

UW

Portfolio Analysis in R

Econ 424/CFRM 462

Eric Zivot

Summer 2014

Updated: August 12, 2014

R Functions for Portfolio Analysis

- My R functions (on class webpage in `portfolio.r` and `portfolio_noshorts.r`)
- R package **PortfolioAnalytics** (on R-forge)
 - Extensive collection of functions
- Rmetrics package **fPortfolio**
 - Extensive collection of functions
- R package **quadprog**
 - `solve.QP()` for quadratic programming.

R functions in `portfolio.r`

Function	Description
<code>getPortfolio</code>	Create portfolio object
<code>efficient.portfolio</code>	Compute minimum variance portfolio subject to target return
<code>globalMin.portfolio</code>	Compute global minimum variance portfolio
<code>tangency.portfolio</code>	Compute tangency portfolio
<code>efficient.frontier</code>	Compute efficient frontier of risky asset only portfolios

Note: these functions are based on matrix algebra solutions to portfolio calculations that allow short sales

3 Firm Example Data

```
> er
```

```
MSFT      NORD      SBUX  
0.0427 0.0015 0.0285
```

```
> covmat
```

```
          MSFT      NORD      SBUX  
MSFT 0.0100 0.0018 0.0011  
NORD 0.0018 0.0109 0.0026  
SBUX 0.0011 0.0026 0.0199
```

```
> r.free
```

```
[1] 0.005
```

Create Arbitrary Portfolio

```
# compute equally weighted portfolio
> ew = rep(1,3)/3
> equalWeight.portfolio =
+ getPortfolio(er=er,cov.mat=covmat,weights=ew)

> equalWeight.portfolio
```

Call:

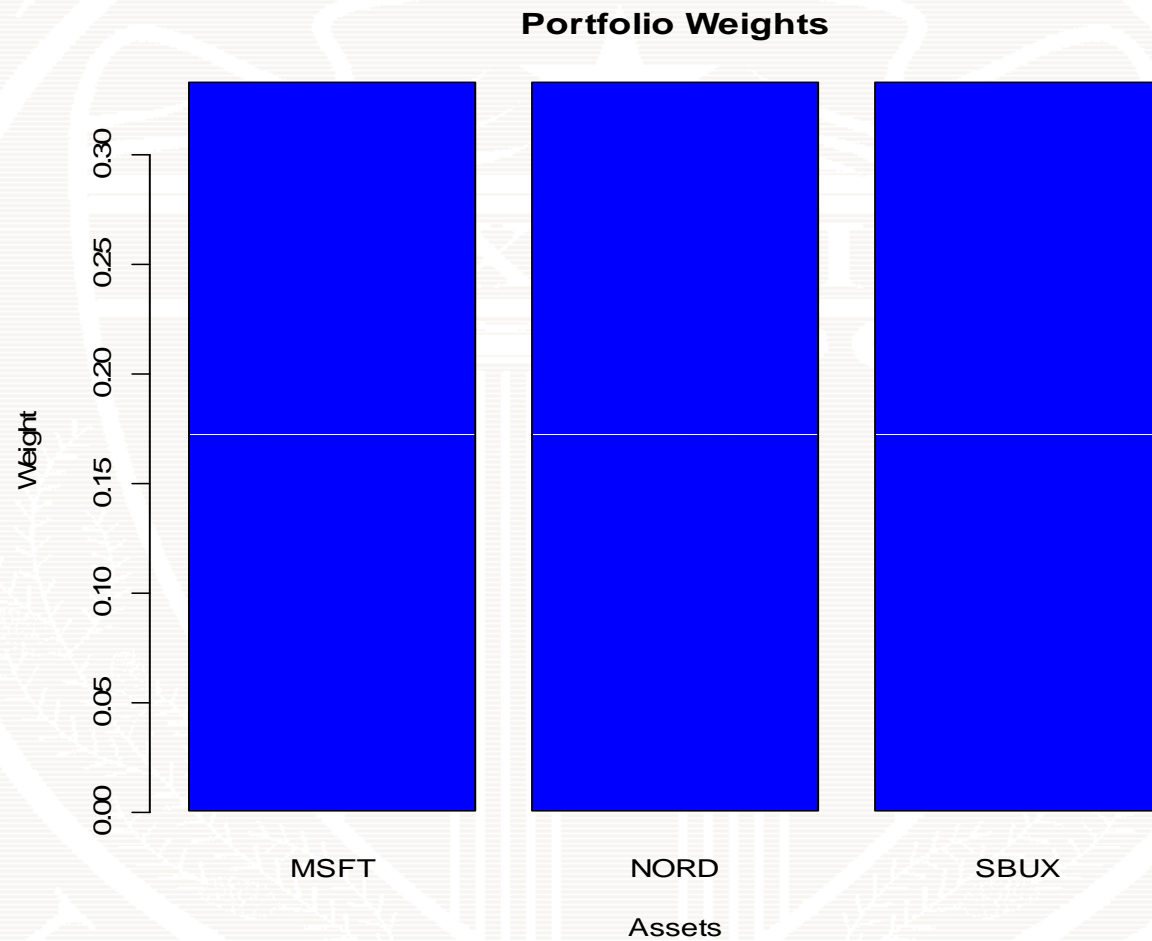
```
getPortfolio(er = er, cov.mat = covmat,
weights = ew)
```

