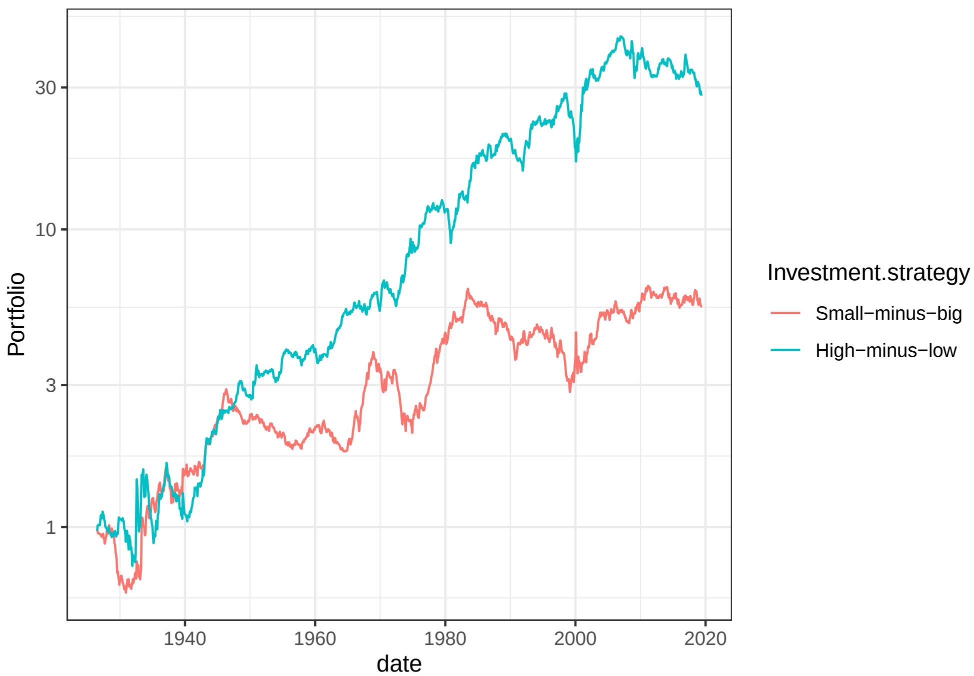
Financial Econometrics

Simple Regression

Darya Yuferova

There are very few as well established empirical facts in Finance as (1) small stocks earn a higher return than big stocks and (2) stocks with high book to market (BM) ratio earn a higher return than stocks with low book to market ratio. The plot below examplifies this by showing the value of two portfolios from 1926 to today. The size of the portfolio at the beginning is 1 USD. The red (blue) portfolio invests in small (high BM) by shorting big (low BM) firms. Note that the plot uses log scales for the y-axis.



In this lab session, you will use data on S&P500 firms from 2010 to 2015 to revisit the aforementioned findings in this particular dataset. This is interesting because, as you may recognize in the plot, the pattern seemed to have disappeared or even reversed in the most recent times. In particular, using the fiscal year 2014, you want to estimate following two regression equations:

$$Return_{Next\ fiscal\ year} = \alpha_0 + \beta_1 log(BM) + u$$

$$Return_{Next\ fiscal\ year} = \alpha_0 + \beta_1 log(MKTV_{End\ of\ current\ fiscal\ year}) + u$$

This exercise takes you through the standard steps in econometric analysis: load data, construct variables, limit the sample, and interpret the result.

You are provided with a csv file *Data_lab_simple_regession.csv* that includes following variables:

- lpermno: firm identifier
- datadate: date when the data was supplied to authorities
- fyear: fiscal year
- BE: book value of equity
- MKTV.june.of.current.fiscal.year, MKTV.end.of.current.fiscal.year, MKTV.end.of.subsequent.fiscal.year: market value recorded at different times
- FF.49: industry classification following Fama and French's 49 industries

Your analysis proceeds in following steps:

