Introductory Financial Econometrics

Spring 2019

Wasin Siwasarit

Project Spring 2019, Faculty of Economics, Thammasat University.

Academic Year Spring 2019

Road Map of this class:

1. Financial Time Series and Their Characteristics

Financial time series (FTS) analysis

Financial time series (FTS) analysis is concerned with theory and practice of asset valuation over time.

What is the difference, if any, from traditional time series analysis?

Two topics are highly related, but FTS has added uncertainty, because it must deal with the ever-changing business & economic environment and the fact that volatility is not directly observed.

1.1 The Objectives of this chapter

1. to access financial data online and to process the embedded information
2. to provide basic knowledge of FTS data such as skewness, heavy tails, and measure of dependence

between asset returns

3. to introduce statistical tools econometric models useful for analyzing these series.
4. to gain experience in analyzing FTS

1.2 Examples of financial time series

1. Daily log returns of Apple stock: 2007 to 2018 (12 years). Data downloaded using quantmod

2. The VIX index.

3. CDS spreads: Daily 3-year CDS spreads of JP Morgan from July 20, 2004 to September 19, 2018.

4. Quarterly earnings of Coca-Cola Company: 1983-2009 Seasonal time series useful in

• earning forecasts

• pricing weather related derivatives (e.g. energy) • modeling intraday behavior of asset returns

5. US monthly interest rates (3m & 6m Treasury bills)

Relations between the two asset returns? Term structure of interest rates.

6. Exchange rate between US Dollar vs Euro Fixed income, hedging, carry trade.

7. Size of insurance claims.

8. High-frequency financial data:
Tick-by-tick data of Caterpillars stock: January 04, 2010.

1.3 Asset Returns

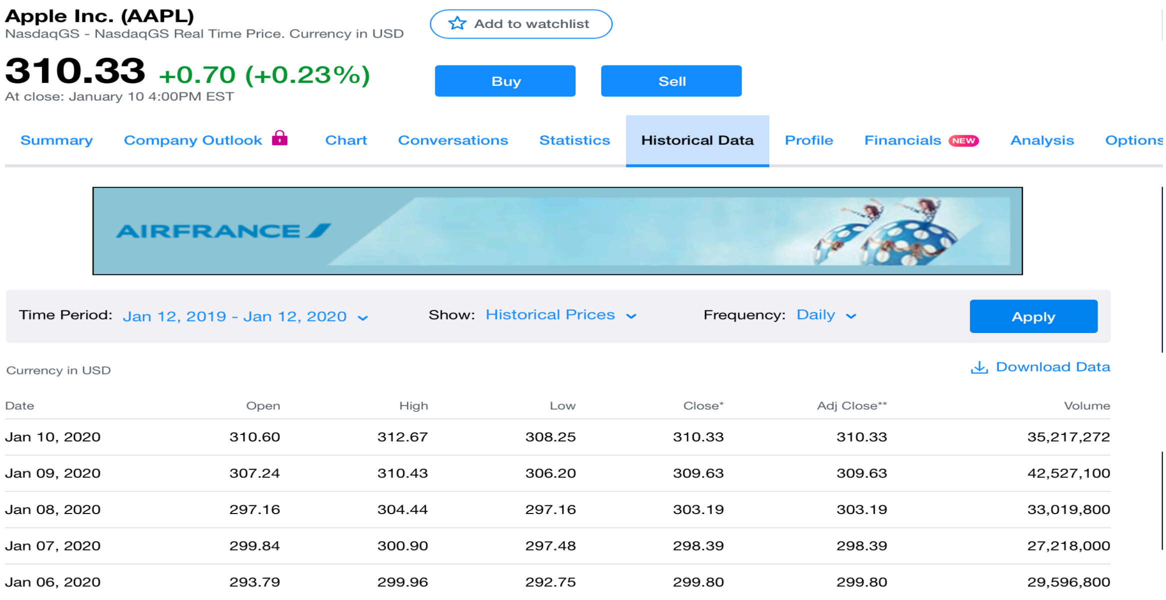
Let P_t be the price of an asset at time t, and assume no dividend. One-period simple return: Gross return

$$1+R_t = \frac{P_t}{P_{t-1}}$$

One-Period Simple Net Return or Simple Return:

Multiperiod simple return: Gross return)

Example:Table below gives 5 daily prices of Apple stock in January 2020.



