Financial Statistics
Homework 4

Chenghua Liu
liuch18@mails.tsinghua.edu.cn
Department of Computer Science
Tsinghua University

1 5.13

Consider the following portfolio optimization problem with a risk-free asset having return r_0 :

$$\min \alpha^{\mathrm{T}} \boldsymbol{\Sigma} \boldsymbol{\alpha}, \quad \text{s.t. } \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{\mu} + \left(1 - \boldsymbol{\alpha}^{\mathrm{T}} \mathbf{1}\right) r_0 = \mu$$

That is, we minimize the variance of the portfolio consisting of allocation vector α on risky assets with return vector μ and allocation $\left(1 - \boldsymbol{\alpha}^{\mathrm{T}} \mathbf{1}\right)$ on the risk-free bond with return r_0, subject to the constraint that the portfolio's expected return is μ.

(a) The optimal solution is

$$\boldsymbol{\alpha} = P^{-1} \left(\mu - r_0\right) \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_0$$

where $P = \mu_0^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_0$ is the squared Sharpe ratio, and $\boldsymbol{\mu}_0 = \boldsymbol{\mu} - r_0 \mathbf{1}$ is the vector of excess returns.

(b) The variance of this portfolio is $\sigma^2 = \left(\mu - r_0\right)^2 / P$.

(c) When $r_0 < \mu$, show that $r_0 + P^{1/2} \sigma = \mu$, namely, the optimal allocation for the risky asset α is the tangent portfolio.

Solution:

(a) According to method of Lagrange multipliers, we have Lagrangian function:

$$\min \frac{1}{2} \alpha^T \Sigma \alpha + \lambda \left(\mu - \alpha^T \vec{\mu} - \left(1 - \alpha^T \hat{1}\right) r_0\right)$$

$$\frac{\partial L}{\partial \alpha} = 0 \Rightarrow \Sigma \alpha^* - \lambda \left(\vec{u} - r_0 \vec{1}\right) = 0 \Rightarrow \alpha^* = \lambda \Sigma^{-1} \vec{\mu_0}, \quad \text{where } \vec{\mu_0} = \vec{\mu} - r_0 \vec{1}$$

Then,

$$
\begin{aligned}
&\alpha^{*T}\vec{\mu}+\left(1-\alpha^{*}\vec{1}\right)r_0=\mu\\
\Rightarrow&\lambda\mu_0^T\Sigma^{-1}\vec{\mu}-\lambda\mu_0^T\Sigma^{-1}r_0\vec{1}=\mu-r_0\\
\Rightarrow&\lambda\mu_0^T\Sigma^{-1}\mu_0=\mu-r_0\\
\Rightarrow&\lambda=P^{-1}\left(\mu-r_0\right),\text{where}P=\mu_0^T\Sigma^{-1}\mu_0
\end{aligned}
$$

So,

$$
\alpha^{*}=P^{-1}\left(\mu-r_0\right)\Sigma^{-1}\vec{\mu_0},\quad \text{where } \vec{\mu_0}=\vec{\mu}-r_0\vec{1},P=\mu_0^T\Sigma^{-1}\mu_0
$$

(b) The variance of this portfolio is

$$
{\alpha^{*}}^{T}\Sigma\alpha^{*}=P^{-2}\left(\mu-r_0\right)^2\mu_0^T\Sigma^{-1}\Sigma\Sigma^{-1}\mu_0=P^{-2}\left(\mu-r_0\right)^2P=\left(\mu-r_0\right)^2/P
$$

(c)

$$
r_0+P^{1/2}\sigma=r_0+P^{\frac{1}{2}}\cdot\left(\mu-r_0\right)/P^{\frac{1}{2}}=r_0+\mu-r_0=\mu
$$

■

2 5.14

Show that given $\boldsymbol{\Sigma}_0$ and $\boldsymbol{B}_0,\hat{\gamma}_0$ given by (5.59) is the maximum likelihood estimator.

(a) Download the monthly data of 8 stocks: Dell, Ford, GE, IBM, Intel Johnson & Johnson, Merck, Microsoft from January 2006 to December 2019 . Use the threemonth treasury bill rates as a proxy for the risk-free rate and the S&P 500 index as a proxy of the market portfolio.

(b) Construct the optimal allocations of the 8 stocks, if an investor is willing to invest 20% in riskless asset, using the the monthly data between 2006 and 2016 .

