Portfolio managment

Jovanni Rivera

2/5/2021

```r
library(quantmod)
```

```
## Warning: package 'quantmod' was built under R version 4.0.3

## Loading required package: xts

## Warning: package 'xts' was built under R version 4.0.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.0.3

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: TTR

## Warning: package 'TTR' was built under R version 4.0.3

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

Getting Apple stock data

```r
dt='2020-2-1'
aapl=getSymbols.yahoo('AAPL',from=dt,auto.assign=F)
head(aapl)
```

```
##            AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
## 2020-02-03   76.0750   78.3725  75.5550    77.1650   173985600      76.39016
## 2020-02-04   78.8275   79.9100  78.4075    79.7125   136616400      78.91208
## 2020-02-05   80.8800   81.1900  79.7375    80.3625   118826800      79.55555
## 2020-02-06   80.6425   81.3050  80.0650    81.3025   105425600      80.48611
## 2020-02-07   80.5925   80.8500  79.5000    80.0075   117684000      79.39209
## 2020-02-10   78.5450   80.3875  78.4625    80.3875   109348800      79.76917
```

Filtering only Close Column

```
aaplclose=getSymbols.yahoo('AAPL',from=dt,auto.assign=F)[,6]
head(aaplclose)
```

```
##            AAPL.Adjusted
## 2020-02-03      76.39016
## 2020-02-04      78.91208
## 2020-02-05      79.55555
## 2020-02-06      80.48611
## 2020-02-07      79.39209
## 2020-02-10      79.76917
```

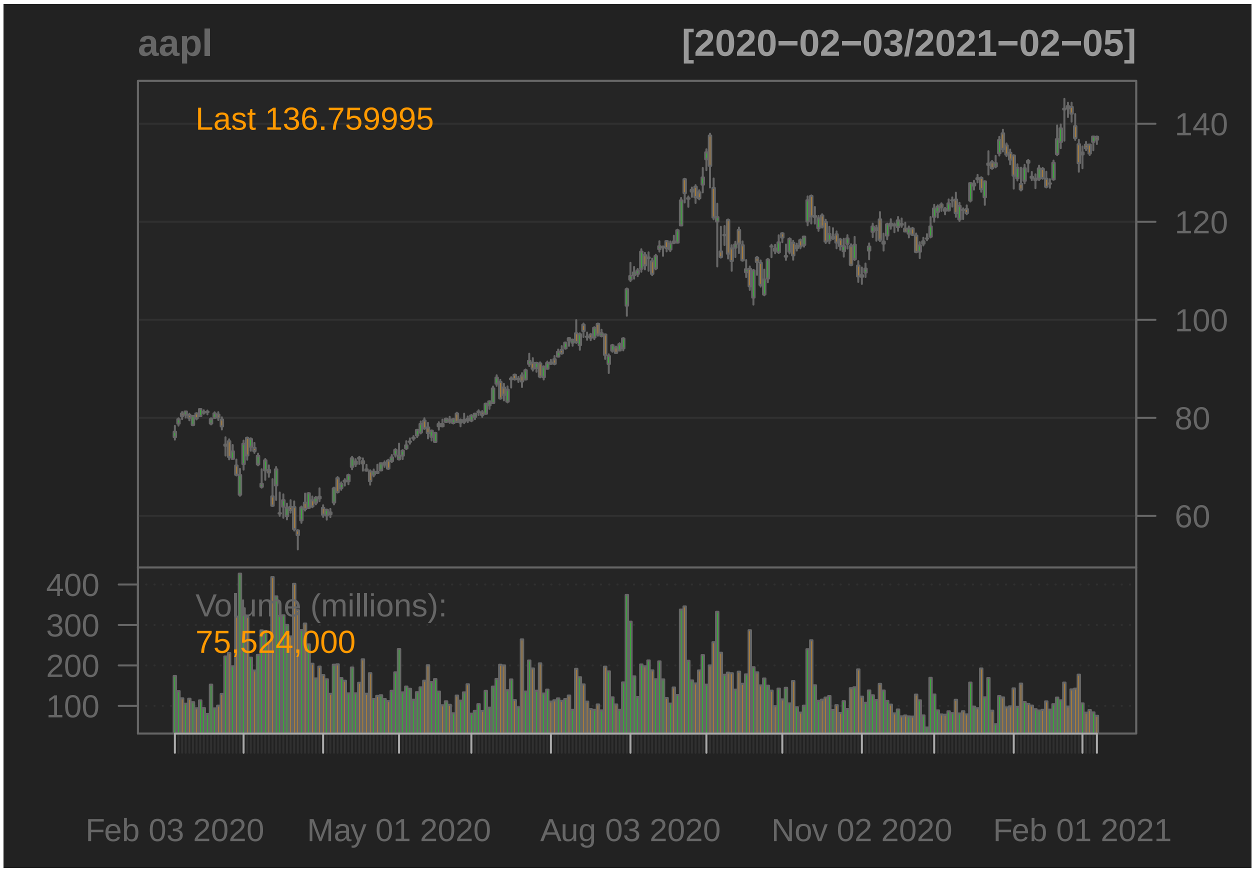
Creating Daily returns Column

