FERM-504-Project_III.R

selim

2021-01-19

```
# I have tried doing homework completely on R.
# Necessary libraries for code
library(quantmod)
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: TTR
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
library(ggplot2)
library(readxl)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::first()  masks xts::first()
## x dplyr::lag()    masks stats::lag()
## x dplyr::last()   masks xts::last()
```

```
# Set working directory
setwd("~/Documents/Ozu/FERM 504/HW")

# Load data
data<- as.data.frame(read_excel("BESequityfundreturns.xlsx"))


# Create a summary data-frame for project
```

```r
results_df <- data.frame(matrix(NA, nrow=27, ncol=1))
rownames(results_df)<-colnames(data)[2:28]
colnames(results_df)<-"Perf2018"

# I. Best Performing funds in 2018
data_2018=data[49:60,2:28]
for (x in 1:27){
  results_df[x,1]=cumprod(1 + data_2018[,x])[12]-1
}

# All funds beat BIST100 return but none beat risk free return in 2018
results_df %>% filter(Perf2018 > results_df[26,1])
```

```
##               Perf2018
## AEB        -0.06162035
## AEH        -0.16578531
## AG3        -0.08582746
## AGH        -0.10755922
## AH5        -0.14638742
## AHB        -0.07276703
## ALH        -0.09090282
## ANS        -0.14558108
## AVH        -0.13376372
## AZH        -0.07268026
## BBH        -0.14381811
## BEH        -0.15481482
## BPH        -0.10964543
## CHH        -0.09051369
## EIH        -0.12788181
## GEG        -0.10890605
## GEH        -0.11110365
## GHH        -0.16329314
## HES        -0.11646251
## HHB        -0.16443402
## HHM        -0.11545495
## IEH        -0.14300392
## MHH        -0.09889221
## VEH        -0.12684245
## ZHB        -0.10200594
## kydon_brut  0.18443363
```

```r
results_df %>% filter(Perf2018 > results_df[26,1]) %>%dim()
```

```
## [1] 26  1
```

```r
# All of the funds have negative returns. This is not acceptable from my point of view.

# II. Funds performance statistics
# Deduct risk free return to find excess return for each fund
data_rfree=data
for (x in 2:27){
  data_rfree[,x]=data[,x]-data[,28]
}

# Calculate mean return, standard deviation of excess returns and portfolios
```

```r
mean_excess_return <-c()
sd_excess_return <-c()
mean_portfolio<-c()
sd_portfolio<-c()

for (x in 2:27){
  mean_excess_return=c(mean_excess_return,mean(data_rfree[,x]))
  sd_excess_return=c(sd_excess_return,sd(data_rfree[,x]))
  mean_portfolio=c(mean_portfolio,mean(data[,x]))
  sd_portfolio=c(sd_portfolio,sd(data[,x]))
}
# Calculate t statistic for risk free returns of each portfolio
t_stats <- mean_excess_return/sd_excess_return

# Calculate Sharpe ratio of each portfolio
# Multiply with sqrt(12) to annualize
sharpe_ratios=mean_excess_return / sd_portfolio * sqrt(12)

results_df<-as.data.frame(cbind(mean_excess_return,t_stats,sharpe_ratios))
rownames(results_df)<-colnames(data)[2:27]
results_df
```

```
##              mean_excess_return      t_stats sharpe_ratios
## AEB               2.305233e-03  0.045952313   0.159620381
## AEH              -8.381667e-05 -0.001651903  -0.005761016
## AG3               2.515283e-03  0.050098697   0.174480167
## AGH               8.596833e-04  0.020829716   0.073001665
## AH5               1.069967e-03  0.020970983   0.073152766
## AHB               4.309617e-03  0.082948030   0.288772877
## ALH               2.430467e-03  0.047222494   0.164393566
## ANS              -6.107833e-04 -0.012305627  -0.042933764
## AVH               5.050667e-04  0.010384834   0.036205227
## AZH               4.270550e-03  0.082307949   0.286549948
## BBH              -6.797000e-04 -0.013754946  -0.047969335
## BEH               7.031667e-05  0.001415110   0.004935846
## BPH               1.118717e-03  0.022316440   0.077715846
## CHH               5.082467e-03  0.097003452   0.337791698
## EIH               1.251683e-03  0.023369926   0.081325280
## GEG               2.672900e-03  0.053630946   0.186939662
## GEH               2.552867e-03  0.051077483   0.178035045
## GHH               1.410833e-03  0.026889347   0.093876661
## HES               1.797583e-03  0.035427343   0.123387824
## HHB              -7.787667e-04 -0.014188736  -0.049408191
## HHM               7.921667e-05  0.001911853   0.006738613
## IEH              -4.104667e-04 -0.007974612  -0.027787423
## MHH               6.101667e-05  0.001221289   0.004252610
## VEH              -6.663833e-04 -0.012595833  -0.043806073
## ZHB               1.347350e-03  0.026744248   0.093082580
## getiri_xu100     -6.936333e-04 -0.012439865  -0.043342776
```