- Task 1: Load the data into R.
  - Investigate the structure of the data using the functions *str()*. Are all variables of the correct object type? If not, please readjust them [optional].
  - Summarise the data using the function *summary()*
- Task 2: Construct new variables
  - Book to market is defined book value of equity divided by market value as of June of the current fiscal year
  - Return during the next fiscal year is defined as the difference of log market capitalization of end of current and end of subsequent fiscal year
- Task 3: Estimate the regression equations as outlined above, using only the fiscal year 2014. Note that you can use the logarithm function *log()* within the regression function *lm()*.
- Task 4: Having estimated both models, combine them in a output table with the package *stargazer*. Google for additional information.
- Task 5: Complete following sentences for each model,
  - An increase in log(BM) of 1, leads to a ... of ... in returns at the ... significance level.
  - An increase in log(MKTV) of 1, leads to a ... of ... in returns at the ... significance level.
- Task 6 [optional]: You want to know, if the estimated coefficient $\hat{\beta}_1$ for the link between BM and future return depends on the year the regression is estimated for. Use following coding strategy:
  - Declare a 5x4 matrix. Each row of this matrix will be populated by the regression output for each fiscal year.
  - Write a loop that iterates over the fiscal years 2010 to 2014
  - For each iteration estimate the regression in Task 3
  - Store the estimated coefficient $\hat{\beta}_1$ on log(BM).
  - Inspect the output. Does the relationship between BM and future return seem to be robust?

```r
# packages needed
require(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.5.2
```

```r
require(stargazer)
```

```
## Warning: package 'stargazer' was built under R version 4.5.2
```

```r
# Task 1: get to know the data
comp <- read.csv("C:/Users/s13163/Dropbox/FIE401/data/data_labs/Data_lab_simple_regression.csv")

# investigate structure
str(comp)
```

```
## 'data.frame':    2478 obs. of  8 variables:
##  $ lpermno                           : int  10104 10104 10104 10104 10104 10107 10107 10107 10107 10
##  $ datadate                          : chr  "31/05/2011" "31/05/2012" "31/05/2013" "31/05/2014" ...
##  $ fyear                             : int  2010 2011 2012 2013 2014 2010 2011 2012 2013 2014 ...
##  $ BE                                : num  39835 43736 44821 47136 49043 ...
##  $ MKTV.end.of.current.fiscal.year   : num  158141 128913 157747 172071 197480 ...
##  $ MKTV.end.of.subsequent.fiscal.year: num  128913 157747 172071 197480 153471 ...
##  $ MKTV.june.of.current.fiscal.year  : num  107858 166706 145010 142679 180557 ...
##  $ FF.49                             : chr  "36 Softw" "36 Softw" "36 Softw" "36 Softw" ...
```

```r
# note that the date is not in date format and that the business description is a character
# adjust
comp$datadate <- as.Date(comp$datadate,format="%d/%m/%Y")

# summarise data
summary(comp)
```

```
##     lpermno         datadate              fyear            BE        
##  Min.   :10104   Min.   :2010-06-30   Min.   :2010   Min.   :-11080  
##  1st Qu.:27887   1st Qu.:2011-12-31   1st Qu.:2011   1st Qu.:  2671  
##  Median :64282   Median :2012-12-31   Median :2012   Median :  5855  
##  Mean   :57592   Mean   :2012-12-21   Mean   :2012   Mean   : 13427  
##  3rd Qu.:83821   3rd Qu.:2013-12-31   3rd Qu.:2013   3rd Qu.: 13432  
##  Max.   :93422   Max.   :2015-05-31   Max.   :2014   Max.   :224162  
##  MKTV.end.of.current.fiscal.year MKTV.end.of.subsequent.fiscal.year
##  Min.   :   234.6                Min.   :   193.1                  
##  1st Qu.:  7529.2                1st Qu.:  7859.9                  
##  Median : 13473.0                Median : 14305.2                  
##  Mean   : 28154.8                Mean   : 30396.0                  
##  3rd Qu.: 26997.0                3rd Qu.: 29489.4                  
##  Max.   :643120.1                Max.   :643120.1                  
##  MKTV.june.of.current.fiscal.year    FF.49          
##  Min.   :   229.2                 Length:2478       
##  1st Qu.:  7120.0                 Class :character  
##  Median : 12593.3                 Mode  :character  
##  Mean   : 26310.6                                   
##  3rd Qu.: 25272.0                                   
##  Max.   :556573.7                                   
```