Portfolio expected return: 0.02423

Portfolio standard deviation: 0.07587

Portfolio weights:

MSFT	NORD	SBUX
0.3333	0.3333	0.3333



```
> plot(equalWeight.portfolio, col="blue")
```

Compute Global Minimum Variance Portfolio

```
> gmin.port <- globalMin.portfolio(er, covmat)
> attributes(gmin.port)
```

```
$names
```

```
[1] "call"      "er"        "sd"        "weights"
```

```
$class
```

```
[1] "portfolio"
```

```
> print(gmin.port)
```

```
Call:
```

```
globalMin.portfolio(er = er, cov.mat = covmat)
```

```
Portfolio expected return:      0.02489
```

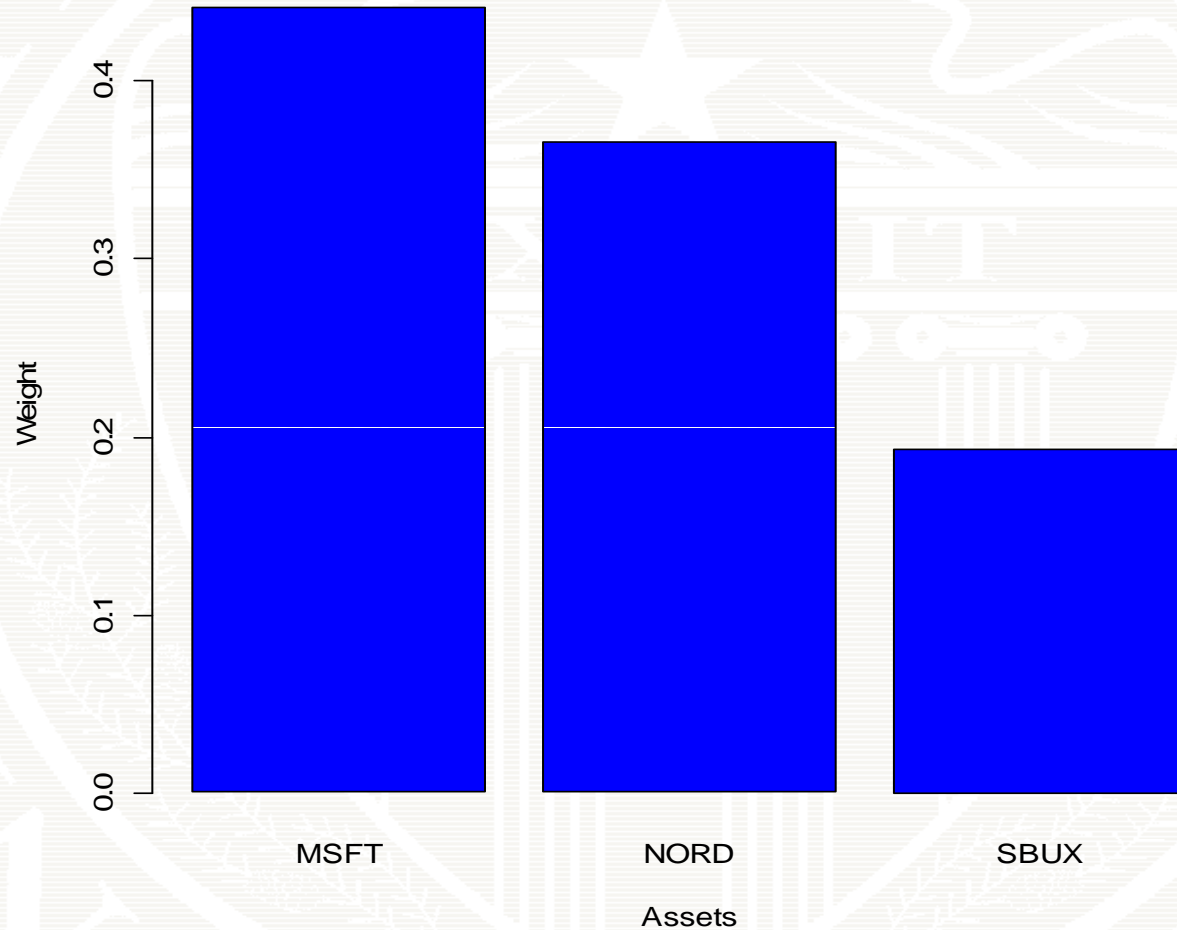
```
Portfolio standard deviation:   0.07268
```

```
Portfolio weights:
```

```
    MSFT    NORD    SBUX
```

```
0.4411 0.3656 0.1933
```