```r
# How many of the funds have positive excess return?
results_df %>% filter(mean_excess_return > 0)
```

```
##     mean_excess_return     t_stats sharpe_ratios
## AEB       2.305233e-03 0.045952313   0.159620381
```

```
## AG3       2.515283e-03 0.050098697   0.174480167
## AGH       8.596833e-04 0.020829716   0.073001665
## AH5       1.069967e-03 0.020970983   0.073152766
## AHB       4.309617e-03 0.082948030   0.288772877
## ALH       2.430467e-03 0.047222494   0.164393566
## AVH       5.050667e-04 0.010384834   0.036205227
## AZH       4.270550e-03 0.082307949   0.286549948
## BEH       7.031667e-05 0.001415110   0.004935846
## BPH       1.118717e-03 0.022316440   0.077715846
## CHH       5.082467e-03 0.097003452   0.337791698
## EIH       1.251683e-03 0.023369926   0.081325280
## GEG       2.672900e-03 0.053630946   0.186939662
## GEH       2.552867e-03 0.051077483   0.178035045
## GHH       1.410833e-03 0.026889347   0.093876661
## HES       1.797583e-03 0.035427343   0.123387824
## HHM       7.921667e-05 0.001911853   0.006738613
## MHH       6.101667e-05 0.001221289   0.004252610
## ZHB       1.347350e-03 0.026744248   0.093082580
```

```
results_df %>% filter(mean_excess_return > 0) %>% dim()
```

```
## [1] 19  3
```

```
# 19 of 25 funds have positive excess return

# How many of the funds have statistically significant positive excess return?
# At 95% confidence level, t statistic must be greater than 2.01
results_df %>% filter(mean_excess_return>0, t_stats >2.01)
```

```
## [1] mean_excess_return t_stats            sharpe_ratios
## <0 rows> (or 0-length row.names)
```

```
# None of the funds has statistically significant positive excess return except for risk free asset

# First sort by excess returns
results_df[order(-results_df$mean_excess_return),]
```

```
##             mean_excess_return      t_stats sharpe_ratios
## CHH               5.082467e-03  0.097003452   0.337791698
## AHB               4.309617e-03  0.082948030   0.288772877
## AZH               4.270550e-03  0.082307949   0.286549948
## GEG               2.672900e-03  0.053630946   0.186939662
## GEH               2.552867e-03  0.051077483   0.178035045
## AG3               2.515283e-03  0.050098697   0.174480167
## ALH               2.430467e-03  0.047222494   0.164393566
## AEB               2.305233e-03  0.045952313   0.159620381
## HES               1.797583e-03  0.035427343   0.123387824
## GHH               1.410833e-03  0.026889347   0.093876661
## ZHB               1.347350e-03  0.026744248   0.093082580
## EIH               1.251683e-03  0.023369926   0.081325280
## BPH               1.118717e-03  0.022316440   0.077715846
## AH5               1.069967e-03  0.020970983   0.073152766
## AGH               8.596833e-04  0.020829716   0.073001665
## AVH               5.050667e-04  0.010384834   0.036205227
## HHM               7.921667e-05  0.001911853   0.006738613
## BEH               7.031667e-05  0.001415110   0.004935846
```

```
## MHH                6.101667e-05  0.001221289   0.004252610
## AEH               -8.381667e-05 -0.001651903  -0.005761016
## IEH               -4.104667e-04 -0.007974612  -0.027787423
## ANS               -6.107833e-04 -0.012305627  -0.042933764
## VEH               -6.663833e-04 -0.012595833  -0.043806073
## BBH               -6.797000e-04 -0.013754946  -0.047969335
## getiri_xu100      -6.936333e-04 -0.012439865  -0.043342776
## HHB               -7.787667e-04 -0.014188736  -0.049408191
```

```
# then  sort by Sharpe ratios
results_df[order(-results_df$sharpe_ratios),]
```

```
##              mean_excess_return      t_stats sharpe_ratios
## CHH                5.082467e-03  0.097003452   0.337791698
## AHB                4.309617e-03  0.082948030   0.288772877
## AZH                4.270550e-03  0.082307949   0.286549948
## GEG                2.672900e-03  0.053630946   0.186939662
## GEH                2.552867e-03  0.051077483   0.178035045
## AG3                2.515283e-03  0.050098697   0.174480167
## ALH                2.430467e-03  0.047222494   0.164393566
## AEB                2.305233e-03  0.045952313   0.159620381
## HES                1.797583e-03  0.035427343   0.123387824
## GHH                1.410833e-03  0.026889347   0.093876661
## ZHB                1.347350e-03  0.026744248   0.093082580
## EIH                1.251683e-03  0.023369926   0.081325280
## BPH                1.118717e-03  0.022316440   0.077715846
## AH5                1.069967e-03  0.020970983   0.073152766
## AGH                8.596833e-04  0.020829716   0.073001665
## AVH                5.050667e-04  0.010384834   0.036205227
## HHM                7.921667e-05  0.001911853   0.006738613
## BEH                7.031667e-05  0.001415110   0.004935846
## MHH                6.101667e-05  0.001221289   0.004252610
## AEH               -8.381667e-05 -0.001651903  -0.005761016
## IEH               -4.104667e-04 -0.007974612  -0.027787423
## ANS               -6.107833e-04 -0.012305627  -0.042933764
## getiri_xu100      -6.936333e-04 -0.012439865  -0.043342776
## VEH               -6.663833e-04 -0.012595833  -0.043806073
## BBH               -6.797000e-04 -0.013754946  -0.047969335
## HHB               -7.787667e-04 -0.014188736  -0.049408191
```

