MLPS2

Hao Ran Li, Feiwen Liang, Leila Lan, Susu Zhu, Yitao Hu

12/04/2020

Question 1

```
#import data and libraries
library('lfe')
library('stargazer')
library(readr)
library(data.table)
library(foreign)
library(knitr)
library(ggplot2)
library(reshape2)
Car_data <- read_csv("imports85_modified.csv",
    col_types = cols(city.mpg = col_double(),
        horsepower = col_double(), price = col_double()))
```

1

Here we regress city.mpg on horsepower without fixed effects.

```
#regression without fixed effects or clustering
city_mpg_no_fixed_effect=felm(Car_data$city.mpg~Car_data$horsepower)
stargazer(city_mpg_no_fixed_effect,type = 'text',report = 'vc*t')
```

```
##
## ===============================================
##                         Dependent variable:
##                     ---------------------------
##                              city.mpg
## -----------------------------------------------
## horsepower                   0.041***
##                              t = 3.201
##
## Constant                     12.229***
##                              t = 8.534
##
## -----------------------------------------------
## Observations                    203
## R2                             0.049
## Adjusted R2                    0.044
## Residual Std. Error      7.252 (df = 201)
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

Under the assumption of i.i.d standard errors and without the fixed effect of another categorical variable, we get a positive and statistically significant slope coefficient,which implies that horsepower has a positive effect on city.mpg. But we also notice that this regression has a low explanatory power, which is concluded from low R square of 4.9%.

2

Here we regress city.mpg on horsepower with the fixed effect of whether the number of cylinders is ‘two’ or ‘four’. Note here, we first subset the data by whether the number of cylinders are ‘two’ or ‘four’.

```
#subset the data to only the cars with 2 or 4 cylinders
Cars_data_two_four_cylinders=Car_data[(Car_data$num.of.cylinders=='two')|(Car_data$num.of.cylinders=='f
#run the regression with the fixed effects
city_mpg_two_four_cylinders=felm(Cars_data_two_four_cylinders$city.mpg~Cars_data_two_four_cylinders$hors

stargazer(city_mpg_two_four_cylinders,type = 'text',report = 'vc*t')
```

```
##
## ===============================================
##                          Dependent variable:
##                      ---------------------------
##                               city.mpg
## -----------------------------------------------
## horsepower                    -0.088***
##                              t = -17.165
##
## -----------------------------------------------
## Observations                     161
## R2                              0.678
## Adjusted R2                     0.674
## Residual Std. Error       1.667 (df = 158)
## ===============================================
## Note:                *p<0.1; **p<0.05; ***p<0.01
```

Note here the slope coefficient becomes statically negative number. This is because the coefficient in this regression is the effect of horsepower on mpg holding number of cylinders constant, or the partial effect of horsepower. We also note that the model has much greater explanatory power with a R square of 67.8%. Comparing this result with our previous model, we conclude that the positive coefficient in part 1 model is actually driven by number of cylinders instead of horsepower.

3

```
Cars_data_two_four_cylinders=data.table(Cars_data_two_four_cylinders)
#demean by groups
demean_Cars_data=Cars_data_two_four_cylinders[,list(horsepower=horsepower-mean(horsepower,na.rm=T),city

#re-run the regression without fixed effects
#run the regression with the fixed effects
demean_two_four_cylinders=felm(demean_Cars_data$city.mpg~demean_Cars_data$horsepower|0|0|0)

stargazer(demean_two_four_cylinders,type = 'text',report = 'vc*t')
```

```
##
## ===============================================
##                          Dependent variable:
##                      ---------------------------
##                               city.mpg
## -----------------------------------------------
## horsepower                    -0.088***
##                              t = -17.219
```

```
##
## Constant                          0.026
##                                 t = 0.195
##
## -----------------------------------------------
## Observations                       161
## R2                                0.651
## Adjusted R2                       0.649
## Residual Std. Error      1.662 (df = 159)
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

The results are the same from those obtained in part 2. We can conclude that adding a fixed effect in linear regression is equivalent to perform a groupped de-mean transformation for both independent and dependent variables.

