Homework – Semester 2

Nonlinear and multivariate time series

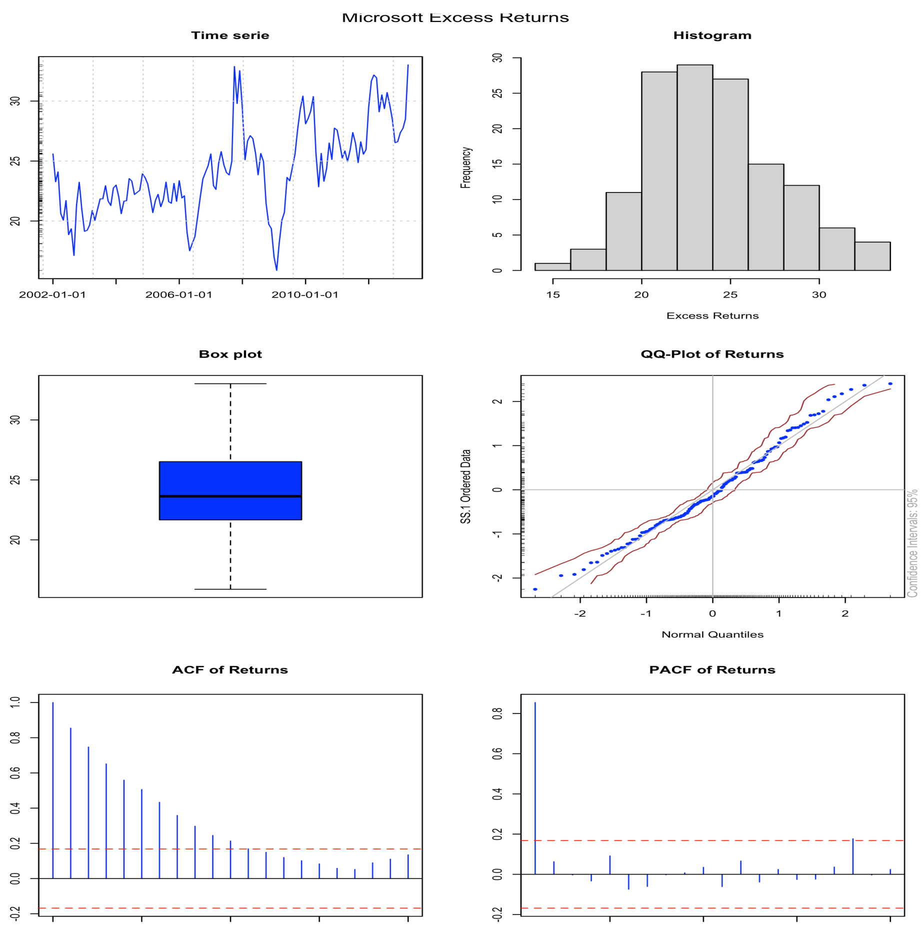
Aldo Caumo, Eleonora Fiorentino

2024-03-24

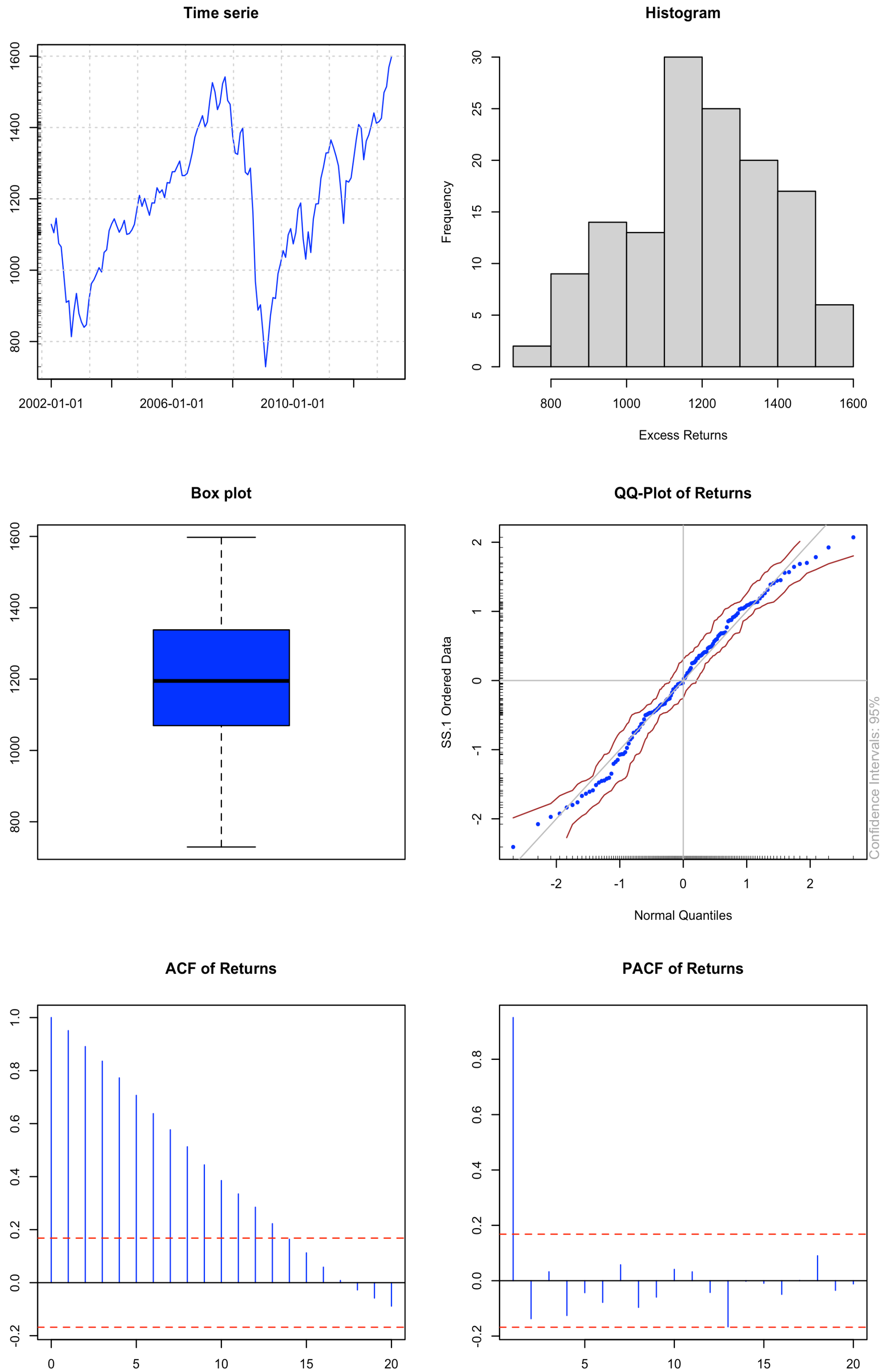
Exercise 1

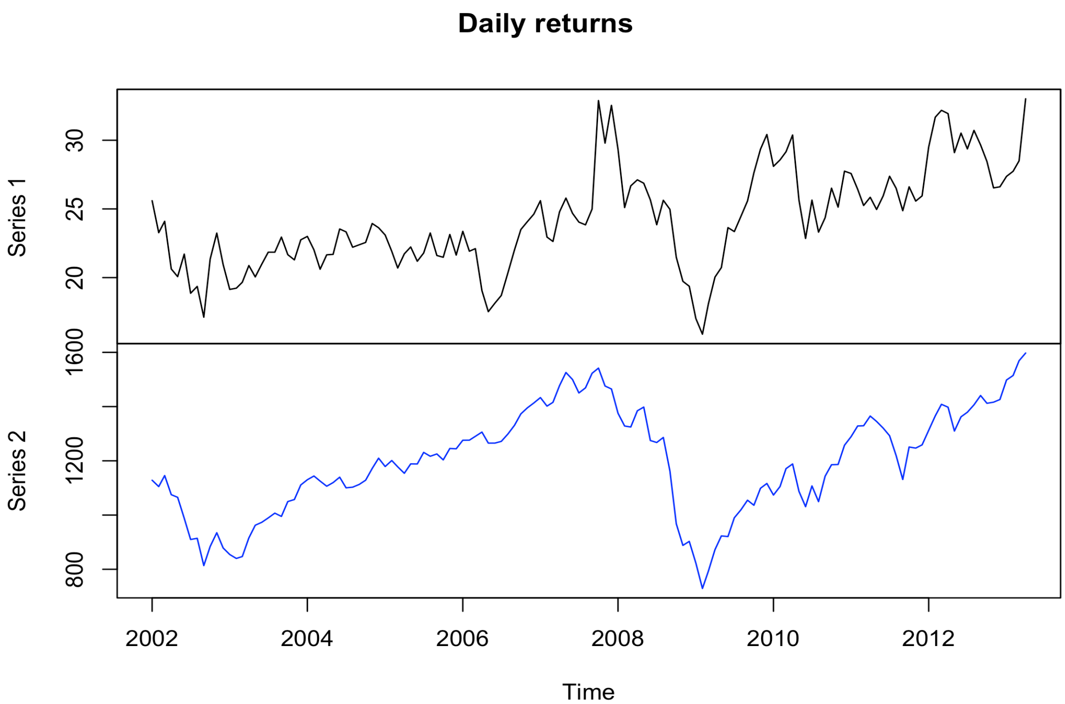
a

Do a preliminary analysis of the excess returns (your asset + market excess returns) based on basics statistics, graphs of the distributions, QQ-plots, box-plot, bivariate scatterplot, graphs of time series, autocorrelation function, pp-plot



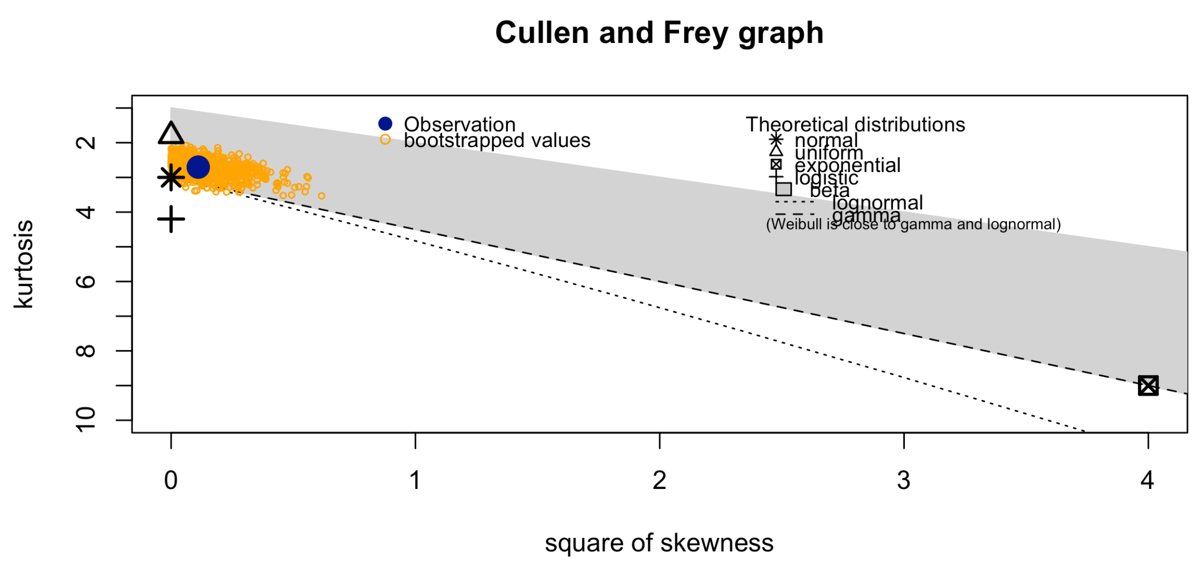
Market Excess Returns



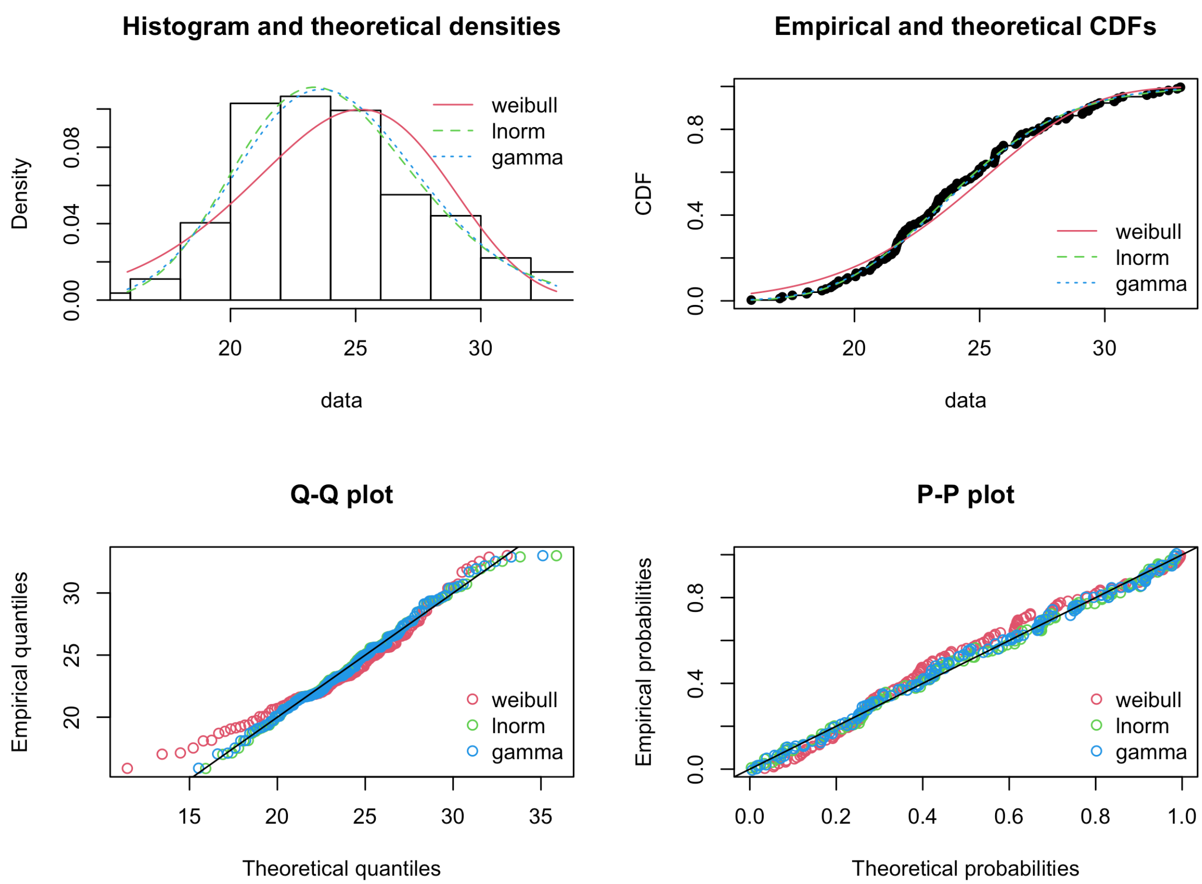

Daily returns
Series 1
30
25
20
Series 2
1600
1200
800
2002
2004
2006
2008
2010
2012
Time

