HW 4

#Question 3.11

install.packages("tseries",repos = "http://cran.us.r-project.org")

## Installing package into 'C:/Users/ahmad/OneDrive/Documents/R/win-library/3.6'  
## (as 'lib' is unspecified)

## package 'tseries' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'tseries'

## Warning in file.copy(savedcopy, lib, recursive = TRUE):  
## problem copying C:\Users\ahmad\OneDrive\Documents\R\win-  
## library\3.6\00LOCK\tseries\libs\x64\tseries.dll to C:  
## \Users\ahmad\OneDrive\Documents\R\win-library\3.6\tseries\libs\x64\tseries.dll:  
## Permission denied

## Warning: restored 'tseries'

##   
## The downloaded binary packages are in  
## C:\Users\ahmad\AppData\Local\Temp\RtmpUdusLq\downloaded\_packages

library(tseries)

## Warning: package 'tseries' was built under R version 3.6.3

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

#a  
x = get.hist.quote(instrument="BBY", start="2009-12-31",end="2015-12-31", quote="AdjClose", compression="m")

## '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.

## time series starts 2010-01-01  
## time series ends 2015-12-01

x

## Adjusted  
## 2010-01-01 26.837444  
## 2010-02-01 26.727596  
## 2010-03-01 31.150465  
## 2010-04-01 33.332611  
## 2010-05-01 31.034174  
## 2010-06-01 24.871407  
## 2010-07-01 25.459040  
## 2010-08-01 23.152834  
## 2010-09-01 30.115646  
## 2010-10-01 31.701454  
## 2010-11-01 31.625866  
## 2010-12-01 25.385088  
## 2011-01-01 25.280964  
## 2011-02-01 23.972300  
## 2011-03-01 21.354977  
## 2011-04-01 23.213869  
## 2011-05-01 23.732065  
## 2011-06-01 23.470537  
## 2011-07-01 20.721733  
## 2011-08-01 19.212646  
## 2011-09-01 17.493347  
## 2011-10-01 19.823248  
## 2011-11-01 20.473192  
## 2011-12-01 17.661816  
## 2012-01-01 18.226936  
## 2012-02-01 18.797726  
## 2012-03-01 18.021460  
## 2012-04-01 16.796183  
## 2012-05-01 14.351789  
## 2012-06-01 16.069096  
## 2012-07-01 13.982364  
## 2012-08-01 13.711832  
## 2012-09-01 13.294450  
## 2012-10-01 11.866911  
## 2012-11-01 10.228480  
## 2012-12-01 9.245425  
## 2013-01-01 12.863050  
## 2013-02-01 12.981712  
## 2013-03-01 17.522543  
## 2013-04-01 20.719477  
## 2013-05-01 21.963120  
## 2013-06-01 21.787737  
## 2013-07-01 24.136269  
## 2013-08-01 28.876886  
## 2013-09-01 30.080105  
## 2013-10-01 34.487804  
## 2013-11-01 32.674778  
## 2013-12-01 32.134888  
## 2014-01-01 19.046120  
## 2014-02-01 21.546228  
## 2014-03-01 21.368229  
## 2014-04-01 21.117889  
## 2014-05-01 22.526829  
## 2014-06-01 25.255135  
## 2014-07-01 24.355598  
## 2014-08-01 26.125132  
## 2014-09-01 27.517818  
## 2014-10-01 28.134720  
## 2014-11-01 32.477707  
## 2014-12-01 32.123348  
## 2015-01-01 29.159676  
## 2015-02-01 31.562029  
## 2015-03-01 31.305229  
## 2015-04-01 29.223904  
## 2015-05-01 29.266071  
## 2015-06-01 27.503359  
## 2015-07-01 27.417376  
## 2015-08-01 31.195869  
## 2015-09-01 31.518518  
## 2015-10-01 29.928350  
## 2015-11-01 27.151667  
## 2015-12-01 26.015369