(c) If the allocation is fixed over the next two years (invested in December 2016), compare the performance of the portfolio over the next 6 -month, one-year, two-year and three-year with the S&P 500 stock, in terms of return, volatility (standard deviation), and Sharpe ratio.

(d) Create a value-weighted portfolio of the 8 stocks using the data in year 2016 . As a proxy, the weight for Dell computer, for example, is proportional to the sum of the volume times closing price (un-adjusted) over the year. Report the percentage of allocation, if 20% of the asset is allocated to the 3 -month treasury bills. Compare the performance of the portfolio over the next 6 -month, one-year two-year, and three-year with the S&P 500 stock, in terms of gain, volatility (standard deviation) and Sharpe ratio.

(e) Create a portfolio with 20% invested on risk-free bond and 10% over each of the 8 stocks. Compare the performance of the portfolio over the last 6 -month, oneyear, two-year and three-year with the S&P 500 stock, in terms of gain, volatility (standard deviation) and Sharpe ratio.

Solution:

(a)

```
library(tidyquant)
library(quantmod)
library(zoo)
library(xts)
library(tidyverse)

options("getSymbols.warning4.0"=FALSE)
options("getSymbols.yahoo.warning"=FALSE)
tickers=c("F","GE","IBM","INTC","JNJ","MRK","MSFT")

stock=tq_get(tickers,from="2006-01-01",to="2019-12-31",get = "stock.prices",
periodicity="monthly")
IRX=tq_get("^IRX",from="2006-01-01",to="2019-12-31",periodicity="monthly")
GSPC=tq_get("^GSPC",from="2006-01-01",to="2019-12-31",get = "stock.prices",
periodicity="monthly")

F_monthly_return<-subset(stock,stock$symbol=="F")%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
GE_monthly_return<-subset(stock,stock$symbol=="GE")%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
IBM_monthly_return<-subset(stock,stock$symbol=="IBM")%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
INTC_monthly_return<-subset(stock,stock$symbol=="INTC")%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
JNJ_monthly_return<-subset(stock,stock$symbol=="JNJ")%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
MRK_monthly_return<-subset(stock,stock$symbol=="MRK")%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
MSFT_monthly_return<-subset(stock,stock$symbol=="MSFT")%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
MKT_monthly_return<-GSPC%>%tq_transmute(
select=adjusted,mutate_fun=periodReturn,period="monthly",col_rename="returns"
)
RF_monthly_return<-data.frame(date=IRX$date,returns=(1+IRX$adjusted*0.01)**(1/3)-1)

list_of_dataframe<-list(MKT_monthly_return,RF_monthly_return,F_monthly_return,
```

```
GE_monthly_return,IBM_monthly_return,INTC_monthly_return,JNJ_monthly_return,
MRK_monthly_return,MSFT_monthly_return)
for(i in list_of_dataframe){
        i=data.frame(i)
}
returns=list_of_dataframe%>%reduce(left_join,by="date")
names(returns)=c("date","MKT","RF",tickers)
returns=data.frame(returns)
returns=xts(returns[,2:10],order.by = as.Date(returns[,1]))
##############################################
> head(returns)
                     MKT         RF           F           GE          IBM        INTC
2006-01-01  0.000000e+00 0.01435948  0.00000000  0.000000000  0.000000000  0.00000000
2006-02-01  4.531576e-04 0.01480312 -0.06019291  0.003664019 -0.013038053 -0.03104431
2006-03-01  1.109581e-02 0.01482901 -0.00125450  0.066030177  0.030371948 -0.05086257
2006-04-01  1.215565e-02 0.01526577 -0.12688433 -0.005462619 -0.001575803  0.02672166
2006-05-01 -3.091692e-02 0.01547593  0.04472582 -0.009540625 -0.029633359 -0.09809835
2006-06-01  8.659623e-05 0.01594442 -0.03212298 -0.037945214 -0.035072120  0.05979404
                    JNJ         MRK        MSFT
2006-01-01  0.000000000  0.00000000  0.00000000
2006-02-01  0.001911540  0.01043509 -0.04547045
2006-03-01  0.033052224  0.01061386  0.01608505
2006-04-01 -0.010300582 -0.01223679 -0.11245873
2006-05-01  0.027469368 -0.03282973 -0.06211143
2006-06-01  0.001194456  0.10701650  0.03270866
```