Date	Open	High	Low	Close*	Adj Close**	Volume
Jan 10, 2020	310.60	312.67	308.25	310.33	310.33	35,217,272
Jan 09, 2020	307.24	310.43	306.20	309.63	309.63	42,527,100
Jan 08, 2020	297.16	304.44	297.16	303.19	303.19	33,019,800
Jan 07, 2020	299.84	300.90	297.48	298.39	298.39	27,218,000
Jan 06, 2020	293.79	299.96	292.75	299.80	299.80	29,596,800

what is one-day gross return of holding the stock from 01/06 to 12/07 and the daily simple return?

what is one-day log return of holding the stock from 01/09 to 01/10 ?

Time interval is important! Default is one year. Annualized (average) return:

Besides the simple return, we can also compute the continuously compounding interest rate where r is the interest rate per annum, C is the initial capital, n is the number of years, and exp is the exponential function.

$$A = C \times exp(r \times n)$$

Continuously compounded (or log) return

$$r_t = ln(1+R_t) = ln\frac{P_t}{P_{t-1}} = p_t - p_{t-1}$$

where $p_t = ln(P_t)$

Multiperiod log return:

Continuously compounding: Illustration of the power of compounding (int. rate 10 % per annum)

Type	#(payment)	Int.	Net
Annual	1	0.1	\$1.10000
Semi-Annual	2	0.05	\$1.10250
Quarterly	4	0.025	\$1.10381
Monthly	12	0.0083	\$1.10471
Weekly	52	$\frac{0.1}{52}$	\$1.10506
Daily	365	$\frac{0.1}{365}$	\$1.10516
Continuously	∞		\$1.10517

Portfolio return: N assets

Dividend payment:

Excess Returns (adjusting for risk)

Remarks:

Example If the monthly log returns of an asset are 4.46 %, −7.34 % and 10.77 %, then what is the corresponding quarterly log return?

Example If the monthly simple returns of an asset are 4.46 %, −7.34 %, and 10.77 %, then what is the corresponding quarterly simple return?

1.4 Distributional Properties of Returns

What is the distribution of
r_{it} where $i = 1, ..., N$; and $t = 1, ..., T$

Some theoretical properties:

Moments of a random variable X with density f(x): l-th moment

$$m_l' = E(X^l) = \int_{\infty}^{\infty} x^l f(x) dx$$

First Moment: mean or expectation of X.

l-th central moment

$$m_l = E[(X - \mu_x)^l] = \int_{\infty}^{\infty} (x - \mu_x)^l f(x) dx$$

2nd central moment.

Standard deviation: square-root of variance

Skewness (Symmestry)

$$S(x) = E\left[\frac{(X-\mu_x)^3}{\sigma_x^3}\right]$$

Kurtosis (Fat-taills)

$$K(x) = E\left[\frac{(X-\mu_x)^4}{\sigma_x^4}\right]$$

Q1: Why study the mean and variance of returns?
They are concerned with long-term return and risk, respectively.
Q2: Why is symmetry important?
Symmetry has important implications in holding short or long financial positions and in risk management.
Q3: Why is kurtosis important?
Related to volatility forecasting, efficiency in estimation and tests High kurtosis implies heavy (or long) tails in distribution.