b

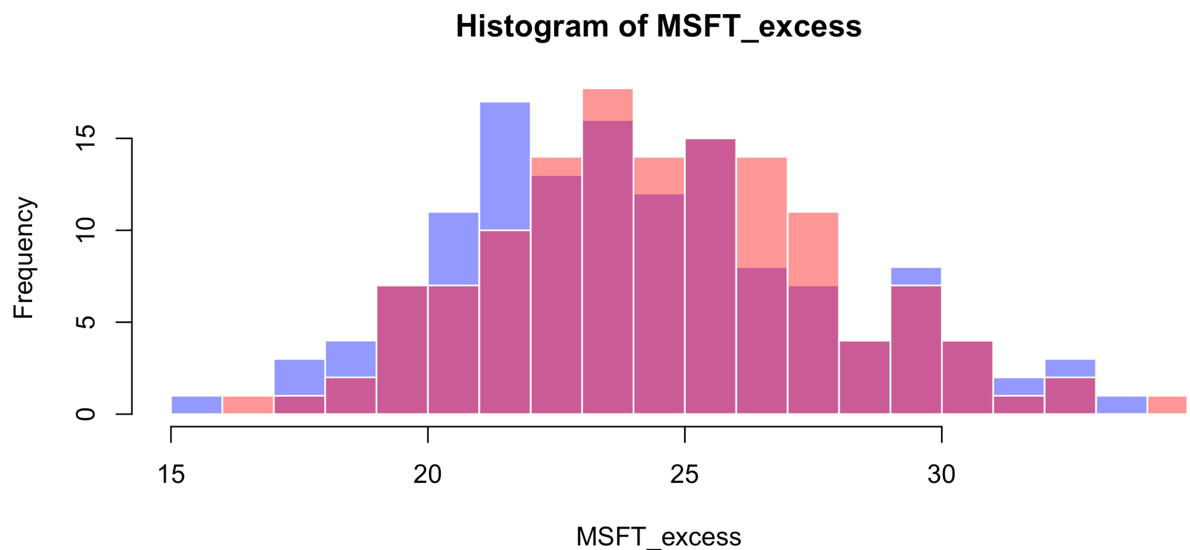
Fit one of the following extreme distribution to the excess returns (your asset + market excess returns): Weibull, log-normal, Gamma.

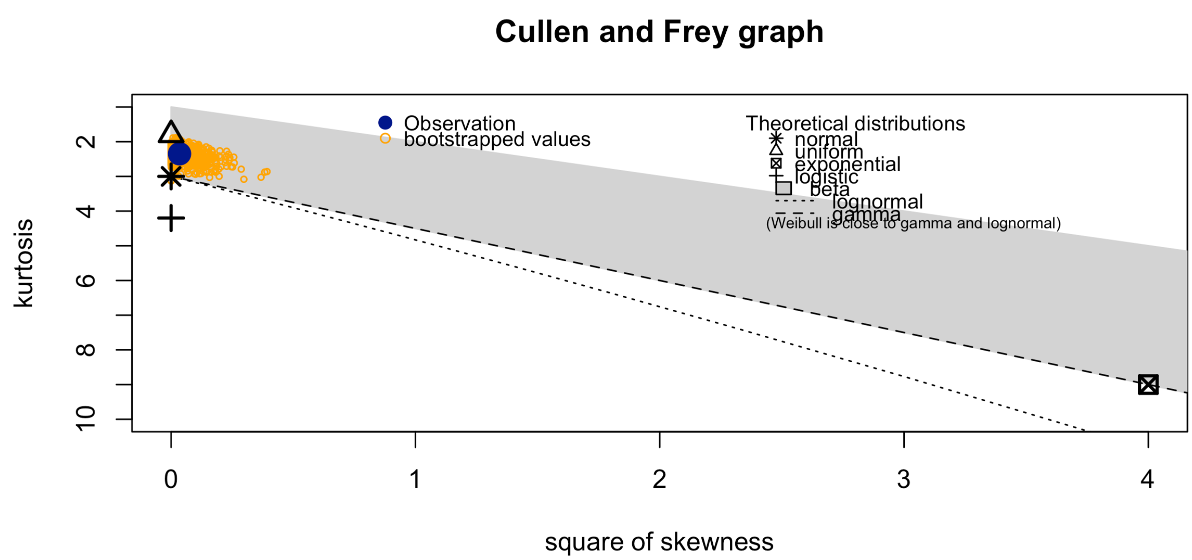


```
## summary statistics
## ------
## min:  15.88   max:  33.02
## median:  23.635
## mean:  24.1764
## estimated sd:  3.679879
## estimated skewness:  0.3333205
## estimated kurtosis:  2.70397
```

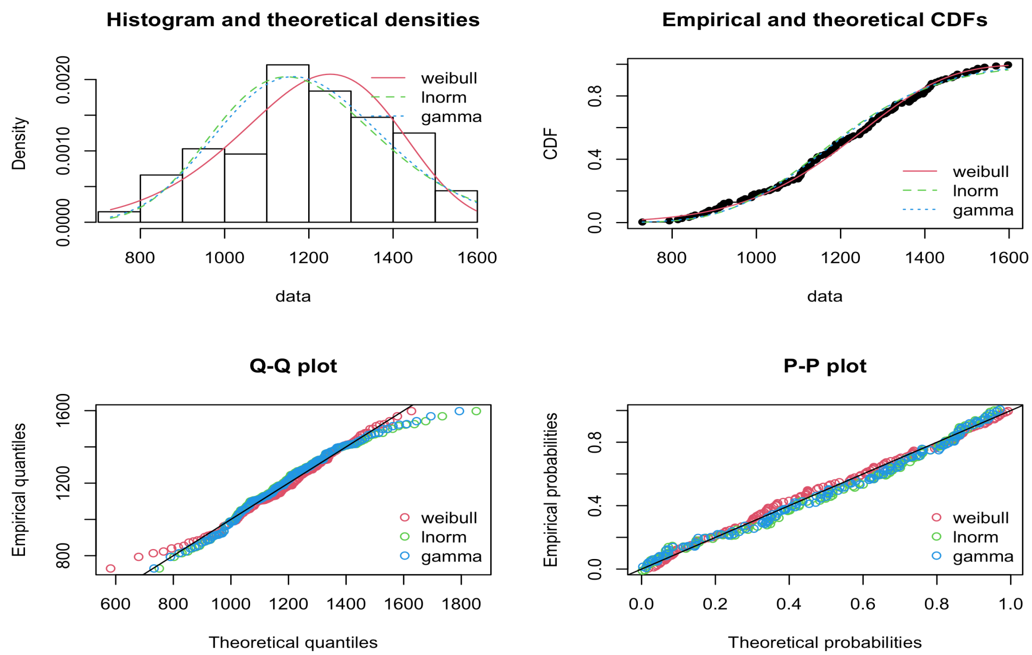


Looking at the qqplots the distribution that seems to best fit our data is a Gamma distribution.

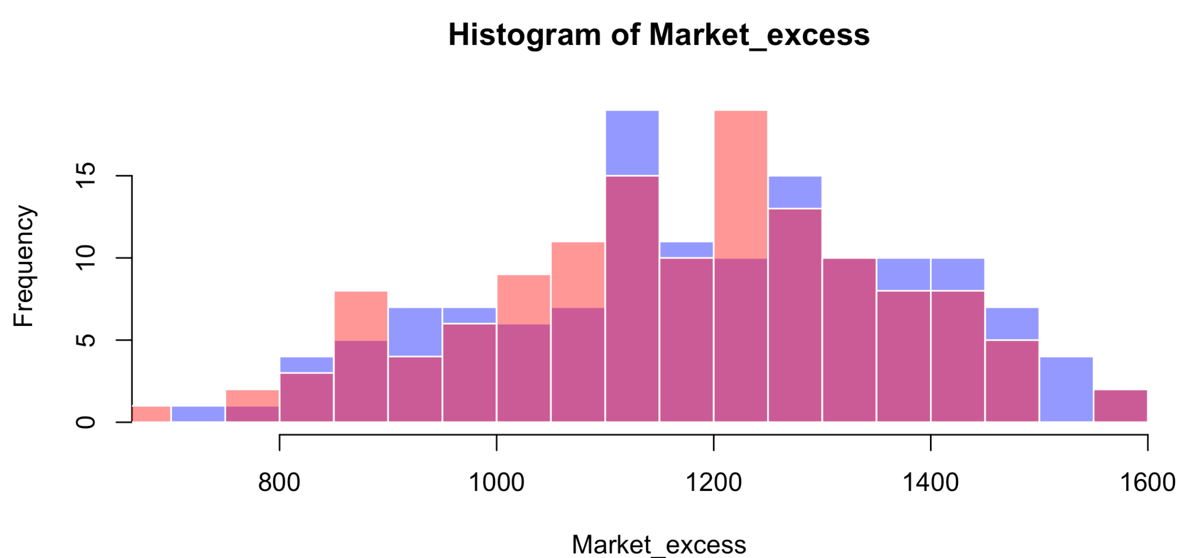





```
## summary statistics
## ------
## min:  729.3   max:  1597.47
## median:  1194.86
## mean:  1195.964
## estimated sd:  193.8403
## estimated skewness:  -0.1854934
## estimated kurtosis:  2.347256
```

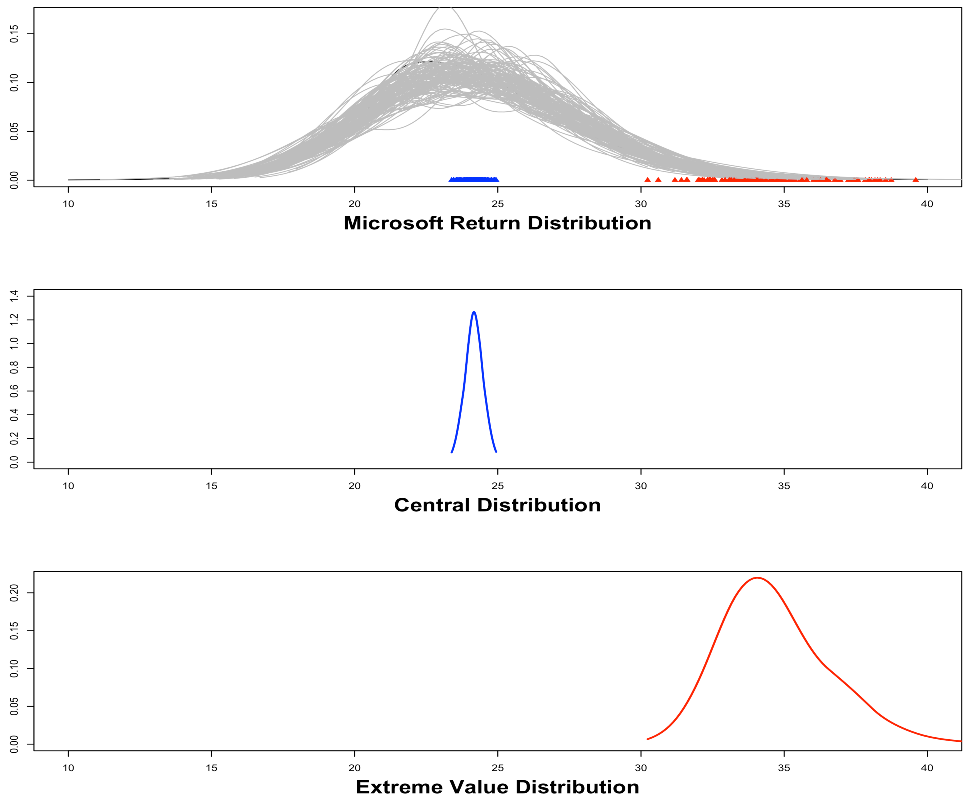


Looking at the qqplots the distribution that seems to best fit our data is a Weibull distribution.

