Chapter 8 Homework

## Answer Key

pacman::p\_load(pacman, tidyverse, knitr, here, psych, tseries)  
knitr::opts\_chunk$set(prompt = F, message = F)

#### Prob. 8.3.

Calculate five years of monthly excess returns ending Dec. 31, 2015 on these five stocks:

* Papa John’s International, PZZA
* Bed, Bath, & Beyond, BBBY
* Netflix, Inc. NFLX
* ~~Time Warner, Inc. TWX~~ substitute AT&T, T
* Verizon Communications, Inc. VZ

For the risk-free rate, use the three-month Treasury bill. Use the S&P 500 index, ^GSPC, as the market index.

**Required**: Estimate the beta for each stock. Which stock is most sensitive?

##### Solution

* The excess returns of the five stocks. Use the data developed in Chapter 6’s homework. (Note: the output incorporates the risk-free rate; so, there’s no need to pull that data again.)

d2 <- read\_csv("fiveStocks.csv")  
d2

## # A tibble: 60 x 5  
## PZZA BBBY NFLX ATT VZ  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0361 -0.0234 0.218 -0.0634 -0.00460  
## 2 0.0167 0.00312 -0.0347 0.0461 0.0500   
## 3 0.0853 0.00249 0.150 0.0785 0.0438   
## 4 -0.0508 0.163 -0.0215 0.0166 -0.0198   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994  
## 6 0.00392 0.0831 -0.0300 -0.00479 0.00809  
## 7 -0.0616 0.00206 0.0125 -0.0685 -0.0521   
## 8 -0.0468 -0.0279 -0.116 -0.0133 0.0383   
## 9 0.0218 0.00791 -0.518 0.00140 0.0174   
## 10 0.111 0.0790 -0.275 0.0277 0.00487  
## # ... with 50 more rows

* The risk-free rate

dat <- read\_csv("Yield2015.csv")  
  
# extract the vector of data we need  
rff <- dat$yield  
  
# convert percentages to proportional monthly returns  
rfree <- (1+rff/100)^(1/12) - 1  
head(rfree)

## [1] 1.249141e-04 1.082688e-04 8.329516e-05 4.998626e-05 3.332722e-05  
## [6] 3.332722e-05

* The market index. Use the S&P500 Index, ^GSPC

# library(tseries)  
x <- get.hist.quote(instrument = "^GSPC",  
 start = "2010-12-01",  
 end = "2015-12-31",  
 quote = "AdjClose",  
 compression = "m")

## time series ends 2015-12-01

sp500 <- as.vector(x)  
n <- length(sp500)  
  
# Net returns  
sp500\_ret <- (sp500[-1] - sp500[-n])/sp500[-n]  
  
# add to d2 data frame  
d2 <- d2 %>%  
 mutate(SP500 = sp500\_ret - rfree)  
d2

## # A tibble: 60 x 6  
## PZZA BBBY NFLX ATT VZ SP500  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0361 -0.0234 0.218 -0.0634 -0.00460 0.0225   
## 2 0.0167 0.00312 -0.0347 0.0461 0.0500 0.0318   
## 3 0.0853 0.00249 0.150 0.0785 0.0438 -0.00113  
## 4 -0.0508 0.163 -0.0215 0.0166 -0.0198 0.0284   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994 -0.0135   
## 6 0.00392 0.0831 -0.0300 -0.00479 0.00809 -0.0183   
## 7 -0.0616 0.00206 0.0125 -0.0685 -0.0521 -0.0215   
## 8 -0.0468 -0.0279 -0.116 -0.0133 0.0383 -0.0568   
## 9 0.0218 0.00791 -0.518 0.00140 0.0174 -0.0718   
## 10 0.111 0.0790 -0.275 0.0277 0.00487 0.108   
## # ... with 50 more rows

stk4 <- as.matrix(d2[-6])  
SP500 <- d2$SP500  
stk4\_betas <- lm(stk4 ~ SP500)$coefficients[2,]  
stk4\_betas

## PZZA BBBY NFLX ATT VZ   
## 0.2172923 0.6509977 1.5662885 0.2953731 0.2575223

Netflix is the most sensitive of the five; Papa John’s is the least sensitive.