Estimation

Sample mean, Sample Variance, Sample Skewness and Sample Kurtosis

1.5 Hypothesis Testing

A random variable under the normal distribution

$$\widehat{S(x)} \sim N(0, \frac{6}{T})$$

$$\widehat{K(x)} - 3 \sim N(0, \frac{24}{T})$$

Test for symmetry

Test for tail thickness

Test for normality :(Jarque-Bera test)

1.6 Empirical work using R program

FE code

```
#EE435 Wasin Siwasarit  Lecture1  Spring/2019
setwd("/Users/wasin_siwasarit/Desktop/EE435")
cat(rep("\n",50))  #clear R Console
#install.packages("quantmod")
#install.packages("fBasics")
#install.packages("sn")
#install.packages("PerformanceAnalytics")
#install.packages("car")
#install.packages("tseries")
#install.packages("forecast")
library(quantmod)
library(fBasics)
library(sn)
library(PerformanceAnalytics)
library(car)
library(tseries)
library(forecast)

getSymbols("^GSPC",from="2000-01-03",to="2020-01-10")
dim(GSPC)
head(GSPC)
tail(GSPC)
da=GSPC
chartSeries(GSPC,theme="white")
price=da[,6]
plot(price,type='l')
```

```
logprice=log(price)
plot(logprice,type='l')
logreturn=diff(log(price))
simplereturn <-exp(logreturn)-1
#1 Plot the series of log return and simple return

par(mfrow=c(1,1))
plot(logreturn,type='l')
plot(simplereturn)

newlogreturn <- logreturn[2:nrow(logreturn),]
newsimplereturn <- simplereturn[2:nrow(logreturn),]

#2 Histogram and sample statistics
hist(logreturn, breaks=100, col="slateblue")
chart.Histogram(logreturn,methods = c("add.normal"))
table.Stats(logreturn)

#3 QQ-plots and tests for normality
#
# use qqnorm function
par(mfrow=c(1,1))
qqnorm(newlogreturn)
qqline(newlogreturn, col = 2)
jarque.bera.test(newlogreturn)
```

FE Print out

```
> #install.packages("quantmod")
> #install.packages("fBasics")
> #install.packages("sn")
> #install.packages("PerformanceAnalytics")
> #install.packages("car")
> #install.packages("tseries")
> #install.packages("forecast")
> library(quantmod)
> library(fBasics)
> library(sn)
> library(PerformanceAnalytics)
> library(car)
> library(tseries)
> library(forecast)
> getSymbols("^GSPC",from="2000-01-03",to="2020-01-10")
[1] "GSPC"
> dim(GSPC)
[1] 5037    6
```

```
> head(GSPC)
GSPC.Open GSPC.High GSPC.Low GSPC.Close GSPC.Volume GSPC.Adjusted
2000-01-03    1469.25    1478.00   1438.36    1455.22    931800000        1455.22
2000-01-04    1455.22    1455.22   1397.43    1399.42   1009000000        1399.42
2000-01-05    1399.42    1413.27   1377.68    1402.11   1085500000        1402.11
2000-01-06    1402.11    1411.90   1392.10    1403.45   1092300000        1403.45
2000-01-07    1403.45    1441.47   1400.73    1441.47   1225200000        1441.47
2000-01-10    1441.47    1464.36   1441.47    1457.60   1064800000        1457.60
> tail(GSPC)
GSPC.Open GSPC.High GSPC.Low GSPC.Close GSPC.Volume GSPC.Adjusted
2020-01-02    3244.67    3258.14   3235.53    3257.85   3458250000        3257.85
2020-01-03    3226.36    3246.15   3222.34    3234.85   3461290000        3234.85
2020-01-06    3217.55    3246.84   3214.64    3246.28   3674070000        3246.28
2020-01-07    3241.86    3244.91   3232.43    3237.18   3420380000        3237.18
2020-01-08    3238.59    3267.07   3236.67    3253.05   3720890000        3253.05
2020-01-09    3266.03    3275.58   3263.67    3274.70   3638390000        3274.70
> da=GSPC
> chartSeries(GSPC,theme="white")
> price=da[,6]
> plot(price,type='l')
> logprice=log(price)
> plot(logprice,type='l')
> logreturn=diff(log(price))
> simplereturn <-exp(logreturn)-1
> par(mfrow=c(1,1))
> plot(logreturn,type='l')
> plot(simplereturn)
> newlogreturn <- logreturn[2:nrow(logreturn),]
> newsimplereturn <- simplereturn[2:nrow(logreturn),]
> #2 Histogram and sample statistics
> hist(logreturn, breaks=100, col="slateblue")
> chart.Histogram(logreturn,methods = c("add.normal"))
> table.Stats(logreturn)
GSPC.Adjusted
Observations        5036.0000
NAs                    1.0000
Minimum               -0.0947
Quartile 1            -0.0047
Median                 0.0006
Arithmetic Mean        0.0002
Geometric Mean         0.0001
Quartile 3             0.0057
Maximum                0.1096
SE Mean                0.0002
LCL Mean (0.95)       -0.0002
UCL Mean (0.95)        0.0005
```