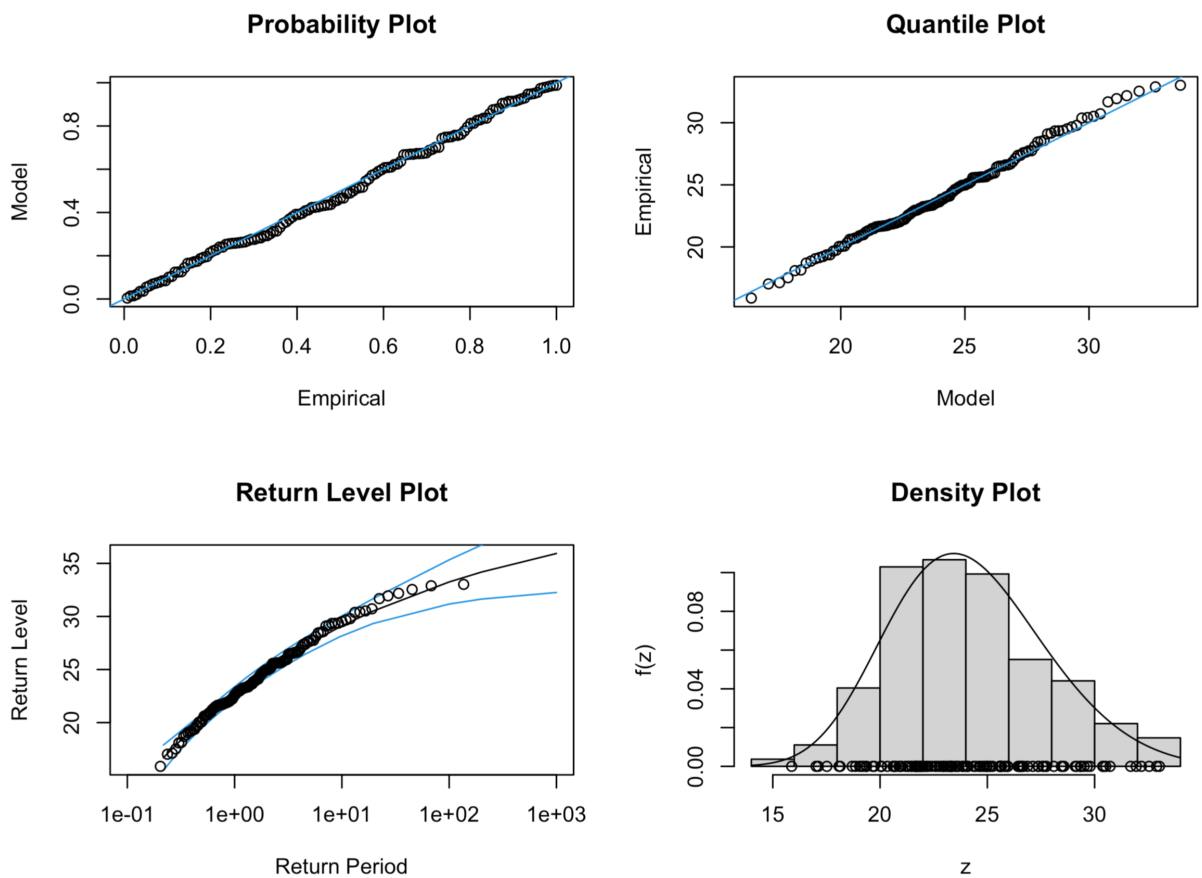


c

Fit a GEV distribution to the excess return series and plot the QQ-plot, density and CDF.

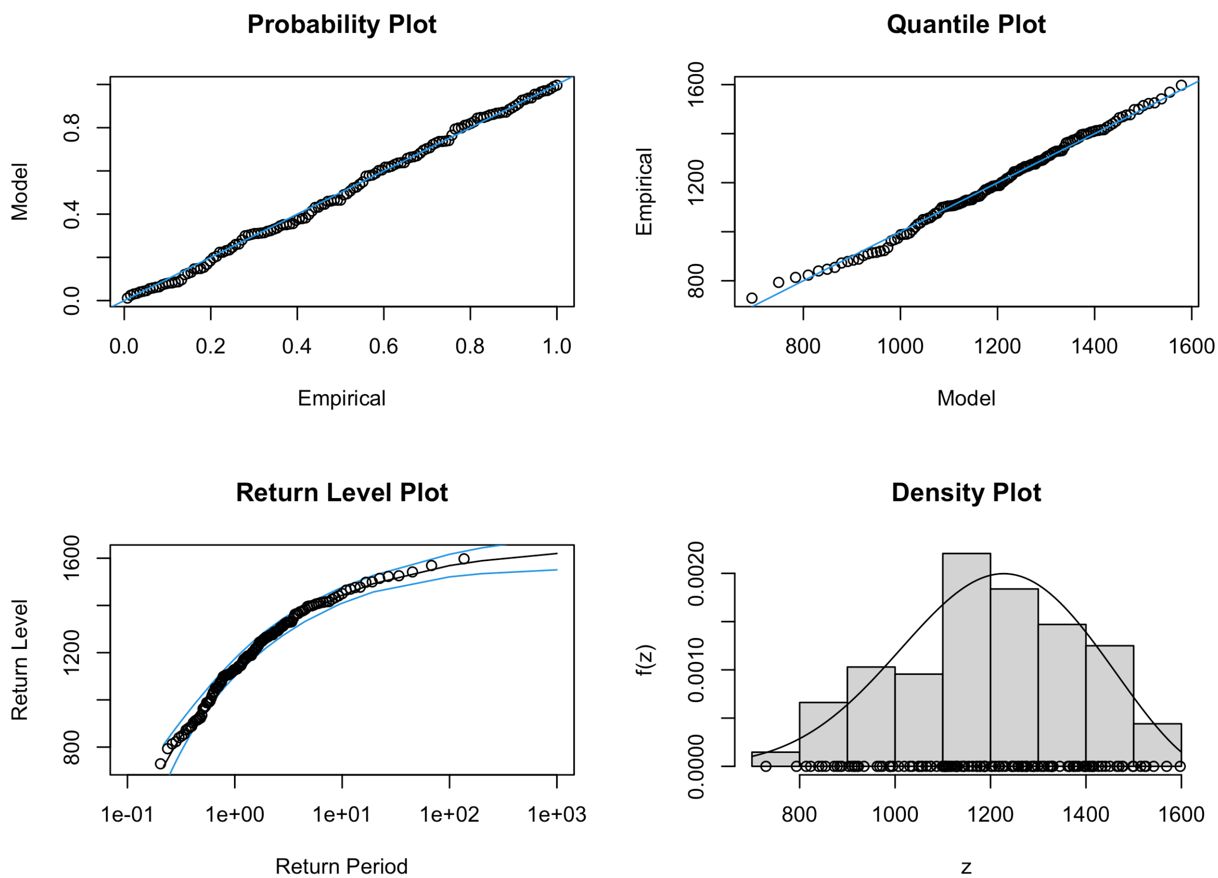



```
## $conv
## [1] 0
##
## $nllh
## [1] 367.5097
##
## $mle
## [1] 22.7356095  3.4124671 -0.1880733
##
## $se
## [1] 0.32832511 0.23243064 0.06314148

MSFT_GEV$mle

## [1] 22.7356095  3.4124671 -0.1880733

gev.diag(MSFT_GEV)
```



It shows the results for the estimated parameters. The shape parameter is -0.18 ($\xi < 0$). So, a Weibull distribution fits the data with high likelihood.

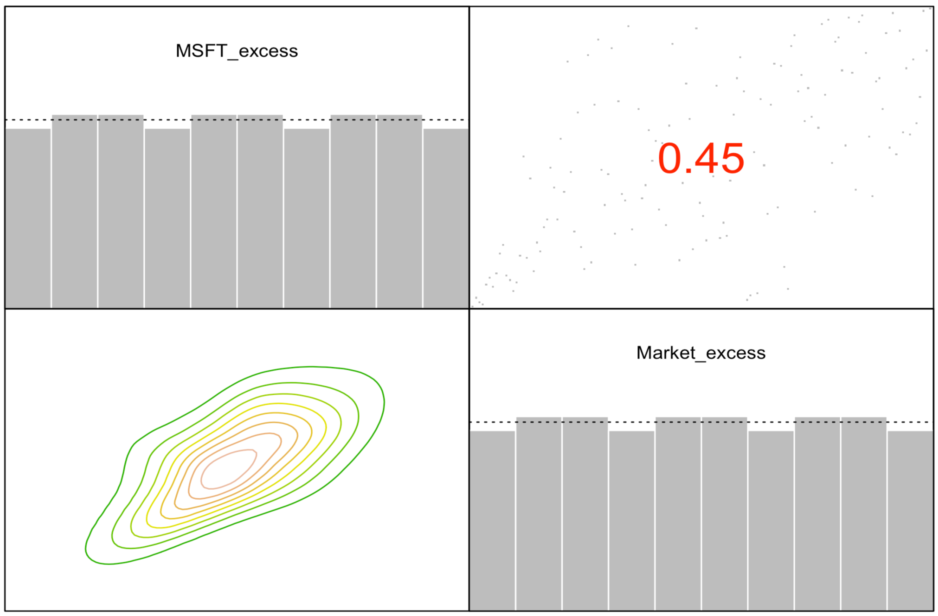
```
## $conv
## [1] 0
##
## $nllh
## [1] 906.2552
##
## $mle
## [1] 1137.648445  201.083884   -0.388338
##
## $se
## [1] 18.92626616 13.97425643  0.05676183

Market_GEV$mle

## [1] 1137.648445  201.083884   -0.388338

gev.diag(Market_GEV)
```



It shows the results for the estimated parameters. The shape parameter is -0.41 ($\xi < 0$). So, a Weibull distribution fits the data with high likelihood.

d

Using elliptical copulas, estimate the correlation between the excess returns of your asset asset and the market excess returns.



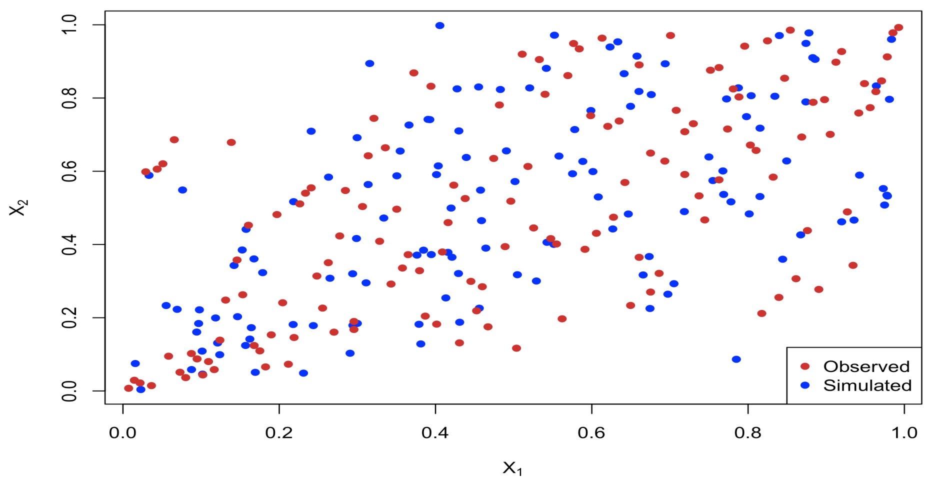

```
##                 MSFT_excess, Market_excess
## Pearson rho                           0.64
## Kendall's tau                         0.45
## Spearman's rho                        0.62
```

e

Fit Archimedian copulas (Clayton or Weibull).

```
## Call: fitCopula(Ccop, data = pseudobs)
## Fit based on "maximum pseudo-likelihood" and 136 2-dimensional
observations.
## Clayton copula, dim. d = 2
##       Estimate Std. Error
## alpha     1.65       0.26
## The maximized loglikelihood is 40.5
## Optimization converged
```

```
## Number of loglikelihood evaluations:
## function gradient
##        3        3
```