#### Prob. 8.6

For the return data on the five stocks given in Exercise 3, use the shrinkage method to estimate the values of beta in the market model with the S&P 500 index as the market index. Compute the estimates using a global value of the weight , as in Example 8.8, and then repeat the analysis using an asset-specific value of , as in Example 8.9.

Estimate of the betas: (See Prob. 8.3)

stk4\_betas <- lm(stk4 ~ SP500)$coefficients[2,]  
stk4\_betas

## PZZA BBBY NFLX ATT VZ   
## 0.2172923 0.6509977 1.5662885 0.2953731 0.2575223

The quantities for and

beta\_bar <- mean(stk4\_betas)  
tausq\_beta <- mean( (stk4\_betas - beta\_bar)^2)  
tibble(beta\_bar, tausq\_beta)

## # A tibble: 1 x 2  
## beta\_bar tausq\_beta  
## <dbl> <dbl>  
## 1 0.597 0.259

Standard errors of for the five stocks:

f\_betase <- function(y){  
 summary(lm(y~SP500))$coefficients[2,2]  
 }  
  
stk4\_betase <- apply(stk4,2,f\_betase)  
stk4\_betase

## PZZA BBBY NFLX ATT VZ   
## 0.2747684 0.2493602 0.7923553 0.1446296 0.1614159

and is given by

sesq\_bar <- mean(stk4\_betase^2)  
tibble(sesq\_bar) %>% kable()

|  |
| --- |
| sesq\_bar |
| 0.1624956 |

It follows that the weight psi using in the shrinkage estimator is given by

stk4\_1psi <- sesq\_bar/(sesq\_bar + tausq\_beta)  
tibble(stk4\_1psi) %>% kable()

|  |
| --- |
| stk4\_1psi |
| 0.3859176 |

and the shrinkage estimates of beta for the four stocks are given by

stk4\_1psi \* beta\_bar + (1 - stk4\_1psi)\*stk4\_betas

## PZZA BBBY NFLX ATT VZ   
## 0.3640191 0.6303500 1.1924140 0.4119672 0.3887236

##### Now, repeat the analysis wiht an asset specific value of the weight .

Let

Then an alternative shrinkage estimator of is given by

Compute shrinkage estimates of beta for all four stocks.

stk4\_psi <- (stk4\_betase^2) / (tausq\_beta + stk4\_betase^2)  
stk4\_psi

## PZZA BBBY NFLX ATT VZ   
## 0.22599695 0.19386103 0.70829310 0.07484384 0.09154269

stk4\_psi\*beta\_bar + (1-stk4\_psi)\*stk4\_betas

## PZZA BBBY NFLX ATT VZ   
## 0.3032169 0.6406256 0.8800986 0.3179851 0.2886443

#### Prob. 8.10

Consider three mutual funds, Copley (symbol COPLX), Edgewood Growth Retail (EGFFX), and ~~Fidelity Contrafund~~ Fidelity New Millennium Fund (FMILX). Using five years of monthly excess returns for the period ending December 31, 2015, and the S&P 500 index as the market index, determine which of these three funds is the most diversified. Justify your answer.

