Empirical Methods in Finance
TA Session

Mahyar Kargar

March 7, 2019

Fama-MacBeth Regressions

In this excerise, we will replicate the main results from Adrian, Etula, and Muir (2014) (AEM hereafter).[1] They propose a single-factor asset pricing model where the pricing factor is shocks to the leverage of securities broker-dealers. We try to test their model in the cross-section of equity and bond portfolios using size/book-to-market, momentum, and bond portfolios as test assets.

We first download the factor time-series (at quarterly frequency) from Professor Tyler Muir's website.

```
library(data.table)
library(zoo)
library(ggplot2)
library(scales)
library(stargazer)

rm(list=ls())
options(max.print=999999)
setwd('E:/Dropbox/PhD/Teaching/9_Winter 2019/Lars MFE Empirical Methods_W2018/TA Sessions/Thursday_03-07-2019/')

# read the leverage factor from Tyler's website
AEM <- fread("./LEVERAGEFACTORDATA.csv")
AEM <- AEM[,.(Date,LevFactor)]
AEM[,`:=`(year=Date%/%100,quarter=Date%%100)]
AEM[,date:=as.yearqtr(paste(year,quarter,sep="-"))][,`:=`(Date=NULL,year=NULL,quarter=NULL)]
setkey(AEM,date)
```

We then get the test assets from Ken French's website and CRSP. AEM use 41 test assets for their cross-sectional regressions: 25 portfolios sorted on size and book-to-market, 10 portfolios sorted on momentum, and 6 Treasury bond portfolios from CRSP. We also need risk-free rate to

[1] Adrian, Tobias, Erkko Etula, and Tyler Muir, 2014, Financial intermediaries and the cross-section of asset returns, *The Journal of Finance* 69, 2557-2596. Here is the link to the paper.

calculate excess returns which we obtain from Fama-French factors data as we did in Homework 5. The sample is quarterly from 1968Q1 to 2009Q4.

Note that all the data are monthly but the factor is quarterly. So we need to compute quarterly returns from monthly returns. As we learned in the very first lecture:

$$1 + R_{t,t+3} = (1 + R_{t+1}) \times (1 + R_{t+2}) \times (1 + R_{t+3})$$

where $1 + R_{t+k} = (P_{t+k} + D_{t+k})/P_{t+k-1}$ is the gross return in month k.

```
########################################
# Replicating results in AEM (LevFact) #
########################################
merged_AEM <- merge(factors,portfolios)
merged_AEM[,year_qtr:=yearqtr(date)]
setkey(merged_AEM,date,year_qtr)

# compute quarterly returns
merged_AEMqtr <- merged_AEM[,lapply(.SD, function(x) 100*(prod(1+(x/100))-1)),
                            by=.(year_qtr), .SDcols=-c("date","year","month")]

setkey(merged_AEMqtr,year_qtr)
setnames(AEM,"date","year_qtr")
setkey(AEM,year_qtr)
merged_AEMqtr <- merge(merged_AEMqtr,AEM)

tmp <- copy(merged_AEMqtr)
# annualize excess returns
merged_AEMqtr <- merged_AEMqtr[,lapply(.SD, function(x) 4*(x-RF)),
                               .SDcols = -c("MktRF","SMB","HML","RF","UMD","LevFactor"),by=year_qtr]
merged_AEMqtr[,LevFactor:=4*tmp$LevFactor]
merged_meltAEM <- melt(merged_AEMqtr,id.vars=c("year_qtr","LevFactor"))
```

Because this factor is *not tradable* we need to run Fama-MacBeth regressions to obtain the price of risk (λ) and assess the model. If the factor is not a portfolio of traded assets, we cannot apply the time-series tests. The reason is that it is no longer the case that the null hypothesis implies that α in the time-series regression is zero. To see this, consider the following one factor model

$$\mathbb{E}_t[R^e_{i,t+1}] = \beta_{i,t}\lambda, \quad \beta_i = \frac{Cov\left(F_{t+1}, R^e_{i,t+1}\right)}{Var(F_{t+1})}$$

