

## Portfolio Returns

### Long Way

```
investment.amounts <- c(AMZN = 50e3, MSFT = 10e3, NFLX = 30e3, IBM = 10e3)
portfolio.names <- c("AMZN", "MSFT", "NFLX", "IBM")
date.from <- '2010-12-31'; date.to <- '2014-1-1'

getSymbols(portfolio.names, from = date.from, to = date.to, auto.assign = T)
```

'getSymbols' currently uses auto.assign=TRUE by default, but will use auto.assign=FALSE in 0.5-0. You will still be able to use 'loadSymbols' to automatically load data. getOption("getSymbols.env") and getOption("getSymbols.auto.assign") will still be 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 details.

```
[1] "AMZN" "MSFT" "NFLX" "IBM"
```

```
n <- nrow(AMZN)

combined.prices <- data.table(AMZN = AMZN[c(1, n), 6],
                             MSFT = MSFT[c(1, n), 6],
                             NFLX = NFLX[c(1, n), 6],
                             IBM = IBM[c(1, n), 6])
colnames(combined.prices) <- portfolio.names

period.return <- data.table(apply(combined.prices, 2, Delt))
period.return <- period.return[2]

portfolio.weights <- investment.amounts / sum(investment.amounts)

portfolio.return <- sum(period.return * portfolio.weights)
```

Portfolio Return over the Period: 101.69%

### Matrix Algebra

```
wgt.mat <- t(as.matrix(portfolio.weights))
ret.mat <- t(as.matrix(period.return))

port.ret <- wgt.mat %*% ret.mat
```

Portfolio Return over the Period: 101.69%

## Benchmark Returns

### Benchmark Portfolios

Benchmark Portfolio with 3 names

```
benchmark.names <- c("AMZN", "MSFT", "IBM")
benchmark.prices <- data.table(Date = index(AMZN),
                                AMZN = AMZN[, 6],
                                MSFT = MSFT[, 6],
                                IBM = IBM[, 6])
colnames(benchmark.prices) <- c("Date", benchmark.names)

ewport <- benchmark.returns <- data.table(Date = index(AMZN), apply(benchmark.prices[, 2:4], 2,
ewport <- ewport[Date >= '2012-12-31' &
                Date <= '2013-12-31',]
rownames(ewport) <- seq(1:nrow(ewport))

ewport
```

