HW11

Factor Models

4/16/2021

pacman::p\_load(pacman, tidyverse, tseries, psych,knitr)  
knitr::opts\_chunk$set(message = FALSE, tidy = T, fig.align = "center")

## Question 10.4

#a. Risk-free rate, rfree  
data1 = read.table(file.choose(),header = T,sep = ",")  
data1

## TimePeriod Yield  
## 1 2011-01 0.15  
## 2 2011-02 0.13  
## 3 2011-03 0.10  
## 4 2011-04 0.06  
## 5 2011-05 0.04  
## 6 2011-06 0.04  
## 7 2011-07 0.04  
## 8 2011-08 0.02  
## 9 2011-09 0.01  
## 10 2011-10 0.02  
## 11 2011-11 0.01  
## 12 2011-12 0.01  
## 13 2012-01 0.03  
## 14 2012-02 0.09  
## 15 2012-03 0.08  
## 16 2012-04 0.08  
## 17 2012-05 0.09  
## 18 2012-06 0.09  
## 19 2012-07 0.10  
## 20 2012-08 0.10  
## 21 2012-09 0.11  
## 22 2012-10 0.10  
## 23 2012-11 0.09  
## 24 2012-12 0.07  
## 25 2013-01 0.07  
## 26 2013-02 0.10  
## 27 2013-03 0.09  
## 28 2013-04 0.06  
## 29 2013-05 0.04  
## 30 2013-06 0.05  
## 31 2013-07 0.04  
## 32 2013-08 0.04  
## 33 2013-09 0.02  
## 34 2013-10 0.05  
## 35 2013-11 0.07  
## 36 2013-12 0.07  
## 37 2014-01 0.04  
## 38 2014-02 0.05  
## 39 2014-03 0.05  
## 40 2014-04 0.03  
## 41 2014-05 0.03  
## 42 2014-06 0.04  
## 43 2014-07 0.03  
## 44 2014-08 0.03  
## 45 2014-09 0.02  
## 46 2014-10 0.02  
## 47 2014-11 0.02  
## 48 2014-12 0.03  
## 49 2015-01 0.03  
## 50 2015-02 0.02  
## 51 2015-03 0.03  
## 52 2015-04 0.02  
## 53 2015-05 0.02  
## 54 2015-06 0.02  
## 55 2015-07 0.03  
## 56 2015-08 0.07  
## 57 2015-09 0.02  
## 58 2015-10 0.02  
## 59 2015-11 0.11  
## 60 2015-12 0.22

rff1 = data1$Yield  
table(is.na(rff1))

##   
## FALSE   
## 60

rff1

## [1] 0.15 0.13 0.10 0.06 0.04 0.04 0.04 0.02 0.01 0.02 0.01 0.01 0.03 0.09 0.08  
## [16] 0.08 0.09 0.09 0.10 0.10 0.11 0.10 0.09 0.07 0.07 0.10 0.09 0.06 0.04 0.05  
## [31] 0.04 0.04 0.02 0.05 0.07 0.07 0.04 0.05 0.05 0.03 0.03 0.04 0.03 0.03 0.02  
## [46] 0.02 0.02 0.03 0.03 0.02 0.03 0.02 0.02 0.02 0.03 0.07 0.02 0.02 0.11 0.22

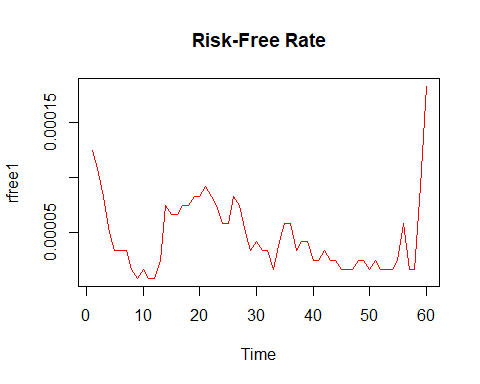
# convert from percentages to proportional monthly returns  
rfree1 = (1 + rff1/100)^(1/12) - 1  
head(rfree1)

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

length(rfree1)

## [1] 60

ts.plot(rfree1,col=2,main="Risk-Free Rate")



#b. S&P 500  
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\_m\_ret = (sp500[-1] - sp500[-n])/sp500[-n]  
# Excess returns  
d = tibble(SP500 = sp500\_m\_ret - rfree1)  
  
#PZZA  
x1 = get.hist.quote(instrument = "PZZA", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

pzza\_m = as.vector(x1)  
n = length(pzza\_m)  
  
pzza\_m\_ret = (pzza\_m[-1] - pzza\_m[-n])/pzza\_m[-n]  
  
stks = tibble(PZZA = pzza\_m\_ret - rfree1)  
  
  
#BBBY  
x2 = get.hist.quote(instrument = "BBBY", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

bbby\_m = as.vector(x2)  
n = length(bbby\_m)  
  
bbby\_m\_ret = (bbby\_m[-1] - bbby\_m[-n])/bbby\_m[-n]  
  
stks = stks %>% mutate(BBBY = bbby\_m\_ret - rfree1)  
  
#NFLX  
  
x3 = get.hist.quote(instrument = "NFLX", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

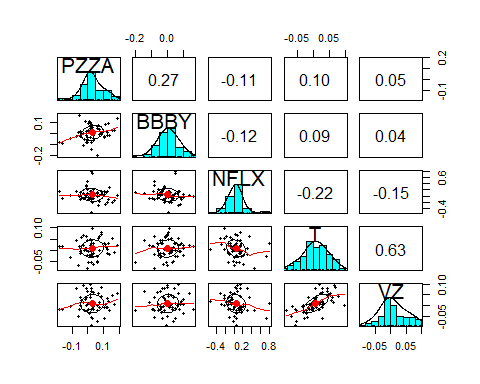
nflx\_m = as.vector(x3)  
n = length(nflx\_m)  
  
nflx\_m\_ret = (nflx\_m[-1] - nflx\_m[-n])/nflx\_m[-n]  
  
stks = stks %>% mutate(NFLX = nflx\_m\_ret - rfree1)  
  
  
#T  
  
X4 = get.hist.quote(instrument = "T", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

t\_m = as.vector(X4)  
n = length(t\_m)  
  
t\_m\_ret = (t\_m[-1] - t\_m[-n])/t\_m[-n]  
  
stks = stks %>% mutate(T = t\_m\_ret - rfree1)  
  
  
#VZ  
  
X5 = get.hist.quote(instrument = "VZ", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

vz\_m = as.vector(X5)  
n = length(vz\_m)  
  
vz\_m\_ret = (vz\_m[-1] - vz\_m[-n])/vz\_m[-n]  
  
stks = stks %>% mutate(VZ = vz\_m\_ret - rfree1)  
  