Here, F_{t+1} is the factor, e.g. the leverage factor here. λ_t is the conditional risk premium (the price of risk) associated with "1 risk unit" exposure to the factor. Let's consider the time-series regression

$$R^e_{i,t+1} = \alpha_i + \beta_i F_{t+1} + \varepsilon_{i,t+1}$$

Taking unconditional expectations, this gives

$$\mathbb{E}[R^e_{i,t+1}] = \beta_i\lambda = \alpha_i + \beta_i\mathbb{E}[F_{t+1}]$$
$$\alpha_i = \beta_i\left(\lambda - \mathbb{E}[F_{t+1}]\right)$$

In the case that F_{t+1} is the excess return of a traded asset, $\lambda - \mathbb{E}[F_{t+1}] = 0$. This is because for the traed factor as a test asset itself, the beta coefficient is trivially equal to 1. However, this will not be the case in general for non-traded factors such as consumption growth, leverge factor, etc. In this case, we need to estimate the factor price of risk λ in cross-sectional tests. There are two main procedures: Two pass regression and Fama-MacBeth regressions. These procedures can of course also be used when the factor(s) are traded assets. Today we go over Fama-MacBeth regressions.[2]

The asset pricing model we want to estimate is:

$$\mathbb{E}[R^e_{i,t+1}] = \beta_{i,\text{LevFac}}\,\lambda_{\text{LevFac}},$$

where LevFac is AEM's broker-dealer leverage factor. For each asset i, we first estimate betas from time-series regressions of portfolio excess returns on the risk factor (LevFac):

$$R^e_{i,t} = a_i + \beta_{i,\text{LevFac}}\,\text{LevFac}_t + \epsilon_{i,t}, \quad i = 1, \ldots, N, \tag{1}$$

where N is the number of test portfolios ($N = 41$ in this case: 25 size/book-to-market, 10 momentum, and 6 Treasury bond portfolios).

Next, in order to estimate the factor risk price, λ_{LevFac}, we run T cross-sectional regressions (for each quarter $t = 1, \ldots, T$) of average excess returns (N observations) on the estimated betas $\hat{\beta}_{i,\text{LevFac}}$ from the time-series regressions:

$$\mathbb{E}\left[R^e_{i,t}\right] = \alpha_i + \widehat{\beta}_{i,\text{LevFac}}\,\lambda_{\text{LevFac},t} + \varepsilon_{i,t}, \quad i = 1, \ldots, N, \tag{2}$$

The final estimate is

$$\widehat{\lambda}_{\text{LevFac}} = \frac{1}{T}\sum_{t=1}^{T}\widehat{\lambda}_{\text{LevFac},t}$$

Assuming $\lambda_{\text{LevFac,t}}$s are uncorrelated across time, the standard error of the estimated price of risk is:

$$sd\left(\widehat{\lambda}_{\text{LevFac}}\right) = \frac{sd\left(\widehat{\lambda}_{\text{LevFac,t}}\right)}{\sqrt{T}}.$$

Here are the cross-sectional regression results. we can compare them to the 5th column of Table III on page 2571 of the paper. The results we get are very close to the paper. The price of risk (λ_{LevFac}) is positive and highly statistically significant. The cross-sectional R^2 of 75% is quite high.