First, compute the risk-free rate

dat <- read\_csv("Yield2015.csv")  
  
# extract the vector of data we need  
rff <- dat$yield  
  
# convert percentages to proportional monthly returns  
rfree <- (1+rff/100)^(1/12) - 1  
head(rfree)

## [1] 1.249141e-04 1.082688e-04 8.329516e-05 4.998626e-05 3.332722e-05  
## [6] 3.332722e-05

Copley, COPLX

X <- get.hist.quote(instrument = "COPLX",  
 start = "2010-12-01",  
 end = "2015-12-31",  
 quote = "AdjClose",  
 compression = "m") # monthly

## time series ends 2015-12-01

coplx\_m <- as.vector(X)  
n <- length(coplx\_m)  
  
coplx\_ret <- (coplx\_m[-1] - coplx\_m[-n]) / coplx\_m[-n]  
  
d3 <- tibble(COPLX = coplx\_ret - rfree)  
d3

## # A tibble: 60 x 1  
## COPLX  
## <dbl>  
## 1 0.0158   
## 2 0.0222   
## 3 0.00986  
## 4 0.0190   
## 5 0.00249  
## 6 -0.00674  
## 7 -0.0129   
## 8 0.00319  
## 9 -0.0181   
## 10 0.0466   
## # ... with 50 more rows

Edgewood Growth Retail (EGFFX)

X <- get.hist.quote(instrument = "EGFFX",  
 start = "2010-12-01",  
 end = "2015-12-31",  
 quote = "AdjClose",  
 compression = "m") # monthly

## time series ends 2015-12-01

egffx\_m <- as.vector(X)  
n <- length(egffx\_m)  
  
egffx\_ret <- (egffx\_m[-1] - egffx\_m[-n]) / egffx\_m[-n]  
  
d3 <- d3 %>%  
 mutate(EGFFX = egffx\_ret - rfree)   
d3

## # A tibble: 60 x 2  
## COPLX EGFFX  
## <dbl> <dbl>  
## 1 0.0158 0.0171   
## 2 0.0222 0.0435   
## 3 0.00986 0.00760   
## 4 0.0190 0.0186   
## 5 0.00249 -0.0125   
## 6 -0.00674 -0.000875  
## 7 -0.0129 -0.00425   
## 8 0.00319 -0.0499   
## 9 -0.0181 -0.0615   
## 10 0.0466 0.102   
## # ... with 50 more rows

Fidelity New Millennium Fund (FMILX)

X <- get.hist.quote(instrument = "FMILX",  
 start = "2010-12-01",  
 end = "2015-12-31",  
 quote = "AdjClose",  
 compression = "m") # monthly

## time series ends 2015-12-01

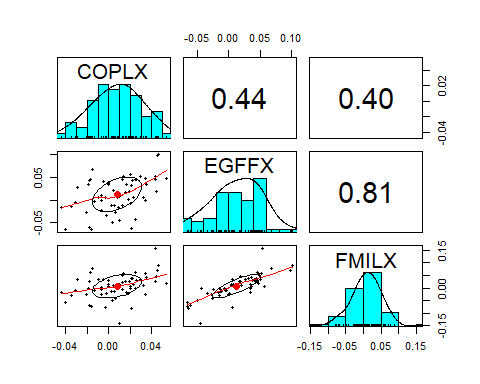
fmilx\_m <- as.vector(X)  
n <- length(fmilx\_m)  
  
fmilx\_ret <- (fmilx\_m[-1] - fmilx\_m[-n]) / fmilx\_m[-n]  
  
d3 <- d3%>%  
 mutate(FMILX = fmilx\_ret - rfree)   
d3 %>% round(4)

## # A tibble: 60 x 3  
## COPLX EGFFX FMILX  
## <dbl> <dbl> <dbl>  
## 1 0.0158 0.0171 0.0282  
## 2 0.0222 0.0435 0.0385  
## 3 0.0099 0.0076 0.0145  
## 4 0.019 0.0186 0.0273  
## 5 0.0025 -0.0125 -0.0118  
## 6 -0.0067 -0.0009 -0.0232  
## 7 -0.0129 -0.0042 -0.0097  
## 8 0.0032 -0.0499 -0.0487  
## 9 -0.0181 -0.0615 -0.0695  
## 10 0.0466 0.102 0.0883  
## # ... with 50 more rows

##### Exploratory data analysis

Optional for the assignment, but a good practice.

pairs.panels(d3)