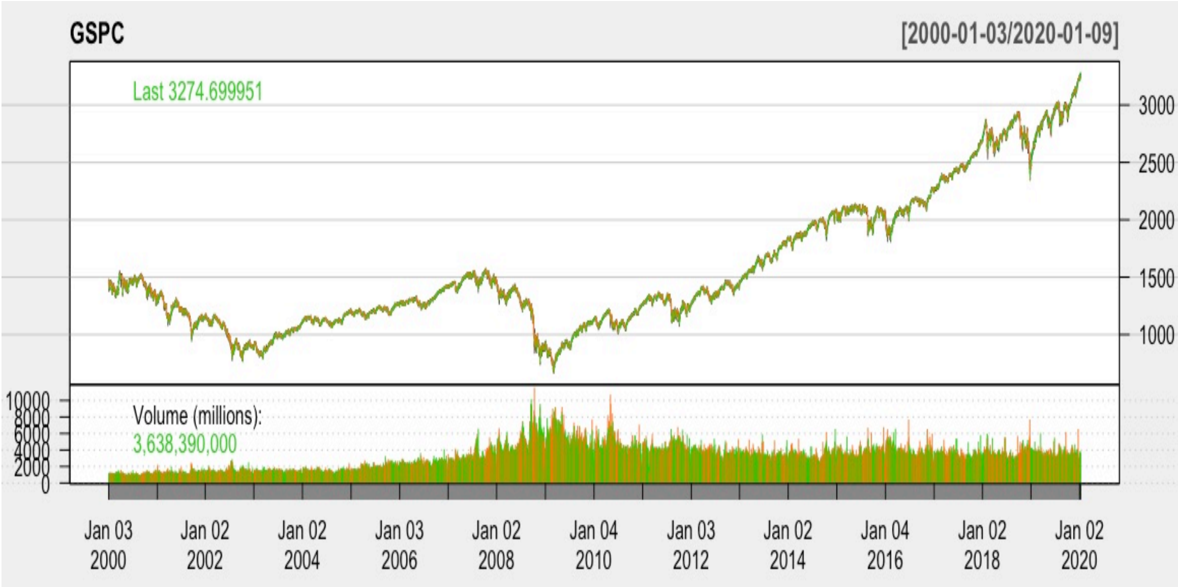
```
Variance                0.0001
Stdev                   0.0119
Skewness               -0.2302
Kurtosis                8.6521
> #3 QQ-plots and tests for normality
> #
> # use qqnorm function
> par(mfrow=c(1,1))
> qqnorm(newlogreturn)
> qqline(newlogreturn, col = 2)
> jarque.bera.test(newlogreturn)

Jarque Bera Test

data:  newlogreturn
X-squared = 15752, df = 2, p-value < 2.2e-16
```

FE code (Cont.)

```
#4 Test mean = 0
t.test(newlogreturn)

#5 Test Skewness = 0
T=length(newlogreturn)
s3=skewness(newlogreturn)
tst = s3/sqrt(6/T)
tst
pv = 2*pnorm(tst)
pv

#6 Test excess kurtosis =0
k4 = kurtosis(newlogreturn)
tst = k4/sqrt(24/T)
tst
pv = 2*(1-pnorm(tst))
pv
```