#Exploratory Data Analysis  
pairs.panels(stks)



#c.Load and prepare the SMB and HML factors  
ff = read.table(file.choose(),header = T,sep = ",")  
ff = ff[3:4]/100  
SMB = ff$SMB  
HML = ff$HML  
  
stks1 = stks %>%   
 mutate(SMB,  
 HML,   
 SP500 = d$SP500)  
stks1

## # A tibble: 60 x 8  
## PZZA BBBY NFLX T VZ SMB HML SP500  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0360 -0.0235 0.218 -0.0634 -0.00460 -0.0252 0.00820 0.0225   
## 2 0.0166 0.00302 -0.0347 0.0461 0.0500 0.0153 0.0129 0.0318   
## 3 0.0852 0.00241 0.150 0.0785 0.0438 0.0258 -0.0176 -0.00113  
## 4 -0.0509 0.163 -0.0215 0.0166 -0.0198 -0.0037 -0.0243 0.0284   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994 -0.00580 -0.0205 -0.0135   
## 6 0.00389 0.0831 -0.0300 -0.00479 0.00809 -0.0011 -0.0042 -0.0183   
## 7 -0.0617 0.00202 0.0125 -0.0685 -0.0521 -0.0137 -0.0088 -0.0215   
## 8 -0.0468 -0.0279 -0.116 -0.0133 0.0383 -0.0303 -0.0256 -0.0568   
## 9 0.0218 0.00791 -0.518 0.00140 0.0174 -0.0335 -0.0171 -0.0718   
## 10 0.111 0.0790 -0.275 0.0277 0.00487 0.0332 0.0013 0.108   
## # ... with 50 more rows

# Factor Sensitivity  
#BIG5  
big5 = tibble(PZZA = stks1$PZZA, BBBY = stks1$BBBY, NFLX = stks1$NFLX, T = stks1$T, VZ = stks1$VZ)  
  
big5

## # A tibble: 60 x 5  
## PZZA BBBY NFLX T VZ  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0360 -0.0235 0.218 -0.0634 -0.00460  
## 2 0.0166 0.00302 -0.0347 0.0461 0.0500   
## 3 0.0852 0.00241 0.150 0.0785 0.0438   
## 4 -0.0509 0.163 -0.0215 0.0166 -0.0198   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994  
## 6 0.00389 0.0831 -0.0300 -0.00479 0.00809  
## 7 -0.0617 0.00202 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

big5\_mat = as.matrix(big5)  
  
big5\_fact = lm(big5\_mat ~ SP500 + SMB + HML, data = stks1)  
  
big5\_fact\_betas = t(big5\_fact$coefficients)[,-1]  
big5\_fact\_betas

## SP500 SMB HML  
## PZZA 0.1874806 0.74633754 -0.83640471  
## BBBY 0.6712400 0.09870395 -0.41838163  
## NFLX 1.4473745 0.64580754 0.33999260  
## T 0.3465500 -0.34217176 0.01039033  
## VZ 0.3654715 -0.66843958 -0.16675084

#Eliminate the rows with Inf  
stks1 = stks1 %>%  
 filter\_all(all\_vars(is.finite(.)))  
# Revised data frame  
stks1

## # A tibble: 60 x 8  
## PZZA BBBY NFLX T VZ SMB HML SP500  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0360 -0.0235 0.218 -0.0634 -0.00460 -0.0252 0.00820 0.0225   
## 2 0.0166 0.00302 -0.0347 0.0461 0.0500 0.0153 0.0129 0.0318   
## 3 0.0852 0.00241 0.150 0.0785 0.0438 0.0258 -0.0176 -0.00113  
## 4 -0.0509 0.163 -0.0215 0.0166 -0.0198 -0.0037 -0.0243 0.0284   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994 -0.00580 -0.0205 -0.0135   
## 6 0.00389 0.0831 -0.0300 -0.00479 0.00809 -0.0011 -0.0042 -0.0183   
## 7 -0.0617 0.00202 0.0125 -0.0685 -0.0521 -0.0137 -0.0088 -0.0215   
## 8 -0.0468 -0.0279 -0.116 -0.0133 0.0383 -0.0303 -0.0256 -0.0568   
## 9 0.0218 0.00791 -0.518 0.00140 0.0174 -0.0335 -0.0171 -0.0718   
## 10 0.111 0.0790 -0.275 0.0277 0.00487 0.0332 0.0013 0.108   
## # ... with 50 more rows

#Fit the factor model  
options(digits = 4)  
stks\_fact = lm(cbind(PZZA,BBBY,NFLX,T,VZ) ~ SP500 + SMB + HML, data=stks1)  
  
stks\_fact$coefficients

## PZZA BBBY NFLX T VZ  
## (Intercept) 0.02482 -0.004331 0.03493 0.004248 0.004477  
## SP500 0.18748 0.671240 1.44737 0.346550 0.365471  
## SMB 0.74634 0.098704 0.64581 -0.342172 -0.668440  
## HML -0.83640 -0.418382 0.33999 0.010390 -0.166751

#Estimate the residual standard deviations  
f.sighat = function(y) {  
 summary(lm(y ~ SP500+SMB+HML, data=stks1))$sigma  
}  
  
y = matrix(c(stks1$PZZA, stks1$BBBY, stks1$NFLX,  
 stks1$T, stks1$VZ),nrow=60)  
   
apply(y,2,f.sighat)

## [1] 0.06968 0.06538 0.20862 0.03746 0.04002

#Correlation  
cor(cbind(stks1$SP500, stks1$SMB, stks1$HML))

## [,1] [,2] [,3]  
## [1,] 1.0000 0.22374 0.15672  
## [2,] 0.2237 1.00000 0.05941  
## [3,] 0.1567 0.05941 1.00000

## Question 10.7

Factor\_Premium1 = 0.002  
Factor\_Premium2 = 0.001  
Factor\_Premium3 = 0.0025  
Alpha = 0.005  
Beta1 = 1.0  
Beta2 = 0.75  
Beta3 = -0.10  
Expected\_return = Alpha+(Beta1\*Factor\_Premium1)+(Beta2\*Factor\_Premium2)+(Beta3\*Factor\_Premium3)  
Expected\_return

## [1] 0.0075

## Question 10.8

#a. Risk-free rate, rfree  
data1 = read.table(file.choose(),header = T,sep = ",")  
data1