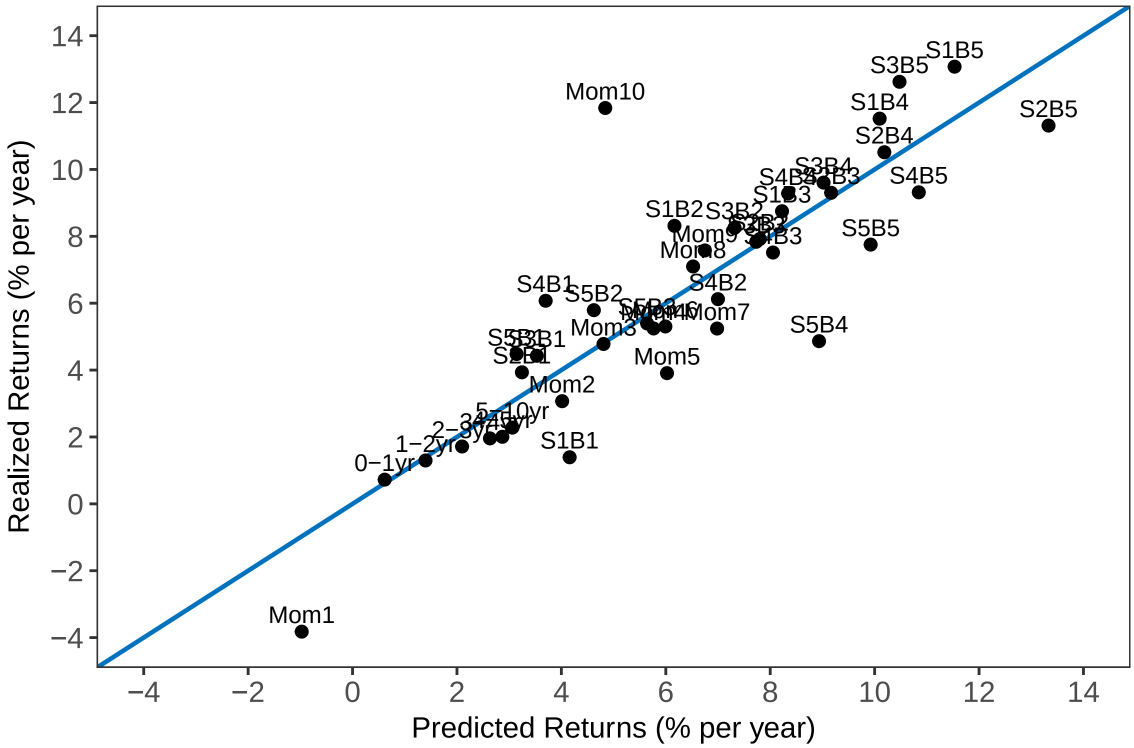
[2] For more details, see "*A Note on Linear Factor Models: Time series and cross-sectional tests*" posted under Week 9 on CCLE.

```
##
## ========================================
##                  Dependent variable:
##              ---------------------------
##                          ret
## ----------------------------------------
## LevFac                60.92***
##                        (13.16)
## Constant                0.21
##                        (2.01)
## ----------------------------------------
## Observations             41
## R2                      0.76
## Adjusted R2             0.75
## ========================================
## Note:        *p<0.1; **p<0.05; ***p<0.01
```

Compute the mean absolute pricing errors (MAPEs) total and by portfolio group.

```
##    portfolios     MAPE
## 1:   Size B/M 1.264875
## 2:        MOM 1.728811
## 3:       Bond 0.485593
## 4:      Total 1.263989
```

Plot predicted vsersus realized returns and compare with Figure 1 of the paper on page 2559. This figure shows the leverage factor's pricing performance in a cross-section that spans 35 common equity portfolios sorted on size, book-to-market, and momentum, and 6 Treasury bond portfolios sorted by maturity. The single-factor model explains 75% of the variation in average returns in these cross-sections, with an average absolute pricing error around 1.2% per annum.