Question 2

```
#import data
StockRetAcct_DT= as.data.table(read.dta("StockRetAcct_insample.dta")) #set the key as firm ID and year
setkey(x = StockRetAcct_DT,FirmID,year)
#compute the excess returns
StockRetAcct_DT[,ExRet:=exp(lnAnnRet)-exp(lnRf),]
```

1

Perform Fama-MacBeth Regression and report the results

```
port_ret = NULL
for (i in 1980:2014)
{
StockRetAcct_yr =StockRetAcct_DT[year==i,]
Model_yr =lm(StockRetAcct_yr$ExRet ~ StockRetAcct_yr$lnInv)
port_ret = rbind(port_ret,Model_yr$coef[2])
}
 fm_output = list(MeanReturn = mean(port_ret),
                  StdReturn = sqrt(var(port_ret)),
                  SR_Return = mean(port_ret)/sqrt(var(port_ret)))
fm_output
```

```
## $MeanReturn
## [1] -0.08679146
##
## $StdReturn
##                         StockRetAcct_yr$lnInv
## StockRetAcct_yr$lnInv               0.1486441
##
## $SR_Return
##                         StockRetAcct_yr$lnInv
## StockRetAcct_yr$lnInv              -0.5838877
```

2

Recall in Fama-MacBeth Regression, at each time step t, we compute the coefficients (Portfolio Returns) in the following way as we did in regular linear regression:

$$\vec{\lambda}_t = (X_{t-1}^T X_{t-1})^{-1} X_{t-1} \vec{r}_t$$

where $\vec{\lambda}_t$ is a vector of excess portfolio returns for all factor-based long-short strategies,$\vec{r}_t$ is a vector of excess returns for each asset for the time period t, and X_{t-1} is the feature matrix at time t-1.

We know

$$r_{port} = \vec{w}^T \vec{r}$$

Therefore, the portfolio weights to construct all factor-based long-short strategies are computed as:

$$W_t = (X_{t-1}^T X_{t-1})^{-1} X_{t-1}$$

where each row vector of the matrix W_t is the portfolio weights to construct the corresponded factor-based trading strategy at time t-1.

In this case, the second row of W_t is the portfolio weights to construct lnInv-based long-short strategy at time t-1.

3

To reduce the industry-based noise, we add the Industries as a categorical variable in our original Fama-MacBeth Regression. In this case, the return of our long-short portfolio would be the “pure” return from Investment factor.

```r
Cleaned_port_ret = NULL
for (i in 1980:2014)
{
StockRetAcct_yr =StockRetAcct_DT[year==i,]
Model_yr =lm(StockRetAcct_yr$ExRet ~ StockRetAcct_yr$lnInv+as.factor(StockRetAcct_yr$ff_ind))
Cleaned_port_ret = rbind(Cleaned_port_ret,Model_yr$coef[2])
}
Cleaned_fm_output = list(MeanReturn = mean(Cleaned_port_ret),
                   StdReturn = sqrt(var(Cleaned_port_ret)),
                   SR_Return=mean(Cleaned_port_ret)/sqrt(var(Cleaned_port_ret)))
Cleaned_fm_output
```

```
## $MeanReturn
## [1] -0.08257762
##
## $StdReturn
##                        StockRetAcct_yr$lnInv
## StockRetAcct_yr$lnInv              0.1019642
##
## $SR_Return
##                        StockRetAcct_yr$lnInv
## StockRetAcct_yr$lnInv             -0.8098685
```

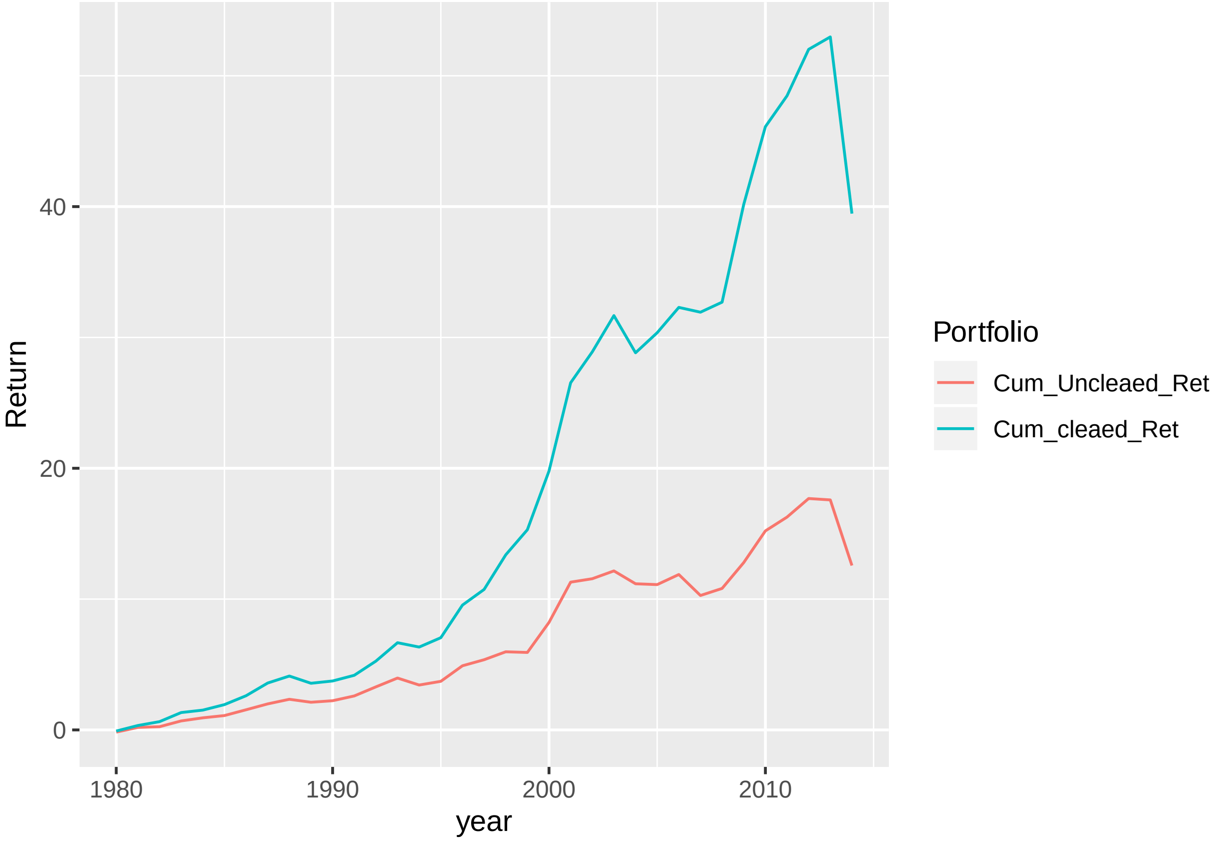
Note here, the Sharpe Ratio increases significantly from 0.58 to 0.81 (We are shorting the lnInv factor).