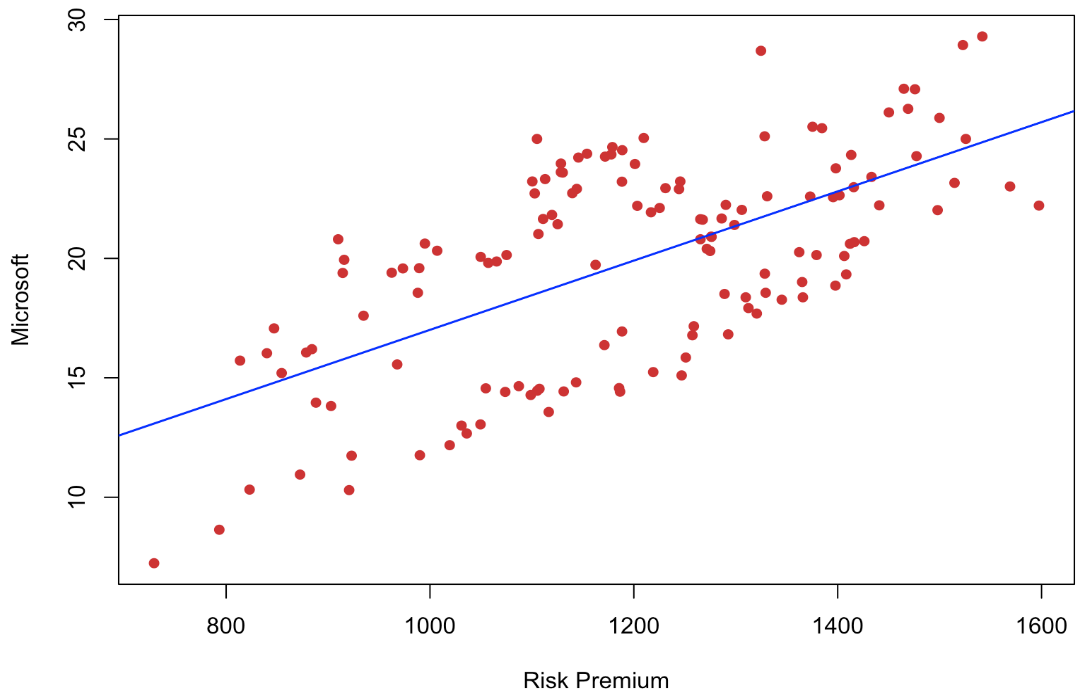


Exercise 2

We consider a market economy containing N financial assets, and we denote by $E(R_{it})$ the expected log-return on the i-th asset on day t. For the purposes of the present discussion, we assume the existence of a market portfolio, and we denote by R_{mt} its expected log-return on day t. Finally, we also assume the existence of lending and borrowing at the same risk-free rate which we denote by R_{ft}, and which we assume to be deterministic. The Sharpe-Lintner version of the CAPM states that the excess return (over the risk-free rate) of each asset i is, up to noise, a linear function of the excess return of the market portfolio. In other words, for each i:

$$E(R_{it}) - R_{ft} = \alpha_i + \beta_i(L)\left[E(R_{mt}) - R_{ft}\right] + \varepsilon_{it}$$

In order to test the CAPM, we consider the daily returns on GE stocks. We use the S&P 500 index as a proxy for the market portfolio, and we use the yield on the 3-month US Treasury bond as a proxy for the risk-free rate of borrowing from which the excess returns are computed.



1

Estimate the above equation using a Markov-switching model with 2 regimes:

a.

Estimate the linear model and use tests on residuals to show that the dynamics is probably nonlinear.

Just to visualize it again, we start by plotting MSFT ts:

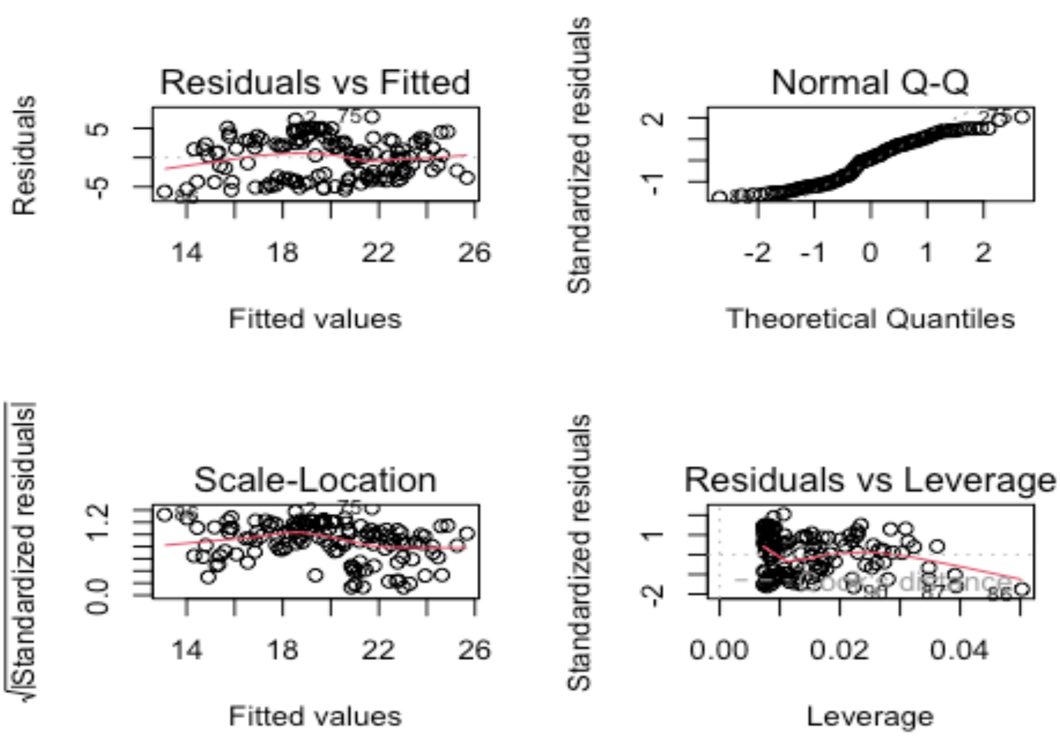
Then we start calculating the actual linear model

```
##
## Call:
## lm(formula = GE_excess ~ Market_excess, data = capm)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.846 -3.334  0.359  2.883  6.974
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.51495    1.84245    1.37     0.17
## Market_excess   0.01449    0.00152    9.53   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.4 on 134 degrees of freedom
## Multiple R-squared:  0.404,  Adjusted R-squared:   0.4
## F-statistic: 90.8 on 1 and 134 DF,  p-value: <2e-16
```

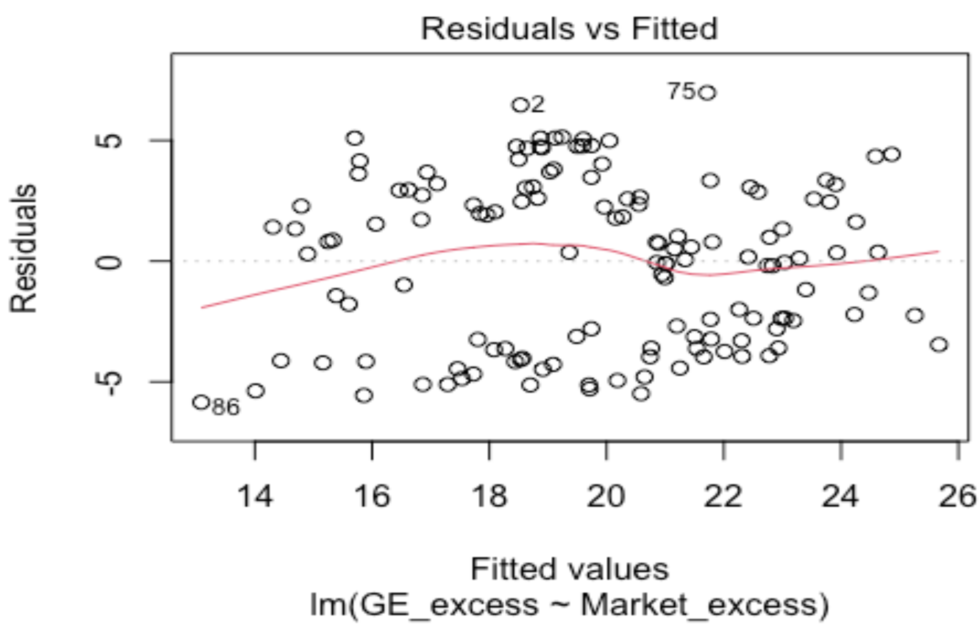
We can notice that there is a poor linear adjustment (low Adj R^2, 0.4)

```
head(model.diag.metrics)

##   GE_excess[,1] Market_excess .fitted .resid    .hat .sigma .cooksd
.std.resid
##           <dbl>         <dbl>   <dbl>  <dbl>   <dbl>  <dbl>   <dbl>
<dbl>
## 1          24.0         1128.    18.9   5.10 0.00825   3.41 0.00929
1.49
## 2          25           1105.    18.5   6.47 0.00898   3.39 0.0163
1.90
## 3          24.2         1146.    19.1   5.10 0.00785   3.41 0.00884
1.49
## 4          20.1         1075.    18.1   2.04 0.0102    3.43 0.00185
0.599
## 5          19.9         1065.    18.0   1.91 0.0107    3.43 0.00171
0.561
## 6          18.6          988.    16.8   1.72 0.0159    3.43 0.00207
0.507
```

Let’s do some diagnostic plots

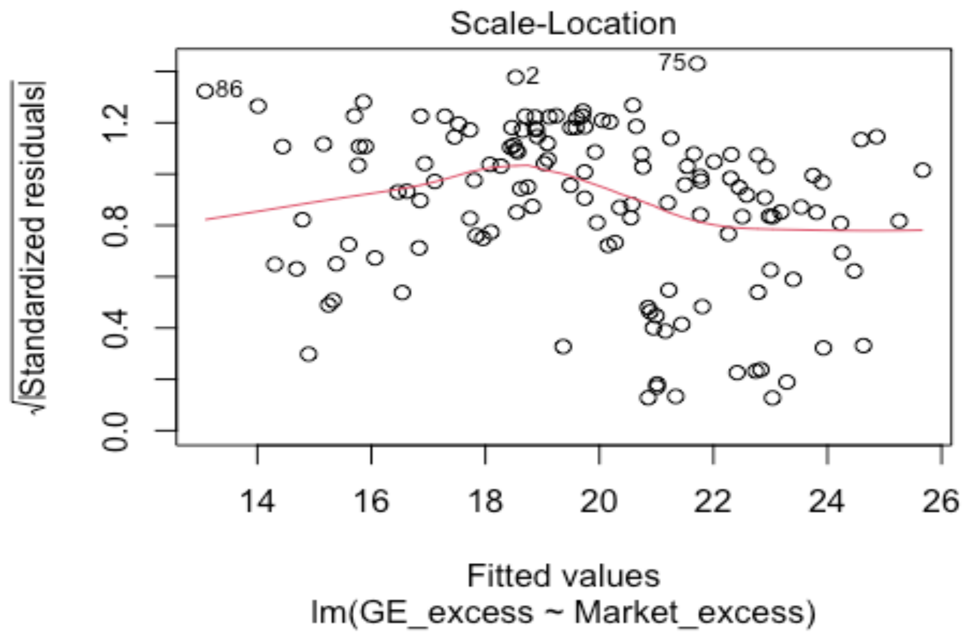


The linearity assumption can be checked by inspecting the Residuals vs Fitted plot (1st plot):



This plot clearly shows that our points are not iid/white noise (in fact they are mostly not near the line but spread all around)

Variance homogeneity can be checked by examining the scale-location plot, also known as the spread-location plot.



This plot shows if residuals are spread equally along the ranges of predictors. It's good if you see a horizontal line with equally spread points. In our example, this is not the case. This suggests non-constant variances in the residuals errors (or heteroscedasticity).

Let's now see if there's a non-linearity problem:

```
##         ** Teraesvirta's neural network test  **
##         Null hypothesis: Linearity in "mean"
##         X-squared =  23  df =  2  p-value =  1.3e-05
##
##         ** White neural network test  **
##         Null hypothesis: Linearity in "mean"
##         X-squared =  20  df =  2  p-value =  5e-05
##
##         ** Keenan's one-degree test for nonlinearity  **
##         Null hypothesis: The time series follows some AR process
##         F-stat =  0.15  p-value =  0.7
##
##         ** McLeod-Li test  **
##         Null hypothesis: The time series follows some ARIMA process
##         Maximum p-value =  1.1e-14
##
##         ** Tsay's Test for nonlinearity **
##         Null hypothesis: The time series follows some AR process
##         F-stat =  1.3  p-value =  0.17
##
```

```
##       ** Likelihood ratio test for threshold nonlinearity **
##       Null hypothesis: The time series follows some AR process
##       Alternativce hypothesis: The time series follows some TAR process
##       X-squared =  27  p-value =  0.066

## $Terasvirta
##
##  Teraesvirta Neural Network Test
##
## data:  ts(time.series)
## X-squared = 23, df = 2, p-value = 1e-05
##
##
## $White
##
##  White Neural Network Test
##
## data:  ts(time.series)
## X-squared = 20, df = 2, p-value = 5e-05
##
##
## $Keenan
## $Keenan$test.stat
## [1] 0.15
##
## $Keenan$df1
## [1] 1
##
## $Keenan$df2
## [1] 114
##
## $Keenan$p.value
## [1] 0.7
##
## $Keenan$order
## [1] 10
##
##
## $McLeodLi
## $McLeodLi$p.values
##  [1] 1.1e-14 0.0e+00 0.0e+00 0.0e+00 0.0e+00 0.0e+00 0.0e+00 0.0e+00
0.0e+00
## [10] 0.0e+00 0.0e+00 0.0e+00 0.0e+00 0.0e+00 0.0e+00 0.0e+00 0.0e+00
0.0e+00
## [19] 0.0e+00 0.0e+00 0.0e+00
##
##
## $Tsay
## $Tsay$test.stat
## [1] 1.3
```

```
##
## $Tsay$p.value
## [1] 0.17
##
## $Tsay$order
## [1] 10
##
##
## $TarTest
## $TarTest$percentiles
## [1] 25 75
##
## $TarTest$test.statistic
## [1] 27
##
## $TarTest$p.value
## [1] 0.066
```

According to the output:

- Teraesvirta's Neural Network Test: examines the null hypothesis that the mean of the time series behaves linearly. The low p-value suggests strong evidence to reject this null hypothesis, indicating the presence of nonlinearity in the mean behavior of the time series.
- White Neural Network Test: similar to Teraesvirta's test, this also assesses the linearity in the mean of the time series. The low p-value provides further evidence against linearity, consistent with the previous test.
- Keenan's One-Degree Test for Nonlinearity: This test investigates whether the time series follows an autoregressive (AR) process, implying linearity. The high p-value (0.698) suggests no significant evidence to reject the null hypothesis, indicating that the time series may behave linearly.
- McLeod-Li Test: This test assesses whether the time series follows an autoregressive integrated moving average (ARIMA) process, implying linearity. The extremely low maximum p-value strongly suggests rejection of the null hypothesis, indicating nonlinearity in the time series.
- Tsay's Test for Nonlinearity: Similar to Keenan's test, Tsay's test evaluates whether the time series follows an AR process.
- Likelihood Ratio Test for Threshold Nonlinearity: This test explores whether the time series follows a threshold autoregressive (TAR) process instead of a simple AR process. The borderline p-value (0.0660172) suggests weak evidence against the null hypothesis of linearity.