```
# Except for BIST 100, rankings do not change

# III. Make regression analysis for each stock against BIST100 returns
# Compute alpha, regression standard error, alpha t statistic and p value
alpha<-c()
reg_sd<-c()
alpha_t<-c()
alpha_p<-c()
for (x in 2:26){
  formula=paste0(colnames(data_rfree)[x],"~","getiri_xu100")
  regstats <- lm(formula=formula, data=data_rfree)
  alpha<-c(alpha,regstats$coefficients[1])
  reg_sd <-c(reg_sd,as.numeric(summary(regstats)[6]))
  alpha_t<-c(alpha_t,summary(regstats)[4][[1]][[5]])
  alpha_p<-c(alpha_p,summary(regstats)[4][[1]][[7]])
```

```
}

# Calculate Information ratio
inf_ratio=alpha/reg_sd*sqrt(12)

# Create a summary data-frame for regression statistics
regs_df <- data.frame(matrix(NA, nrow=25, ncol=5))
regs_df=as.data.frame(cbind(alpha,reg_sd,alpha_t,alpha_p,inf_ratio))
rownames(regs_df)=rownames(results_df)[1:25]
regs_df
```

```
##             alpha      reg_sd      alpha_t      alpha_p    inf_ratio
## AEB  2.919441e-03 0.008952966  2.525656411 1.430235e-02  1.129596758
## AEH  5.381975e-04 0.008695114  0.479410787 6.334492e-01  0.214415891
## AG3  3.134147e-03 0.006825422  3.556568810 7.561669e-04  1.590671075
## AGH  1.314844e-03 0.019259195  0.528783045 5.989738e-01  0.236497574
## AH5  1.698018e-03 0.007427463  1.770693354 8.186652e-02  0.791940449
## AHB  4.951078e-03 0.006413868  5.978898536 1.476616e-07  2.674055099
## ALH  3.065404e-03 0.006679864  3.554360824 7.613903e-04  1.589683556
## ANS  3.876780e-06 0.004750215  0.006321197 9.949781e-01  0.002827148
## AVH  1.101394e-03 0.008282070  1.030018851 3.072774e-01  0.460674679
## AZH  4.911235e-03 0.006342197  5.997805844 1.374360e-07  2.682511369
## BBH -7.043938e-05 0.006624719 -0.082354906 9.346481e-01 -0.036833132
## BEH  6.818372e-04 0.007312384  0.722209462 4.730676e-01  0.323007304
## BPH  1.729296e-03 0.010280703  1.302830361 1.977817e-01  0.582689294
## CHH  5.711640e-03 0.013797727  3.206230557 2.189006e-03  1.433982717
## EIH  1.886724e-03 0.016345358  0.894036901 3.749971e-01  0.399856916
## GEG  3.282139e-03 0.009319555  2.727741375 8.423792e-03  1.219979012
## GEH  3.163234e-03 0.009600767  2.551918590 1.337036e-02  1.141342486
## GHH  2.050330e-03 0.010588494  1.499792233 1.390923e-01  0.670780252
## HES  2.419634e-03 0.008681319  2.158764947 3.501842e-02  0.965504997
## HHB -1.057001e-04 0.009303257 -0.087999845 9.301799e-01 -0.039357823
## HHM  5.433581e-04 0.018174666  0.231558743 8.176958e-01  0.103564366
## IEH  2.218109e-04 0.008192378  0.209707703 8.346307e-01  0.093791515
## MHH  6.787756e-04 0.005525274  0.951511139 3.452934e-01  0.425562200
## VEH -2.051995e-05 0.010254849 -0.015498455 9.876877e-01 -0.006931665
## ZHB  1.963880e-03 0.009120446  1.667786139 1.007491e-01  0.745915322
```

```
# How many of the funds does have positive alpha?
regs_df %>% filter(alpha > 0)
```

```
##            alpha      reg_sd     alpha_t      alpha_p   inf_ratio
## AEB 2.919441e-03 0.008952966 2.525656411 1.430235e-02 1.129596758
## AEH 5.381975e-04 0.008695114 0.479410787 6.334492e-01 0.214415891
## AG3 3.134147e-03 0.006825422 3.556568810 7.561669e-04 1.590671075
## AGH 1.314844e-03 0.019259195 0.528783045 5.989738e-01 0.236497574
## AH5 1.698018e-03 0.007427463 1.770693354 8.186652e-02 0.791940449
## AHB 4.951078e-03 0.006413868 5.978898536 1.476616e-07 2.674055099
## ALH 3.065404e-03 0.006679864 3.554360824 7.613903e-04 1.589683556
## ANS 3.876780e-06 0.004750215 0.006321197 9.949781e-01 0.002827148
## AVH 1.101394e-03 0.008282070 1.030018851 3.072774e-01 0.460674679
## AZH 4.911235e-03 0.006342197 5.997805844 1.374360e-07 2.682511369
## BEH 6.818372e-04 0.007312384 0.722209462 4.730676e-01 0.323007304
## BPH 1.729296e-03 0.010280703 1.302830361 1.977817e-01 0.582689294
```

```
## CHH 5.711640e-03 0.013797727 3.206230557 2.189006e-03 1.433982717
## EIH 1.886724e-03 0.016345358 0.894036901 3.749971e-01 0.399856916
## GEG 3.282139e-03 0.009319555 2.727741375 8.423792e-03 1.219979012
## GEH 3.163234e-03 0.009600767 2.551918590 1.337036e-02 1.141342486
## GHH 2.050330e-03 0.010588494 1.499792233 1.390923e-01 0.670780252
## HES 2.419634e-03 0.008681319 2.158764947 3.501842e-02 0.965504997
## HHM 5.433581e-04 0.018174666 0.231558743 8.176958e-01 0.103564366
## IEH 2.218109e-04 0.008192378 0.209707703 8.346307e-01 0.093791515
## MHH 6.787756e-04 0.005525274 0.951511139 3.452934e-01 0.425562200
## ZHB 1.963880e-03 0.009120446 1.667786139 1.007491e-01 0.745915322
```

```
regs_df %>% filter(alpha > 0) %>% dim()
```

```
## [1] 22  5
```

```
# 22 of 25 funds have positive alpha

