

```

# testport.r          illustrate portfolio functions in portfolio.r and
portfolio_noshorts.r
# updates:
# August 15, 2012
#   updated to cover portfolio functions that impose no short sale constraints
# August 11, 2011
#   updated to work with R 2.13.1
options(digits=4, width=70)
source("http://faculty.washington.edu/ezivot/econ424/portfolio_noshorts.r")
# Functions:
#   1. efficient.portfolio          compute minimum variance portfolio
#                                   subject to target
return
#   2. globalMin.portfolio          compute global minimum variance portfolio
#   3. tangency.portfolio           compute tangency portfolio
#   4. efficient.frontier           compute Markowitz bullet
#   5. getPortfolio                create portfolio object
#
#
# Examples from Introduction to Financial Econometrics
#
asset.names = c("MSFT", "NORD", "SBUX")
er = c(0.0427, 0.0015, 0.0285)
names(er) = asset.names
covmat = matrix(c(0.0100, 0.0018, 0.0011,
                  0.0018, 0.0109, 0.0026,
                  0.0011, 0.0026, 0.0199),
                nrow=3, ncol=3)

r.free = 0.005
dimnames(covmat) = list(asset.names, asset.names)
er
covmat
r.free

#
# compute equally weighted portfolio
ew = rep(1,3)/3
equalWeight.portfolio = getPortfolio(er=er,cov.mat=covmat,weights=ew)
class(equalWeight.portfolio)
names(equalWeight.portfolio)
equalWeight.portfolio
summary(equalWeight.portfolio)
plot(equalWeight.portfolio, col="blue")

#
# compute global minimum variance portfolio
gmin.port = globalMin.portfolio(er, covmat)
attributes(gmin.port)
print(gmin.port)
summary(gmin.port, risk.free=r.free)
plot(gmin.port, col="blue")

#
# compute global minimum variance portfolio with no short sales
gmin.port.ns = globalMin.portfolio(er, covmat, shorts=FALSE)
attributes(gmin.port.ns)
print(gmin.port.ns)
summary(gmin.port.ns, risk.free=r.free)
plot(gmin.port.ns, col="blue")

```

```

#
# compute efficient portfolio subject to target return
target.return = er["MSFT"]
e.port.msft = efficient.portfolio(er, covmat, target.return)
e.port.msft
summary(e.port.msft, risk.free=r.free)
plot(e.port.msft, col="blue")

#
# compute efficient portfolio subject to target return with no short sales
target.return = er["MSFT"]
e.port.msft.ns = efficient.portfolio(er, covmat, target.return, shorts=FALSE)
e.port.msft.ns
summary(e.port.msft.ns, risk.free=r.free)
plot(e.port.msft.ns, col="blue")

#
# compute tangency portfolio
tan.port <- tangency.portfolio(er, covmat, r.free)
tan.port
summary(tan.port, risk.free=r.free)
plot(tan.port, col="blue")

#
# compute tangency portfolio with no short sales
tan.port.ns <- tangency.portfolio(er, covmat, r.free, shorts=FALSE)
tan.port.ns
summary(tan.port.ns, risk.free=r.free)
plot(tan.port.ns, col="blue")

#
# compute portfolio frontier
ef <- efficient.frontier(er, covmat, alpha.min=-2,
                        alpha.max=1.5, nport=20)

attributes(ef)
ef

plot(ef)
plot(ef, plot.assets=TRUE, col="blue", pch=16)
points(gmin.port$sd, gmin.port$er, col="green", pch=16, cex=2)
points(tan.port$sd, tan.port$er, col="red", pch=16, cex=2)
text(gmin.port$sd, gmin.port$er, labels="GLOBAL MIN", pos=2)
text(tan.port$sd, tan.port$er, labels="TANGENCY", pos=2)
sr.tan = (tan.port$er - r.free)/tan.port$sd
abline(a=r.free, b=sr.tan, col="green", lwd=2)

# plot portfolio frontier with tangency portfolio
sd.vals = sqrt(diag(covmat))
mu.vals = er
plot(ef$sd, ef$er, ylim=c(0, max(ef$er)), xlim=c(0, max(ef$sd)),
     xlab="portfolio sd", ylab="portfolio er", main="Efficient Portfolios")
text(sd.vals, mu.vals, labels=names(mu.vals))
abline(a=r.free, b=sr.tan)

#
# compute portfolio frontier with no short sales
ef.ns <- efficient.frontier(er, covmat, alpha.min=0,
                           alpha.max=1, nport=20, shorts=FALSE)

```

```
attributes(ef.ns)
ef.ns
summary(ef.ns)

plot(ef.ns)
plot(ef.ns, plot.assets=TRUE, col="blue", pch=16)
points(gmin.port.ns$sd, gmin.port.ns$er, col="green", pch=16, cex=2)
points(tan.port.ns$sd, tan.port.ns$er, col="red", pch=16, cex=2)
text(gmin.port.ns$sd, gmin.port.ns$er, labels="GLOBAL MIN", pos=2)
text(tan.port.ns$sd, tan.port.ns$er, labels="TANGENCY", pos=2)
sr.tan.ns = (tan.port.ns$er - r.free)/tan.port.ns$sd
abline(a=r.free, b=sr.tan.ns, col="green", lwd=2)

# plot portfolio frontier with tangency portfolio
sd.vals = sqrt(diag(covmat))
mu.vals = er
plot(ef.ns$sd, ef.ns$er, ylim=c(0, max(ef.ns$er)), xlim=c(0, max(ef.ns$sd)),
     xlab="portfolio sd", ylab="portfolio er", main="Efficient Portfolios")
text(sd.vals, mu.vals, labels=names(mu.vals))
abline(a=r.free, b=sr.tan.ns)
```