Realized Returns (% per year)
Predicted Returns (% per year)

R code

```
library(data.table)
library(zoo)
library(ggplot2)
library(scales)
library(stargazer)

rm(list=ls())
options(max.print=999999)
setwd('E:/Dropbox/PhD/Teaching/9_Winter 2019/Lars MFE Empirical Methods_W2018/TA Sessions/Thursday_03-07-2019/')

# read the leverage factor from Tyler's website
AEM <- fread("./LEVERAGEFACTORDATA.csv")
AEM <- AEM[,.(Date,LevFactor)]
AEM[,`:=`(year=Date%/%100,quarter=Date%%100)]
AEM[,date:=as.yearqtr(paste(year,quarter,sep="-"))][,`:=`(Date=NULL,year=NULL,quarter=NULL)]
setkey(AEM,date)

#####################################
# Read FF 3- and FF 4 factor returns
#####################################
FF3_fact <- fread("F-F_Research_Data_Factors.txt",skip = 3)
setnames(FF3_fact,c("date","MktRF","SMB","HML","RF"))

FF_mom <- fread("F-F_Momentum_Factor.TXT", skip = 13)
setnames(FF_mom,c("date","UMD"))

setkey(FF_mom,date)
setkey(FF3_fact,date)

FF_factors <- merge(FF3_fact,FF_mom)
FF_factors[,`:=`(year=date%/%100,month=date%%100)]
FF_factors[,date:=as.yearmon(paste(year,month,sep = "-"))]

# get the test assets
##################################################
# Read 6 maturity-sorted Fama bond portfolios
# (12 month intervals up to 5 years)
##################################################
library(RPostgres)

wrds <- dbConnect(Postgres(),
                  host='wrds-pgdata.wharton.upenn.edu',
                  port=9737,
                  user = rstudioapi::askForPassword("wrds username:"),
                  password = rstudioapi::askForPassword("wrds password:"),
                  sslmode='require',
```

```
                  dbname='wrds')

res <- dbSendQuery(wrds, "SELECT KYTREASNOX,MCALDT,TMEWRETD FROM crspa.tfz_mth_bp")
data <- dbFetch(res)
dbClearResult(res)
Fama_bonds <- as.data.table(data)
Fama_bonds[,mcaldt:=as.Date(mcaldt,format="%Y-%m-%d")]

####################################################
# Read 6 maturity-sorted Fama bond portfolios
# (12 month intervals up to 5 years)
####################################################
bond40 <- Fama_bonds[kytreasnox==2000040]  # 0-1  year
bond41 <- Fama_bonds[kytreasnox==2000041]  # 1-2  year
bond42 <- Fama_bonds[kytreasnox==2000042]  # 2-3  year
bond43 <- Fama_bonds[kytreasnox==2000043]  # 3-4  year
bond44 <- Fama_bonds[kytreasnox==2000044]  # 4-5  year
bond38 <- Fama_bonds[kytreasnox==2000038]  # 5-10 year

bond_port_12m <- copy(bond40)
bond_port_12m[,`:=`(port2_12m=bond41$tmewretd,port3_12m=bond42$tmewretd,
                    port4_12m=bond43$tmewretd,port5_12m=bond44$tmewretd,
                    port6_12m=bond38$tmewretd)]
setnames(bond_port_12m,"tmewretd","port1_12m")
bond_port_12m[,date:=as.yearmon(mcaldt)]
bond_port_12m[,`:=`(mcaldt=NULL,kytreasnox=NULL)]

##############################################################################
# Read 25 Portfolios Formed on Size and Book-to-Market (5 x 5) (Value-Weighted)
##############################################################################
FF25 <- fread("25_Portfolios_5x5.CSV",fill = TRUE,skip = 15,nrows = 1100)
FF25 <- FF25[,lapply(.SD, function(x) as.numeric(x))]
setnames(FF25,c("date","S1B1","S1B2","S1B3","S1B4","S1B5",
                "S2B1","S2B2","S2B3","S2B4","S2B5",