```r
# Task 2: construct new variables and keep only positive book-to-market values
comp %>%
  mutate(return = log(MKTV.end.of.subsequent.fiscal.year/
```

```r
       MKTV.end.of.current.fiscal.year)*100,
       BM = BE / MKTV.june.of.current.fiscal.year) %>%
       filter(BM>0)-> comp


# Task 3: regression
# Note that I limit the data supplied to the regression in the regression formula
fit.1 <- lm(return ~ log(BM),
            data = comp[comp$fyear == 2014,])

fit.2 <- lm(return ~ log(MKTV.end.of.current.fiscal.year),
            data = comp[comp$fyear == 2014,])


# look at the output
summary(fit.1)
```

```
##
## Call:
## lm(formula = return ~ log(BM), data = comp[comp$fyear == 2014,
##     ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -196.827  -15.608    6.499   19.693   90.467
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -15.210      2.359  -6.447 2.78e-10 ***
## log(BM)       -4.178      1.775  -2.354    0.019 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.56 on 481 degrees of freedom
## Multiple R-squared:  0.01139,    Adjusted R-squared:  0.009332
## F-statistic: 5.541 on 1 and 481 DF,  p-value: 0.01898
```

```r
summary(fit.2)
```

```
##
## Call:
## lm(formula = return ~ log(MKTV.end.of.current.fiscal.year), data = comp[comp$fyear ==
##     2014, ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -190.297  -15.627    4.396   20.277   97.286
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           -55.796     14.686  -3.799 0.000164 ***
## log(MKTV.end.of.current.fiscal.year)    4.516      1.472   3.068 0.002275 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 33.43 on 481 degrees of freedom
## Multiple R-squared:  0.0192, Adjusted R-squared:  0.01716
## F-statistic: 9.414 on 1 and 481 DF,  p-value: 0.002275
```

```r
# Task 4

# make a nice stargazer table
# we will talk about other important parameters in stargazer later
stargazer(fit.1, fit.2,
          type = "text")
```

```
##
## ==================================================================
##                                            Dependent variable:
##                                       ----------------------------
##                                                  return
##                                            (1)            (2)
## ------------------------------------------------------------------
## log(BM)                                  -4.178**
##                                          (1.775)
##
## log(MKTV.end.of.current.fiscal.year)                   4.516***
##                                                         (1.472)
##
## Constant                                -15.210***    -55.796***
##                                          (2.359)       (14.686)
##
## ------------------------------------------------------------------
## Observations                               483            483
## R2                                        0.011          0.019
## Adjusted R2                               0.009          0.017
## Residual Std. Error (df = 481)            33.559         33.426
## F Statistic (df = 1; 481)                5.541**        9.414***
## ==================================================================
## Note:                                  *p<0.1; **p<0.05; ***p<0.01
```

```r
# Task 5

# An increase in log(BM) of 1, leads to a decrease of 4.2% in returns at the 5% significance level
# An increase in log(MKTV) of 1, leads to a increase of 4.5% in returns at the 1% significance level.

# Task 6

# declare matrix
coef.mat <- matrix(NA, nrow = 5, ncol = 4)

# declare a vector with years
years <- c(2010:2014)

# loop over all years
for(i in 1:5){
  # estimate regression for that year
  fit <- lm(return ~ log(BM),
            data = comp[comp$fyear == years[i],])
```

```r
  # extract the coefficients
  coef.mat[i,] <- summary(fit)$coefficients[2,]
}

# inspect result
colnames(coef.mat)<-colnames(summary(fit)$coefficients)
rownames(coef.mat)<-c(2010:2014)
print(coef.mat)
```

```
##       Estimate Std. Error   t value    Pr(>|t|)
## 2010 -1.522460   1.471354 -1.034734 0.301306538
## 2011  4.048296   1.241552  3.260673 0.001189674
## 2012  2.510030   1.198954  2.093516 0.036824011
## 2013 -3.379441   1.374886 -2.457979 0.014318646
## 2014 -4.178302   1.775094 -2.353849 0.018981523
```

```r
# looking at the first column which holds the coefficient, it is obvious
# that the relationship is not stable
```