(b)

```
library(PerformanceAnalytics)
returns_train=returns[1:132,]
beta=rep(0,7)
for(i in 1:7){
        beta[i]=CAPM.beta(Ra=returns_train[,i+2],Rb=returns_train$MKT,
        Rf=returns_train$RF)
}
betas=data.frame(company=tickers,beta=beta)
##############################################
> betas
  company      beta
1       F 2.0067189
2      GE 1.5069746
3     IBM 0.7191471
4    INTC 1.0879984
5     JNJ 0.6058957
6     MRK 0.6993596
7    MSFT 1.0428936
##############################################
mean=apply(returns_train,2,mean)
return_mean=data.frame(type=c("MKT","RF",tickers),mean=mean)
##############################################
> return_mean
     type         mean
```

```
MKT    MKT 0.005150767
RF      RF 0.003306431
F        F 0.015242202
GE      GE 0.006018579
IBM    IBM 0.008606097
INTC  INTC 0.009073959
JNJ    JNJ 0.008630123
MRK    MRK 0.009190248
MSFT  MSFT 0.010446560
> cov(returns_train[,3:9])
                 F          GE          IBM        INTC          JNJ           MRK        MSFT
F     2.622379e-02 0.006012876 0.0027382958 0.003438552 0.0013310426 -3.569684e-05 0.003830423
GE    6.012876e-03 0.006636545 0.0018372736 0.002693537 0.0017344962  1.392713e-03 0.002793171
IBM   2.738296e-03 0.001837274 0.0028476806 0.001847870 0.0007597348  6.700513e-04 0.001259756
INTC  3.438552e-03 0.002693537 0.0018478699 0.004958638 0.0012966946  1.619087e-03 0.002837282
JNJ   1.331043e-03 0.001734496 0.0007597348 0.001296695 0.0016742244  1.183007e-03 0.001064940
MRK  -3.569684e-05 0.001392713 0.0006700513 0.001619087 0.0011830070  3.874579e-03 0.001544072
MSFT  3.830423e-03 0.002793171 0.0012597556 0.002837282 0.0010649399  1.544072e-03 0.005061665

##########################################
a=solve(cov(returns_train[,3:9]))%*%as.matrix(return_mean$mean[3:9]-mean[2])
A=sum(a)/0.8
alpha=1/A*a
##########################################
> alpha
           [,1]
F     0.07572213
GE   -0.27195028
IBM   0.22535875
INTC -0.05996150
JNJ   0.53646262
MRK   0.12311490
MSFT  0.17125339
> sum(alpha)
[1] 0.8
```

(c)

```
returns_test = returns[133:168,]
range = c(6,12,24,36)
# Portfolio
e_return = rep(0,4)
e_sd = rep(0,4)
e_spr = rep(0,4)
# S&P 500
sp_return = rep(0,4)
sp_sd = rep(0,4)
sp_spr = rep(0,4)
for(i in 1:4){
        mean = apply(returns_test[1:range[i]],2,mean)
        e_return[i] = t(as.matrix(alpha)) %*% as.matrix(mean[3:9])+mean[2]*0.2
        sp_return[i] = mean[1]
```

```
        cov = cov(returns_train[1:range[i],3:9])
        e_sd[i] = (t(as.matrix(alpha)) %*% cov %*% as.matrix(alpha))**0.5
        sp_sd[i] = (var(returns_test[1:range[i],1]))**0.5
        e_spr[i] = (e_return[i]-mean[2])/e_sd[i]
        sp_spr[i] = (sp_return[i]-mean[2])/sp_sd[i]
}
comparison = data.frame(e_return = e_return,sp500_return = sp_return,
e_sd=e_sd, sp500_sd = sp_sd,
e_sharpe = e_spr, sp500_sharpe = sp_spr)
##########################################
> comparison
    e_return sp500_return        e_sd    sp500_sd  e_sharpe sp500_sharpe
1 0.02426612   0.013362426 0.01950032 0.01320765 1.1169858   0.82360485
2 0.02642580   0.014956266 0.02867411 0.01121747 0.8128836   1.05542390
3 0.02136367   0.005260406 0.03135997 0.03302727 0.5289924   0.01471286
4 0.01772334   0.010842765 0.03846013 0.03494036 0.3206201   0.15599474
```

Our portfolio has higher return and lower volatility than S&P 500 .