4

```r
#set risk constraint
SD_constraint=0.15
#compute the cumulative return after adding the leverage
Un_cleaned_Ret=(-port_ret)*(SD_constraint/as.numeric(fm_output$StdReturn))
Cum_Uncleaed_Ret=cumprod(1+Un_cleaned_Ret)-1

Cleaned_Ret=(-Cleaned_port_ret)*(SD_constraint/as.numeric(Cleaned_fm_output$StdReturn))
Cum_cleaed_Ret=cumprod(1+Cleaned_Ret)-1

Ret_dt=data.frame(cbind(1980:2014,Cum_Uncleaed_Ret,Cum_cleaed_Ret))
Ret_dt=melt(data = Ret_dt,id=1)
colnames(Ret_dt)=c('year','Portfolio','Return')
#plot the two series
ggplot(Ret_dt,aes(x = year,y = Return,col=Portfolio))+geom_line()
```



Question 3

1

```r
#define next year rv
StockRetAcct_DT[,leadrv:=shift(rv,type = 'lead'),]
#define five year ahead rv
StockRetAcct_DT[,lead5rv:=shift(rv,type = 'lead',n = 5),]
#run regressions
rvModelno=felm(formula = leadrv~rv+lnProf+lnLever+lnBM+lnROE+lnInv,StockRetAcct_DT)
```

```r
#with fixed effect on year
rvModel1=felm(formula = leadrv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year|0|0,StockRetAcct_DT)
#with fixed effect on year and industry
rvModel2=felm(formula = leadrv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year+ff_ind|0|0,StockRetAcct_DT)
#with auto-covariance but no industry fixed effect
rvModel3=felm(formula = leadrv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year|0|year,StockRetAcct_DT)
#with auto-covariance and cross-sectional clustering but no industry fixed effect
rvModel4=felm(formula = leadrv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year|0|year+FirmID,StockRetAcct_DT)
stargazer(rvModelno,rvModel1,rvModel2,rvModel3,rvModel4,type = 'text',report = 'vc*t')
```

```
##
## ===================================================================================================
##                                                         Dependent variable:
##                     -------------------------------------------------------------------------------
##                                                                leadrv
##                            (1)                (2)                (3)                (4)
## ---------------------------------------------------------------------------------------------------
## rv                       0.409***           0.527***           0.495***           0.527***
##                        t = 105.230        t = 128.205        t = 116.900         t = 4.675
##
## lnProf                  -0.039***          -0.022***          -0.027***           -0.022*
##                        t = -11.134         t = -7.955         t = -9.723         t = -1.902
##
## lnLever                 -0.007***          -0.006***           -0.002**            -0.006
##                         t = -7.054         t = -7.786         t = -2.497         t = -0.937
##
## lnBM                    -0.030***          -0.017***          -0.013***           -0.017**
##                        t = -35.493        t = -24.111        t = -16.575         t = -2.524
##
## lnROE                   -0.029***          -0.025***          -0.021***          -0.025***
##                         t = -6.589         t = -7.184         t = -5.869         t = -2.841
##
## lnInv                    0.066***           0.037***           0.035***           0.037***
##                         t = 26.624         t = 18.363         t = 17.356         t = 2.903
##
## Constant                 0.084***
##                         t = 54.028
##
## ---------------------------------------------------------------------------------------------------
## Observations              57,440             57,440             57,440             57,440
## R2                        0.266              0.546              0.552              0.546
## Adjusted R2               0.266              0.545              0.552              0.545
## Residual Std. Error 0.154 (df = 57433) 0.121 (df = 57399) 0.120 (df = 57388) 0.121 (df = 57399) 0.12
## ===================================================================================================
## Note:                                                                                 *p<0.1; **p<0
```