## TimePeriod Yield  
## 1 2011-01 0.15  
## 2 2011-02 0.13  
## 3 2011-03 0.10  
## 4 2011-04 0.06  
## 5 2011-05 0.04  
## 6 2011-06 0.04  
## 7 2011-07 0.04  
## 8 2011-08 0.02  
## 9 2011-09 0.01  
## 10 2011-10 0.02  
## 11 2011-11 0.01  
## 12 2011-12 0.01  
## 13 2012-01 0.03  
## 14 2012-02 0.09  
## 15 2012-03 0.08  
## 16 2012-04 0.08  
## 17 2012-05 0.09  
## 18 2012-06 0.09  
## 19 2012-07 0.10  
## 20 2012-08 0.10  
## 21 2012-09 0.11  
## 22 2012-10 0.10  
## 23 2012-11 0.09  
## 24 2012-12 0.07  
## 25 2013-01 0.07  
## 26 2013-02 0.10  
## 27 2013-03 0.09  
## 28 2013-04 0.06  
## 29 2013-05 0.04  
## 30 2013-06 0.05  
## 31 2013-07 0.04  
## 32 2013-08 0.04  
## 33 2013-09 0.02  
## 34 2013-10 0.05  
## 35 2013-11 0.07  
## 36 2013-12 0.07  
## 37 2014-01 0.04  
## 38 2014-02 0.05  
## 39 2014-03 0.05  
## 40 2014-04 0.03  
## 41 2014-05 0.03  
## 42 2014-06 0.04  
## 43 2014-07 0.03  
## 44 2014-08 0.03  
## 45 2014-09 0.02  
## 46 2014-10 0.02  
## 47 2014-11 0.02  
## 48 2014-12 0.03  
## 49 2015-01 0.03  
## 50 2015-02 0.02  
## 51 2015-03 0.03  
## 52 2015-04 0.02  
## 53 2015-05 0.02  
## 54 2015-06 0.02  
## 55 2015-07 0.03  
## 56 2015-08 0.07  
## 57 2015-09 0.02  
## 58 2015-10 0.02  
## 59 2015-11 0.11  
## 60 2015-12 0.22

rff1 = data1$Yield  
table(is.na(rff1))

##   
## FALSE   
## 60

rff1

## [1] 0.15 0.13 0.10 0.06 0.04 0.04 0.04 0.02 0.01 0.02 0.01 0.01 0.03 0.09 0.08  
## [16] 0.08 0.09 0.09 0.10 0.10 0.11 0.10 0.09 0.07 0.07 0.10 0.09 0.06 0.04 0.05  
## [31] 0.04 0.04 0.02 0.05 0.07 0.07 0.04 0.05 0.05 0.03 0.03 0.04 0.03 0.03 0.02  
## [46] 0.02 0.02 0.03 0.03 0.02 0.03 0.02 0.02 0.02 0.03 0.07 0.02 0.02 0.11 0.22

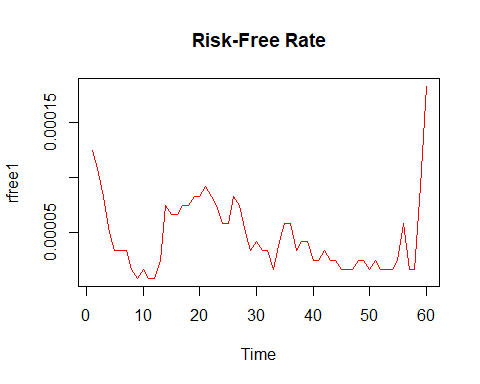
# convert from percentages to proportional monthly returns  
rfree1 = (1 + rff1/100)^(1/12) - 1  
head(rfree1)

## [1] 1.249e-04 1.083e-04 8.330e-05 4.999e-05 3.333e-05 3.333e-05

length(rfree1)

## [1] 60

ts.plot(rfree1,col=2,main="Risk-Free Rate")



#b. S&P 500  
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\_m\_ret = (sp500[-1] - sp500[-n])/sp500[-n]  
# Excess returns  
d = tibble(SP500 = sp500\_m\_ret - rfree1)  
  
#PZZA  
x1 = get.hist.quote(instrument = "PZZA", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

pzza\_m = as.vector(x1)  
n = length(pzza\_m)  
  
pzza\_m\_ret = (pzza\_m[-1] - pzza\_m[-n])/pzza\_m[-n]  
  
stks = tibble(PZZA = pzza\_m\_ret - rfree1)  
  
  
#BBBY  
x2 = get.hist.quote(instrument = "BBBY", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

bbby\_m = as.vector(x2)  
n = length(bbby\_m)  
  
bbby\_m\_ret = (bbby\_m[-1] - bbby\_m[-n])/bbby\_m[-n]  
  
stks = stks %>% mutate(BBBY = bbby\_m\_ret - rfree1)  
  
#NFLX  
  
x3 = get.hist.quote(instrument = "NFLX", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

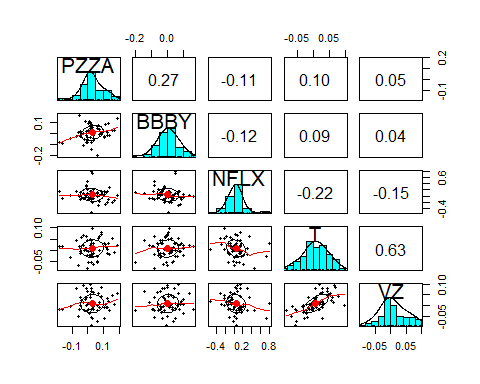
nflx\_m = as.vector(x3)  
n = length(nflx\_m)  
  
nflx\_m\_ret = (nflx\_m[-1] - nflx\_m[-n])/nflx\_m[-n]  
  
stks = stks %>% mutate(NFLX = nflx\_m\_ret - rfree1)  
  
  
#T  
  
X4 = get.hist.quote(instrument = "T", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

t\_m = as.vector(X4)  
n = length(t\_m)  
  
t\_m\_ret = (t\_m[-1] - t\_m[-n])/t\_m[-n]  
  
stks = stks %>% mutate(T = t\_m\_ret - rfree1)  
  
  
#VZ  
  
X5 = get.hist.quote(instrument = "VZ", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

vz\_m = as.vector(X5)  
n = length(vz\_m)  
  
vz\_m\_ret = (vz\_m[-1] - vz\_m[-n])/vz\_m[-n]  
  
stks = stks %>% mutate(VZ = vz\_m\_ret - rfree1)  
  
#Exploratory Data Analysis  
pairs.panels(stks)