                "S3B1","S3B2","S3B3","S3B4","S3B5",
                "S4B1","S4B2","S4B3","S4B4","S4B5",
                "S5B1","S5B2","S5B3","S5B4","S5B5"))
FF25[,`:=`(year=date%/%100,month=date%%100)]
FF25[,date:=as.yearmon(paste(year,month,sep = "-"))]
FF25[,`:=`(year=NULL,month=NULL)]
#########################################################
# Read 10 Portfolios Formed on Momentum (Value-Weighted)
#########################################################
FFmom <- fread("10_Portfolios_Prior_12_2.txt",fill = TRUE, skip = 11,nrows = 1100-6)
setnames(FFmom,c("date","Mom1","Mom2","Mom3","Mom4",
                 "Mom5","Mom6","Mom7","Mom8","Mom9","Mom10"))
FFmom <- FFmom[,lapply(.SD, function(x) as.numeric(x))]
FFmom[,`:=`(year=date%/%100,month=date%%100)]
```

```r
FFmom[,date:=as.yearmon(paste(year,month,sep = "-"))]
FFmom[,`:=`(year=NULL,month=NULL)]


# merging factors
setkey(FF_factors,date)
factors <- copy(FF_factors)

# merging portfolios
setkey(FF25,date)
setkey(FFmom,date)
# express bond returns in percentages
bond_port_12m <- bond_port_12m[,lapply(.SD, function(x) x*100),by=date]
setkey(bond_port_12m,date)
bonds <- copy(bond_port_12m)
portfolios <- merge(FF25,FFmom)
portfolios <- merge(portfolios,bonds)

########################################
# Replicating results in AEM (LevFact) #
########################################
merged_AEM <- merge(factors,portfolios)
merged_AEM[,year_qtr:=yearqtr(date)]
setkey(merged_AEM,date,year_qtr)

# compute quarterly returns
merged_AEMqtr <- merged_AEM[,lapply(.SD, function(x) 100*(prod(1+(x/100))-1)),
                            by=.(year_qtr), .SDcols=-c("date","year","month")]

setkey(merged_AEMqtr,year_qtr)
setnames(AEM,"date","year_qtr")
setkey(AEM,year_qtr)
merged_AEMqtr <- merge(merged_AEMqtr,AEM)

tmp <- copy(merged_AEMqtr)
# annualize excess returns
merged_AEMqtr <- merged_AEMqtr[,lapply(.SD, function(x) 4*(x-RF)),
                               .SDcols = -c("MktRF","SMB","HML","RF",
                                            "UMD","LevFactor"),by=year_qtr]
merged_AEMqtr[,LevFactor:=4*tmp$LevFactor]
merged_meltAEM <- melt(merged_AEMqtr,id.vars=c("year_qtr","LevFactor"))

# run time-series regressions to get betas
tseries_AEM <- function(x){
  reg <- lm(value ~ LevFactor,data=x)
  coef(reg)[2]
}
```

```r
betas_AEM <- merged_meltAEM[,tseries_AEM(.SD),by=variable]
setnames(betas_AEM,"V1","beta")

###########################################################
# Fama-MacBeth Regressions: T corss-sectional regressions #
###########################################################
FM_reg_AEM <- copy(merged_meltAEM)
FM_reg_AEM[,LevFactor:=NULL]
setnames(FM_reg_AEM,c("date","Portfolio","ret"))

FM_beta_AEM <- copy(betas_AEM)
setnames(FM_beta_AEM,"variable","Portfolio")

FMB_AEM <- function(x){
  reg <- lm(ret ~ FM_beta_AEM$beta,data=x)
  data.frame(t(coef(reg)[1:2]))
}

lambda_AEM <- FM_reg_AEM[,FMB_AEM(.SD),by=.(date)]
setnames(lambda_AEM,c("date","constant","lambda_AEM"))
FM_results_AEM <- lambda_AEM[,.(alpha_hat = mean(constant),
                                lambda_AEMhat = mean(lambda_AEM),
                                se_alpha = sd(constant)/sqrt(length(date)),
