Carlos Arias

```r
library(data.table,quietly=T)
library(bizdays,quietly=T)
```

```
##
## Attaching package: 'bizdays'
##
## The following object is masked from 'package:stats':
##
##     offset
```

```r
library(tables,quietly=T)
```

```
##
## Attaching package: 'Hmisc'
##
## The following objects are masked from 'package:base':
##
##     format.pval, round.POSIXt, trunc.POSIXt, units
```

```r
library(knitr,quietly=T)


setwd('~/spring2016/viz/vizrepo/assignments/0214/data/')

# getting the data
apple <- fread('APPLE.csv')
boa <- fread('BOA.csv')
walmart <- fread('WALMART.csv')
ge <- fread('GE.csv')
exxon <- fread('EXXON.csv')
market <- fread('NYSE(composite).csv')
```

```
## Warning in fread("NYSE(composite).csv"): Some columns have been read as
## type 'integer64' but package bit64 isn't loaded. Those columns will display
## as strange looking floating point data. There is no need to reload the
## data. Just require(bit64) to obtain the integer64 print method and print
## the data again.
```

```r
# market Volume is huge- coerced to int64 and makes binding problematic
#  am removing that column before binding
apple[, Volume := NULL]
boa[, Volume := NULL]
walmart[, Volume := NULL]
ge[, Volume := NULL]
exxon[, Volume := NULL]
market[, Volume := NULL]

apple[, company := 'apple']
boa[, company := 'boa']
```

```r
walmart[, company := 'walmart']
ge[, company := 'ge']
exxon[, company := 'exxon']
market[, company := 'market']

all.data <- rbind(apple, boa, walmart, ge, exxon, market)
all.data <- all.data[, c('Open', 'High', 'Low', 'Close') := NULL]
names(all.data)[2] <- 'adj_close'
all.data <- all.data[, Date := as.Date(Date) ]
all.data <- all.data[ Date >= as.Date('2010-01-29'), ]


# getting important dates
#  since we are talking about years since 1/29/10
#  each "adjusted year" begins and ends on the next business day

years <- 2010:2016
dates <- ISOdate(years, 1, 29)
cal <- Calendar(weekdays=c('sunday', 'saturday'))
dates <- adjust.next(dates, cal)

# setting year values (is from 1-29 instead of 1-1)
map <- as.data.table(dates)
map[, year := year(dates)]
all.data$year <- map$year[findInterval(x = all.data$Date, vec = map$dates)]

# add current day to important dates so we get 2016 so far
dates <- c(dates, all.data$Date[1])

# getting changes by adj_year
year.data <- all.data[ Date %in% dates, ]
year.data[, change := 100 * adj_close / shift(adj_close, type = 'lead') - 100 ]

# adjusting year value since date marks when change happens
year.data[Date != last(dates), year := as.integer(year) - 1]
```

```
## Warning in `[.data.table`(year.data, Date != last(dates), `:=`(year,
## as.integer(year) - : Coerced 'double' RHS to 'integer' to match the
## column's type; may have truncated precision. Either change the target
## column to 'double' first (by creating a new 'double' vector length 48
## (nrows of entire table) and assign that; i.e. 'replace' column), or coerce
## RHS to 'integer' (e.g. 1L, NA_[real|integer]_, as.*, etc) to make your
## intent clear and for speed. Or, set the column type correctly up front when
## you create the table and stick to it, please.
```

```r
# removing oldest year as it has meaningless change
year.data <- year.data[Date != dates[1] , ]

# getting relative to market changes
market.year.data <- year.data[company == 'market',]
year.data[, rel_change := ((change / 100 + 1) / (market.year.data$change / 100 + 1)) * 100 - 100, by = c

# historical volatility: sd of daily prices
```

```
all.data[, volatility := sd(adj_close), by = list(company, year)]
total_volatilities <- all.data[, sd(adj_close), by = list(company)]
volat <- all.data
volat[, c('Date', 'adj_close') := NULL ]
volat <- unique(volat)

# merging tables
setkey(year.data, company, year)
setkey(volat, company, year)
final.data <- year.data[volat, ]
final.data[, c('Date', 'adj_close') := NULL ]
```

```
setwd("~/spring2016/viz/vizrepo/assignments/0214/data/")
final_table <- read.csv("final_table.csv")

kable(final_table,
      caption="Annual Return is computed by the change in value from the beginning of year to end. Rela
      col.names = c("Company", " ", "2010","2011","2012",   "2013", "2014","2015","2016","Total"),
      padding=2,
      escape = TRUE,
      format.args = list(na.encode=FALSE,
                         n.rgroup=c(3,3,3,3,3,3,1),
                         n.cgroup = c(2,7))
)
```

Table 1: Annual Return is computed by the change in value from the beginning of year to end. Relative Return is computed by dividing the Annual Return by the market return. Historical Volatility is computed by taking the standard deviation for the given period of time.

Company		2010	2011	2012	2013	2014	2015	2016	Total
Apple	Annual Return	76.67	33.51	2.06	12.01	69.71	-16.74	-2.92	281.01
	Relative Return	4.20	-8.95	0.15	1.05	9.53	1.69	0.70	7.03
	Historical Volatility	5.28	3.89	7.54	6.83	13.54	8.66	1.12	28.35
B of A	Annual Return	-9.28	-48.26	63.31	45.60	-6.82	-7.24	-15.49	-3.53
	Relative Return	-0.51	12.89	4.50	3.97	-0.93	0.73	3.70	-0.09
	Historical Volatility	1.98	3.11	1.31	1.44	0.88	0.99	0.91	3.52
Exxon	Annual Return	28.61	8.47	10.06	6.48	-5.31	-8.06	5.03	42.33
	Relative Return	1.57	-2.26	0.72	0.56	-0.73	0.81	-1.20	1.06
	Historical Volatility	4.74	3.95	3.21	3.86	3.85	4.61	2.18	11.87
GE	Annual Return	28.78	-2.88	23.51	16.04	-1.42	25.02	-2.89	120.92
	Relative Return	1.58	0.77	1.67	1.39	-0.19	-2.53	0.69	3.03
	Historical Volatility	1.17	1.55	1.20	1.66	0.71	2.14	0.48	4.95
Walmart	Absolute Return	7.36	12.32	16.82	8.63	21.35	-22.25	-0.27	44.38
	Relative Return	0.40	-3.29	1.20	0.75	2.92	2.25	0.06	1.11
	Historical Volatility	1.69	2.72	5.57	2.81	4.51	8.35	0.63	12.07
Total	Annual Return	132.14	3.16	115.76	88.76	77.51	-29.27	-16.54	1124.53
	Relative Return	7.24	-0.84	8.24	7.72	10.60	2.95	3.95	12.14
	Historical Volatility	7.65	7.09	10.07	8.62	14.82	13.10	2.73	33.57
Market	Annual Return	18.24	-3.74	14.06	11.50	7.31	-9.90	-4.18	39.95

```
latex(final_table,center='centering',numeric.dollar=FALSE,colheads=c("Company", " ", "2010","2011","2012
```