#c.Load and prepare the SMB and HML factors  
ff = read.table(file.choose(),header = T,sep = ",")  
ff = ff[3:4]/100  
SMB = ff$SMB  
HML = ff$HML  
  
stks1 = stks %>%   
 mutate(SMB,  
 HML,   
 SP500 = d$SP500)  
stks1

## # A tibble: 60 x 8  
## PZZA BBBY NFLX T VZ SMB HML SP500  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0360 -0.0235 0.218 -0.0634 -0.00460 -0.0252 0.00820 0.0225   
## 2 0.0166 0.00302 -0.0347 0.0461 0.0500 0.0153 0.0129 0.0318   
## 3 0.0852 0.00241 0.150 0.0785 0.0438 0.0258 -0.0176 -0.00113  
## 4 -0.0509 0.163 -0.0215 0.0166 -0.0198 -0.0037 -0.0243 0.0284   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994 -0.00580 -0.0205 -0.0135   
## 6 0.00389 0.0831 -0.0300 -0.00479 0.00809 -0.0011 -0.0042 -0.0183   
## 7 -0.0617 0.00202 0.0125 -0.0685 -0.0521 -0.0137 -0.0088 -0.0215   
## 8 -0.0468 -0.0279 -0.116 -0.0133 0.0383 -0.0303 -0.0256 -0.0568   
## 9 0.0218 0.00791 -0.518 0.00140 0.0174 -0.0335 -0.0171 -0.0718   
## 10 0.111 0.0790 -0.275 0.0277 0.00487 0.0332 0.0013 0.108   
## # ... with 50 more rows

# Factor Sensitivity  
#BIG5  
big5 = tibble(PZZA = stks1$PZZA, BBBY = stks1$BBBY, NFLX = stks1$NFLX, T = stks1$T, VZ = stks1$VZ)  
  
big5

## # A tibble: 60 x 5  
## PZZA BBBY NFLX T VZ  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0360 -0.0235 0.218 -0.0634 -0.00460  
## 2 0.0166 0.00302 -0.0347 0.0461 0.0500   
## 3 0.0852 0.00241 0.150 0.0785 0.0438   
## 4 -0.0509 0.163 -0.0215 0.0166 -0.0198   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994  
## 6 0.00389 0.0831 -0.0300 -0.00479 0.00809  
## 7 -0.0617 0.00202 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

big5\_mat = as.matrix(big5)  
  
big5\_fact = lm(big5\_mat ~ SP500 + SMB + HML, data = stks1)  
  
big5\_fact\_betas = t(big5\_fact$coefficients)[,-1]  
big5\_fact\_betas

## SP500 SMB HML  
## PZZA 0.1875 0.7463 -0.83640  
## BBBY 0.6712 0.0987 -0.41838  
## NFLX 1.4474 0.6458 0.33999  
## T 0.3465 -0.3422 0.01039  
## VZ 0.3655 -0.6684 -0.16675

#Eliminate the rows with Inf  
stks1 = stks1 %>%  
 filter\_all(all\_vars(is.finite(.)))  
# Revised data frame  
stks1

## # A tibble: 60 x 8  
## PZZA BBBY NFLX T VZ SMB HML SP500  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0360 -0.0235 0.218 -0.0634 -0.00460 -0.0252 0.00820 0.0225   
## 2 0.0166 0.00302 -0.0347 0.0461 0.0500 0.0153 0.0129 0.0318   
## 3 0.0852 0.00241 0.150 0.0785 0.0438 0.0258 -0.0176 -0.00113  
## 4 -0.0509 0.163 -0.0215 0.0166 -0.0198 -0.0037 -0.0243 0.0284   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994 -0.00580 -0.0205 -0.0135   
## 6 0.00389 0.0831 -0.0300 -0.00479 0.00809 -0.0011 -0.0042 -0.0183   
## 7 -0.0617 0.00202 0.0125 -0.0685 -0.0521 -0.0137 -0.0088 -0.0215   
## 8 -0.0468 -0.0279 -0.116 -0.0133 0.0383 -0.0303 -0.0256 -0.0568   
## 9 0.0218 0.00791 -0.518 0.00140 0.0174 -0.0335 -0.0171 -0.0718   
## 10 0.111 0.0790 -0.275 0.0277 0.00487 0.0332 0.0013 0.108   
## # ... with 50 more rows

head(big5\_fact\_betas,10)

## SP500 SMB HML  
## PZZA 0.1875 0.7463 -0.83640  
## BBBY 0.6712 0.0987 -0.41838  
## NFLX 1.4474 0.6458 0.33999  
## T 0.3465 -0.3422 0.01039  
## VZ 0.3655 -0.6684 -0.16675

big5\_mean = apply(big5\_mat, 2, mean)  
  
fit = summary(lm(big5\_mean ~ big5\_fact\_betas))  
fit

##   
## Call:  
## lm(formula = big5\_mean ~ big5\_fact\_betas)  
##   
## Residuals:  
## PZZA BBBY NFLX T VZ   
## 0.00377 -0.00905 0.00305 -0.00734 0.00957   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 0.0291 0.0309 0.94 0.52  
## big5\_fact\_betasSP500 -0.0115 0.0391 -0.29 0.82  
## big5\_fact\_betasSMB 0.0301 0.0228 1.32 0.41  
## big5\_fact\_betasHML 0.0319 0.0417 0.77 0.58  
##   
## Residual standard error: 0.0158 on 1 degrees of freedom  
## Multiple R-squared: 0.805, Adjusted R-squared: 0.218   
## F-statistic: 1.37 on 3 and 1 DF, p-value: 0.544

fact\_prem = fit$coefficients[,2]  
fact\_prem

## (Intercept) big5\_fact\_betasSP500 big5\_fact\_betasSMB   
## 0.03092 0.03912 0.02282   
## big5\_fact\_betasHML   
## 0.04169

big5\_fit = lm(big5\_mean ~ big5\_fact\_betas)$fitted.values  
round(big5\_fit,4)

## PZZA BBBY NFLX T VZ   
## 0.0227 0.0110 0.0427 0.0152 -0.0005

mean(abs(big5\_fit - big5\_mean))

## [1] 0.006556

cor(big5\_fit, big5\_mean)

## [1] 0.897

spcoef = lm(t(big5\_mat) ~ big5\_fact\_betas)$coef  
str(spcoef)

## num [1:4, 1:60] -0.0263 0.1028 0.0917 0.0562 0.0659 ...  
## - attr(\*, "dimnames")=List of 2  
## ..$ : chr [1:4] "(Intercept)" "big5\_fact\_betasSP500" "big5\_fact\_betasSMB" "big5\_fact\_betasHML"  
## ..$ : NULL

apply(spcoef, 1, mean)

## (Intercept) big5\_fact\_betasSP500 big5\_fact\_betasSMB   
## 0.02909 -0.01149 0.03006   
## big5\_fact\_betasHML   
## 0.03191

## Question 10.9

#a. Risk-free rate, rfree  
data1 = read.table(file.choose(),header = T,sep = ",")  
data1