bby\_m\_logprices = log(x)  
bby\_m\_logprices

## Adjusted  
## 2010-01-01 3.289798  
## 2010-02-01 3.285697  
## 2010-03-01 3.438829  
## 2010-04-01 3.506536  
## 2010-05-01 3.435089  
## 2010-06-01 3.213719  
## 2010-07-01 3.237071  
## 2010-08-01 3.142117  
## 2010-09-01 3.405045  
## 2010-10-01 3.456363  
## 2010-11-01 3.453975  
## 2010-12-01 3.234162  
## 2011-01-01 3.230052  
## 2011-02-01 3.176899  
## 2011-03-01 3.061285  
## 2011-04-01 3.144750  
## 2011-05-01 3.166827  
## 2011-06-01 3.155746  
## 2011-07-01 3.031183  
## 2011-08-01 2.955569  
## 2011-09-01 2.861821  
## 2011-10-01 2.986855  
## 2011-11-01 3.019116  
## 2011-12-01 2.871405  
## 2012-01-01 2.902900  
## 2012-02-01 2.933736  
## 2012-03-01 2.891563  
## 2012-04-01 2.821152  
## 2012-05-01 2.663875  
## 2012-06-01 2.776898  
## 2012-07-01 2.637797  
## 2012-08-01 2.618259  
## 2012-09-01 2.587347  
## 2012-10-01 2.473754  
## 2012-11-01 2.325176  
## 2012-12-01 2.224129  
## 2013-01-01 2.554359  
## 2013-02-01 2.563542  
## 2013-03-01 2.863488  
## 2013-04-01 3.031074  
## 2013-05-01 3.089365  
## 2013-06-01 3.081347  
## 2013-07-01 3.183716  
## 2013-08-01 3.363041  
## 2013-09-01 3.403864  
## 2013-10-01 3.540606  
## 2013-11-01 3.486603  
## 2013-12-01 3.469942  
## 2014-01-01 2.946863  
## 2014-02-01 3.070201  
## 2014-03-01 3.061905  
## 2014-04-01 3.050121  
## 2014-05-01 3.114707  
## 2014-06-01 3.229030  
## 2014-07-01 3.192762  
## 2014-08-01 3.262898  
## 2014-09-01 3.314834  
## 2014-10-01 3.337004  
## 2014-11-01 3.480554  
## 2014-12-01 3.469583  
## 2015-01-01 3.372787  
## 2015-02-01 3.451955  
## 2015-03-01 3.443785  
## 2015-04-01 3.374987  
## 2015-05-01 3.376429  
## 2015-06-01 3.314308  
## 2015-07-01 3.311177  
## 2015-08-01 3.440286  
## 2015-09-01 3.450575  
## 2015-10-01 3.398806  
## 2015-11-01 3.301438  
## 2015-12-01 3.258687

bby\_m\_log = as.vector(bby\_m\_logprices)  
bby\_m\_log

## [1] 3.289798 3.285697 3.438829 3.506536 3.435089 3.213719 3.237071 3.142117  
## [9] 3.405045 3.456363 3.453975 3.234162 3.230052 3.176899 3.061285 3.144750  
## [17] 3.166827 3.155746 3.031183 2.955569 2.861821 2.986855 3.019116 2.871405  
## [25] 2.902900 2.933736 2.891563 2.821152 2.663875 2.776898 2.637797 2.618259  
## [33] 2.587347 2.473754 2.325176 2.224129 2.554359 2.563542 2.863488 3.031074  
## [41] 3.089365 3.081347 3.183716 3.363041 3.403864 3.540606 3.486603 3.469942  
## [49] 2.946863 3.070201 3.061905 3.050121 3.114707 3.229030 3.192762 3.262898  
## [57] 3.314834 3.337004 3.480554 3.469583 3.372787 3.451955 3.443785 3.374987  
## [65] 3.376429 3.314308 3.311177 3.440286 3.450575 3.398806 3.301438 3.258687

head(bby\_m\_log)

## [1] 3.289798 3.285697 3.438829 3.506536 3.435089 3.213719

length(bby\_m\_log)