# How many of the funds have statistically significant positive alpha?
# at 95% confidence level
regs_df %>% filter(alpha>0, alpha_p < 0.05)
```

```
##           alpha      reg_sd  alpha_t      alpha_p inf_ratio
## AEB 0.002919441 0.008952966 2.525656 1.430235e-02  1.129597
## AG3 0.003134147 0.006825422 3.556569 7.561669e-04  1.590671
## AHB 0.004951078 0.006413868 5.978899 1.476616e-07  2.674055
## ALH 0.003065404 0.006679864 3.554361 7.613903e-04  1.589684
## AZH 0.004911235 0.006342197 5.997806 1.374360e-07  2.682511
## CHH 0.005711640 0.013797727 3.206231 2.189006e-03  1.433983
## GEG 0.003282139 0.009319555 2.727741 8.423792e-03  1.219979
## GEH 0.003163234 0.009600767 2.551919 1.337036e-02  1.141342
## HES 0.002419634 0.008681319 2.158765 3.501842e-02  0.965505
```

```
regs_df %>% filter(alpha>0, alpha_p < 0.05) %>% dim()
```

```
## [1] 9 5
```

```
# 9 of the funds have statistically significant positive alpha

# Sorting of funds by their alpha
regs_df[order(-regs_df$alpha),]
```

```
##             alpha      reg_sd      alpha_t      alpha_p    inf_ratio
## CHH  5.711640e-03 0.013797727  3.206230557 2.189006e-03  1.433982717
## AHB  4.951078e-03 0.006413868  5.978898536 1.476616e-07  2.674055099
## AZH  4.911235e-03 0.006342197  5.997805844 1.374360e-07  2.682511369
## GEG  3.282139e-03 0.009319555  2.727741375 8.423792e-03  1.219979012
## GEH  3.163234e-03 0.009600767  2.551918590 1.337036e-02  1.141342486
## AG3  3.134147e-03 0.006825422  3.556568810 7.561669e-04  1.590671075
## ALH  3.065404e-03 0.006679864  3.554360824 7.613903e-04  1.589683556
## AEB  2.919441e-03 0.008952966  2.525656411 1.430235e-02  1.129596758
## HES  2.419634e-03 0.008681319  2.158764947 3.501842e-02  0.965504997
## GHH  2.050330e-03 0.010588494  1.499792233 1.390923e-01  0.670780252
## ZHB  1.963880e-03 0.009120446  1.667786139 1.007491e-01  0.745915322
## EIH  1.886724e-03 0.016345358  0.894036901 3.749971e-01  0.399856916
## BPH  1.729296e-03 0.010280703  1.302830361 1.977817e-01  0.582689294
## AH5  1.698018e-03 0.007427463  1.770693354 8.186652e-02  0.791940449
## AGH  1.314844e-03 0.019259195  0.528783045 5.989738e-01  0.236497574
```

```
## AVH  1.101394e-03 0.008282070  1.030018851 3.072774e-01  0.460674679
## BEH  6.818372e-04 0.007312384  0.722209462 4.730676e-01  0.323007304
## MHH  6.787756e-04 0.005525274  0.951511139 3.452934e-01  0.425562200
## HHM  5.433581e-04 0.018174666  0.231558743 8.176958e-01  0.103564366
## AEH  5.381975e-04 0.008695114  0.479410787 6.334492e-01  0.214415891
## IEH  2.218109e-04 0.008192378  0.209707703 8.346307e-01  0.093791515
## ANS  3.876780e-06 0.004750215  0.006321197 9.949781e-01  0.002827148
## VEH -2.051995e-05 0.010254849 -0.015498455 9.876877e-01 -0.006931665
## BBH -7.043938e-05 0.006624719 -0.082354906 9.346481e-01 -0.036833132
## HHB -1.057001e-04 0.009303257 -0.087999845 9.301799e-01 -0.039357823
```

```
# Sorting of funds by their information ratio
regs_df[order(-regs_df$inf_ratio),]
```

```
##             alpha      reg_sd      alpha_t      alpha_p    inf_ratio
## AZH  4.911235e-03 0.006342197  5.997805844 1.374360e-07  2.682511369
## AHB  4.951078e-03 0.006413868  5.978898536 1.476616e-07  2.674055099
## AG3  3.134147e-03 0.006825422  3.556568810 7.561669e-04  1.590671075
## ALH  3.065404e-03 0.006679864  3.554360824 7.613903e-04  1.589683556
## CHH  5.711640e-03 0.013797727  3.206230557 2.189006e-03  1.433982717
## GEG  3.282139e-03 0.009319555  2.727741375 8.423792e-03  1.219979012
## GEH  3.163234e-03 0.009600767  2.551918590 1.337036e-02  1.141342486
## AEB  2.919441e-03 0.008952966  2.525656411 1.430235e-02  1.129596758
## HES  2.419634e-03 0.008681319  2.158764947 3.501842e-02  0.965504997
## AH5  1.698018e-03 0.007427463  1.770693354 8.186652e-02  0.791940449
## ZHB  1.963880e-03 0.009120446  1.667786139 1.007491e-01  0.745915322
## GHH  2.050330e-03 0.010588494  1.499792233 1.390923e-01  0.670780252
## BPH  1.729296e-03 0.010280703  1.302830361 1.977817e-01  0.582689294
## AVH  1.101394e-03 0.008282070  1.030018851 3.072774e-01  0.460674679
## MHH  6.787756e-04 0.005525274  0.951511139 3.452934e-01  0.425562200
## EIH  1.886724e-03 0.016345358  0.894036901 3.749971e-01  0.399856916
## BEH  6.818372e-04 0.007312384  0.722209462 4.730676e-01  0.323007304
## AGH  1.314844e-03 0.019259195  0.528783045 5.989738e-01  0.236497574
## AEH  5.381975e-04 0.008695114  0.479410787 6.334492e-01  0.214415891
## HHM  5.433581e-04 0.018174666  0.231558743 8.176958e-01  0.103564366
## IEH  2.218109e-04 0.008192378  0.209707703 8.346307e-01  0.093791515
## ANS  3.876780e-06 0.004750215  0.006321197 9.949781e-01  0.002827148
## VEH -2.051995e-05 0.010254849 -0.015498455 9.876877e-01 -0.006931665
## BBH -7.043938e-05 0.006624719 -0.082354906 9.346481e-01 -0.036833132
## HHB -1.057001e-04 0.009303257 -0.087999845 9.301799e-01 -0.039357823
```