In summary, while some tests indicate evidence of nonlinearity (Teraesvirta's and White's tests, McLeod-Li test), others (Keenan's and Tsay's tests) suggest no strong evidence against linearity. The Likelihood Ratio Test provides however somewhat evidence of nonlinearity.

b.

Fit a Markov-switching model (select the optimal lag structure using tests based on information criteria).

Let's start with a basic msm with no lags:

```
## Markov Switching Model
##
## Call: msmFit(object = GE.fit, k = 2, sw = c(TRUE, TRUE, TRUE))
##
##   AIC BIC logLik
##   506 538   -249
##
## Coefficients:
##
## Regime 1
## ---------
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)      -5.991      0.686   -8.73   <2e-16 ***
## Market_excess(S)     0.018      0.001   18.00   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8
## Multiple R-squared: 0.955
##
## Standardized Residuals:
##      Min       Q1      Med       Q3      Max
## -1.8e+00  2.7e-16  5.6e-10  7.0e-05  1.7e+00
##
## Regime 2
## ---------
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)       5.721      1.337    4.28  1.9e-05 ***
## Market_excess(S)     0.014      0.001   14.00  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.9
## Multiple R-squared: 0.645
##
## Standardized Residuals:
##      Min       Q1      Med       Q3      Max
## -4.30315 -0.48469 -0.00012  0.44432  4.74320
##
## Transition probabilities:
##          Regime 1 Regime 2
## Regime 1    0.981  5.1e-06
## Regime 2    0.019  1.0e+00
```

Now we select the best lag structure for the msm:

First we need to calculate all the lagged version of the variable `GE_excess`.

Selecting the best lag:

```
##    Model AIC
## 1   lag1 507
## 2   lag2 502
## 3   lag3 500
## 4   lag4 500
## 5   lag5 501
## 6   lag6 496
## 7   lag7 488
## 8   lag8 488
## 9   lag9 495
## 10 lag10 556
```

The best model is the one with 8 lags. We will now fit this model:

```
bestModel <- lm(y ~ X + lag1 + lag2 + lag3+ lag4+ lag5+ lag6+ lag7+ lag8,
data = matrix)
msm1 = msmFit(bestModel, k = 2, sw = rep(TRUE, 11))
summary(msm1)
```

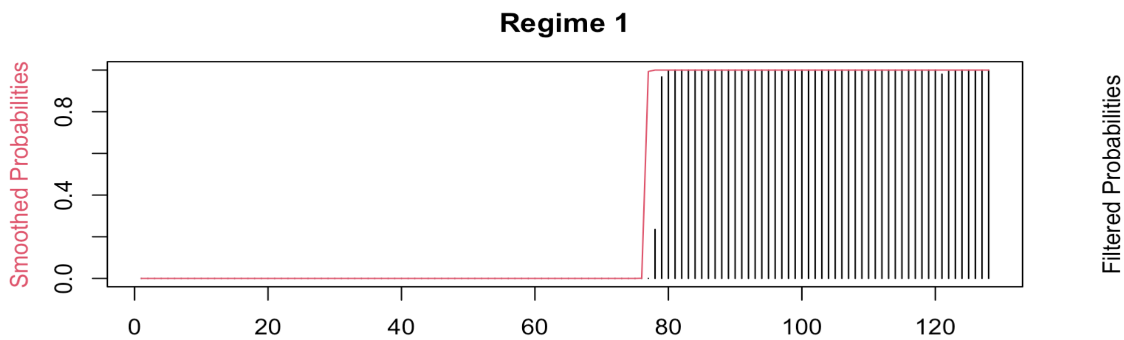
```
## Markov Switching Model
##
## Call: msmFit(object = bestModel, k = 2, sw = rep(TRUE, 11))
##
##   AIC BIC logLik
##   484 638   -222
##
## Coefficients:
##
## Regime 1
## ---------
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)   -6.039      0.803   -7.52  5.4e-14 ***
## X(S)              0.020      0.002   10.00  < 2e-16 ***
## lag1(S)          -0.003      0.003   -1.00     0.32
## lag2(S)           0.003      0.003    1.00     0.32
## lag3(S)          -0.003      0.003   -1.00     0.32
## lag4(S)           0.002      0.003    0.67     0.50
## lag5(S)           0.002      0.003    0.67     0.50
## lag6(S)          -0.001      0.002   -0.50     0.62
## lag7(S)          -0.003      0.003   -1.00     0.32
## lag8(S)           0.002      0.002    1.00     0.32
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

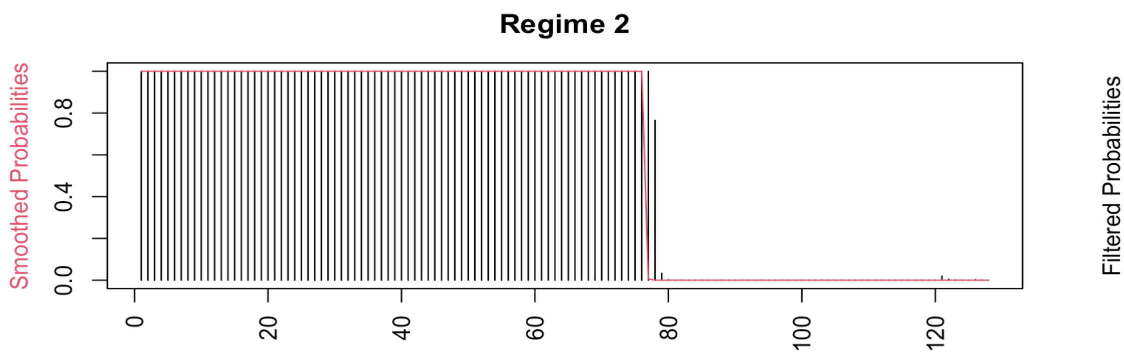
```
## Residual standard error: 0.76
## Multiple R-squared: 0.959
##
## Standardized Residuals:
##      Min       Q1      Med       Q3      Max
## -1.5e+00  4.8e-18  1.2e-10  2.5e-05  1.9e+00
##
## Regime 2
## ---------
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)    6.963      1.370    5.08  3.7e-07 ***
## X(S)              0.018      0.004    4.50  6.8e-06 ***
## lag1(S)          -0.001      0.005   -0.20     0.84
## lag2(S)          -0.004      0.008   -0.50     0.62
## lag3(S)           0.002      0.010    0.20     0.84
## lag4(S)          -0.001      0.013   -0.08     0.94
## lag5(S)           0.007      0.018    0.39     0.70
## lag6(S)           0.001      0.021    0.05     0.96
## lag7(S)          -0.008      0.010   -0.80     0.42
## lag8(S)          -0.003      0.005   -0.60     0.55
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.7
## Multiple R-squared: 0.736
##
## Standardized Residuals:
##      Min       Q1      Med       Q3      Max
## -2.9e+00 -4.0e-01 -3.6e-05  1.7e-01  4.9e+00
##
## Transition probabilities:
##          Regime 1 Regime 2
## Regime 1    0.981  1.5e-06
## Regime 2    0.019  1.0e+00
```

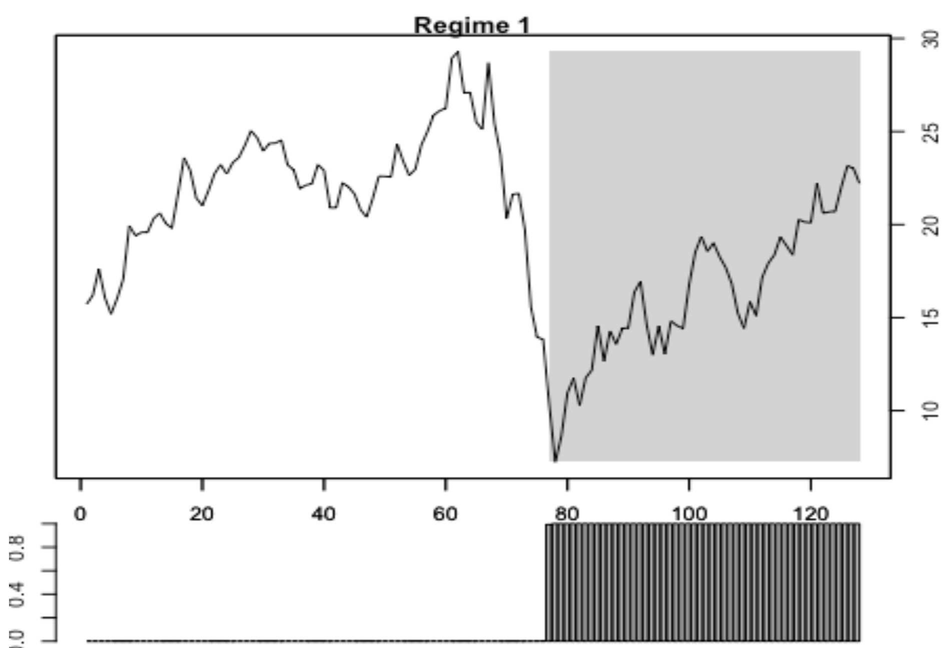
The transition probabilities describe the likelihood that the current regime stays the same or changes (i.e the probability that the regime transitions to another regime). Looking at the transition probabilities of this model, we notice that we have two distinct absorbent regimes, which means that, once you fall into one of the two, there is a high probability that you will remain in that specific regime.

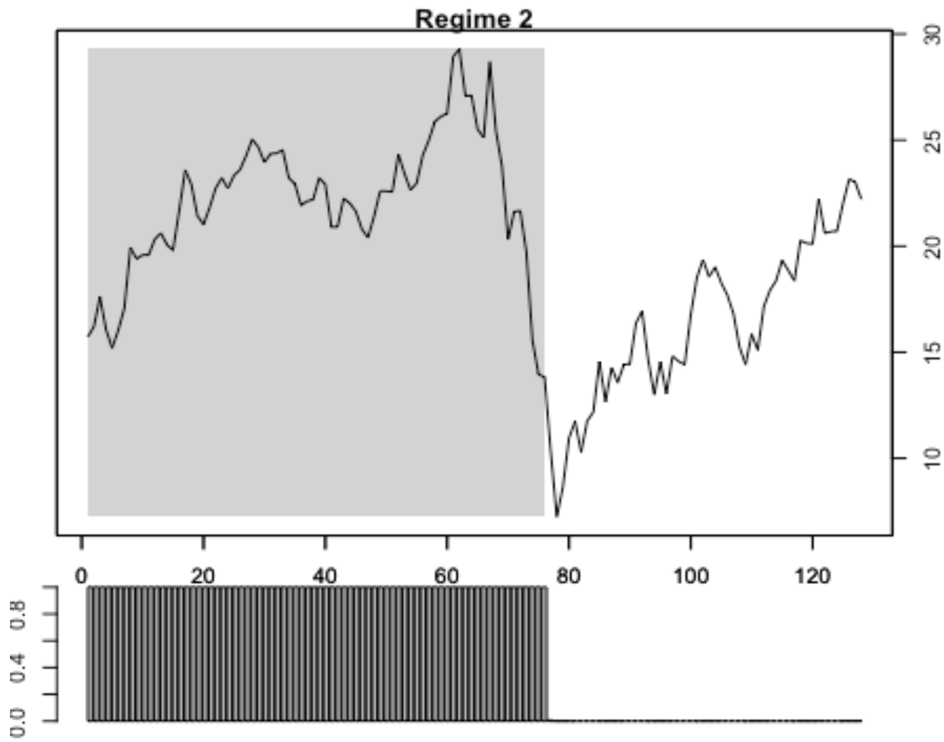
c.

Plot graphs of the posterior probabilities of each regime.









d.

Comment your findings.

In regime 1 there is a clear positive trend, while in regime 2 we observe higher values and a changing trend (positive at the beginning and negative in a second time).

2

We wish to apply a STAR model to the excess return on assets, the transition variable being the market excess return delayed by one or more periods (to be determined). After applying the appropriate tests, estimate an ESTAR or LSTAR model. Make a graphical representation of the transition function. Comment on your results.

First we calculate the star model:

```
## Using only first 134 elements of thVar
## Testing linearity...   p-Value =  0.81
## The series is linear. Using linear model instead.