                                se_lambdaAEM = sd(lambda_AEM)/sqrt(length(date)))]
FM_results_AEM[,`:=`(tstat_alpha=alpha_hat/se_alpha,tstat_lambdaAEM=lambda_AEMhat/se_lambdaAEM)]


##################################
# comparewith 2-pass regressions #
##################################
# 1 cross-sectional regression
avg_return <- merged_AEMqtr[,lapply(.SD, function(x) mean(x,na.rm=TRUE)),
                            .SDcols=-c("year_qtr","LevFactor")]
betas_AEM[,ret:=t(avg_return[1,])]
FMB_AEM <- lm(ret ~  beta , data=betas_AEM)
summary(FMB_AEM)

# correct standard error in the regression output: Fama-MacBeth standard errors
stargazer(FMB_AEM,type="text",
          se = list(c(FM_results_AEM$se_alpha, FM_results_AEM$se_lambdaAEM)),
          covariate.labels = c("LevFac"),align=TRUE,no.space = TRUE,
          digits = 2,omit.stat=c("LL","ser","f"))

##########################################
# get mean absolute pricing erros (MAPE) #
##########################################
MAPE_SizeBM_AEM <- mean(abs(FMB_AEM$residuals[1:25]))
MAPE_MOM_AEM <- mean(abs(FMB_AEM$residuals[26:35]))
```

```r
MAPE_bond_AEM <- mean(abs(FMB_AEM$residuals[36:41]))
MAPE_total_AEM <- mean(abs(FMB_AEM$residuals))
MAPE_AEM <- data.table(c("Size B/M","MOM","Bond","Total"),
                       c(MAPE_SizeBM_AEM,MAPE_MOM_AEM,MAPE_bond_AEM,MAPE_total_AEM))
setnames(MAPE_AEM,c("portfolios","MAPE"))
MAPE_AEM

###############################
# plot realized vs. predicted #
###############################
# get average returns
setnames(avg_return,c("port1_12m","port2_12m","port3_12m","port4_12m","port5_12m","port6_12m"),
         c("0-1yr","1-2yr","2-3yr","3-4yr","4-5yr","5-10yr"))
total <- avg_return[,rowMeans(.SD)]
size_BM <- avg_return[, rowMeans(.SD),
                      .SDcols=c("S1B1","S1B2","S1B3","S1B4","S1B5","S2B1",
                                "S2B2","S2B3","S2B4","S2B5","S3B1","S3B2",
                                "S3B3","S3B4","S3B5","S4B1","S4B2","S4B3",
                                "S4B4","S4B5","S5B1","S5B2","S5B3","S5B4","S5B5")]
mom <- avg_return[, rowMeans(.SD), .SDcols=c("Mom1","Mom2","Mom3","Mom4","Mom5",
                                             "Mom6","Mom7","Mom8","Mom9","Mom10")]
bond <- avg_return[, rowMeans(.SD), .SDcols=c("0-1yr","1-2yr","2-3yr","3-4yr","4-5yr","5-10yr")]

dt_AEM <- data.table(FMB_AEM$fitted.values)
dt_AEM[,`:=`(realized_ret=t(avg_return[1,]),names=names(avg_return))]
setnames(dt_AEM,"V1","predicted_ret")
dt_AEM[,`:=`(predicted_ret=predicted_ret,realized_ret=realized_ret)]

ggplot(dt_AEM,aes(x=predicted_ret,y=realized_ret)) +
  geom_abline(color=rgb(0,0.447,0.741), size = .8) + theme_bw() +
  geom_point(size = 1.8) +
  xlab("Predicted Returns (% per year)") + ylab("Realized Returns (% per year)") +
  scale_x_continuous(limits = c(-4,14),breaks = seq(-4,14,by = 2)) +
  scale_y_continuous(limits = c(-4,14),breaks = seq(-4,14,by = 2)) +
  geom_text(aes(label=names),hjust=.5, vjust=-0.5, size = 3) +
  theme(axis.text.x = element_text(size=11,face="plain"),
        axis.text.y = element_text(size=11,face="plain"),
        axis.title.x = element_text(size=11,face="plain"),
        axis.title.y = element_text(size=11,face="plain")) +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```