hw3.R

Dipro

2021-03-29

```
#install.packages("quantmod")

library(quantmod)
```

```
## Warning: package 'quantmod' was built under R version 4.0.4
## Loading required package: xts
## Warning: package 'xts' was built under R version 4.0.4
## Loading required package: zoo
## Warning: package 'zoo' was built under R version 4.0.4
##
## 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 4.0.4
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
#Question-1

#1.1

VIX <- getOptionChain("^VIX" , NULL)


#1.2
LastQuotePrice <- getQuote("^VIX")$Last
LastQuotePrice
```

```
## [1] 20.66
```

```
#1.3

for(i in 1:length(VIX)){
  VIX[[i]]$calls$Price <- 0.5*(VIX[[i]]$calls$Bid + VIX[[i]]$calls$Ask)
  VIX[[i]]$puts$Price <- 0.5 * (VIX[[i]]$puts$Bid + VIX[[i]]$puts$Bid)
}
```

```r
# 1.4

for(i in 1:length(VIX)){
  VIX[[i]]$calls$"In-The-Money" <- (VIX[[i]]$calls$Strike < (rep(c(LastQuotePrice), length(VIX[[i]]$cal
}

for(i in 1:length(VIX)){
  VIX[[i]]$puts$"In-The-Money" <-  VIX[[i]]$puts$Strike > (rep(c(LastQuotePrice), length(VIX[[i]]$puts$
}



#1.5

for(i in 1:length(VIX)){
  VIX[[i]]$calls <- VIX[[i]]$calls[c("Strike" , "Bid" , "Ask" , "Price" , "In-The-Money")]
  VIX[[i]]$puts <- VIX[[i]]$puts[c("Strike" , "Bid" , "Ask" , "Price" , "In-The-Money")]
}



today <- format(Sys.Date(), "%Y-%m-%d")
Exp <- names(VIX)
Exp <- as.Date(Exp, format = "%b.%d.%Y") # convert to date object
Exp <- format(Exp, "%Y_%m_%d") # convert to chars with certain format


for (i in 1:length(Exp)){
  write.csv(VIX[[i]]$calls, file = paste("VIXdata" , today, "Exp" , Exp[i] , "calls.csv" , sep = ""))
  write.csv(VIX[[i]]$puts, file = paste("VIXdata" , today, "Exp" , Exp[i] , "puts.csv" , sep = ""))
}



# Question 2

#2.1

sample.skewness <- function(x , adjusted){

  x.m3.origin <- mean(x^3)# 3rd sample moment about the origin
  x.m2.origin <- mean(x^2)
  x.sm3 <- x.m3.origin /  (x.m2.origin)^(3/2)

  n <- length(x)

  if(adjusted == TRUE){
    coff <- sqrt(n * (n-1)) / (n-2)
    x.sm3.adj <- coff * x.sm3
    return(x.sm3.adj)

  }else{
```

```
    return(x.sm3)
  }

}

#2.2

sample.kurtosis <- function(x , adjusted){

  x.m4.origin <- mean(x^4)# 4th sample moment about the origin
  x.m2.origin <- mean(x^2)
  x.sm4 <- x.m4.origin /  (x.m2.origin)^(4/2)
  n <- length(x)

  if(adjusted == TRUE){
    coff <- (n-1) / ((n-2) * (n-3))
    x.sm4.adj <- coff * ((n+1)*x.sm4 - 3*(n-1)) + 3
    return(x.sm4.adj)
  }else{
    return(x.sm4)
  }
}


#2.3

getSymbols(Symbols = "SPY" , from = "2012-01-01" , to = "2013-12-31")
```

```
## '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] "SPY"
```

```
SPY <- data.frame(SPY)
SPY <- SPY[nrow(SPY):1,]


head(SPY)
```

```
##            SPY.Open SPY.High SPY.Low SPY.Close SPY.Volume SPY.Adjusted
## 2013-12-30   183.87   184.02  183.58    183.82   56857000     159.8652
## 2013-12-27   184.10   184.18  183.66    183.85   61814000     159.8912
## 2013-12-26   183.34   183.96  183.32    183.86   63365000     159.8999
## 2013-12-24   182.54   183.01  182.53    182.93   45368800     159.0911
## 2013-12-23   182.45   182.64  182.07    182.53   85598000     158.7432
## 2013-12-20   180.69   181.99  180.57    181.56  197087000     157.8997
```

```
spy.price <- SPY$SPY.Adjusted
spy.log.price <- log(spy.price)
```

```r
spy.log.return <- diff(spy.log.price)

#2.4

spy.skewness <- sample.skewness(spy.log.return , FALSE)
spy.skewness
```

```
## [1] -0.1682616
```

```r
spy.skewness.adj <- sample.skewness(spy.log.return , TRUE)
spy.skewness.adj
```

```
## [1] -0.1687683
```

```r
#2.5

spy.kurtosis <- sample.kurtosis(spy.log.return , FALSE)
spy.kurtosis
```

```
## [1] 4.014448
```

```r
spy.kurtosis.adj <- sample.kurtosis(spy.log.return , TRUE)
spy.kurtosis.adj
```

```
## [1] 4.036763
```