##
## Non linear autoregressive model
##
## AR model
## Coefficients:
## const phi.1 phi.2
##   1.2   1.0  -0.1
```

```
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -3.96  -0.79   0.12   0.92   3.97
##
## Fit:
## residuals variance = 1.939,  AIC = 96, MAPE = 6.163%
##
## Coefficient(s):
##        Estimate  Std. Error  t value Pr(>|t|)
## const    1.2018      0.5656     2.12    0.035 *
## phi.1    1.0378      0.0861    12.06   <2e-16 ***
## phi.2   -0.0995      0.0860    -1.16    0.249
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This produces some unclear results: in fact, if we do a linearity test, the output is:

```
##       ** Teraesvirta's neural network test  **
##       Null hypothesis: Linearity in "mean"
##       X-squared =  6.7  df =  2  p-value =  0.035
##
##       ** White neural network test  **
##       Null hypothesis: Linearity in "mean"
##       X-squared =  6.1  df =  2  p-value =  0.048
##
##       ** Keenan's one-degree test for nonlinearity  **
##       Null hypothesis: The time series follows some AR process
##       F-stat =  4.8  p-value =  0.03
##
##       ** McLeod-Li test  **
##       Null hypothesis: The time series follows some ARIMA process
##       Maximum p-value =  0.39
##
##       ** Tsay's Test for nonlinearity **
##       Null hypothesis: The time series follows some AR process
##       F-stat =  2.5  p-value =  0.00063
##
##       ** Likelihood ratio test for threshold nonlinearity **
##       Null hypothesis: The time series follows some AR process
##       Alternativce hypothesis: The time series follows some TAR process
##       X-squared =  13  p-value =  0.33

## $Terasvirta
##
##  Teraesvirta Neural Network Test
##
## data:  ts(time.series)
## X-squared = 7, df = 2, p-value = 0.04
##
```

```
##
## $White
##
##  White Neural Network Test
##
## data:  ts(time.series)
## X-squared = 6, df = 2, p-value = 0.05
##
##
## $Keenan
## $Keenan$test.stat
## [1] 4.8
##
## $Keenan$df1
## [1] 1
##
## $Keenan$df2
## [1] 118
##
## $Keenan$p.value
## [1] 0.03
##
## $Keenan$order
## [1] 7
##
##
## $McLeodLi
## $McLeodLi$p.values
##  [1] 1.7e-01 3.9e-01 6.4e-03 1.4e-03 2.4e-03 3.0e-03 3.0e-05 3.7e-05
1.9e-05
## [10] 1.7e-05 2.6e-05 4.7e-05 8.9e-05 1.4e-04 2.3e-04 2.7e-04 4.6e-04
7.1e-04
## [19] 1.1e-03 1.5e-03 2.2e-03
##
##
## $Tsay
## $Tsay$test.stat
## [1] 2.5
##
## $Tsay$p.value
## [1] 0.00063
##
## $Tsay$order
## [1] 7
##
##
## $TarTest
## $TarTest$percentiles
## [1] 24 76
##
```

```
## $TarTest$test.statistic
## [1] 13
##
## $TarTest$p.value
## [1] 0.33
```

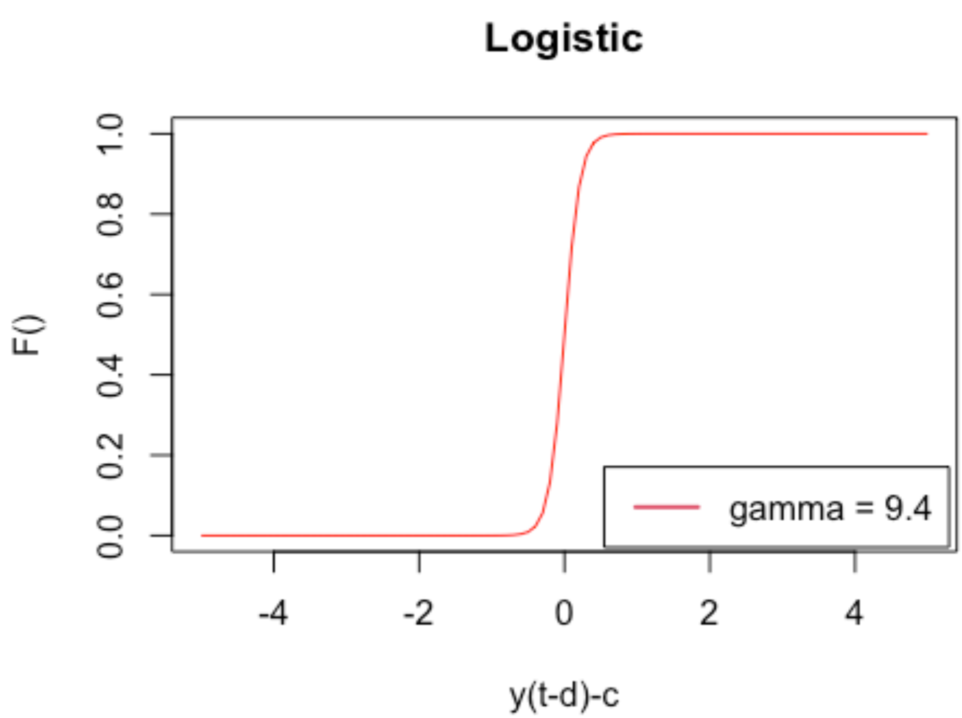
As we can see, the p-value of the Teraesvirta's neural network test is very low (0.035) and so we reject the null hypothesis of linearity.

Because the p-value is < 0.05, we choose a LSTAR model.

```
## Performing grid search for starting values...
## Starting values fixed: gamma =  8.6 , th =  25 ; SSE =  191
## Optimization algorithm converged
## Optimized values fixed for regime 2  : gamma =  9.4 , th =  25 ; SSE =
190

summary(lstar_mod)

##
## Non linear autoregressive model
##
## LSTAR model
## Coefficients:
## Low regime:
## const.L  phiL.1  phiL.2  phiL.3  phiL.4  phiL.5
##    0.72    1.08   -0.21    0.10    0.16   -0.16
##
## High regime:
##  const.H    phiH.1    phiH.2    phiH.3    phiH.4    phiH.5
## -26.6111  -0.6687   0.3965  -0.0032   0.1572   1.0473
##
## Smoothing parameter: gamma = 9.372
##
## Threshold
## Variable: Z(t) = + (0) X(t) + (0) X(t-1)+ (0) X(t-2)+ (0) X(t-3)+ (0)
X(t-4)+ (1) X(t-5)
##
## Value: 25.09
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6385 -0.6940  0.0611  0.8324  2.8710
##
## Fit:
## residuals variance = 1.396,  AIC = 73, MAPE = 5.467%
##
## Coefficient(s):
##          Estimate  Std. Error  t value Pr(>|z|)
## const.L   0.71587     0.54874     1.30   0.1920
```

```
## phiL.1    1.07972     0.08462    12.76   < 2e-16 ***
## phiL.2   -0.20823     0.12598    -1.65    0.0983 .
## phiL.3    0.10039     0.12466     0.81    0.4206
## phiL.4    0.16268     0.12398     1.31    0.1895
## phiL.5   -0.16354     0.08565    -1.91    0.0562 .
## const.H -26.61113     6.80191    -3.91   9.1e-05 ***
## phiH.1   -0.66871     0.23049    -2.90    0.0037 **
## phiH.2    0.39649     0.29301     1.35    0.1760
## phiH.3   -0.00322     0.26678    -0.01    0.9904
## phiH.4    0.15718     0.25809     0.61    0.5425
## phiH.5    1.04726     0.33531     3.12    0.0018 **
## gamma     9.37161    10.95257     0.86    0.3922
## th       25.08547     0.12261   204.60   < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Non-linearity test of full-order LSTAR model against full-order AR model
##  F = 2.06 ; p-value = 0.0634
##
## Threshold
## Variable: Z(t) = + (0) X(t) + (0) X(t-1)+ (0) X(t-2)+ (0) X(t-3)+ (0)
X(t-4)+ (1) X(t-5)
```

Finally, we plot the transition function for the lstar model:



Exercise 3

Loading the data

```
## New names:
## • `Date` -> `Date...1`
## • `Date` -> `Date...3`
## • `Date` -> `Date...5`
## • `Date` -> `Date...7`
## • `Date` -> `Date...9`
## • `Date` -> `Date...11`
## • `Date` -> `Date...13`
## • `Date` -> `Date...15`
## • `Date` -> `Date...17`
## • `Date` -> `Date...19`
## • `Date` -> `Date...21`
## • `Date` -> `Date...23`
## • `Date` -> `Date...25`
## • `Date` -> `Date...27`
## • `Date` -> `Date...29`
## • `Date` -> `Date...31`
## • `Date` -> `Date...33`
## • `Date` -> `Date...35`
## • `Date` -> `Date...37`
## • `Date` -> `Date...39`
## • `Date` -> `Date...41`
## • `Date` -> `Date...43`
```

3

Estimate the above equation using a Markov-switching model with 2 regimes

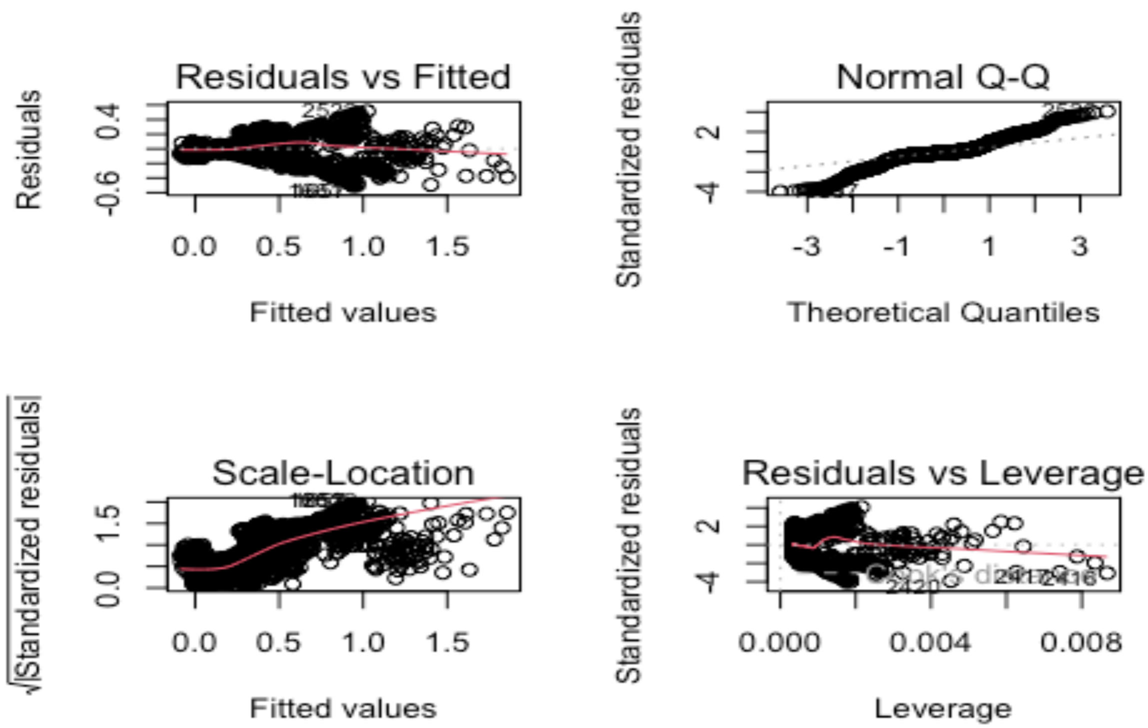
a.

Estimate the linear model and use tests on residuals to show that the dynamics is probably nonlinear.

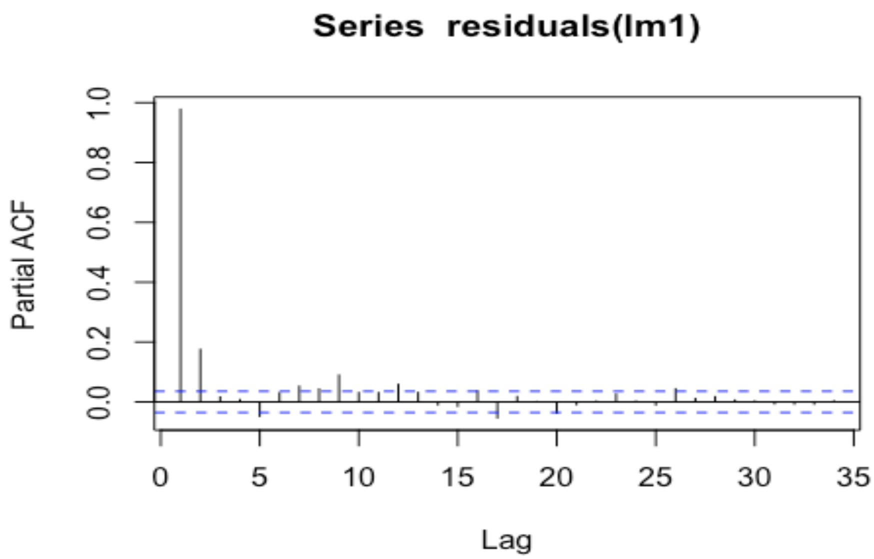
```
## Call:
## lm(formula = Spread_Fr ~ Spread_Be, data = Spread_FI)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5083 -0.0409 -0.0138  0.0386  0.5149
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.02982    0.00301     9.9   <2e-16 ***
## Spread_Be    0.51040    0.00378   135.1   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.13 on 3051 degrees of freedom
## Multiple R-squared:  0.857,  Adjusted R-squared:  0.857
## F-statistic: 1.82e+04 on 1 and 3051 DF,  p-value: <2e-16
```

The R-squared is not so good: let's do some diagnostic plots



From the Q-Q plot it's clear that we have a linearity problem, especially with the tails. This graph suggests that there may be some heteroskedasticity in our data.