```
# Sorting by Information ratio gives very different results from sorting by alpha

### PART IV
# Repeat PART II and III by half data
data=data[1:30,]

# II. Funds performance statistics
# Deduct risk free return to find excess return for each fund
data_rfree=data
for (x in 2:27){
  data_rfree[,x]=data[,x]-data[,28]
}

# Calculate mean return, standard deviation of excess returns and portfolios
```

```r
mean_excess_return <-c()
sd_excess_return <-c()
mean_portfolio<-c()
sd_portfolio<-c()

for (x in 2:27){
  mean_excess_return=c(mean_excess_return,mean(data_rfree[,x]))
  sd_excess_return=c(sd_excess_return,sd(data_rfree[,x]))
  mean_portfolio=c(mean_portfolio,mean(data[,x]))
  sd_portfolio=c(sd_portfolio,sd(data[,x]))
}
# Calculate t statistic for risk free returns of each portfolio
t_stats <- mean_excess_return/sd_excess_return

# Calculate Sharpe ratio of each portfolio
# Multiply with sqrt(12) to annualize
sharpe_ratios=mean_excess_return / sd_portfolio * sqrt(12)

results_df<-as.data.frame(cbind(mean_excess_return,t_stats,sharpe_ratios))
rownames(results_df)<-colnames(data)[2:27]
results_df
```

```
##              mean_excess_return       t_stats sharpe_ratios
## AEB               2.933667e-04  0.0060748339   0.020932227
## AEH               3.021000e-04  0.0062555837   0.021561697
## AG3               1.578600e-03  0.0306952946   0.105831284
## AGH               6.272167e-03  0.1656680341   0.570303920
## AH5               1.307067e-03  0.0254169576   0.087641355
## AHB               4.112667e-03  0.0786045945   0.270982264
## ALH               7.190333e-04  0.0139280663   0.048016514
## ANS               8.995000e-04  0.0181371752   0.062528527
## AVH               3.986000e-04  0.0083036242   0.028621400
## AZH               4.061733e-03  0.0777124193   0.267910030
## BBH              -8.832667e-04 -0.0180618547  -0.062250702
## BEH               4.817667e-04  0.0099529544   0.034306733
## BPH               1.249167e-03  0.0251607030   0.086759603
## CHH               4.066133e-03  0.0750179701   0.258952726
## EIH               4.547100e-03  0.0890043851   0.306789074
## GEG               3.268167e-03  0.0658457513   0.227133149
## GEH               2.983433e-03  0.0602468933   0.207826762
## GHH               3.121367e-03  0.0627638459   0.216518933
## HES               1.856067e-03  0.0361465870   0.124630412
## HHB               7.513333e-04  0.0136530065   0.047109727
## HHM               2.336433e-03  0.0596179479   0.205334864
## IEH               1.502667e-03  0.0290870065   0.100241646
## MHH              -2.415333e-04 -0.0047906601  -0.016517809
## VEH              -7.674000e-04 -0.0139821327  -0.048232298
## ZHB              -3.069000e-04 -0.0059030749  -0.020360120
## getiri_xu100     -2.566667e-05 -0.0004582061  -0.001580249
```

```r
# How many of the funds have positive excess return?
results_df %>% filter(mean_excess_return > 0)
```

```
##     mean_excess_return     t_stats sharpe_ratios
## AEB       0.0002933667 0.006074834    0.02093223
```

```
## AEH       0.0003021000 0.006255584    0.02156170
## AG3       0.0015786000 0.030695295    0.10583128
## AGH       0.0062721667 0.165668034    0.57030392
## AH5       0.0013070667 0.025416958    0.08764136
## AHB       0.0041126667 0.078604595    0.27098226
## ALH       0.0007190333 0.013928066    0.04801651
## ANS       0.0008995000 0.018137175    0.06252853
## AVH       0.0003986000 0.008303624    0.02862140
## AZH       0.0040617333 0.077712419    0.26791003
## BEH       0.0004817667 0.009952954    0.03430673
## BPH       0.0012491667 0.025160703    0.08675960
## CHH       0.0040661333 0.075017970    0.25895273
## EIH       0.0045471000 0.089004385    0.30678907
## GEG       0.0032681667 0.065845751    0.22713315
## GEH       0.0029834333 0.060246893    0.20782676
## GHH       0.0031213667 0.062763846    0.21651893
## HES       0.0018560667 0.036146587    0.12463041
## HHB       0.0007513333 0.013653007    0.04710973
## HHM       0.0023364333 0.059617948    0.20533486
## IEH       0.0015026667 0.029087006    0.10024165
```

```
results_df %>% filter(mean_excess_return > 0) %>% dim()
```

```
## [1] 21  3
```

```
# 21 of 25 funds have positive excess return