Portfolio Weights



```
> plot(gmin.port, col="blue")
```


Compute Efficient Portfolio

```
# compute efficient portfolio subject to
# target return equal to  $E[R_{msft}] = 0.0427$ 
> target.return = er["MSFT"]
> e.port.msft = efficient.portfolio(er, covmat,
                                   target.return)

> e.port.msft
```

Call:

```
efficient.portfolio(er = er, cov.mat = covmat,
target.return = target.return)
```

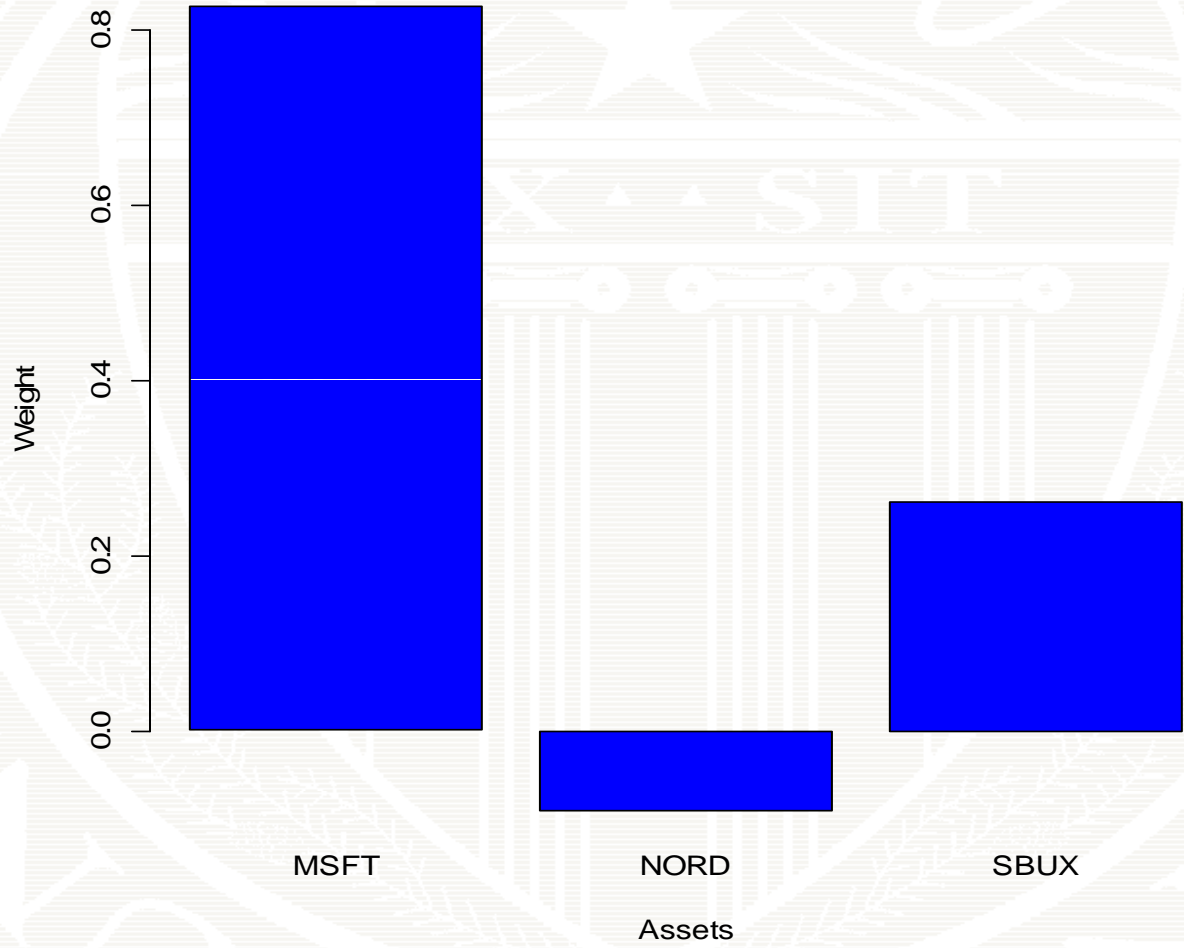
Portfolio expected return: 0.0427

Portfolio standard deviation: 0.09166

Portfolio weights:

MSFT	NORD	SBUX
0.8275	-0.0907	0.2633

Portfolio Weights



Compute Tangency Portfolio

```
> tan.port <- tangency.portfolio(er, covmat,  
+                               r.free)  
> print(tan.port)
```

Call:

```
tangency.portfolio(er = er, cov.mat = covmat,  
risk.free = r.free)
```