When considering the results of all these tests, they suggest that the assumption of linearity in the average of the residuals does not hold. This indicates the possible existence of nonlinear patterns within the time series data. Consequently, it appears that a simple linear model may not adequately explain the relationship present in the data, implying the necessity of employing a more sophisticated modeling approach.

b.

Fit a Markov-switching model (select the optimal lag structure using tests based on information criteria).

We start with a simple model without lags:

```
## Markov Switching Model
##
## Call: msmFit(object = lm1, k = 2, sw = c(TRUE, TRUE, TRUE), control =
list(parallel = FALSE))
##
##       AIC    BIC logLik
##    -8023 -7967   4016
##
## Coefficients:
##
## Regime 1
## ---------
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)     0.009      0.001       9   <2e-16 ***
## Spread_Be(S)       0.702      0.003     234   <2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05
## Multiple R-squared: 0.975
##
## Standardized Residuals:
##      Min       Q1      Med       Q3      Max
## -0.15410 -0.02341 -0.00044  0.00782  0.17696
##
## Regime 2
## ---------
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)   -0.022      0.010    -2.2    0.028 *
## Spread_Be(S)      0.454      0.007    64.9   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.12
## Multiple R-squared: 0.856
##
## Standardized Residuals:
##      Min       Q1      Med       Q3      Max
## -0.35707  0.00012  0.00032  0.00116  0.53420
##
## Transition probabilities:
##          Regime 1 Regime 2
## Regime 1   0.9966   0.0079
## Regime 2   0.0034   0.9921
```

Again we have an absorbing state, which is not the best for us as it can lead to the model becoming trapped in this state, thereby limiting its ability to accurately capture transitions between different regimes or states over time.

We created the lags and now we would like to select the model with the smallest AIC

```
##    Model   AIC
## 1   lag1 -8375
## 2   lag2 -8671
## 3   lag3 -8682
## 4   lag4 -8694
## 5   lag5 -8697
## 6   lag6 -8472
## 7   lag7 -8695
## 8   lag8 -8696
## 9   lag9 -8697
## 10 lag10 -8703
```

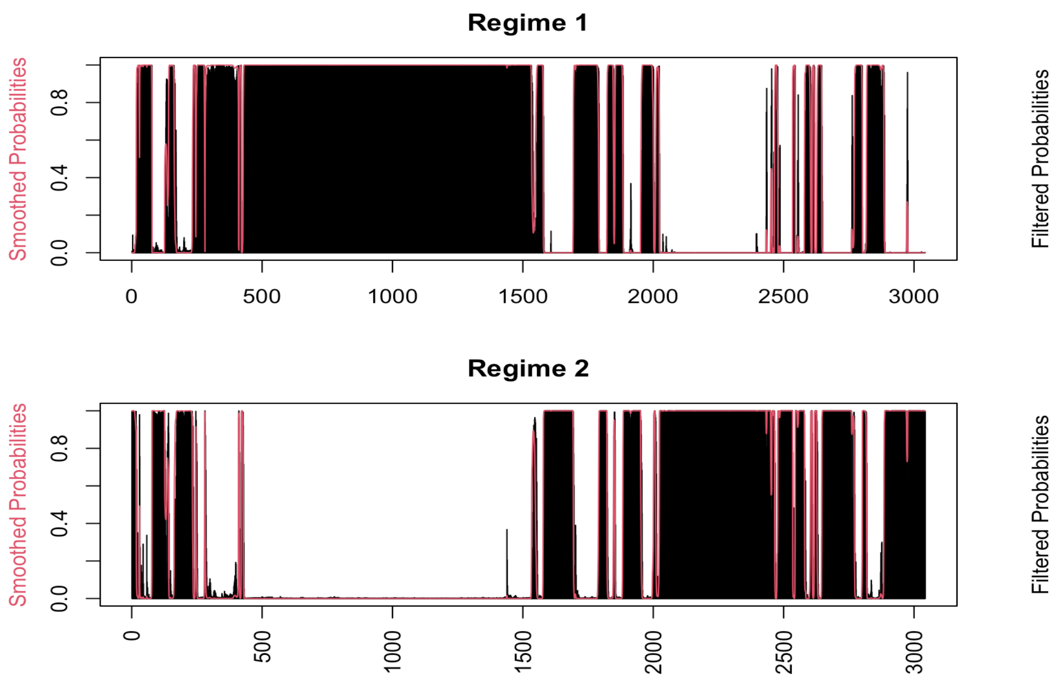
We will choose the model with 5 lags, as it has one of the smallest AIC with the smallest computational cost (and time).

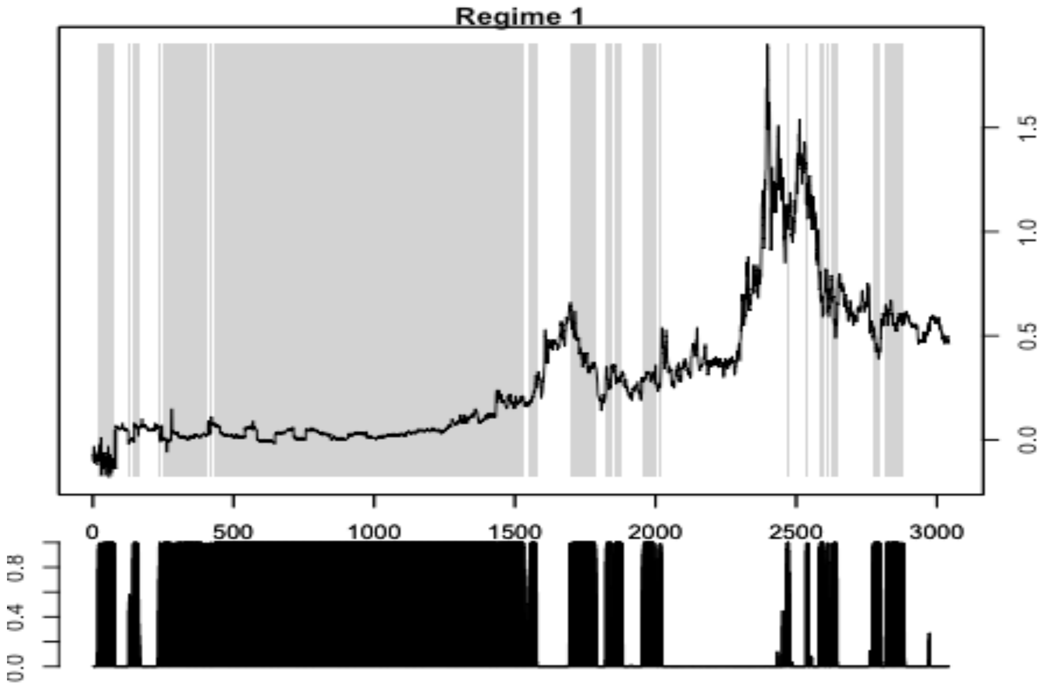
```
## Markov Switching Model
##
## Call: msmFit(object = BestModel5, k = 2, sw = rep(TRUE, 8), control =
list(parallel = FALSE))
##
##     AIC   BIC logLik
##   -8473 -8276   4250
##
## Coefficients:
##
## Regime 1
## ---------
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)    0.004      0.001    4.00  6.3e-05 ***
## X(S)              0.272      0.014   19.43  < 2e-16 ***
## lag1(S)           0.116      0.012    9.67  < 2e-16 ***
## lag2(S)           0.048      0.039    1.23   0.2183
## lag3(S)           0.052      0.022    2.36   0.0181 *
## lag4(S)           0.065      0.024    2.71   0.0068 **
## lag5(S)           0.069      0.015    4.60  4.2e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.025
## Multiple R-squared: 0.987
##
## Standardized Residuals:
##       Min       Q1      Med       Q3      Max
## -8.4e-02 -7.2e-03 -1.0e-25  4.9e-03  9.1e-02
##
## Regime 2
## ---------
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)(S)    0.076      0.009    8.44   <2e-16 ***
## X(S)              0.408      0.131    3.12   0.0018 **
## lag1(S)          -0.007      0.031   -0.23   0.8212
## lag2(S)          -0.031      0.096   -0.32   0.7467
## lag3(S)          -0.034      0.229   -0.15   0.8823
## lag4(S)          -0.016      0.071   -0.22   0.8220
## lag5(S)           0.143      0.100    1.43   0.1527
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.19
## Multiple R-squared: 0.723
##
## Standardized Residuals:
##       Min       Q1      Med       Q3      Max
## -0.48371 -0.01042 -0.00045  0.00014  0.56707
```

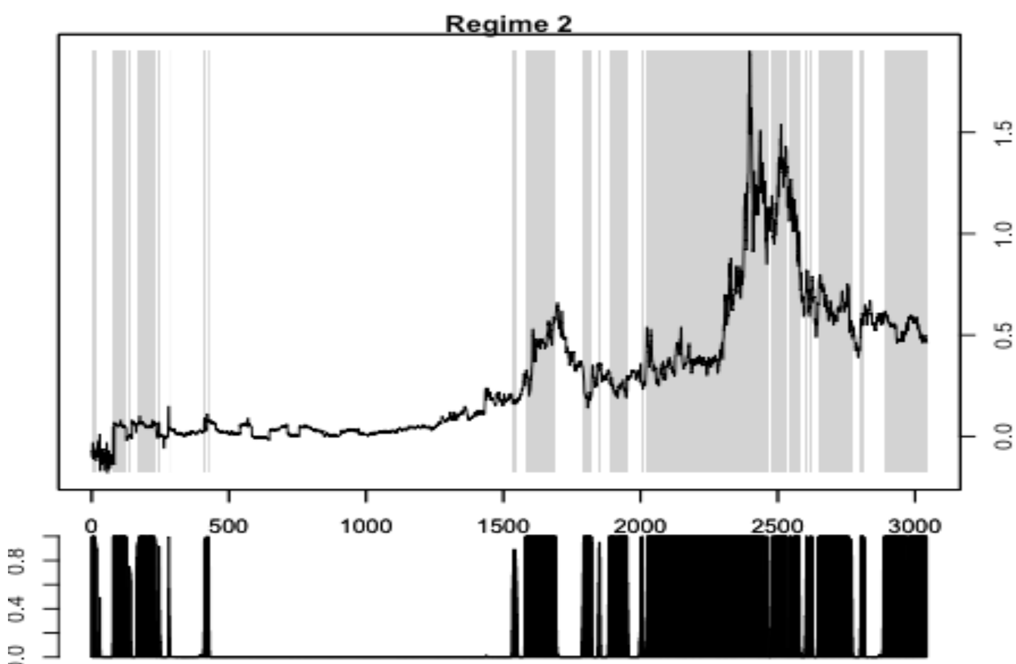
```
##
## Transition probabilities:
##          Regime 1 Regime 2
## Regime 1    0.988    0.018
## Regime 2    0.012    0.982
```

c.

Plot graphs of the posterior probabilities of each regime




Regime 1
0
500
1000
1500
2000
2500
3000
0.0
0.5
1.0
1.5
0.0
0.4
0.8


Regime 2
0
500
1000
1500
2000
2500
3000
0.0
0.5
1.0
1.5
0.0
0.4
0.8

d.

Comment your findings.

Regime 2 is characterized by growth, whereas regime 1 appears to be relatively stable or possibly even declining. Given this observation, we think it might be useful to try introducing a third regime into the model, to provide a more meaningful explanation of the varying phases observed within the data. This additional regime could help to better capture the complexity and dynamics of the underlying processes driving the observed patterns.

4

We wish to apply a STAR model to the excess return on assets, the transition variable being the market excess return delayed by one or more periods (to be determined). After applying the appropriate tests, estimate an ESTAR or LSTAR model. Make a graphical representation of the transition function. Comment on your results.

First we need to calculate the excess return, as we did in the previous points.