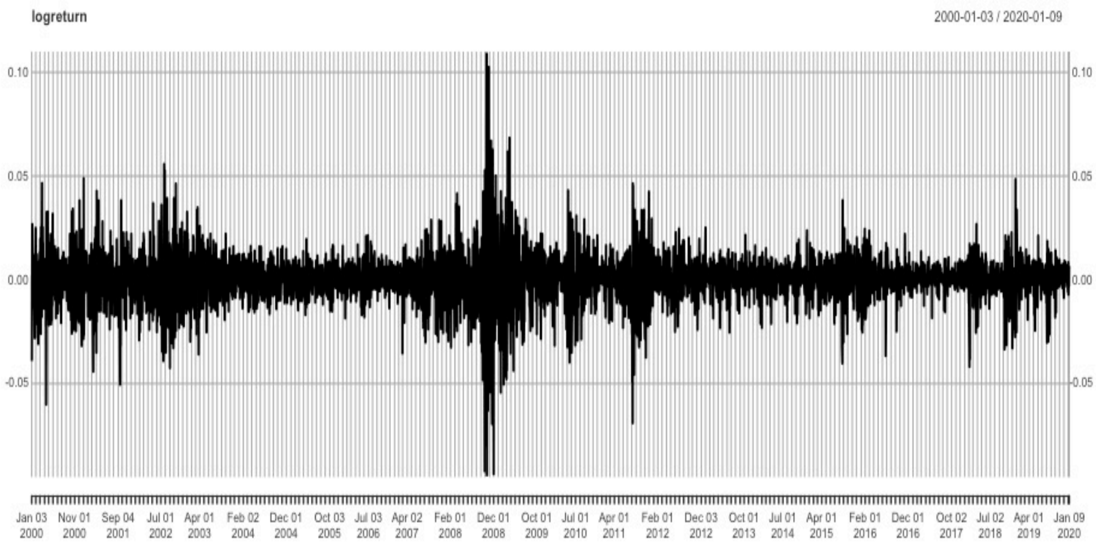
FE Print out (Cont.)

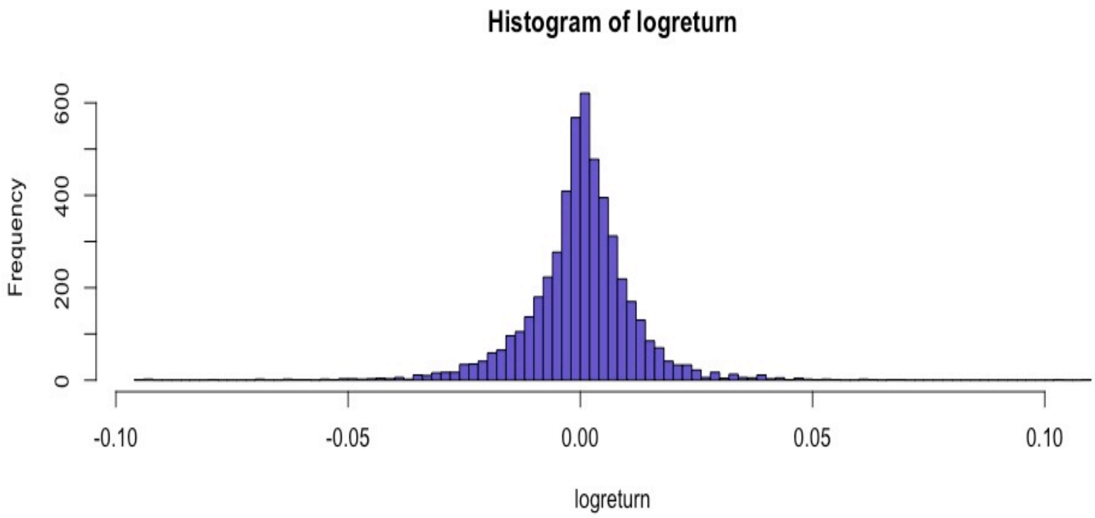
```
> t.test(newlogreturn)

One Sample t-test

data:  newlogreturn
t = 0.96103, df = 5035, p-value = 0.3366
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
-0.0001674840  0.0004895925
sample estimates:
mean of x
0.0001610542


> #5 Test Skewness = 0
> T=length(newlogreturn)
> s3=skewness(newlogreturn)
> s3
[1] -0.230244
> tst = s3/sqrt(6/T)
> tst
[1] -6.670456
> pv = 2*pnorm(tst)
> pv
[1] 2.550093e-11
> #6 Test excess kurtosis =0
> k4 = kurtosis(newlogreturn)
> k4
[1] 8.652073
> tst = k4/sqrt(24/T)
> tst
[1] 125.3307
> pv = 2*(1-pnorm(tst))
> pv
[1] 0
```