Portfolio expected return: 0.05189

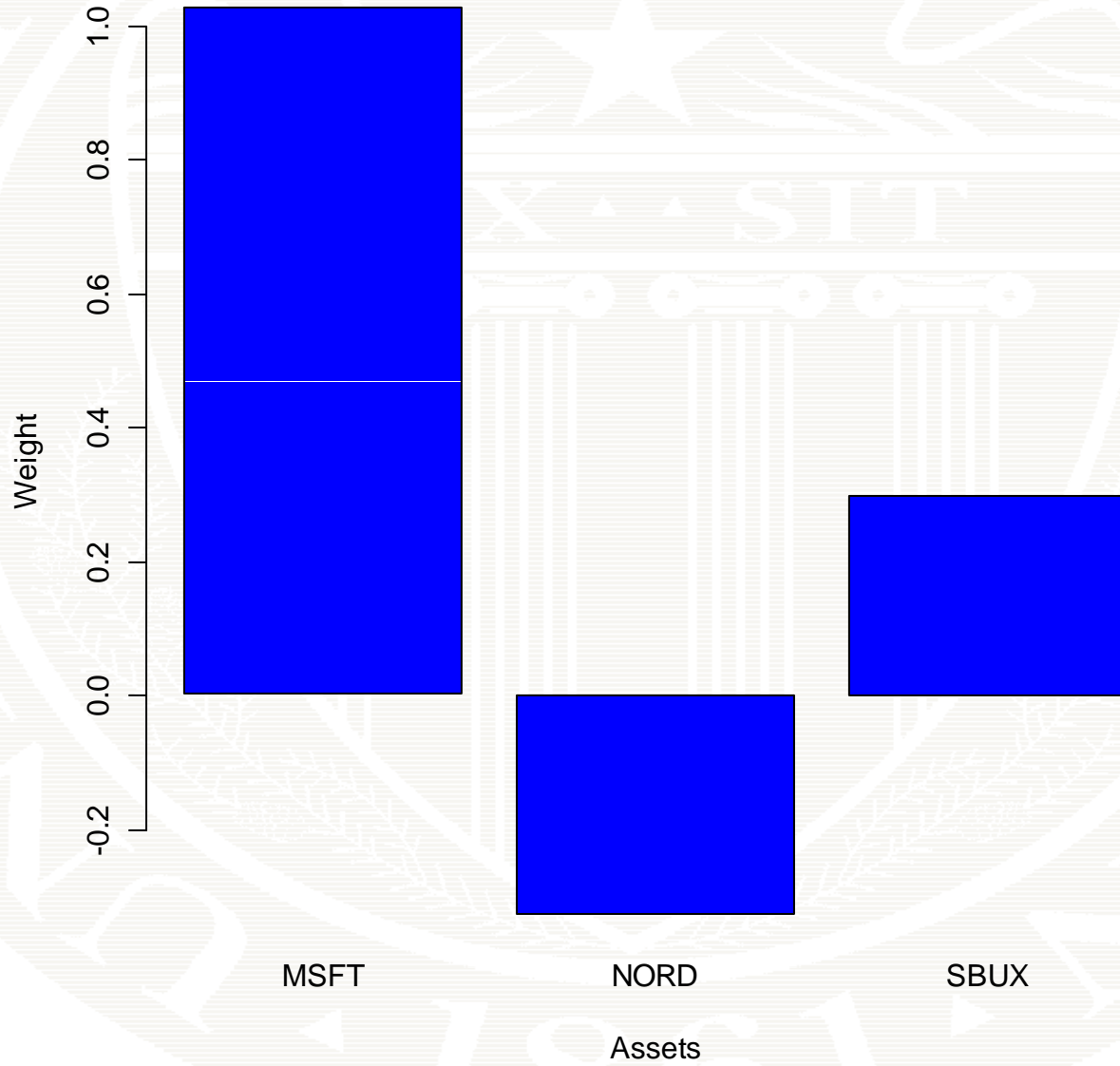
Portfolio standard deviation: 0.1116

Portfolio Sharpe Ratio: 0.4202

Portfolio weights:

MSFT	NORD	SBUX
1.0268	-0.3263	0.2994

Portfolio Weights



Compute Efficient Frontier

```
> ef = efficient.frontier(er, covmat,  
+ alpha.min=-2, alpha.max=1.5,  
+ nport=20)  
> attributes(ef)  
$names  
[1] "call"      "er"        "sd"        "weights"  
  
$class  
[1] "Markowitz"
```

Frontier portfolios

```
> summary(ef)
```

```
Frontier portfolios' expected returns and standard  
deviations
```

```
    port 1 port 2 port 3 port 4 port 5 port 6  
ER 0.0783 0.0750 0.0718 0.0685 0.0652 0.0619  
SD 0.1826 0.1732 0.1640 0.1548 0.1458 0.1370  
  
    port 7 port 8 port 9 port 10 port 11 port 12  
ER 0.0586 0.0554 0.0521  0.0488  0.0455  0.0422  
SD 0.1284 0.1200 0.1120  0.1044  0.0973  0.0908  
  
    port 13 port 14 port 15 port 16 port 17  
ER  0.039  0.0357  0.0324  0.0291  0.0258  
SD  0.085  0.0802  0.0764  0.0739  0.0727  
  
    port 18 port 19 port 20  
ER  0.0225  0.0193  0.0160  
SD  0.0730  0.0748  0.0779
```

Frontier portfolios

Portfolio weights:

	MSFT	NORD	SBUX
port 1	1.6001	-1.0035	0.4033
port 2	1.5290	-0.9194	0.3904
port 3	1.4578	-0.8354	0.3775
port 4	1.3866	-0.7513	0.3646
port 5	1.3155	-0.6672	0.3517
	...		
	...		
port 18	0.3903	0.4257	0.1841
port 19	0.3191	0.5097	0.1712
port 20	0.2479	0.5938	0.1583

Plot efficient frontier

```
> plot(ef, plot.assets=T, 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)
```


Efficient Frontier