(d)

```

returns_2016 = returns[121:132]
volume = c(8997474600,9311752320,1018605700,5763208600,1903010600,2566389500,7819726800)
alpha_d = 0.8/sum(volume)*volume
##########################################
> alpha_d
[1] 0.19256146 0.19928754 0.02179992 0.12334259 0.04072771 0.05492516 0.16735563
##########################################
returns_test = returns[133:168,]
range = c(6,12,24,36)
# Portfolio
e_return = rep(0,4)
e_sd = rep(0,4)
e_spr = rep(0,4)
# S&P 500
sp_return = rep(0,4)
sp_sd = rep(0,4)
sp_spr = rep(0,4)
for(i in 1:4){
        mean = apply(returns_test[1:range[i]],2,mean)
        e_return[i] = t(as.matrix(alpha_d)) %*% as.matrix(mean[3:9])+mean[2]*0.2
        sp_return[i] = mean[1]
        cov = cov(returns_train[1:range[i],3:9])
        e_sd[i] = (t(as.matrix(alpha_d)) %*% cov %*% as.matrix(alpha_d))**0.5
        sp_sd[i] = (var(returns_test[1:range[i],1]))**0.5
        e_spr[i] = (e_return[i]-mean[2])/e_sd[i]
        sp_spr[i] = (sp_return[i]-mean[2])/sp_sd[i]
}
comparison_d = data.frame(e_return = e_return,sp500_return = sp_return,
e_sd=e_sd, sp500_sd = sp_sd,
e_sharpe = e_spr, sp500_sharpe = sp_spr)
##########################################
```

```
> print(comparison_d)
      e_return sp500_return       e_sd   sp500_sd      e_sharpe sp500_sharpe
1 -0.001636302   0.013362426 0.02004036 0.01320765 -0.2056271748   0.82360485
2  0.001861933   0.014956266 0.03123078 0.01121747 -0.0401894785   1.05542390
3 -0.005290343   0.005260406 0.03352992 0.03302727 -0.3001743802   0.01471286
4  0.005373999   0.010842765 0.05194068 0.03494036 -0.0003514267   0.15599474
> apply(returns_test[1:12],2,mean)
          MKT            RF             F            GE           IBM          INTC
 1.495627e-02  3.117082e-03  7.630900e-03 -4.457045e-02 -2.493257e-03  2.450783e-02
          JNJ           MRK          MSFT
 1.884002e-02 -3.080783e-05  2.938240e-02
> apply(returns_test[1:24],2,mean)
         MKT           RF            F           GE          IBM         INTC          JNJ
 0.005260406  0.004774480 -0.012164534 -0.052816086 -0.009983871  0.014527938  0.007934645
         MRK         MSFT
 0.014879232  0.023352064
```

The return is poorer than S&P 500 and the portfolio has higher volatility.

(e)

```
alpha_e = rep(0.8/7,7)
returns_test = returns[133:168,]
range = c(6,12,24,36)
# Portfolio
e_return = rep(0,4)
e_sd = rep(0,4)
e_spr = rep(0,4)
# S&P 500
sp_return = rep(0,4)
sp_sd = rep(0,4)
sp_spr = rep(0,4)
for(i in 1:4){
        mean = apply(returns_test[1:range[i]],2,mean)
        e_return[i] = t(as.matrix(alpha_e)) %*% as.matrix(mean[3:9])+mean[2]*0.2
        sp_return[i] = mean[1]
        cov = cov(returns_train[1:range[i],3:9])
        e_sd[i] = (t(as.matrix(alpha_e)) %*% cov %*% as.matrix(alpha_d))**0.5
        sp_sd[i] = (var(returns_test[1:range[i],1]))**0.5
        e_spr[i] = (e_return[i]-mean[2])/e_sd[i]
        sp_spr[i] = (sp_return[i]-mean[2])/sp_sd[i]
}
comparison_e = data.frame(e_return = e_return,sp500_return = sp_return,
e_sd=e_sd, sp500_sd = sp_sd,
e_sharpe = e_spr, sp500_sharpe = sp_spr)
###############################################
> print(comparison_e)
      e_return sp500_return       e_sd   sp500_sd    e_sharpe sp500_sharpe
1  0.002142186  0.013362426 0.01781731 0.01320765 -0.01921465   0.82360485
2  0.004425318  0.014956266 0.02762175 0.01121747  0.04736252   1.05542390
3 -0.000676031  0.005260406 0.03033464 0.03302727 -0.17967946   0.01471286
4  0.007086275  0.010842765 0.04661317 0.03494036  0.03634216   0.15599474
```

The expected return is poorer than S&P 500 index.But the portfolio has largest volatility in all situations.