# How many of the funds have statistically significant positive excess return?
# At 95% confidence level, t statistic must be greater than 2.01
results_df %>% filter(mean_excess_return>0, t_stats >2.01)
```

```
## [1] mean_excess_return t_stats            sharpe_ratios
## <0 rows> (or 0-length row.names)
```

```
# None of the funds has statistically significant positive excess return except for risk free asset

# First sort by excess returns
results_df[order(-results_df$mean_excess_return),]
```

```
##              mean_excess_return       t_stats sharpe_ratios
## AGH               6.272167e-03  0.1656680341   0.570303920
## EIH               4.547100e-03  0.0890043851   0.306789074
## AHB               4.112667e-03  0.0786045945   0.270982264
## CHH               4.066133e-03  0.0750179701   0.258952726
## AZH               4.061733e-03  0.0777124193   0.267910030
## GEG               3.268167e-03  0.0658457513   0.227133149
## GHH               3.121367e-03  0.0627638459   0.216518933
## GEH               2.983433e-03  0.0602468933   0.207826762
## HHM               2.336433e-03  0.0596179479   0.205334864
## HES               1.856067e-03  0.0361465870   0.124630412
## AG3               1.578600e-03  0.0306952946   0.105831284
## IEH               1.502667e-03  0.0290870065   0.100241646
## AH5               1.307067e-03  0.0254169576   0.087641355
## BPH               1.249167e-03  0.0251607030   0.086759603
## ANS               8.995000e-04  0.0181371752   0.062528527
## HHB               7.513333e-04  0.0136530065   0.047109727
```

```
## ALH               7.190333e-04  0.0139280663   0.048016514
## BEH               4.817667e-04  0.0099529544   0.034306733
## AVH               3.986000e-04  0.0083036242   0.028621400
## AEH               3.021000e-04  0.0062555837   0.021561697
## AEB               2.933667e-04  0.0060748339   0.020932227
## getiri_xu100     -2.566667e-05 -0.0004582061  -0.001580249
## MHH              -2.415333e-04 -0.0047906601  -0.016517809
## ZHB              -3.069000e-04 -0.0059030749  -0.020360120
## VEH              -7.674000e-04 -0.0139821327  -0.048232298
## BBH              -8.832667e-04 -0.0180618547  -0.062250702
```

```
# then  sort by Sharpe ratios
results_df[order(-results_df$sharpe_ratios),]
```

```
##              mean_excess_return       t_stats sharpe_ratios
## AGH               6.272167e-03  0.1656680341   0.570303920
## EIH               4.547100e-03  0.0890043851   0.306789074
## AHB               4.112667e-03  0.0786045945   0.270982264
## AZH               4.061733e-03  0.0777124193   0.267910030
## CHH               4.066133e-03  0.0750179701   0.258952726
## GEG               3.268167e-03  0.0658457513   0.227133149
## GHH               3.121367e-03  0.0627638459   0.216518933
## GEH               2.983433e-03  0.0602468933   0.207826762
## HHM               2.336433e-03  0.0596179479   0.205334864
## HES               1.856067e-03  0.0361465870   0.124630412
## AG3               1.578600e-03  0.0306952946   0.105831284
## IEH               1.502667e-03  0.0290870065   0.100241646
## AH5               1.307067e-03  0.0254169576   0.087641355
## BPH               1.249167e-03  0.0251607030   0.086759603
## ANS               8.995000e-04  0.0181371752   0.062528527
## ALH               7.190333e-04  0.0139280663   0.048016514
## HHB               7.513333e-04  0.0136530065   0.047109727
## BEH               4.817667e-04  0.0099529544   0.034306733
## AVH               3.986000e-04  0.0083036242   0.028621400
## AEH               3.021000e-04  0.0062555837   0.021561697
## AEB               2.933667e-04  0.0060748339   0.020932227
## getiri_xu100     -2.566667e-05 -0.0004582061  -0.001580249
## MHH              -2.415333e-04 -0.0047906601  -0.016517809
## ZHB              -3.069000e-04 -0.0059030749  -0.020360120
## VEH              -7.674000e-04 -0.0139821327  -0.048232298
## BBH              -8.832667e-04 -0.0180618547  -0.062250702
```

```
# Except for BIST 100 and 4th and 5th ranked funds, rankings do not change

# III. Make regression analysis for each stock against BIST100 returns
# Compute alpha, regression standard error, alpha t statistic and p value
alpha<-c()