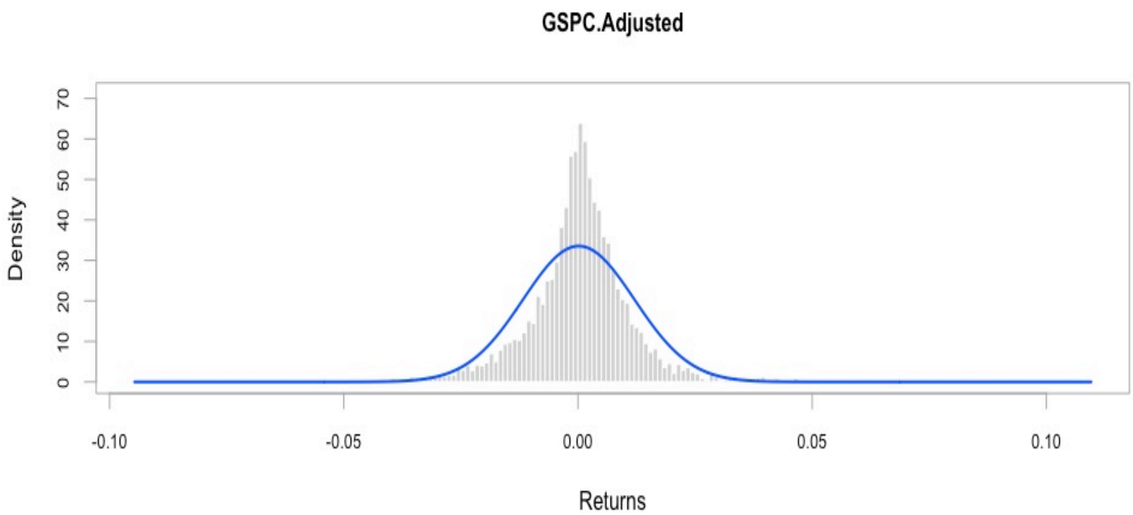

GSPC
[2000-01-03/2020-01-09]
Last 3274.699951
3000
2500
2000
1500
1000
Volume (millions):
3,638,390,000
10000
8000
6000
4000
2000
0
Jan 03 2000
Jan 02 2002
Jan 02 2004
Jan 03 2006
Jan 02 2008
Jan 04 2010
Jan 03 2012
Jan 02 2014
Jan 04 2016
Jan 02 2018
Jan 02 2020

