Midterm Part II

Nicholas Colonna

3/25/2018

## R Markdown

#install.packages("quantmod")  
#install.packages("tseries")  
require("quantmod")

## Loading required package: quantmod

## Loading required package: xts

## Loading required package: zoo

##   
## Attaching package: 'zoo'

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

## Loading required package: TTR

## Version 0.4-0 included new data defaults. See ?getSymbols.

require("tseries") #for normality test

## Loading required package: tseries

#1 Download daily data for the past 3 years for 5 equities  
getSymbols("JPM", src="yahoo", from=as.Date("2015-03-01"), to=as.Date("2018-3-01"))

## '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.

##   
## WARNING: There have been significant changes to Yahoo Finance data.  
## Please see the Warning section of '?getSymbols.yahoo' for details.  
##   
## This message is shown once per session and may be disabled by setting  
## options("getSymbols.yahoo.warning"=FALSE).

## [1] "JPM"

getSymbols("GS", src="yahoo", from=as.Date("2015-03-01"), to=as.Date("2018-3-01"))

## [1] "GS"

getSymbols("BAC", src="yahoo", from=as.Date("2015-03-01"), to=as.Date("2018-3-01"))

## [1] "BAC"

getSymbols("C", src="yahoo", from=as.Date("2015-03-01"), to=as.Date("2018-3-01"))

## [1] "C"

getSymbols("WFC", src="yahoo", from=as.Date("2015-03-01"), to=as.Date("2018-3-01"))

## [1] "WFC"

#\*\*Note: All calculations done using adjusted closing prices for each equity\*\*

#2 Calculate montly log returns and output the mean monthly return for each of the 5 equities  
returnsJPM <- monthlyReturn(Ad(JPM), type="log")  
mean(returnsJPM)

## [1] 0.01952774

returnsGS <- monthlyReturn(Ad(GS), type="log")  
mean(returnsGS)

## [1] 0.00989461

returnsBAC <- monthlyReturn(Ad(BAC), type="log")  
mean(returnsBAC)

## [1] 0.02052758

returnsC <- monthlyReturn(Ad(C), type="log")  
mean(returnsC)

## [1] 0.01041419

returnsWFC <- monthlyReturn(Ad(WFC), type="log")  
mean(returnsWFC)

## [1] 0.003773735

#3 Calculate correlation between monthly returns. Which pair has the highest correlation?  
JPMvsGS <- cor(returnsJPM, returnsGS)  
JPMvsGS[1]

## [1] 0.9178383

JPMvsBAC <- cor(returnsJPM, returnsBAC)  
JPMvsBAC[1]

## [1] 0.9290794

JPMvsC <- cor(returnsJPM, returnsC)  
JPMvsC[1]

## [1] 0.8868562

JPMvsWFC <- cor(returnsJPM, returnsWFC)  
JPMvsWFC[1]

## [1] 0.8311147

GSvsBAC <- cor(returnsGS, returnsBAC)  
GSvsBAC[1]

## [1] 0.908706

GSvsC <- cor(returnsGS, returnsC)  
GSvsC[1]

## [1] 0.8357428

GSvsWFC <- cor(returnsGS, returnsWFC)  
GSvsWFC[1]

## [1] 0.8011588

BACvsC <- cor(returnsBAC, returnsC)  
BACvsC[1]

## [1] 0.8914001

BACvsWFC <- cor(returnsBAC, returnsWFC)  
BACvsWFC[1]

## [1] 0.8258725

CvsWFC <- cor(returnsC, returnsWFC)  
CvsWFC[1]

## [1] 0.7302742

#The pair with the highest correlation is JPM and BAC, with a correlation of 0.929  
  
  
#4 Linear regression model between highest correlated monthly return equities  
linearRegression <- lm(Ad(JPM) ~ Ad(BAC))   
#Regression done on adjusted closing prices for each equity (data wasn't specified on exam)  
summary(linearRegression)

##   
## Call:  
## lm(formula = Ad(JPM) ~ Ad(BAC))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -6.3637 -2.3707 0.2866 2.1058 6.5724   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.43600 0.35786 37.55 <2e-16 \*\*\*  
## Ad(BAC) 3.15855 0.01787 176.72 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.67 on 754 degrees of freedom  
## Multiple R-squared: 0.9764, Adjusted R-squared: 0.9764   
## F-statistic: 3.123e+04 on 1 and 754 DF, p-value: < 2.2e-16

#5 Interpret the results from the linar regression

# (Adjusted Close JPM) = Intercept + Beta \* (Adjusted Close BAC)  
# Therefore, our regression model is: JPM = 13.436 + 3.159 \* BAC  
# The parameters in the model are significant. When looking at the summary of the regression, you can see that the p-value is very small, which means the parameters are significant. They were give '\*\*\*' in the significance test  
# Therefore, there is evidence that the closing price of BAC has explanatory power when determining the closing price of JPM  
# The coefficient of multiple determination, also known as R-squared, for the model is 0.9764  
  
  
#Bonus: Test returns for normality  
#Used Jarque Bera Test, as shown in the sample midterm   
jarque.bera.test(returnsJPM)

##   
## Jarque Bera Test  
##   
## data: returnsJPM  
## X-squared = 0.090306, df = 2, p-value = 0.9559

jarque.bera.test(returnsGS)

##   
## Jarque Bera Test  
##   
## data: returnsGS  
## X-squared = 1.8786, df = 2, p-value = 0.3909

jarque.bera.test(returnsBAC)

##   
## Jarque Bera Test  
##   
## data: returnsBAC  
## X-squared = 3.6355, df = 2, p-value = 0.1624

jarque.bera.test(returnsC)

##   
## Jarque Bera Test  
##   
## data: returnsC  
## X-squared = 4.6119, df = 2, p-value = 0.09966

jarque.bera.test(returnsWFC)

##   
## Jarque Bera Test  
##   
## data: returnsWFC  
## X-squared = 0.23787, df = 2, p-value = 0.8879

# Since the p-value for the JB test is greater than 0.05 for all of the equities, we can assume that the log returns are not normally distributed  
  
# I ran out of time to do any more research on this topic