## TimePeriod Yield  
## 1 2011-01 0.15  
## 2 2011-02 0.13  
## 3 2011-03 0.10  
## 4 2011-04 0.06  
## 5 2011-05 0.04  
## 6 2011-06 0.04  
## 7 2011-07 0.04  
## 8 2011-08 0.02  
## 9 2011-09 0.01  
## 10 2011-10 0.02  
## 11 2011-11 0.01  
## 12 2011-12 0.01  
## 13 2012-01 0.03  
## 14 2012-02 0.09  
## 15 2012-03 0.08  
## 16 2012-04 0.08  
## 17 2012-05 0.09  
## 18 2012-06 0.09  
## 19 2012-07 0.10  
## 20 2012-08 0.10  
## 21 2012-09 0.11  
## 22 2012-10 0.10  
## 23 2012-11 0.09  
## 24 2012-12 0.07  
## 25 2013-01 0.07  
## 26 2013-02 0.10  
## 27 2013-03 0.09  
## 28 2013-04 0.06  
## 29 2013-05 0.04  
## 30 2013-06 0.05  
## 31 2013-07 0.04  
## 32 2013-08 0.04  
## 33 2013-09 0.02  
## 34 2013-10 0.05  
## 35 2013-11 0.07  
## 36 2013-12 0.07  
## 37 2014-01 0.04  
## 38 2014-02 0.05  
## 39 2014-03 0.05  
## 40 2014-04 0.03  
## 41 2014-05 0.03  
## 42 2014-06 0.04  
## 43 2014-07 0.03  
## 44 2014-08 0.03  
## 45 2014-09 0.02  
## 46 2014-10 0.02  
## 47 2014-11 0.02  
## 48 2014-12 0.03  
## 49 2015-01 0.03  
## 50 2015-02 0.02  
## 51 2015-03 0.03  
## 52 2015-04 0.02  
## 53 2015-05 0.02  
## 54 2015-06 0.02  
## 55 2015-07 0.03  
## 56 2015-08 0.07  
## 57 2015-09 0.02  
## 58 2015-10 0.02  
## 59 2015-11 0.11  
## 60 2015-12 0.22

rff1 = data1$Yield  
table(is.na(rff1))

##   
## FALSE   
## 60

rff1

## [1] 0.15 0.13 0.10 0.06 0.04 0.04 0.04 0.02 0.01 0.02 0.01 0.01 0.03 0.09 0.08  
## [16] 0.08 0.09 0.09 0.10 0.10 0.11 0.10 0.09 0.07 0.07 0.10 0.09 0.06 0.04 0.05  
## [31] 0.04 0.04 0.02 0.05 0.07 0.07 0.04 0.05 0.05 0.03 0.03 0.04 0.03 0.03 0.02  
## [46] 0.02 0.02 0.03 0.03 0.02 0.03 0.02 0.02 0.02 0.03 0.07 0.02 0.02 0.11 0.22

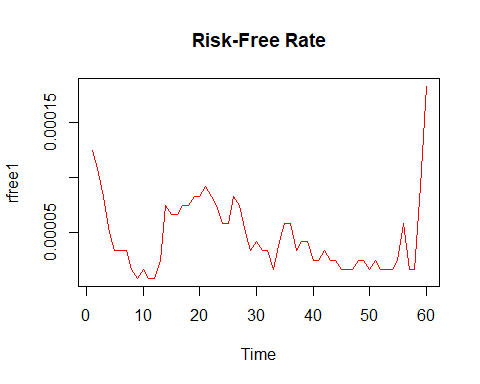
# convert from percentages to proportional monthly returns  
rfree1 = (1 + rff1/100)^(1/12) - 1  
head(rfree1)

## [1] 1.249e-04 1.083e-04 8.330e-05 4.999e-05 3.333e-05 3.333e-05

length(rfree1)

## [1] 60

ts.plot(rfree1,col=2,main="Risk-Free Rate")



#b. S&P 500  
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\_m\_ret = (sp500[-1] - sp500[-n])/sp500[-n]  
# Excess returns  
stks3 = tibble(SP500 = sp500\_m\_ret - rfree1)  
  
#PZZA  
x1 = get.hist.quote(instrument = "PZZA", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

pzza\_m = as.vector(x1)  
n = length(pzza\_m)  
  
pzza\_m\_ret = (pzza\_m[-1] - pzza\_m[-n])/pzza\_m[-n]  
  
stks = tibble(PZZA = pzza\_m\_ret - rfree1)  
  
  
#BBBY  
x2 = get.hist.quote(instrument = "BBBY", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

bbby\_m = as.vector(x2)  
n = length(bbby\_m)  
  
bbby\_m\_ret = (bbby\_m[-1] - bbby\_m[-n])/bbby\_m[-n]  
  
stks = stks %>% mutate(BBBY = bbby\_m\_ret - rfree1)  
  
#NFLX  
  
x3 = get.hist.quote(instrument = "NFLX", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

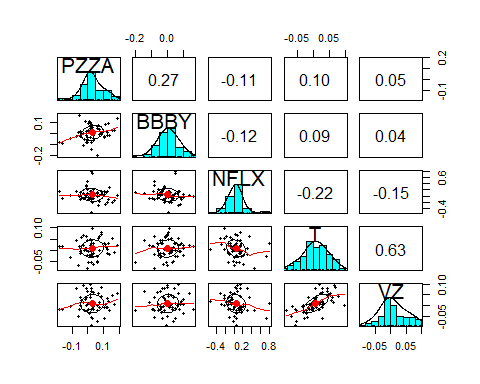
nflx\_m = as.vector(x3)  
n = length(nflx\_m)  
  
nflx\_m\_ret = (nflx\_m[-1] - nflx\_m[-n])/nflx\_m[-n]  
  
stks = stks %>% mutate(NFLX = nflx\_m\_ret - rfree1)  
  
  
#T  
  
X4 = get.hist.quote(instrument = "T", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

t\_m = as.vector(X4)  
n = length(t\_m)  
  
t\_m\_ret = (t\_m[-1] - t\_m[-n])/t\_m[-n]  
  
stks = stks %>% mutate(T = t\_m\_ret - rfree1)  
  
  
#VZ  
  
X5 = get.hist.quote(instrument = "VZ", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

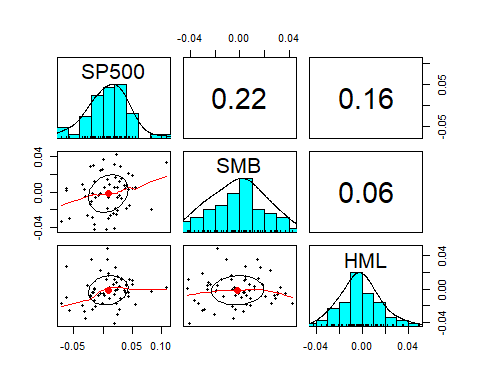
vz\_m = as.vector(X5)  
n = length(vz\_m)  
  
vz\_m\_ret = (vz\_m[-1] - vz\_m[-n])/vz\_m[-n]  
  
stks = stks %>% mutate(VZ = vz\_m\_ret - rfree1)  
  