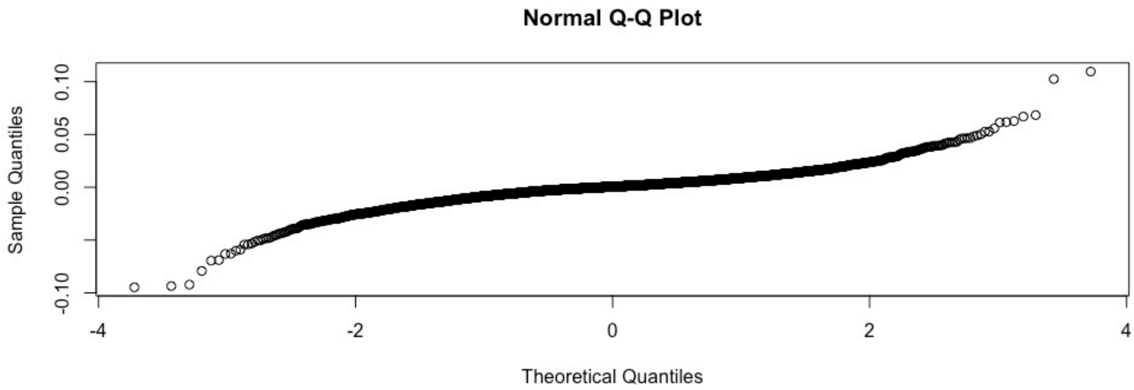

price
2000-01-03 / 2020-01-09
3000
2500
2000
1500
1000
Jan 03 2000
Nov 01 2000
Sep 04 2001
Jul 01 2002
Apr 01 2003
Feb 02 2004
Dec 01 2004
Oct 03 2005
Jul 03 2006
Apr 02 2007
Feb 01 2008
Dec 01 2008
Oct 01 2009
Jul 01 2010
Apr 01 2011
Feb 01 2012
Dec 03 2012
Oct 01 2013
Jul 01 2014
Apr 01 2015
Feb 01 2016
Dec 01 2016
Oct 02 2017
Jul 02 2018
Apr 01 2019
Jan 09 2020

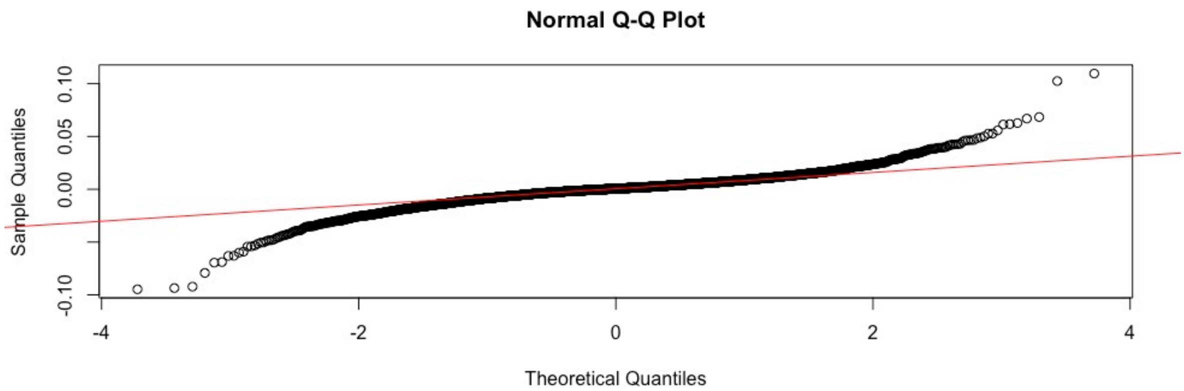

logprice
2000-01-03 / 2020-01-09
8.0
7.5
7.0
Jan 03 2000
Nov 01 2000
Sep 04 2001
Jul 01 2002
Apr 01 2003
Feb 02 2004
Dec 01 2004
Oct 03 2005
Jul 03 2006
Apr 02 2007
Feb 01 2008
Dec 01 2008
Oct 01 2009
Jul 01 2010
Apr 01 2011
Feb 01 2012
Dec 03 2012
Oct 01 2013
Jul 01 2014
Apr 01 2015
Feb 01 2016
Dec 01 2016
Oct 02 2017
Jul 02 2018
Apr 01 2019
Jan 09 2020


logreturn
2000-01-03 / 2020-01-09
0.10
0.05
0.00
-0.05


Histogram of logreturn
Frequency
0
200
400
600
-0.10
-0.05
0.00
0.05
0.10
logreturn


GSPC.Adjusted
Density
0
10
20
30
40
50
60
70
-0.10
-0.05
0.00
0.05
0.10
Returns


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles
-0.10
0.00
0.05
0.10
-4
-2
0
2
4


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles
-0.10
0.00
0.05
0.10
-4
-2
0
2
4