Regress the funds on the market index

funds3 <- as.matrix(d3)  
funds3\_mm <- lm(funds3 ~ SP500)  
summary(funds3\_mm)

## Response COPLX :  
##   
## Call:  
## lm(formula = COPLX ~ SP500)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.044725 -0.009025 0.003531 0.011601 0.039643   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.004027 0.002391 1.684 0.0975 .   
## SP500 0.427507 0.069143 6.183 6.79e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.01794 on 58 degrees of freedom  
## Multiple R-squared: 0.3973, Adjusted R-squared: 0.3869   
## F-statistic: 38.23 on 1 and 58 DF, p-value: 6.791e-08  
##   
##   
## Response EGFFX :  
##   
## Call:  
## lm(formula = EGFFX ~ SP500)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.064007 -0.009303 -0.001174 0.010348 0.066623   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.004325 0.002670 1.62 0.111   
## SP500 0.973858 0.077208 12.61 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02003 on 58 degrees of freedom  
## Multiple R-squared: 0.7328, Adjusted R-squared: 0.7282   
## F-statistic: 159.1 on 1 and 58 DF, p-value: < 2.2e-16  
##   
##   
## Response FMILX :  
##   
## Call:  
## lm(formula = FMILX ~ SP500)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.122923 -0.008655 0.003542 0.010913 0.106912   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.0009427 0.0042557 -0.222 0.825   
## SP500 1.0246009 0.1230551 8.326 1.74e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.03192 on 58 degrees of freedom  
## Multiple R-squared: 0.5445, Adjusted R-squared: 0.5366   
## F-statistic: 69.33 on 1 and 58 DF, p-value: 1.742e-11

**Conclusion**. Based on the results, the Edgewood Fund with is the most diversified of the three funds; the Copley Fund, with is the least diversified.

#### Prob. 8.14

Consider the returns on the three mutual funds described in Exercise 10. For each fund, estimate the appraisal ratio using the market model with the S&P 500 index as the market index; see Example 8.17. According to these results, rank the performances of the funds. Compare the results to the ranking based on the estimated Sharpe ratios.

funds3.mm <- lm(funds3 ~ SP500)  
  
funds3\_alpha <- funds3.mm$coefficients[1,]   
funds3\_alpha

## COPLX EGFFX FMILX   
## 0.0040271825 0.0043254231 -0.0009427462

The market model error standard deviation for each fund may be calculated using…

f.sighat <- function(y){summary(lm(y~SP500))$sigma}  
  
funds3\_sig <- apply(funds3, 2, f.sighat)  
funds3\_sig

## COPLX EGFFX FMILX   
## 0.01793529 0.02002737 0.03191996

and the appraisal ratios may be calculated by

funds3\_appr <- funds3\_alpha/funds3\_sig  
funds3\_appr

## COPLX EGFFX FMILX   
## 0.22453952 0.21597560 -0.02953469

**Conclusion:** According to the appraisal ratios, the ranking of the funds is Copley, followed by Edgewood and Fidelity.

The Sharpe ratios of the funds are given by…

Sharpe <- function(x, ind){  
 mean(x[ind])/sd(x[ind])  
 }  
  
# computations  
copley\_sharpe <- Sharpe(d3$COPLX,1:60)  
edgewood\_sharpe <- Sharpe(d3$EGFFX,1:60)  
fidelity\_sharpe <- Sharpe(d3$FMILX,1:60)  
  
tibble(copley\_sharpe, edgewood\_sharpe, fidelity\_sharpe)

## # A tibble: 1 x 3  
## copley\_sharpe edgewood\_sharpe fidelity\_sharpe  
## <dbl> <dbl> <dbl>  
## 1 0.337 0.332 0.169

**Conclusion:** According to the Sharpe ratios, the ranking of the funds is Copley, followed by Edgewood and Fidelity.

##### Alternative method to estimate the Sharpe ratio.

Copley

p\_load(boot)  
boot(d3$COPLX,Sharpe,1000)