#Exploratory Data Analysis  
pairs.panels(stks)



#c.Load and prepare the SMB and HML factors  
ff = read.table(file.choose(),header = T,sep = ",")  
ff = ff[3:4]/100  
SMB = ff$SMB  
HML = ff$HML  
  
stks3 = stks3 %>%   
 mutate(SMB,  
 HML,   
 SP500 = stks3$SP500)  
stks3

## # A tibble: 60 x 3  
## SP500 SMB HML  
## <dbl> <dbl> <dbl>  
## 1 0.0225 -0.0252 0.00820  
## 2 0.0318 0.0153 0.0129   
## 3 -0.00113 0.0258 -0.0176   
## 4 0.0284 -0.0037 -0.0243   
## 5 -0.0135 -0.00580 -0.0205   
## 6 -0.0183 -0.0011 -0.0042   
## 7 -0.0215 -0.0137 -0.0088   
## 8 -0.0568 -0.0303 -0.0256   
## 9 -0.0718 -0.0335 -0.0171   
## 10 0.108 0.0332 0.0013   
## # ... with 50 more rows

pairs.panels(stks3)



apply(stks3,2,summary)

## SP500 SMB HML  
## Min. -0.071770 -0.042700 -0.042100  
## 1st Qu. -0.015211 -0.018625 -0.011125  
## Median 0.010726 0.000250 -0.002400  
## Mean 0.008637 -0.001738 -0.001558  
## 3rd Qu. 0.028767 0.011675 0.008600  
## Max. 0.107706 0.042800 0.049000

big5 = tibble(PZZA = stks$PZZA, BBBY = stks$BBBY, NFLX = stks$NFLX, T = stks$T, VZ = stks$VZ)  
  
big5

## # A tibble: 60 x 5  
## PZZA BBBY NFLX T VZ  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0360 -0.0235 0.218 -0.0634 -0.00460  
## 2 0.0166 0.00302 -0.0347 0.0461 0.0500   
## 3 0.0852 0.00241 0.150 0.0785 0.0438   
## 4 -0.0509 0.163 -0.0215 0.0166 -0.0198   
## 5 0.102 -0.0399 0.164 0.0285 -0.00994  
## 6 0.00389 0.0831 -0.0300 -0.00479 0.00809  
## 7 -0.0617 0.00202 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

big5\_mat = as.matrix(big5)  
big5\_fact = lm(big5\_mat ~ SP500 + SMB + HML, data = stks3)  
  
big5\_fact\_betas = t(big5\_fact$coefficients)[,-1]  
big5\_fact\_betas

## SP500 SMB HML  
## PZZA 0.1875 0.7463 -0.83640  
## BBBY 0.6712 0.0987 -0.41838  
## NFLX 1.4474 0.6458 0.33999  
## T 0.3465 -0.3422 0.01039  
## VZ 0.3655 -0.6684 -0.16675

f.sighat.hat = function(y){  
 summary(lm(y ~ SP500 + SMB + HML, data = stks3))$sigma^2  
}  
  
Sig.FF1 = cov(stks3)   
  
big5\_Sig\_fact =  
 big5\_fact\_betas %\*% Sig.FF1 %\*% t(big5\_fact\_betas) + diag(apply(big5\_mat, 2,f.sighat.hat))  
  
big5\_mean = apply(big5,2,mean)  
lambda = 10  
p\_load(quadprog)  
big5\_ra10 = solve.QP(  
 Dmat = lambda\*big5\_Sig\_fact,  
 dvec = big5\_mean,  
 A = cbind(rep(1,5)),  
 bvec = c(1),  
 meq = 1)$solution  
  
big5\_ra10

## [1] 0.43991 -0.09597 0.08487 0.26108 0.31011

big8\_ra10 = solve.QP(  
 Dmat = lambda\*big5\_Sig\_fact,  
 dvec = big5\_mean,  
 A = cbind(rep(1,5),diag(5)),  
 bvec = c(1, rep(0,5)),  
 meq=1  
)$solution  
  
tibble(Stocks = names(big5), Weights=round(big5\_ra10,4))

## # A tibble: 5 x 2  
## Stocks Weights  
## <chr> <dbl>  
## 1 PZZA 0.440   
## 2 BBBY -0.096   
## 3 NFLX 0.0849  
## 4 T 0.261   
## 5 VZ 0.310

t(big5\_fact\_betas) %\*% big5\_ra10

## [,1]  
## SP500 0.34471  
## SMB 0.07703  
## HML -0.34793

A = cbind(rep(1,5), big5\_fact\_betas[ ,c(2,3)], diag(5))  
t(A)

## PZZA BBBY NFLX T VZ  
## 1.0000 1.0000 1.0000 1.00000 1.0000  
## SMB 0.7463 0.0987 0.6458 -0.34217 -0.6684  
## HML -0.8364 -0.4184 0.3400 0.01039 -0.1668  
## 1.0000 0.0000 0.0000 0.00000 0.0000  
## 0.0000 1.0000 0.0000 0.00000 0.0000  
## 0.0000 0.0000 1.0000 0.00000 0.0000  
## 0.0000 0.0000 0.0000 1.00000 0.0000  
## 0.0000 0.0000 0.0000 0.00000 1.0000

big5\_ra10\_con = solve.QP(  
 Dmat = lambda\*big5\_Sig\_fact,  
 dvec = big5\_mean,  
 Amat = A,  
 bvec = c(1, 0, 0, rep(0,5)),  
 meq = 3  
)$solution  
  
tibble(Stocks = names(big5), Weights=round(big5\_ra10\_con,4))

## # A tibble: 5 x 2  
## Stocks Weights  
## <chr> <dbl>  
## 1 PZZA 0.103  
## 2 BBBY 0   
## 3 NFLX 0.233  
## 4 T 0.664  
## 5 VZ 0

round(t(big5\_fact\_betas) %\*% big5\_ra10\_con,4)

## [,1]  
## SP500 0.5866  
## SMB 0.0000  
## HML 0.0000

Port1\_rtns = sum(big5\_ra10\*big5\_mean)  
Port1\_sds = (t(big5\_ra10) %\*% big5\_Sig\_fact %\*% big5\_ra10)^0.5  
  
Port2\_rtns = sum(big5\_ra10\_con\*big5\_mean)  
Port2\_sds = (t(big5\_ra10\_con) %\*% big5\_Sig\_fact %\*% big5\_ra10\_con)^0.5  
  
tibble(Port1\_rtns, Port1\_sds, Port2\_rtns, Port2\_sds)

## # A tibble: 1 x 4  
## Port1\_rtns Port1\_sds[,1] Port2\_rtns Port2\_sds[,1]  
## <dbl> <dbl> <dbl> <dbl>  
## 1 0.0202 0.0413 0.0186 0.0585