```
aaplrets=dailyReturn(aaplclose,type='log')
head(aaplrets)
```

```
##            daily.returns
## 2020-02-03   0.000000000
## 2020-02-04   0.032480417
## 2020-02-05   0.008121211
## 2020-02-06   0.011629066
## 2020-02-07  -0.013685845
## 2020-02-10   0.004738285
```

Visualizing

```
chartSeries(aapl)
```




```
library(PerformanceAnalytics)
```

```
## Warning: package 'PerformanceAnalytics' was built under R version 4.0.3
```

```
##
## Attaching package: 'PerformanceAnalytics'
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

Creating Portfolio with each stock weight

```
tickers=c('FB','AAPL','AMZN','NFLX')
weights=c(0.25,0.25,0.25,0.25)
```

Getting Portfolio Prices

```
PortfolioPrices=NULL
for(ticker in tickers){
  PortfolioPrices=cbind(PortfolioPrices,getSymbols.yahoo(ticker,from='2020-2-1',periodicity='daily',auto
}
head(PortfolioPrices)
```

```
##            FB.Close AAPL.Close AMZN.Close NFLX.Close
## 2020-02-03   204.19    77.1650    2004.20     358.00
## 2020-02-04   209.83    79.7125    2049.67     369.01
## 2020-02-05   210.11    80.3625    2039.87     369.67
## 2020-02-06   210.85    81.3025    2050.23     366.95
## 2020-02-07   212.33    80.0075    2079.28     366.77
## 2020-02-10   213.06    80.3875    2133.91     371.07
```

Calculating Market Return Vs Portfolio Return

```
benchmarkPrices=getSymbols.yahoo('^GSPC',from='2020-2-1',periodicity='daily',auto.assign=FALSE)
head(benchmarkPrices)
```

```
##            GSPC.Open GSPC.High GSPC.Low GSPC.Close GSPC.Volume GSPC.Adjusted
## 2020-02-03   3235.66   3268.44  3235.66    3248.92  3757910000       3248.92
## 2020-02-04   3280.61   3306.92  3280.61    3297.59  3995320000       3297.59
## 2020-02-05   3324.91   3337.58  3313.75    3334.69  4117730000       3334.69
## 2020-02-06   3344.92   3347.96  3334.39    3345.78  3868370000       3345.78
## 2020-02-07   3335.54   3341.42  3322.12    3327.71  3730650000       3327.71
## 2020-02-10   3318.28   3352.26  3317.77    3352.09  3450350000       3352.09
```

```
benchmarkReturns=na.omit(ROC(benchmarkPrices))
PortfolioReturns=na.omit(ROC(PortfolioPrices))
```

```
PortfolioReturn=Return.portfolio(PortfolioReturns)
head(PortfolioReturn)
```

```
##            portfolio.returns
## 2020-02-04       0.028112945
## 2020-02-05       0.001628174
## 2020-02-06       0.003217412
## 2020-02-07       0.001016229
## 2020-02-10       0.011436480
## 2020-02-11      -0.004723314
```

```
CAPM.beta(PortfolioReturn,benchmarkReturns,0.35/252)
```

```
##                      portfolio.returns
## Beta: GSPC.Open             0.22921100
## Beta: GSPC.High             0.72867658
## Beta: GSPC.Low              0.79317511
## Beta: GSPC.Close            0.85906724
## Beta: GSPC.Volume          -0.03366936
## Beta: GSPC.Adjusted         0.85906724
```

```
CAPM.jensenAlpha(PortfolioReturn,benchmarkReturns,0.35/252)
```

```
##                      portfolio.returns
## Beta: GSPC.Open              0.5232845
## Beta: GSPC.High              0.7638832
## Beta: GSPC.Low               0.7949530
## Beta: GSPC.Close             0.8266940
## Beta: GSPC.Volume            0.3966517
## Beta: GSPC.Adjusted          0.8266940
```

```r
SharpeRatio(PortfolioReturn,0.35/252)
```

```
##                                 portfolio.returns
## StdDev Sharpe (Rf=0.1%, p=95%):       0.011857034
## VaR Sharpe (Rf=0.1%, p=95%):          0.007070709
## ES Sharpe (Rf=0.1%, p=95%):           0.004237291
```

```r
table.AnnualizedReturns(PortfolioReturn)
```

```
##                           portfolio.returns
## Annualized Return                    0.4143
## Annualized Std Dev                   0.3904
## Annualized Sharpe (Rf=0%)            1.0610
```

```r
table.CalendarReturns(PortfolioReturn)
```

```
##      Jan Feb  Mar  Apr May Jun Jul Aug Sep  Oct Nov Dec portfolio.returns
## 2020  NA 0.1 -3.7 -3.2 1.7 3.8 5.3 2.7 2.5 -5.9 2.7 0.4               6.1
## 2021 2.2 0.1   NA   NA  NA  NA  NA  NA  NA   NA  NA  NA               2.3
```

From all of the calculations, it shows that this portfolio consisting of Amazon, Apple, Netflix, and facebook outperformed the market. This particular portfolio had 41% annualized return, which is very good. Had a beta less than 1, so the volatility was less than the general market and had a good sharpe ratio. Overall this particular Portfolio performed quite well in the past year