■

3 6.3

Let $\boldsymbol{Y}_t$ be a vector of excess returns of N assets. Consider the multivariate linear regression model

$$\boldsymbol{Y}_t = \boldsymbol{\alpha} + \boldsymbol{\beta} Y_t^m + \varepsilon_t$$

where $\varepsilon_t \sim N(0, \boldsymbol{\Sigma})$ and $\operatorname{cov}(Y_t^m, \varepsilon_t) = 0$.

(a) Derive the maximum likelihood estimators for α and $\boldsymbol{\beta}$. (You do not need to derive the MLE for Σ, since this part is hard; you just take for granted that $\hat{\Sigma}$ is the MLE).

(b) Show that the maximum likelihood ratio test for the null hypothesis: $H_0 : \boldsymbol{\alpha} = 0$ is

$$T_2 = T\left[\log(|\hat{\boldsymbol{\Sigma}}_0|) - \log(|\hat{\boldsymbol{\Sigma}}|)\right]$$

where $\hat{\Sigma}_0$ is the MLE under H_0. Give explicitly the expression for $\hat{\boldsymbol{\Sigma}}_0$.

Solution:

(a) The likelihood function is

$$\begin{aligned} f\left(Y_1, \ldots, Y_T \mid Y_1^m, \ldots Y_T^m\right) &= \prod_{t=1}^{T} f\left(Y_t \mid Y_t^m\right) \\ &= \prod_{t=1}^{T} (2\pi)^{-\frac{N}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left(Y_t - \alpha - \beta Y_t^m\right)^T \Sigma^{-1}\left(Y_t - \alpha - \beta Y_t^m\right)\right] \end{aligned}$$

Then we get the log-likelihood function

$$l(\alpha, \beta, \Sigma) = -\frac{NT}{2}\log(2\pi) - \frac{T}{2}\log|\Sigma| - \frac{1}{2}\sum_{i=1}^{T}\left(Y_t - \alpha - \beta Y_t^m\right)^T \Sigma^{-1}\left(Y_t - \alpha - \beta Y_t^m\right)$$

One trick is that

$$\begin{aligned} \sum_{i=1}^{T}\left(Y_t - \alpha - \beta Y_t^m\right)^T \Sigma^{-1}\left(Y_t - \alpha - \beta Y_t^m\right) &= \sum_{i=1}^{T} tr\left(\left(Y_t - \alpha - \beta Y_t^m\right)^T \Sigma^{-1}\left(Y_t - \alpha - \beta Y_t^m\right)\right) \\ &= \sum_{i=1}^{T} tr\left(\varepsilon_t^T \Sigma^{-1} \varepsilon_t\right) = \sum_{i=1}^{T} tr\left(\Sigma^{-1}\varepsilon_t \varepsilon_t^T\right) \end{aligned}$$

Then we drive the partial derivative and let them be zero

$$\left\{\begin{array}{l} \frac{\partial l}{\partial \alpha} = \sum_{i=1}^{T}(\Sigma^{-1}(Y_t - \alpha - \beta Y_t^m)) = 0 \\ \frac{\partial L}{\partial \beta} = \sum_{t=1}^{T}(\Sigma^{-1}(Y_t - \alpha - \beta Y_t^m) Y_t^m) = 0 \end{array}\right. \Rightarrow \left\{\begin{array}{l} \hat{\alpha} = \bar{Y} - \hat{\beta}\bar{Y}_m \\ \hat{\beta} = \frac{\sum_{i=1}^{T}\left(Y_t - \bar{Y}\right)\left(Y_t^m - \bar{Y}_m\right)}{\sum_{i=1}^{T}\left(Y_t^m - \bar{Y}_m\right)^2} \end{array}\right.$$

where$\bar{Y} = \frac{1}{T}\sum_{t=1}^{T} Y_t \quad \bar{Y}_m = \frac{1}{T}\sum_{i=1}^{T} Y_t^m$

(b) Under $H_0 : \alpha = 0$,

$$\begin{aligned}
f\left(Y_1, \ldots, Y_T \mid Y_1^m, \ldots Y_T^m\right) &= \prod_{t=1}^{T} f\left(Y_t \mid Y_t^m\right) \\
&= \prod_{t=1}^{T} (2\pi)^{-\frac{N}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left(Y_t - \beta Y_t^m\right)^T \Sigma^{-1}\left(Y_t - \beta Y_t^m\right)\right]
\end{aligned}$$

and the log-likelihood function

$$l(\beta, \Sigma) = -\frac{NT}{2}\log(2\pi) - \frac{T}{2}\log|\Sigma| - \frac{1}{2}\sum_{i=1}^{T}\left(Y_t - \beta Y_t^m\right)^T \Sigma^{-1}\left(Y_t - \beta Y_t^m\right)$$