reg_sd<-c()
alpha_t<-c()
alpha_p<-c()
for (x in 2:26){
  formula=paste0(colnames(data_rfree)[x],"~","getiri_xu100")
  regstats <- lm(formula=formula, data=data_rfree)
  alpha<-c(alpha,regstats$coefficients[1])
  reg_sd <-c(reg_sd,as.numeric(summary(regstats)[6]))
```

```
  alpha_t<-c(alpha_t,summary(regstats)[4][[1]][[5]])
  alpha_p<-c(alpha_p,summary(regstats)[4][[1]][[7]])
}

# Calculate Information ratio
inf_ratio=alpha/reg_sd*sqrt(12)

# Create a summary data-frame for regression statistics
regs_df <- data.frame(matrix(NA, nrow=25, ncol=5))
regs_df=as.data.frame(cbind(alpha,reg_sd,alpha_t,alpha_p,inf_ratio))
rownames(regs_df)=rownames(results_df)[1:25]
regs_df
```

```
##             alpha      reg_sd    alpha_t      alpha_p  inf_ratio
## AEB  0.0003152360 0.007488788  0.2305605 0.8193325690  0.1458193
## AEH  0.0003239431 0.007862991  0.2256532 0.8231101503  0.1427156
## AG3  0.0016019440 0.007146169  1.2278197 0.2297368317  0.7765415
## AGH  0.0062874142 0.018376300  1.8740215 0.0713978482  1.1852354
## AH5  0.0013303877 0.007484661  0.9735688 0.3386069456  0.6157391
## AHB  0.0041365079 0.005590383  4.0527788 0.0003646994  2.5632026
## ALH  0.0007425985 0.004569430  0.8901284 0.3809815973  0.5629667
## ANS  0.0009221174 0.004892555  1.0323123 0.3107605291  0.6528917
## AVH  0.0004203093 0.007852847  0.2931585 0.7715628062  0.1854097
## AZH  0.0040855535 0.005504371  4.0654045 0.0003525299  2.5711878
## BBH -0.0008611038 0.007331416 -0.6433217 0.5252495749 -0.4068724
## BEH  0.0005037122 0.007131952  0.3868429 0.7017970513  0.2446609
## BPH  0.0012714977 0.009639238  0.7224927 0.4759832624  0.4569445
## CHH  0.0040900152 0.015140878  1.4795663 0.1501531403  0.9357600
## EIH  0.0045692756 0.016654299  1.5027322 0.1441037598  0.9504114
## GEG  0.0032905131 0.009385919  1.9202041 0.0650715190  1.2144439
## GEH  0.0030057141 0.009533665  1.7268252 0.0952177599  1.0921402
## GHH  0.0031437694 0.009260603  1.8593966 0.0735090712  1.1759858
## HES  0.0018793272 0.007859309  1.3097205 0.2009353591  0.8283401
## HHB  0.0007762589 0.008466646  0.5021758 0.6194703705  0.3176039
## HHM  0.0023522712 0.018796777  0.6854323 0.4987071607  0.4335055
## IEH  0.0015261041 0.007373952  1.1335598 0.2665920817  0.7169262
## MHH -0.0002184893 0.003622383 -0.3303669 0.7435820372 -0.2089424
## VEH -0.0007425419 0.008461315 -0.4806663 0.6344881928 -0.3040001
## ZHB -0.0002833447 0.007895549 -0.1965591 0.8455922776 -0.1243149
```

```
# How many of the funds does have positive alpha?
regs_df %>% filter(alpha > 0)
```

```
##            alpha      reg_sd   alpha_t      alpha_p inf_ratio
## AEB 0.0003152360 0.007488788 0.2305605 0.8193325690 0.1458193
## AEH 0.0003239431 0.007862991 0.2256532 0.8231101503 0.1427156
## AG3 0.0016019440 0.007146169 1.2278197 0.2297368317 0.7765415
## AGH 0.0062874142 0.018376300 1.8740215 0.0713978482 1.1852354
## AH5 0.0013303877 0.007484661 0.9735688 0.3386069456 0.6157391
## AHB 0.0041365079 0.005590383 4.0527788 0.0003646994 2.5632026
## ALH 0.0007425985 0.004569430 0.8901284 0.3809815973 0.5629667
## ANS 0.0009221174 0.004892555 1.0323123 0.3107605291 0.6528917
## AVH 0.0004203093 0.007852847 0.2931585 0.7715628062 0.1854097
## AZH 0.0040855535 0.005504371 4.0654045 0.0003525299 2.5711878
```

```
## BEH 0.0005037122 0.007131952 0.3868429 0.7017970513 0.2446609
## BPH 0.0012714977 0.009639238 0.7224927 0.4759832624 0.4569445
## CHH 0.0040900152 0.015140878 1.4795663 0.1501531403 0.9357600
## EIH 0.0045692756 0.016654299 1.5027322 0.1441037598 0.9504114
## GEG 0.0032905131 0.009385919 1.9202041 0.0650715190 1.2144439
## GEH 0.0030057141 0.009533665 1.7268252 0.0952177599 1.0921402
## GHH 0.0031437694 0.009260603 1.8593966 0.0735090712 1.1759858
## HES 0.0018793272 0.007859309 1.3097205 0.2009353591 0.8283401
## HHB 0.0007762589 0.008466646 0.5021758 0.6194703705 0.3176039
## HHM 0.0023522712 0.018796777 0.6854323 0.4987071607 0.4335055
## IEH 0.0015261041 0.007373952 1.1335598 0.2665920817 0.7169262
```

```
regs_df %>% filter(alpha > 0) %>% dim()
```

```
## [1] 21  5
```

```
# 21 of 25 funds have positive alpha

