Introduction to the package stockPortfolio

As short introduction of the package stockPortfolio.

#===> ((((( 1 )))) ) <===#
#===> quick example ) <===#
# load package
library(stockPortfolio)

# select stocks
#IBM: International Business Machines Copr.
#WFC: Wells Fargo & Comnpay
#JPM: JPMorgan Chase & Company
#LUV: Southwest Airlines Co.
#XOM: Exxon Mobil Corporation

ticker <- c("IBM", "WFC", "JPM", "LUV", "XOM")

#Get stock data
gr <- getReturns(ticker, start="2005-03-31", end="2010-03-31")

#gr is a "list" object and we find what it contains by typing the following:
names(gr)

#We can access each component of gr by typing:
gr$R
gr$ticker
etc.

#Obtain the variance-covariance matrix of the returns:
> cov(gr$R)
                  IBM       WFC       JPM       LUV      XOM
IBM  0.0040317283  0.0011198635  0.0019111776  0.0018376731  0.0005219823
WFC  0.0011198635  0.0140009223  0.0090748082  0.0045614185  0.0002158645
JPM  0.0019111776  0.0090748082  0.0087172088  0.0040831109 -0.0001559581
LUV  0.0018376731  0.0045614185  0.0040831109  0.0082593478 -0.0004506132
XOM  0.0005219823  0.0002158645 -0.0001559581 -0.0004506132  0.0026259464

#Obtain the correlation matrix of the returns:
cor(gr$R)

#We can find summary statistics as follows:
summary(gr$R)

#To find the mean, variance, and standard deviation of a particular stock
mean(gr$R[,4])
var(gr$R[,4])
sd(gr$R[,4])
#To find the means of all five stocks:
xx <- as.data.frame(gr$R)
mean(xx)

#To find the covariance and correlation between two stocks:
cov(gr$R[,4], gr$R[,5])
cor(gr$R[,4], gr$R[,5])

Use two stocks to find the minimum risk portfolio (its composition, expected return, and standard deviation). Note: The following lines until the end of this page can be done outside the package!

Let’s work with two stocks: IBM and LUV.

#Find the composition of the minimum risk portfolio:
x_IBM <- (var(gr$R[,4]) - cov(gr$R[,1], gr$R[,4])) / 
          (var(gr$R[,1]) + var(gr$R[,4]) - 2*cov(gr$R[,1], gr$R[,4]))
x_LUV <- 1-x_IBM

# Find the mean and sd of the minimum risk portfolio:
mean_min <- x_IBM*mean(gr$R[,1]) + x_LUV*mean(gr$R[,4])

var_min <- x_IBM^2*var(gr$R[,1]) + x_LUV^2*var(gr$R[,4]) + 
          2*x_IBM*x_LUV*cov(gr$R[,1], gr$R[,4])

sd_min <- var_min^0.5

# Construct the portfolio possibilities curve and identify the efficient frontier:
a <- seq(0,1,.01)
b <- 1-a
mean_p <- a*mean(gr$R[,1]) + b*mean(gr$R[,4])
var_p <- a^2*var(gr$R[,1]) + b^2*var(gr$R[,4]) + 
          2*a*b*cov(gr$R[,1], gr$R[,4])

sd_p <- var_p^0.5

plot(sd_p,mean_p, type="l", xlab="Portfolio standard deviation (risk)", 
ylab="Portfolio expected return")
points(sd_min, mean_min, pch=19, col="green")

# Identify the efficient frontier:
xx <- cbind(sd_p,mean_p)
xxx <- xx[which(xx[,2]>mean_min),]
points(xxx, type="l", col="blue", lwd=3)
We will use now the stockPortfolio package to find the point of tangency (point \( G \)):

**Step 1:** Select a model. At this point we only know one model (short sales + risk free rate). We call this no model, denoted with "none".

```r
port_model <- stockModel(gr, model="none", Rf=0.001)
```

**Step 2:** Optimize, i.e. find the composition of the point of tangency.

```r
op <- optimalPort(port_model)
```

**Note:** \( op \) is a class of "optimalPortfolio" and contains the following:
- **model**: The model used (here "none").
- **X**: The composition of the point of tangency.
- **R**: The expected return of the point of tangency.
- **risk**: The standard deviation of the point of tangency.

All the above information can be accessed using \( op\$\text{name} \), where name is one of (model, X, R, risk).

For example:

```r
> op$X
     IBM    WFC    JPM    LUV    XOM
0.489511049 0.001947768 0.337708840 -0.181017319 0.351849663

> op$R
[1] 0.009327074

> op$risk
[1] 0.05073789
```

**Visualization!**

```r
# Add the portfolio possibilities curve:
portPossCurve(port_model, xlim=c(0,0.25), ylim=c(-0.005,0.02))

# Add a cloud of many portfolios:
portCloud(port_model, add=TRUE)

# Add the five stocks plus the point of tangency:
points(op, pch=19, add=TRUE)
points(op$risk, op$R, pch=19, col="green")

# Add the tangent (the following will draw the line only up to \( G \)):
Rf <- 0.001
segments(0, Rf, op$risk, op$R)

# If you want to extend the tangent beyond \( G \):
slope <- (op$R-Rf)/op$risk
segments(0, Rf, 1.4*op$risk, Rf+slope*1.4*op$risk)
```
The graph of the commands above is shown below:

So far we have use the following functions of the stockPortfolio package:

getReturns
stockModel
optimalPort
portPossCurve
PortCloud