We drive the partial derivative and let it be zero

$$\frac{\partial l}{\partial \beta} = \sum_{i=1}^{T}\left(\Sigma^{-1}\left(Y_t - \beta Y_t^m\right) Y_t^m\right) = 0 \quad \Rightarrow \quad \hat{\beta} = \frac{\sum_{i=1}^{T} Y_t Y_t^m}{\sum_{t=1}^{1}\left(Y_t^m\right)^2}$$

Then ,we get

$$\left|\hat{\Sigma}_0\right| = \frac{\sum_{t=1}^{T}\left(Y_t - \hat{\beta} Y_t^m\right)\left(Y_t - \hat{\beta} Y_t^m\right)^T}{T}$$

Use the trick mentioned in (a) again

$$\begin{aligned}
\sum_{t=1}^{T} \hat{\varepsilon}_t^T \hat{\Sigma}^{-1} \hat{\varepsilon}_t =& \sum_{t=1}^{T} \operatorname{tr}\left(\hat{\varepsilon}_t^T \hat{\Sigma}^{-1} \hat{\varepsilon}_t\right) = \sum_{i=1}^{T} \operatorname{tr}\left(\hat{\Sigma}^{-1} \hat{\varepsilon}_t \hat{\varepsilon}_t^T\right) \\
=& tr\left[\hat{\Sigma}^{-1}\left(\sum_{t=1}^{T} \hat{\varepsilon}_t \hat{\varepsilon}_t^T\right)\right] = tr\left[\left(\frac{1}{T}\sum_{t=1}^{T} \hat{\varepsilon}\hat{\varepsilon}_t^T\right)^{-1}\left(\sum_{t=1}^{T} \hat{\varepsilon}_t \hat{\varepsilon}_t^T\right)\right] = NT
\end{aligned}$$

Then the log-likelihood function can be simplified to

$$l(\beta, \Sigma) = -\frac{NT}{2}\log(2\pi) - \frac{T}{2}\log|\Sigma| - \frac{NT}{2}$$

Thus,

$$T_2 = 2\left\{\max \ell - \max_{\mathrm{H}_0} \ell\right\} = T\left(\log\left|\hat{\Sigma}_0\right| - \log\left|\hat{\Sigma}\right|\right) \overset{a}{\sim} \chi_N^2$$

■

4 6.4

Consider the multi-factor model

$$\boldsymbol{Y}_t = \boldsymbol{a} + \boldsymbol{B}\boldsymbol{X}_t + \varepsilon_t$$

with observable factor $\boldsymbol{X}_t$, where $E\varepsilon_t = 0$ and $\operatorname{cov}\left(\boldsymbol{X}_t, \varepsilon_t\right) = 0$

(a) Based on 20 stock portfolios over a period of 60 months on the three factors, it was computed that $\left|\hat{\Sigma}_0\right| = 2.375$ and $|\hat{\Sigma}| = 1.624$. Test if the multifactor model is consistent with the empirical data, i.e. $H_0 : a = 0$.

(b) Suppose that the beta's of the GE stock over the S&P 500 index (X_1), the size effect X_2 and book-to-market effect X_3 are respectively $1.3, 0.3$ and -0.4. Assume further that over the last 10 years the average risk free interest is 4%, the average return of the S&P 500 is 11%, the average difference of returns between the small large capitalization is 3%, and the average difference of returns between the high and low book-to-market is 2%, what is the expected return of the GE stock using the Fama-French model?

Solution:

(a) Under H_0,

$$T_0 = \left(T - \frac{N}{2} - k - 1\right)\left(\log\left|\hat{\Sigma}_0\right| - \log|\Sigma|\right) \overset{a}{\sim} \chi^2_N$$

So we have $T_0^* = \left(60 - \frac{20}{2} - 3 - 1\right)\left(\log\frac{2.375}{1.624}\right) = 17.485$ and $P - value = 0.379$. So we can't reject H_0 at 1% significance level.

(b) $EY_t = 1.3 \times (0.11 - 0.04) + 0.3 \times 0.03 - 0.4 \times 0.02 = 0.092$. Therefore, the expected return of the GE stock using the Fama-French model is 13.2%.

■