	Date	AMZN	MSFT	IBM
1:	2012-12-31	0.023207448	0.006026504	0.009060607
2:	2013-01-02	0.025670679	0.034069827	0.025058809
3:	2013-01-03	0.004547095	-0.013396361	-0.005500527
4:	2013-01-04	0.002592011	-0.018715624	-0.006554782
5:	2013-01-07	0.035925129	-0.001869558	-0.004381866
---				
249:	2013-12-24	-0.009232604	0.012561526	0.005432995
250:	2013-12-26	0.013001009	0.009708327	0.011625435
251:	2013-12-27	-0.015603818	-0.004005964	-0.001456609
252:	2013-12-30	-0.011831773	0.000000000	0.007185980
253:	2013-12-31	0.013778412	0.003217372	0.006222679

```
ewport$AMZN <- 1 + ewport$AMZN
ewport$MSFT <- 1 + ewport$MSFT
ewport$IBM <- 1 + ewport$IBM

# Q1

ew.q1 <- subset(ewport,
                ewport$Date >= as.Date("2012-12-31") &
                ewport$Date <= as.Date("2013-03-31"))

ew.q1[1, 2:4] <- 1
ew.cum.q1 <- ew.q1[, lapply(.SD, cumprod), .SDcols = benchmark.names][, Date := ew.q1$Date]
```

```
num.sec <- length(colnames(ew.q1)) - 1

ew.idx.q1 <- ew.cum.q1[, lapply(.SD, function(x)(1/num.sec)*x), .SDcols = benchmark.names][, Date := ew.q1$Date]
ew.idx.q1[, Value := AMZN + MSFT + IBM]

# Q2

ew.q2 <- subset(ewport,
               ewport$Date >= as.Date("2013-3-31") &
               ewport$Date <= as.Date("2013-6-30"))

q2.value <- tail(ew.idx.q1, 1)$Value

ew.cum.q2 <- ew.q2[, lapply(.SD, cumprod), .SDcols = benchmark.names][, Date := ew.q2$Date]

ew.idx.q2 <- ew.cum.q2[, lapply(.SD, function(x)(q2.value/num.sec)*x), .SDcols = benchmark.names]
ew.idx.q2[, Value := AMZN + MSFT + IBM]

# Q3

ew.q3 <- subset(ewport,
               ewport$Date >= as.Date("2013-6-30") &
               ewport$Date <= as.Date("2013-9-30"))

q3.value <- tail(ew.idx.q2, 1)$Value

ew.cum.q3 <- ew.q3[, lapply(.SD, cumprod), .SDcols = benchmark.names][, Date := ew.q3$Date]

ew.idx.q3 <- ew.cum.q3[, lapply(.SD, function(x)(q3.value/num.sec)*x), .SDcols = benchmark.names]
ew.idx.q3[, Value := AMZN + MSFT + IBM]

# Q4

ew.q4 <- subset(ewport,
               ewport$Date >= as.Date("2013-9-30") &
               ewport$Date <= as.Date("2013-12-31"))

q4.value <- tail(ew.idx.q3, 1)$Value

ew.cum.q4 <- ew.q4[, lapply(.SD, cumprod), .SDcols = benchmark.names][, Date := ew.q4$Date]

ew.idx.q4 <- ew.cum.q4[, lapply(.SD, function(x)(q4.value/num.sec)*x), .SDcols = benchmark.names]
ew.idx.q4[, Value := AMZN + MSFT + IBM]
```

```

benchmark.returns <- data.table(Date = index(AMZN), apply(benchmark.prices[, 2:4], 2, Delt))
benchmark.returns[, ':='](Year = year(Date), Quarter = quarter(Date), Month = month(Date))
benchmark.returns[1, c(2:4)] <- 0

benchmark.cumret <- benchmark.returns
benchmark.cumret[, ':='](AMZN = AMZN + 1, MSFT = MSFT + 1, IBM = IBM + 1)]
quarterly.returns <- benchmark.cumret[, lapply(.SD, cumprod),
  by = list(Year, Quarter),
  .SDcols = benchmark.names[, Date := benchmark.returns$Date]
rownames(quarterly.returns) <- seq(1:nrow(quarterly.returns))

n.sec <- 3

quarterly.returns[, lapply(.SD, function(x) c(head(x, 1), tail(x, 1))),
  by = list(Year, Quarter),
  .SDcols = "Date"]

```

	Year	Quarter	Date
1:	2010	4	2010-12-31
2:	2010	4	2010-12-31
3:	2011	1	2011-01-03
4:	2011	1	2011-03-31
5:	2011	2	2011-04-01
6:	2011	2	2011-06-30
7:	2011	3	2011-07-01
8:	2011	3	2011-09-30
9:	2011	4	2011-10-03
10:	2011	4	2011-12-30
11:	2012	1	2012-01-03
12:	2012	1	2012-03-30
13:	2012	2	2012-04-02
14:	2012	2	2012-06-29
15:	2012	3	2012-07-02
16:	2012	3	2012-09-28
17:	2012	4	2012-10-01
18:	2012	4	2012-12-31
19:	2013	1	2013-01-02
20:	2013	1	2013-03-28
21:	2013	2	2013-04-01
22:	2013	2	2013-06-28
23:	2013	3	2013-07-01
24:	2013	3	2013-09-30
25:	2013	4	2013-10-01
26:	2013	4	2013-12-31
	Year	Quarter	Date

```
ew.port.value <- rbind(ew.idx.q1[, .(Date, Value)],  
  ew.idx.q2[, .(Date, Value)],  
  ew.idx.q3[, .(Date, Value)],  
  ew.idx.q4[, .(Date, Value)])
```

```
ggplot(ew.port.value[, Quarter := quarter(Date)], aes(Date, Value)) +  
  geom_line(aes(col = Quarter))
```



```
ewport$Quarter <- quarter(ewport$Date)  
  
ewport <- ewport[2:nrow(ewport)]
```