## [1] 72

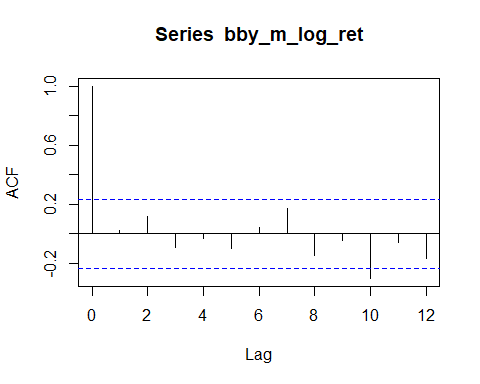
#b  
bby\_m\_log\_ret = log(bby\_m\_log[-1]) - log(bby\_m\_log[-72])  
bby\_m\_log\_ret

## [1] -0.0012475069 0.0455523756 0.0194976636 -0.0205858898 -0.0666140498  
## [6] 0.0072400927 -0.0297720358 0.0803612699 0.0149586416 -0.0006909119  
## [11] -0.0657560109 -0.0012716826 -0.0165925809 -0.0370708568 0.0268996585  
## [16] 0.0069958033 -0.0035052846 -0.0402718934 -0.0252619013 -0.0322330798  
## [21] 0.0427631163 0.0107430552 -0.0501627159 0.0109089452 0.0105662542  
## [26] -0.0144793840 -0.0246520887 -0.0573635102 0.0415527693 -0.0513904188  
## [31] -0.0074343963 -0.0118767420 -0.0448960828 -0.0619410810 -0.0444304253  
## [36] 0.1384359529 0.0035884822 0.1106508011 0.0568765300 0.0190483954  
## [41] -0.0025985329 0.0326820244 0.0547968106 0.0120654862 0.0393865756  
## [46] -0.0153697863 -0.0047900795 -0.1633966111 0.0410016031 -0.0027056215  
## [51] -0.0038562358 0.0209539880 0.0360465417 -0.0112953428 0.0217293995  
## [56] 0.0157917783 0.0066660554 0.0421179337 -0.0031570045 -0.0282951027  
## [61] 0.0232013296 -0.0023694756 -0.0201797253 0.0004271265 -0.0185697035  
## [66] -0.0009451896 0.0382508079 0.0029864401 -0.0151167053 -0.0290659842  
## [71] -0.0130337652

#c  
summary(bby\_m\_log\_ret)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -0.1633966 -0.0203828 -0.0012717 -0.0001338 0.0202258 0.1384360

#d  
print(acf(bby\_m\_log\_ret, lag.max = 12))



##   
## Autocorrelations of series 'bby\_m\_log\_ret', by lag  
##   
## 0 1 2 3 4 5 6 7 8 9 10   
## 1.000 0.023 0.115 -0.092 -0.031 -0.102 0.046 0.172 -0.150 -0.048 -0.303   
## 11 12   
## -0.057 -0.166

#Question 3.12

Box.test(bby\_m\_log\_ret, lag = 12, type = "L")

##   
## Box-Ljung test  
##   
## data: bby\_m\_log\_ret  
## X-squared = 17.68, df = 12, p-value = 0.1258

#Yes the result is consistent with the hypothesis that the log-prices of Best Buy stock follow RW3 on the grounds that the p-value is more noteworthy than 0.05, implying that there is a high probability that our outcome will be consistent with the null hypothesis

