を使うことでも同じことができる:

```
reg.rob <- lm_robust(cumgpa~sat+hsperc+tothrs+female+black+white,
                     data=gpa3, subset=(spring==1))
summary(reg.rob)
```

```
##
## Call:
## lm_robust(formula = cumgpa ~ sat + hsperc + tothrs + female +
##     black + white, data = gpa3, subset = (spring == 1))
##
## Standard error type:  HC2
##
## Coefficients:
##              Estimate Std. Error t value  Pr(>|t|)   CI Lower  CI Upper  DF
## (Intercept)  1.470065  0.2238293  6.5678 1.794e-10  1.0298834  1.910246 359
## sat          0.001141  0.0001925  5.9268 7.249e-09  0.0007622  0.001519 359
## hsperc      -0.008566  0.0014237 -6.0172 4.380e-09 -0.0113661 -
0.005767 359
## tothrs       0.002504  0.0007413  3.3776 8.112e-04  0.0010461  0.003962 359
## female       0.303433  0.0592945  5.1174 5.058e-07  0.1868250  0.420042 359
## black       -0.128284  0.1230101 -1.0429 2.977e-01 -0.3701947  0.113627 359
## white       -0.058722  0.1152462 -0.5095 6.107e-01 -0.2853642  0.167921 359
```

```
##
## Multiple R-squared:  0.4006 ,    Adjusted R-squared:  0.3905
## F-statistic: 39.17 on 6 and 359 DF,  p-value: < 2.2e-16
```

```
linearHypothesis(reg.rob, nullhypo)
```

```
## Linear hypothesis test
##
## Hypothesis:
## white = 0
## black = 0
##
## Model 1: restricted model
## Model 2: cumgpa ~ sat + hsperc + tothrs + female + black + white
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    361
## 2    359  2 1.4178     0.4922
```

また、Breush-Pagan test を用いることで帰無仮説を分散均一 ($\sigma_i^2 = \sigma^2$) として検定することができる:

```
bptest(reg)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  reg
## BP = 44.557, df = 6, p-value = 5.732e-08
```

以上のように OLS 推定量を用いても分散不均一に対してロバストな検定は可能である。 robust-covariance matrix を使った推定は便利ではあるが、明らかに分散不均一が認められる場合、efficient ではない。したがって、BP 検定などで分散不均一が認められた場合、WLS・GLS を用いる必要がある。

WLS

ここでは、$Cov(\varepsilon_i, \varepsilon_j) = 0$ かつ $Var(\varepsilon_i) = \sigma^2 w_i$ と書けるような誤差項を考える。

ここで、$y_i^* = w_i y_i, x_i^* = w_i x_i, \varepsilon^* = w_i \varepsilon_i$ とすることで y_i^* を x_i^* に回帰することで BLUE な推定量を得る。

```r
data("k401ksubs");help(k401ksubs)

ols.401 <-lm(nettfa~inc+male+e401k+age, data=k401ksubs, subset=(fsize==1)) # OLS
wls.401 <- lm(nettfa~inc+male+e401k+age, data=k401ksubs, subset=(fsize==1),
   weight=1/inc) # WLS

summary(ols.401)
```

```
##
## Call:
## lm(formula = nettfa ~ inc + male + e401k + age, data = k401ksubs,
##     subset = (fsize == 1))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -176.92  -14.13   -3.58    6.26 1109.44
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -45.57805    4.41592 -10.321  < 2e-16 ***
## inc           0.75095    0.06169  12.172  < 2e-16 ***
## male          2.10574    2.04973   1.027  0.30439
## e401k         6.29173    2.12831   2.956  0.00315 **
## age           0.85675    0.09385   9.129  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 44.6 on 2012 degrees of freedom
## Multiple R-squared:  0.1235, Adjusted R-squared:  0.1217
## F-statistic: 70.86 on 4 and 2012 DF,  p-value: < 2.2e-16
```

```r
summary(wls.401)
```

```
##
## Call:
## lm(formula = nettfa ~ inc + male + e401k + age, data = k401ksubs,
##     subset = (fsize == 1), weights = 1/inc)
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -26.156  -2.572  -0.880   0.926 177.839
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -33.95919    3.38838 -10.022  < 2e-16 ***
## inc           0.72666    0.06448  11.270  < 2e-16 ***
## male          1.53471    1.56354   0.982  0.32643
## e401k         4.72256    1.70670   2.767  0.00571 **
## age           0.60143    0.07014   8.575  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.08 on 2012 degrees of freedom
## Multiple R-squared:  0.1077, Adjusted R-squared:  0.1059
## F-statistic: 60.72 on 4 and 2012 DF,  p-value: < 2.2e-16
```

FGLS

FGLS(Feasible Generalized Least Square) は、重みの関数形がわかっていないときに行う。

```
df <- gpa3 %>%
  filter(spring == 1) %>%
  as.data.frame()
reg <- lm(cumgpa~sat+hsperc+tothrs+female+black+white, data=df)
df[,'log_resid_sq'] <- log(resid(reg)^2)
var.reg <- lm(log_resid_sq~sat+hsperc+tothrs+female+black+white, data=df)
w <- 1/exp(fitted(var.reg))
fgls.reg <- lm(cumgpa~sat+hsperc+tothrs+female+black+white, data=df, weights=w)
summary(fgls.reg)
```

```
##
## Call:
## lm(formula = cumgpa ~ sat + hsperc + tothrs + female + black +
##     white, data = df, weights = w)
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7946 -1.1846 -0.1055  1.2308  5.6331
##
## Coefficients:
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.3906684  0.2050248   6.783 4.87e-11 ***
## sat          0.0010508  0.0001557   6.750 5.95e-11 ***
## hsperc      -0.0063561  0.0010893  -5.835 1.20e-08 ***
## tothrs       0.0025664  0.0006086   4.217 3.14e-05 ***
## female       0.2903921  0.0486122   5.974 5.59e-09 ***
## black       -0.0660812  0.1373362  -0.481    0.631
## white        0.0266909  0.1333275   0.200    0.841
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.804 on 359 degrees of freedom
## Multiple R-squared:  0.4365, Adjusted R-squared:  0.4271
## F-statistic: 46.35 on 6 and 359 DF,  p-value: < 2.2e-16
```

```
summary(reg)
```

```
##
## Call:
## lm(formula = cumgpa ~ sat + hsperc + tothrs + female + black +
##     white, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.54320 -0.29104 -0.02252  0.28348  1.24872
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.4700648  0.2298031   6.397 4.94e-10 ***
## sat          0.0011407  0.0001786   6.389 5.20e-10 ***
## hsperc      -0.0085664  0.0012404  -6.906 2.27e-11 ***
## tothrs       0.0025040  0.0007310   3.426 0.000685 ***
## female       0.3034333  0.0590203   5.141 4.50e-07 ***
## black       -0.1282837  0.1473701  -0.870 0.384616
## white       -0.0587217  0.1409896  -0.416 0.677295
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4693 on 359 degrees of freedom
## Multiple R-squared:  0.4006, Adjusted R-squared:  0.3905
```

```
## F-statistic: 39.98 on 6 and 359 DF,  p-value: < 2.2e-16
```

若干ではあるが、標準誤差が全体的に小さくなっていることがわかる。