## Question 10.11

#a. Risk-free rate, rfree  
data1 = read.table(file.choose(),header = T,sep = ",")  
data1

## TimePeriod Yield  
## 1 2011-01 0.15  
## 2 2011-02 0.13  
## 3 2011-03 0.10  
## 4 2011-04 0.06  
## 5 2011-05 0.04  
## 6 2011-06 0.04  
## 7 2011-07 0.04  
## 8 2011-08 0.02  
## 9 2011-09 0.01  
## 10 2011-10 0.02  
## 11 2011-11 0.01  
## 12 2011-12 0.01  
## 13 2012-01 0.03  
## 14 2012-02 0.09  
## 15 2012-03 0.08  
## 16 2012-04 0.08  
## 17 2012-05 0.09  
## 18 2012-06 0.09  
## 19 2012-07 0.10  
## 20 2012-08 0.10  
## 21 2012-09 0.11  
## 22 2012-10 0.10  
## 23 2012-11 0.09  
## 24 2012-12 0.07  
## 25 2013-01 0.07  
## 26 2013-02 0.10  
## 27 2013-03 0.09  
## 28 2013-04 0.06  
## 29 2013-05 0.04  
## 30 2013-06 0.05  
## 31 2013-07 0.04  
## 32 2013-08 0.04  
## 33 2013-09 0.02  
## 34 2013-10 0.05  
## 35 2013-11 0.07  
## 36 2013-12 0.07  
## 37 2014-01 0.04  
## 38 2014-02 0.05  
## 39 2014-03 0.05  
## 40 2014-04 0.03  
## 41 2014-05 0.03  
## 42 2014-06 0.04  
## 43 2014-07 0.03  
## 44 2014-08 0.03  
## 45 2014-09 0.02  
## 46 2014-10 0.02  
## 47 2014-11 0.02  
## 48 2014-12 0.03  
## 49 2015-01 0.03  
## 50 2015-02 0.02  
## 51 2015-03 0.03  
## 52 2015-04 0.02  
## 53 2015-05 0.02  
## 54 2015-06 0.02  
## 55 2015-07 0.03  
## 56 2015-08 0.07  
## 57 2015-09 0.02  
## 58 2015-10 0.02  
## 59 2015-11 0.11  
## 60 2015-12 0.22

rff1 = data1$Yield  
table(is.na(rff1))

##   
## FALSE   
## 60

rff1

## [1] 0.15 0.13 0.10 0.06 0.04 0.04 0.04 0.02 0.01 0.02 0.01 0.01 0.03 0.09 0.08  
## [16] 0.08 0.09 0.09 0.10 0.10 0.11 0.10 0.09 0.07 0.07 0.10 0.09 0.06 0.04 0.05  
## [31] 0.04 0.04 0.02 0.05 0.07 0.07 0.04 0.05 0.05 0.03 0.03 0.04 0.03 0.03 0.02  
## [46] 0.02 0.02 0.03 0.03 0.02 0.03 0.02 0.02 0.02 0.03 0.07 0.02 0.02 0.11 0.22

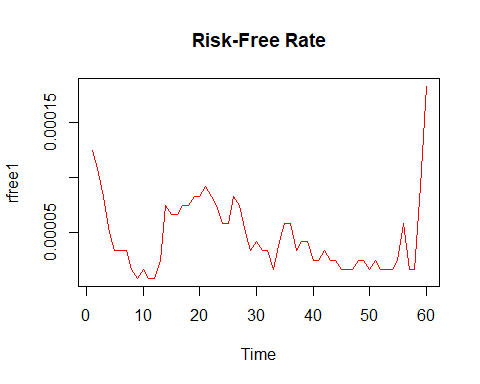
# convert from percentages to proportional monthly returns  
rfree1 = (1 + rff1/100)^(1/12) - 1  
head(rfree1)

## [1] 1.249e-04 1.083e-04 8.330e-05 4.999e-05 3.333e-05 3.333e-05

length(rfree1)

## [1] 60

ts.plot(rfree1,col=2,main="Risk-Free Rate")



#b. S&P 500  
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\_m\_ret = (sp500[-1] - sp500[-n])/sp500[-n]  
# Excess returns  
d = tibble(SP500 = sp500\_m\_ret - rfree1)  
  
#American Funds Income Fund of America Class A (symbol AMECX)  
x1 = get.hist.quote(instrument = "AMECX", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

amecx\_m = as.vector(x1)  
n = length(amecx\_m)  
  
amecx\_m\_ret = (amecx\_m[-1] - amecx\_m[-n])/amecx\_m[-n]  
  
d = d %>% mutate(AMECX = amecx\_m\_ret - rfree1)  
  
  
#Dodge & Cox Stock Fund (DODGX)  
x2 = get.hist.quote(instrument = "DODGX", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

dodgx\_m = as.vector(x2)  
n = length(dodgx\_m)  
  
dodgx\_m\_ret = (dodgx\_m[-1] - dodgx\_m[-n])/dodgx\_m[-n]  
  
d = d %>% mutate(DODGX = dodgx\_m\_ret - rfree1)  
  
# Fidelity Millennium  
  
x3 = 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(x3)  
n = length(fmilx\_m)  
  
fmilx\_m\_ret = (fmilx\_m[-1] - fmilx\_m[-n])/fmilx\_m[-n]  
  
d = d %>% mutate(FMILX = fmilx\_m\_ret - rfree1)  
  
  
#Franklin Income Fund Class A (FKINX)  
  
X4 = get.hist.quote(instrument = "FKINX", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

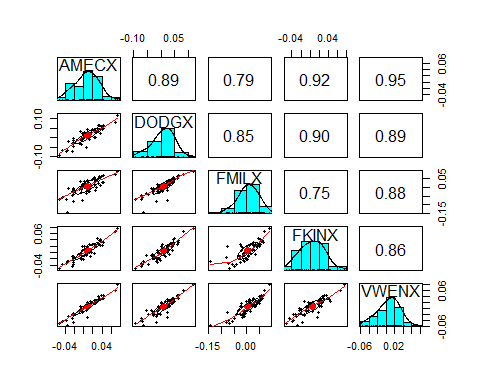
fkinx\_m = as.vector(X4)  
n = length(fkinx\_m)  
  
fkinx\_m\_ret = (fkinx\_m[-1] - fkinx\_m[-n])/fkinx\_m[-n]  
  
d = d %>% mutate(FKINX = fkinx\_m\_ret - rfree1)  
  
  
#Vanguard Wellington Fund (VWENX)  
  
X5 = get.hist.quote(instrument = "VWENX", start = "2010-12-01", end = "2015-12-31",   
 quote = "AdjClose", compression = "m") # monthly