##   
## ORDINARY NONPARAMETRIC BOOTSTRAP  
##   
##   
## Call:  
## boot(data = d3$COPLX, statistic = Sharpe, R = 1000)  
##   
##   
## Bootstrap Statistics :  
## original bias std. error  
## t1\* 0.3370219 0.009536213 0.1428465

Edgewood

boot(d3$EGFFX,Sharpe,1000)

##   
## ORDINARY NONPARAMETRIC BOOTSTRAP  
##   
##   
## Call:  
## boot(data = d3$EGFFX, statistic = Sharpe, R = 1000)  
##   
##   
## Bootstrap Statistics :  
## original bias std. error  
## t1\* 0.3315349 0.01105777 0.1416119

Fidelity

boot(d3$FMILX,Sharpe,1000)

##   
## ORDINARY NONPARAMETRIC BOOTSTRAP  
##   
##   
## Call:  
## boot(data = d3$FMILX, statistic = Sharpe, R = 1000)  
##   
##   
## Bootstrap Statistics :  
## original bias std. error  
## t1\* 0.1686171 0.002509378 0.1338979

#### Prob. 8.15

Consider the returns on the three mutual funds described in Exercise 10. For each fund, use the bootstrap method as described in Example 8.22 to calculate the standard error of the estimated Treynor ratio. Use the estimates from the market model with the S&P 500 index as the market index. Calculate an approximate 95% confidence interval for the true Treynor ratio of each fund. Take the bootstrap sample size to be 10,000.

##### Solution

Treynor ratio for the New Horizons Fund

Treynor <- function(rmat, ind){  
 ret <- rmat[ind,1]  
 mkt <- rmat[ind,2]   
 beta <- lm(ret ~ mkt)$coefficient[2]  
 mean(ret)/beta  
}

*Copley*

p\_load(boot)  
boot(data = cbind(d3$COPLX,SP500),  
 statistic = Treynor,  
 R = 1e4)

##   
## ORDINARY NONPARAMETRIC BOOTSTRAP  
##   
##   
## Call:  
## boot(data = cbind(d3$COPLX, SP500), statistic = Treynor, R = 10000)  
##   
##   
## Bootstrap Statistics :  
## original bias std. error  
## t1\* 0.01805721 0.0002853624 0.007510295

Compute the 95% confidence interval for the true Treynor ratio

copley\_CI95 <- 0.01806 + qnorm(c(0.025,0.975))\*0.00743  
copley\_CI95

## [1] 0.003497468 0.032622532

*Edgewood*

boot(data = cbind(d3$EGFFX,SP500),  
 statistic = Treynor,  
 R = 1e4)

##   
## ORDINARY NONPARAMETRIC BOOTSTRAP  
##   
##   
## Call:  
## boot(data = cbind(d3$EGFFX, SP500), statistic = Treynor, R = 10000)  
##   
##   
## Bootstrap Statistics :  
## original bias std. error  
## t1\* 0.01307859 0.0001181075 0.005403248

Compute the 95% confidence interval for the true Treynor ratio

edgewood\_CI95 <- 0.01308 + qnorm(c(0.025,0.975))\*0.00524  
edgewood\_CI95

## [1] 0.002809789 0.023350211

*Fidelity*

boot(data = cbind(d3$FMILX,SP500),  
 statistic = Treynor,  
 R = 1e4)

##   
## ORDINARY NONPARAMETRIC BOOTSTRAP  
##   
##   
## Call:  
## boot(data = cbind(d3$FMILX, SP500), statistic = Treynor, R = 10000)  
##   
##   
## Bootstrap Statistics :  
## original bias std. error  
## t1\* 0.007716948 4.677012e-05 0.006046545

Compute the 95% confidence interval for the true Treynor ratio

fidelity\_CI95 <- 0.007717 + qnorm(c(0.025,0.975))\*0.006  
fidelity\_CI95

## [1] -0.004042784 0.019476784