# How many of the funds have statistically significant positive alpha?
# at 95% confidence level
regs_df %>% filter(alpha>0, alpha_p < 0.05)
```

```
##           alpha      reg_sd  alpha_t      alpha_p inf_ratio
## AHB 0.004136508 0.005590383 4.052779 0.0003646994  2.563203
## AZH 0.004085553 0.005504371 4.065404 0.0003525299  2.571188
```

```
regs_df %>% filter(alpha>0, alpha_p < 0.05) %>% dim()
```

```
## [1] 2 5
```

```
# 2 of the funds have statistically significant positive alpha

# Sorting of funds by their alpha
regs_df[order(-regs_df$alpha),]
```

```
##             alpha      reg_sd    alpha_t      alpha_p  inf_ratio
## AGH  0.0062874142 0.018376300  1.8740215 0.0713978482  1.1852354
## EIH  0.0045692756 0.016654299  1.5027322 0.1441037598  0.9504114
## AHB  0.0041365079 0.005590383  4.0527788 0.0003646994  2.5632026
## CHH  0.0040900152 0.015140878  1.4795663 0.1501531403  0.9357600
## AZH  0.0040855535 0.005504371  4.0654045 0.0003525299  2.5711878
## GEG  0.0032905131 0.009385919  1.9202041 0.0650715190  1.2144439
## GHH  0.0031437694 0.009260603  1.8593966 0.0735090712  1.1759858
## GEH  0.0030057141 0.009533665  1.7268252 0.0952177599  1.0921402
## HHM  0.0023522712 0.018796777  0.6854323 0.4987071607  0.4335055
## HES  0.0018793272 0.007859309  1.3097205 0.2009353591  0.8283401
## AG3  0.0016019440 0.007146169  1.2278197 0.2297368317  0.7765415
## IEH  0.0015261041 0.007373952  1.1335598 0.2665920817  0.7169262
## AH5  0.0013303877 0.007484661  0.9735688 0.3386069456  0.6157391
## BPH  0.0012714977 0.009639238  0.7224927 0.4759832624  0.4569445
## ANS  0.0009221174 0.004892555  1.0323123 0.3107605291  0.6528917
## HHB  0.0007762589 0.008466646  0.5021758 0.6194703705  0.3176039
## ALH  0.0007425985 0.004569430  0.8901284 0.3809815973  0.5629667
## BEH  0.0005037122 0.007131952  0.3868429 0.7017970513  0.2446609
## AVH  0.0004203093 0.007852847  0.2931585 0.7715628062  0.1854097
## AEH  0.0003239431 0.007862991  0.2256532 0.8231101503  0.1427156
## AEB  0.0003152360 0.007488788  0.2305605 0.8193325690  0.1458193
```

```
## MHH -0.0002184893 0.003622383 -0.3303669 0.7435820372 -0.2089424
## ZHB -0.0002833447 0.007895549 -0.1965591 0.8455922776 -0.1243149
## VEH -0.0007425419 0.008461315 -0.4806663 0.6344881928 -0.3040001
## BBH -0.0008611038 0.007331416 -0.6433217 0.5252495749 -0.4068724
```

```
# Sorting of funds by their information ratio
regs_df[order(-regs_df$inf_ratio),]
```

```
##             alpha      reg_sd    alpha_t      alpha_p  inf_ratio
## AZH  0.0040855535 0.005504371  4.0654045 0.0003525299  2.5711878
## AHB  0.0041365079 0.005590383  4.0527788 0.0003646994  2.5632026
## GEG  0.0032905131 0.009385919  1.9202041 0.0650715190  1.2144439
## AGH  0.0062874142 0.018376300  1.8740215 0.0713978482  1.1852354
## GHH  0.0031437694 0.009260603  1.8593966 0.0735090712  1.1759858
## GEH  0.0030057141 0.009533665  1.7268252 0.0952177599  1.0921402
## EIH  0.0045692756 0.016654299  1.5027322 0.1441037598  0.9504114
## CHH  0.0040900152 0.015140878  1.4795663 0.1501531403  0.9357600
## HES  0.0018793272 0.007859309  1.3097205 0.2009353591  0.8283401
## AG3  0.0016019440 0.007146169  1.2278197 0.2297368317  0.7765415
## IEH  0.0015261041 0.007373952  1.1335598 0.2665920817  0.7169262
## ANS  0.0009221174 0.004892555  1.0323123 0.3107605291  0.6528917
## AH5  0.0013303877 0.007484661  0.9735688 0.3386069456  0.6157391
## ALH  0.0007425985 0.004569430  0.8901284 0.3809815973  0.5629667
## BPH  0.0012714977 0.009639238  0.7224927 0.4759832624  0.4569445
## HHM  0.0023522712 0.018796777  0.6854323 0.4987071607  0.4335055
## HHB  0.0007762589 0.008466646  0.5021758 0.6194703705  0.3176039
## BEH  0.0005037122 0.007131952  0.3868429 0.7017970513  0.2446609
## AVH  0.0004203093 0.007852847  0.2931585 0.7715628062  0.1854097
## AEB  0.0003152360 0.007488788  0.2305605 0.8193325690  0.1458193
## AEH  0.0003239431 0.007862991  0.2256532 0.8231101503  0.1427156
## ZHB -0.0002833447 0.007895549 -0.1965591 0.8455922776 -0.1243149
## MHH -0.0002184893 0.003622383 -0.3303669 0.7435820372 -0.2089424
## VEH -0.0007425419 0.008461315 -0.4806663 0.6344881928 -0.3040001
## BBH -0.0008611038 0.007331416 -0.6433217 0.5252495749 -0.4068724
```

```
# Sorting by Information ratio gives very different results from sorting by alpha

# I select fund AGH as it has the highest mean excess return
# Load data for second half of AGH
data<- as.data.frame(read_excel("BESequityfundreturns.xlsx"))
data=data[31:60,]
data=data["AGH"]

# 1000 TL becomes
return_agh=cumprod(1+data)*1000
return_agh[30,1]
```

```
## [1] 1173.404
```

```
# 1173.40 TL in the second half
```