#Question 3.13

#a  
y = bby\_m\_log\_ret - mean(bby\_m\_log\_ret)  
y

## [1] -0.0011136803 0.0456862022 0.0196314902 -0.0204520632 -0.0664802232  
## [6] 0.0073739193 -0.0296382092 0.0804950965 0.0150924682 -0.0005570853  
## [11] -0.0656221843 -0.0011378560 -0.0164587543 -0.0369370303 0.0270334851  
## [16] 0.0071296299 -0.0033714580 -0.0401380668 -0.0251280747 -0.0320992532  
## [21] 0.0428969429 0.0108768818 -0.0500288893 0.0110427718 0.0107000808  
## [26] -0.0143455574 -0.0245182621 -0.0572296837 0.0416865959 -0.0512565922  
## [31] -0.0073005697 -0.0117429154 -0.0447622562 -0.0618072544 -0.0442965987  
## [36] 0.1385697795 0.0037223088 0.1107846277 0.0570103566 0.0191822220  
## [41] -0.0024647063 0.0328158510 0.0549306372 0.0121993128 0.0395204022  
## [46] -0.0152359597 -0.0046562529 -0.1632627845 0.0411354297 -0.0025717949  
## [51] -0.0037224092 0.0210878146 0.0361803682 -0.0111615162 0.0218632261  
## [56] 0.0159256048 0.0067998820 0.0422517603 -0.0030231779 -0.0281612761  
## [61] 0.0233351562 -0.0022356490 -0.0200458987 0.0005609531 -0.0184358769  
## [66] -0.0008113630 0.0383846345 0.0031202666 -0.0149828787 -0.0289321576  
## [71] -0.0128999386

x6 = y[6:71] + y[5:70] + y[4:69] + y[3:68] + y[2:67] + y[1:66]  
x6

## [1] -0.015354355 -0.043878884 -0.009069990 -0.013609012 0.006285966  
## [6] 0.007144005 -0.001367770 0.011811685 -0.105620442 -0.093679425  
## [11] -0.085992710 -0.023741984 -0.062742194 -0.071411515 -0.066573738  
## [16] -0.050710280 -0.046963028 -0.093620459 -0.042439621 -0.006611465  
## [21] 0.011142231 -0.056272975 -0.124379540 -0.032664055 -0.094963419  
## [26] -0.112964069 -0.110361427 -0.130605421 -0.135182992 -0.221166187  
## [31] -0.031339815 -0.020316936 0.102210607 0.203983219 0.284972696  
## [36] 0.326804588 0.221050660 0.272258988 0.173673673 0.156183719  
## [41] 0.121765537 0.119573991 -0.076504645 -0.090299853 -0.105070960  
## [46] -0.148313772 -0.111989997 -0.071153376 0.080947892 0.061675689  
## [51] 0.080173088 0.090695380 0.111859325 0.072655779 0.055656019  
## [56] 0.057127949 0.038966695 0.012120915 -0.029569892 -0.044982591  
## [61] -0.017632678 -0.002583200 0.002772716 0.007835736 -0.021657375  
## [66] -0.016121437

V6 = (71/66)\*(1/6)\*(sum(x6^2)/66)/var(y)  
V6

## [1] 1.165087

#b  
#considering mean would be 1   
q = 6  
Variance\_V6 = ((2\*((2\*q) - 1)\*(q-1))/(3\*q))\*(1/71)  
Variance\_V6

## [1] 0.08607199

V = 1/Variance\_V6  
V

## [1] 11.61818

Vq =sqrt(V)\*(V6 - 1)  
Vq

## [1] 0.5627074

#c  
p\_value = 2\*(1-pnorm(Vq))  
p\_value

## [1] 0.5736342

#d  
#Therefore, there is no evidence to reject the null hypothesis that Best Buy stock log-prices follow RW3. because p value is larger than V6'(Vq)

#Question 3.14

install.packages("randtests",repos = "http://cran.us.r-project.org")

## Installing package into 'C:/Users/ahmad/OneDrive/Documents/R/win-library/3.6'  
## (as 'lib' is unspecified)

## package 'randtests' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\ahmad\AppData\Local\Temp\RtmpUdusLq\downloaded\_packages

library(randtests)

##   
## Attaching package: 'randtests'

## The following object is masked from 'package:tseries':  
##   
## runs.test

runs.test(bby\_m\_log\_ret)

##   
## Runs Test  
##   
## data: bby\_m\_log\_ret  
## statistic = -1.4448, runs = 30, n1 = 35, n2 = 35, n = 70, p-value =  
## 0.1485  
## alternative hypothesis: nonrandomness

#Based on the runs test results, I conclude that there is no evidence to reject the null hypothesis that random walk hypothesis can be applied to Best Buy stock prices.