## time series ends 2015-12-01

vwenx\_m = as.vector(X5)  
n = length(vwenx\_m)  
  
vwenx\_m\_ret = (vwenx\_m[-1] - vwenx\_m[-n])/vwenx\_m[-n]  
  
d = d %>% mutate(VWENX = vwenx\_m\_ret - rfree1)  
  
#BIG5  
big5 = tibble(AMECX = d$AMECX, DODGX = d$DODGX, FMILX = d$FMILX, FKINX = d$FKINX, VWENX = d$VWENX)  
  
big5

## # A tibble: 60 x 5  
## AMECX DODGX FMILX FKINX VWENX  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0318 0.0331 0.0239 0.0287 0.0274   
## 2 0.0255 0.0364 0.0385 0.0189 0.0231   
## 3 -0.00240 -0.0119 0.0145 0.00533 -0.00847  
## 4 0.0409 0.0416 0.0273 0.0231 0.0348   
## 5 -0.00454 -0.00994 -0.0118 -0.00344 -0.00511  
## 6 -0.0244 -0.0283 -0.0232 -0.0124 -0.0210   
## 7 -0.00549 -0.0340 -0.00967 -0.00821 -0.00505  
## 8 -0.0254 -0.0711 -0.0487 -0.0447 -0.0351   
## 9 -0.0544 -0.0951 -0.0695 -0.0520 -0.0492   
## 10 0.0748 0.118 0.0883 0.0768 0.0805   
## # ... with 50 more rows

#Exploratory Data Analysis  
pairs.panels(big5)



#c.Load and prepare the SMB and HML factors  
ff = read.table(file.choose(),header = T,sep = ",")  
ff = ff[3:4]/100  
SMB = ff$SMB  
HML = ff$HML  
  
d = d %>%   
 mutate(SMB,  
 HML,   
 SP500 = d$SP500)  
d

## # A tibble: 60 x 8  
## SP500 AMECX DODGX FMILX FKINX VWENX SMB HML  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.0225 0.0318 0.0331 0.0239 0.0287 0.0274 -0.0252 0.00820  
## 2 0.0318 0.0255 0.0364 0.0385 0.0189 0.0231 0.0153 0.0129   
## 3 -0.00113 -0.00240 -0.0119 0.0145 0.00533 -0.00847 0.0258 -0.0176   
## 4 0.0284 0.0409 0.0416 0.0273 0.0231 0.0348 -0.0037 -0.0243   
## 5 -0.0135 -0.00454 -0.00994 -0.0118 -0.00344 -0.00511 -0.00580 -0.0205   
## 6 -0.0183 -0.0244 -0.0283 -0.0232 -0.0124 -0.0210 -0.0011 -0.0042   
## 7 -0.0215 -0.00549 -0.0340 -0.00967 -0.00821 -0.00505 -0.0137 -0.0088   
## 8 -0.0568 -0.0254 -0.0711 -0.0487 -0.0447 -0.0351 -0.0303 -0.0256   
## 9 -0.0718 -0.0544 -0.0951 -0.0695 -0.0520 -0.0492 -0.0335 -0.0171   
## 10 0.108 0.0748 0.118 0.0883 0.0768 0.0805 0.0332 0.0013   
## # ... with 50 more rows

big5\_mat = as.matrix(big5)  
big5\_fact = lm(big5\_mat ~ SP500 + SMB + HML, data = d)  
  
big5\_fact\_betas = t(big5\_fact$coefficients)[,-1]  
big5\_fact\_betas

## SP500 SMB HML  
## AMECX 0.7362 -0.19001 -0.06661  
## DODGX 1.1411 0.02017 0.20148  
## FMILX 0.9967 0.25463 -0.18371  
## FKINX 0.6949 -0.07233 0.16884  
## VWENX 0.7307 -0.12519 -0.07303

#Estimate the error covariance matrix of the factors and the error covariance  
Sig.F = with(d,  
 cov(cbind(SP500,SMB,HML)))  
Sig.F

## SP500 SMB HML  
## SP500 1.140e-03 1.609e-04 9.322e-05  
## SMB 1.609e-04 4.536e-04 2.229e-05  
## HML 9.322e-05 2.229e-05 3.102e-04

f.sig = function(y){  
 summary(lm(y ~ d$SP500 + d$SMB + d$HML))$sigma  
}  
  
s4 = d %>%  
 select(c(AMECX,DODGX,FMILX,FKINX,VWENX))  
Sig.eps = diag(apply(s4, 2, f.sig)^2)  
Sig.eps

## [,1] [,2] [,3] [,4] [,5]  
## [1,] 0.0001384 0.0000000 0.0000000 0.0000000 0.0000000  
## [2,] 0.0000000 0.0001145 0.0000000 0.0000000 0.0000000  
## [3,] 0.0000000 0.0000000 0.0006009 0.0000000 0.0000000  
## [4,] 0.0000000 0.0000000 0.0000000 0.0001455 0.0000000  
## [5,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0001714

#Estimate of the covariance matrix of the assets and correlation matrix of the factor model  
Sig = big5\_fact\_betas %\*% Sig.F %\*% t(big5\_fact\_betas) + Sig.eps  
cov2cor(Sig)

## AMECX DODGX FMILX FKINX VWENX  
## AMECX 1.0000 0.8455 0.6987 0.7857 0.7883  
## DODGX 0.8455 1.0000 0.7694 0.8580 0.8327  
## FMILX 0.6987 0.7694 1.0000 0.6972 0.6942  
## FKINX 0.7857 0.8580 0.6972 1.0000 0.7705  
## VWENX 0.7883 0.8327 0.6942 0.7705 1.0000

#Compared to sample correlation matrix  
cor(s4)

## AMECX DODGX FMILX FKINX VWENX  
## AMECX 1.0000 0.8874 0.7895 0.9191 0.9497  
## DODGX 0.8874 1.0000 0.8474 0.8967 0.8934  
## FMILX 0.7895 0.8474 1.0000 0.7460 0.8756  
## FKINX 0.9191 0.8967 0.7460 1.0000 0.8576  
## VWENX 0.9497 0.8934 0.8756 0.8576 1.0000

#the “similar” funds identified in Part (a) are highly correlated compared to other pairs of funds   
#the “different” funds identified in Part (a) have not a low return correlation compared to the other funds  
  
big5\_mean = apply(big5,2,mean)  
big5\_mean

## AMECX DODGX FMILX FKINX VWENX   
## 0.006822 0.008979 0.003640 0.004573 0.005443

big5\_sd = apply(big5,2,sd)  
big5\_sd

## AMECX DODGX FMILX FKINX VWENX   
## 0.02671 0.04071 0.04229 0.02655 0.02723

#Yes for similar funds they are simple and for different funds they are different  
  
# Based on these results, I concluded that the factor sensitivities are useful for identifying portfolios with similar properties because they show that the the properties are some how similar