Portfolio with Rish Free Rate

YIK LUN, KEI
allen29@ucla.edu

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

```r
symbols <-c("BABA","IBM","VIPS","GOOG","MSFT","GE","WFC","AMZN","JD")
days<-252
Rf <- 0.05
stock <- list();daily.return<-list()
fraction<-matrix(NA,nrow=length(symbols),ncol=1)

for (i in 1:length(symbols)) {
  stock[[i]] <- tail(getSymbols(symbols[i],src="yahoo",return.class="xts",auto.assign=FALSE),days)
  daily.return[[i]] <- periodReturn(stock[[i]],period='monthly')
}
```

```
##     As of 0.4-0, 'getSymbols' uses env=parent.frame() and
##  auto.assign=TRUE by default.
##
##  This  behavior  will be  phased out in 0.5-0  when the call  will
##  default to use auto.assign=FALSE. getOption("getSymbols.env") and
##  getOptions("getSymbols.auto.assign") are now checked for alternate defaults
##
##  This message is shown once per session and may be disabled by setting
##  options("getSymbols.warning4.0"=FALSE). See ?getSymbols for more details.
```

```r
daily.return<-as.data.frame(daily.return)
colnames(daily.return)<-symbols
R_ibar <- as.matrix(colMeans(daily.return))
var_covar <- cov(daily.return)
var_covar_inv <- solve(var_covar)
R <- R_ibar-Rf
z <- var_covar_inv %*% R
x <- z/sum(z)
R_Gbar <- t(x) %*% R_ibar
var_G <- t(x) %*% var_covar %*% x
sd_G <- var_G^0.5
x
```

```
##               [,1]
## BABA  0.11142001
## IBM   1.35814193
## VIPS -0.01562768
## GOOG  0.52882847
## MSFT -0.58215475
## GE    0.62511405
## WFC  -0.60484114
## AMZN -0.25792129
## JD   -0.16295959
```

R_Gbar

```
##               [,1]
## [1,] -0.02617691
```

sd_G

```
##              [,1]
## [1,] 0.02424271
```

```r
plot(sd_G,R_Gbar,col="red",pch=19,xlab="Risk",ylab="Return",
     ylim=c(min(colMeans(daily.return),R_Gbar),max(colMeans(daily.return),R_Gbar)),
     xlim=c(min(sqrt(diag(var(daily.return)))),sd_G),max(sqrt(diag(var(daily.return)))),sd_G)))
abline(h=0,col="red")
for (i in 1:length(symbols)){
points(sd(daily.return[,i]),mean(daily.return[,i]),col="blue",pch=17)
text(sd(daily.return[,i]),mean(daily.return[,i]),colnames(daily.return)[i])
}
```