

```
# lab9.r                                lab9 calculations
#
# author: Eric Zivot
# created: November 1, 2008
# revised: August 11, 2011
#
# comments:
# Data for the lab are in the Excel file econ424lab7returns.csv, which contains monthly
# continuously
# compounded returns on Boeing, Nordstrom, Starbucks and Microsoft stocks over
# the period March, 1995 through January, 2000.
options(digits=4, width=70)
library("zoo")
# load the data into a zoo object using the zoo function read.csv
source(file="http://spark-public.s3.amazonaws.com/compfinance/R%20code/portfolio.r")

lab8.df = read.csv("http://spark-
public.s3.amazonaws.com/compfinance/R%20code/lab9returns.csv",
                  stringsAsFactors=F)
colnames(lab9.df)

#
# 2. Create zoo object from data and dates in lab7.df
#

lab9.z = zoo(x=lab9.df[, -1],
            order.by=as.yearmon(lab9.df[, 1], format="%b-%y"))
start(lab9.z)
end(lab9.z)
colnames(lab9.z)

#
# 3. Create timePlots of data
#

# create custom panel function to draw horizontal line at zero in each panel
# of plot
my.panel <- function(...) {
  lines(...)
  abline(h=0)
}
plot(lab9.z, lwd=2, panel=my.panel, col="blue")

# all on the same graph
plot(lab9.z, plot.type = "single", main="lab9 returns",
     col=1:4, lwd=2)
abline(h=0)
legend(x="bottomleft", legend=colnames(lab7.z), col=1:4, lwd=2)

#
# 4. Compute pairwise scatterplots
#

pairs(coredata(lab9.z), col="blue", pch=16)

#
# 5. Compute estimates of CER model parameters
#
```

```
muhat.vals = apply(lab9.z, 2, mean)
muhat.vals
sigma2hat.vals = apply(lab9.z, 2, var)
sigma2hat.vals
sigmahat.vals = apply(lab9.z, 2, sd)
sigmahat.vals
cov.mat = var(lab9.z)
cov.mat
cor.mat = cor(lab9.z)
cor.mat

#
# 6. Export means and covariance matrix to .csv file for
#    import to Excel. Be sure to change the directories to the appropriate ones on your
#    computer
#

write.csv(muhat.vals,
file="C:\\Users\\ezivot\\Documents\\classes\\econ424\\fall2010\\muhatVals.csv")
write.csv(cov.mat,
file="C:\\Users\\ezivot\\Documents\\classes\\econ424\\fall2010\\covMat.csv")

#
# portfolio theory calculations using functions in portfolio.r
#

# compute global minimum variance portfolio with short sales
gmin.port = globalMin.portfolio(muhat.vals, cov.mat)
gmin.port
plot(gmin.port, col="blue")

# compute efficient portfolio with target return equal to highest average return
mu.target = max(muhat.vals)
e1.port = efficient.portfolio(muhat.vals, cov.mat, mu.target)
e1.port
plot(e1.port, col="blue")

# compute covariance b/w min var portfolio and efficient port
t(gmin.port$weights)%*%cov.mat%*%e1.port$weights

# compute efficient frontier of risk assets and plot
e.frontier = efficient.frontier(muhat.vals, cov.mat, alpha.min=-1, alpha.max=1)
summary(e.frontier)
plot(e.frontier, plot.assets=T, col="blue", pch=16, cex=2)

# compute tangency portfolio with rf = 0.005
tan.port = tangency.portfolio(muhat.vals, cov.mat, risk.free=0.005)
summary(tan.port)
plot(tan.port, col="blue")

# efficient portfolio of T-bills + tangency that has the same SD as sbux
names(tan.port)
x.tan = sigmahat.vals["Starbucks"]/tan.port$sd
x.tan
mu.pe = 0.005 + x.tan*(tan.port$er - 0.005)
mu.pe
```