From the table above, we can see that adding a year fixed effect and standard error clustering significantly changed the model explanatory power (R square) and coefficient significance, while an industry fixed effects and cross-sectional clustering standard errors have minor impacts.

I would argue that the year fixed effect and standard error clustering come from the stylized fact of volitility clustering. Because volitility cluters within and across time, it is appropriate to add a year fixed effect and clustering feature in our panel regressional to capture this feature.

2

Regressional results for five year lead realized variance.

```
#run regressions
rvModelno=felm(formula = lead5rv~rv+lnProf+lnLever+lnBM+lnROE+lnInv,StockRetAcct_DT)
#with fixed effect on year
rvModel1=felm(formula = lead5rv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year|0|0,StockRetAcct_DT)
#with fixed effect on year and industry
rvModel2=felm(formula = lead5rv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year+ff_ind|0|0,StockRetAcct_DT)
#with auto-covariance but no industry fixed effect
rvModel3=felm(formula = lead5rv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year|0|year,StockRetAcct_DT)
#with auto-covariance and cross-sectional clustering but no industry fixed effect
rvModel4=felm(formula = lead5rv~rv+lnProf+lnLever+lnBM+lnROE+lnInv |year|0|year+FirmID,StockRetAcct_DT)
stargazer(rvModelno,rvModel1,rvModel2,rvModel3,rvModel4,type = 'text',report = 'vc*t')
```

```
##
## =========================================================================================================
##                                                           Dependent variable:
##                       ----------------------------------------------------------------------------------
##                                                                lead5rv
##                            (1)               (2)               (3)               (4)
## ---------------------------------------------------------------------------------------------------------
## rv                       0.067***          0.239***          0.199***          0.239***
##                        t = 14.558        t = 41.875        t = 33.910        t = 4.177
##
## lnProf                  -0.017***         -0.013***         -0.017***          -0.013
##                        t = -4.168        t = -3.316        t = -4.264        t = -1.569
##
## lnLever                 -0.008***         -0.005***         -0.005***          -0.005
##                        t = -7.107        t = -4.389        t = -3.389        t = -0.964
##
## lnBM                    -0.028***         -0.011***         -0.004***         -0.011**
##                       t = -27.840       t = -10.871        t = -4.068        t = -2.397
##
## lnROE                   -0.070***         -0.026***         -0.024***         -0.026***
##                       t = -13.230        t = -5.155        t = -4.665        t = -4.122
##
## lnInv                    0.053***          0.045***          0.042***          0.045***
##                        t = 18.176        t = 16.441        t = 15.299        t = 6.616
##
## Constant                 0.150***
##                        t = 80.853
##
## ---------------------------------------------------------------------------------------------------------
## Observations              55,539            55,539            55,539            55,539
## R2                        0.048             0.195             0.206             0.195
## Adjusted R2               0.048             0.194             0.205             0.194
## Residual Std. Error 0.182 (df = 55532) 0.168 (df = 55498) 0.167 (df = 55487) 0.168 (df = 55498) 0.16
## =========================================================================================================
## Note:                                                                                      *p<0.1; **p<0
```

From the table above, we observe that only the lnProf and lnLever become statistically insignificant, but the R square decreased significantly from 54.6% to around 19.4%. Therefore, I conclude that we cannot predict 5-year head realized variance at a high confidence level because our model has limited explantory power.

3

The greatest merit of running panel regression is that we increase the number of observations (and hopefully increase predictive power) for our model because coefficients are “trained” by a larger sample size. Also, we reduced the possibility of over-fitting by fitting shared coefficients across all firms. Lastly, we can adjust the fixed effects across time to increase the predictive power of our model.

The largest potential costs is the making simplification assumptions on the variance-covariance matrix of residuals to gain computational efficiency. If we run regressions on each firm, we can estimate the var-cov matrix and compute robust standard errors for all co-efficients, but we cannot implement the same approach because of large sample size. The assumptions we make on var-cov matrix may distort the panel regression results.