Then we calculate the star model:

```
## Using only first 3051 elements of thVar
## Testing linearity...   p-Value =  7.7e-38
## The series is nonlinear. Incremental building procedure:
## Building a 2 regime STAR.
## Using only first 3051 elements of thVar
## Performing grid search for starting values...
## Starting values fixed: gamma =  72 , th =  0.034 ; SSE =  2.5
## Optimization algorithm converged
## Optimized values fixed for regime 2  : gamma =  72 , th =  0.032 ; SSE =
2.5
##
## Testing for addition of regime 3.
##   Estimating gradient matrix...
##   Done. Computing the test statistic...
##   Done. Regime 3 is needed (p-Value =  7.8e-33 ).
## Adding regime  3 .
## Fixing good starting values for regime  3 ...
## Reordering regimes...
## Estimating parameters of regime 3 ...
## Optimized values fixed for regime  3 : gamma =  43 , th =  1.7
## Optimization algorithm converged
## Optimized linear values:
## 0.00085 0.97 0.03
## 525 -761 468
## -527 753 -460
## Ok.
##  Testing for addition of regime  4 .
##   Estimating gradient matrix...
```

```
##   Computing the test statistic...
## Regime  4  is needed (p-Value =  6.9e-05 ).
## Adding regime  4 .
## Fixing good starting values for regime  4 ...
## Reordering regimes...
## Estimating parameters of regime 4 ...
## Optimized values fixed for regime  4 : gamma =  41 , th =  1.8
## Optimization algorithm converged
## Optimized linear values:
## 0.001 0.94 0.052
## 0.38 -11 11
## -0.36 11 -11
## -1.2 -6.8 7
## Ok.
##  Testing for addition of regime  5 .
##   Estimating gradient matrix...
##   Computing the test statistic...
## Regime  5  is needed (p-Value =  3.4e-15 ).
## Adding regime  5 .
## Fixing good starting values for regime  5 ...
## Reordering regimes...
## Estimating parameters of regime 5 ...
## Optimized values fixed for regime  5 : gamma =  41 , th =  1.8
## Optimization algorithm converged
## Optimized linear values:
## 0.001 0.95 0.047
## -0.37 -7.6 8
## 4.2 13 -19
## -3.9 -5.5 12
## -1.1 -6.7 6.8
## Ok.
##  Testing for addition of regime  6 .
##   Estimating gradient matrix...
##   Computing the test statistic...
## Regime  6  is needed (p-Value =  1.9e-13 ).
## Adding regime  6 .
## Fixing good starting values for regime  6 ...
## Reordering regimes...
## Estimating parameters of regime 6 ...
## Optimized values fixed for regime  6 : gamma =  40 , th =  1.9
## Optimization algorithm converged
## Optimized linear values:
## 0.001 0.95 0.045
## -0.91 -8.7 10
## 6.3 14 -24
## -5.5 -5.8 14
## 0.05 -0.4 0.28
## -1.7 -9.5 9.8
## Ok.
##  Testing for addition of regime  7 .
```

```
##   Estimating gradient matrix...
##   Computing the test statistic...
## Regime  7  is needed (p-Value =  1.4e-06 ).
## Adding regime  7 .
## Fixing good starting values for regime  7 ...
## Reordering regimes...
## Estimating parameters of regime 7 ...
## Optimized values fixed for regime  7 : gamma =  43 , th =  1.6
## *** Convergence problem. Code:  1
## Optimized linear values:
## 0.001 0.95 0.042
## -43 951 -709
## 32 193 -243
## -16 -68 93
## -17 -125 150
## 0.069 0.12 -0.27
## 42 -955 714
## Ok.
##  Testing for addition of regime  8 .
##   Estimating gradient matrix...
##   Computing the test statistic...
## Regime  8  is needed (p-Value =  0.018 ).
## Adding regime  8 .
## Fixing good starting values for regime  8 ...
## Reordering regimes...
## Estimating parameters of regime 8 ...
## Optimized values fixed for regime  8 : gamma =  48 , th =  1.7
## Optimization algorithm converged
## Optimized linear values:
## 0.0011 0.95 0.046
## -1296 -9026 9873
## 45 269 -340
## -12 -45 65
## -33 -224 276
## 0.066 -0.15 -0.014
## 334 11418 -11555
## 962 -2395 1684
## Ok.
##  Testing for addition of regime  9 .
##   Estimating gradient matrix...
##   Computing the test statistic...
## Regime  9  is needed (p-Value =  0.047 ).
## Adding regime  9 .
## Fixing good starting values for regime  9 ...
## Reordering regimes...
## Estimating parameters of regime 9 ...
## Optimized values fixed for regime  9 : gamma =  45 , th =  1.8
## *** Convergence problem. Code:  1
## Optimized linear values:
## 0.0011 0.93 0.068
```

```
## 0.0041 0.08 -0.079
## 23 162 -198
## -11 -50 67
## -13 -112 131
## 0.042 0.12 -0.24
## 3.9 23 -28
## -3.7 -23 28
## -1.2 -5.8 6.1
## Ok.
##  Testing for addition of regime  10 .
##   Estimating gradient matrix...
##   Computing the test statistic...
## Regime  10  is needed (p-Value =  0.029 ).
## Adding regime  10 .
## Fixing good starting values for regime  10 ...
## Reordering regimes...
## Estimating parameters of regime 10 ...
## Optimized values fixed for regime  10 : gamma =  40 , th =  1.8
## *** Convergence problem. Code:  1
## Optimized linear values:
## 0.001 0.93 0.068
## 0.0041 0.07 -0.069
## -258 -831 1154
## 690 2250 -3114
## -291 -971 1334
## -142 -448 627
## 0.2 0.61 -0.92
## -150 -853 950
## 149 854 -951
## -0.052 -5.5 5.1
## Ok.
##  Testing for addition of regime  11 .
##   Estimating gradient matrix...
##   Computing the test statistic...
## Regime  11  is NOT accepted (p-Value =  0.091 ).
##
## Finished building a MRSTAR with  10  regimes
```

```
summary(star_mod)
```

```
##
## Non linear autoregressive model
##
## Multiple regime STAR model
##
## Regime  1 :
##     Linear parameters: 0, 0.93, 0.07
##
## Regime  2 :
##     Linear parameters: 0, 0.07, -0.07
```

```
##     Non-linear parameters:
## 76.38, 0.52
## Regime  3 :
##     Linear parameters: -257.8, -830.77, 1153.66
##     Non-linear parameters:
## 43.46, 1.13
## Regime  4 :
##     Linear parameters: 689.97, 2250.06, -3113.58
##     Non-linear parameters:
## 43.99, 1.14
## Regime  5 :
##     Linear parameters: -290.89, -971.49, 1334.24
##     Non-linear parameters:
## 48.1, 1.14
## Regime  6 :
##     Linear parameters: -141.51, -448.47, 626.7
##     Non-linear parameters:
## 40.16, 1.15
## Regime  7 :
##     Linear parameters: 0.2, 0.61, -0.92
##     Non-linear parameters:
## 45.05, 1.28
## Regime  8 :
##     Linear parameters: -150.05, -852.83, 950.41
##     Non-linear parameters:
## 47.3, 1.52
## Regime  9 :
##     Linear parameters: 149.43, 853.91, -951.11
##     Non-linear parameters:
## 45.5, 1.52
## Regime  10 :
##     Linear parameters: -0.05, -5.48, 5.13
##     Non-linear parameters:
## 40.01, 1.84
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18198 -0.00961 -0.00076  0.00878  0.20747
##
## Fit:
## residuals variance = 0.0006447,  AIC = -22334, MAPE = 12.06%
```

This produced a star model with 10 regimes. The next step is to test the residuals in order to choose between a LSTAR or ESTAR model

```
##       ** Teraesvirta's neural network test  **
##       Null hypothesis: Linearity in "mean"
##       X-squared =  3  df =  2  p-value =  0.23
##
##       ** White neural network test  **
##       Null hypothesis: Linearity in "mean"
##       X-squared =  2.9  df =  2  p-value =  0.23
##
##       ** Keenan's one-degree test for nonlinearity  **
##       Null hypothesis: The time series follows some AR process
##       F-stat =  2.5  p-value =  0.11
##
##       ** McLeod-Li test  **
##       Null hypothesis: The time series follows some ARIMA process
##       Maximum p-value =  0
##
##       ** Tsay's Test for nonlinearity **
##       Null hypothesis: The time series follows some AR process
##       F-stat =  6  p-value =  1.3e-208
##
##       ** Likelihood ratio test for threshold nonlinearity **
##       Null hypothesis: The time series follows some AR process
##       Alternativce hypothesis: The time series follows some TAR process
##       X-squared =  142  p-value =  1.5e-13

## $Terasvirta
##
##  Teraesvirta Neural Network Test
##
## data:  ts(time.series)
## X-squared = 3, df = 2, p-value = 0.2
##
##
## $White
##
##  White Neural Network Test
##
## data:  ts(time.series)
## X-squared = 3, df = 2, p-value = 0.2
##
##
## $Keenan
## $Keenan$test.stat
## [1] 2.5
##
## $Keenan$df1
## [1] 1
##
## $Keenan$df2
## [1] 2985
```

```
##
## $Keenan$p.value
## [1] 0.11
##
## $Keenan$order
## [1] 32
##
##
## $McLeodLi
## $McLeodLi$p.values
##  [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##
##
## $Tsay
## $Tsay$test.stat
## [1] 6
##
## $Tsay$p.value
## [1] 1.3e-208
##
## $Tsay$order
## [1] 32
##
##
## $TarTest
## $TarTest$percentiles
## [1] 25 75
##
## $TarTest$test.statistic
## [1] 142
##
## $TarTest$p.value
## [1] 1.5e-13
```

Based on the Terasvirta’s neural network test results, where the p-value is 0.23, we cannot reject the null hypothesis of linearity in the mean. This suggests that there is no significant evidence to support the presence of nonlinearity in the mean of the residuals.

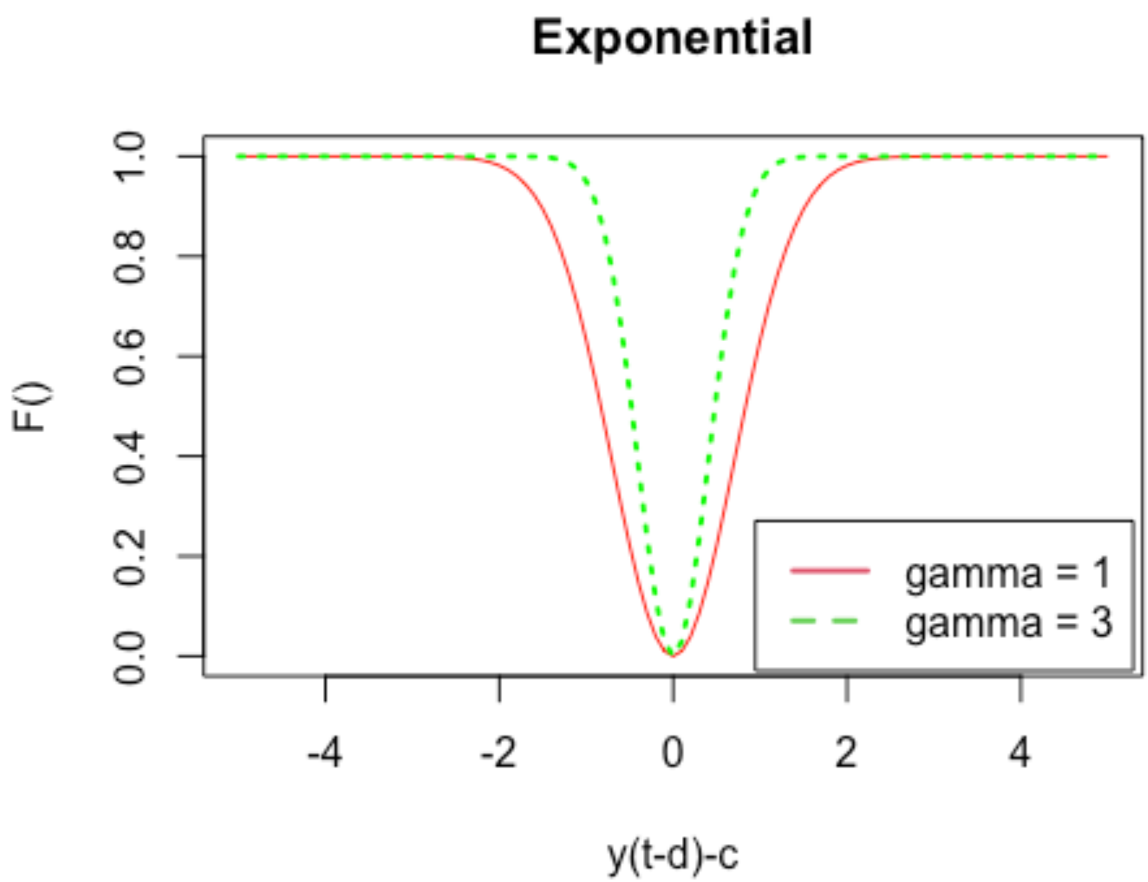
In this case, since the test does not provide evidence against linearity, it is more appropriate to consider an ESTAR (Smooth Transition Autoregressive) model rather than an LSTAR (Threshold Autoregressive) model.

An ESTAR model is suitable when the data suggests a smooth transition between different regimes, while an LSTAR model is preferred when there is evidence of threshold nonlinearity.

Let’s estimate now the ESTAR model:

```
#estar_mod <- ...
#summary(estar_mod)
```

Finally, we plot the transition function for the estar model:



5

Use a bivariate GARCH model to study the correlation between excess return of your assets and excess return on the market. Comment on your results.

Univariate Garch Series

```
##
## *---------------------------------*
## *          DCC GARCH Fit          *
## *---------------------------------*
##
## Distribution         :  mvnorm
## Model                :  DCC(1,1)
## No. Parameters       :  11
## [VAR GARCH DCC UncQ] : [0+8+2+1]
```

```
## No. Series          :  2
## No. Obs.            :  3053
## Log-Likelihood      :  7773
## Av.Log-Likelihood   :  2.5
##
## Optimal Parameters
## -----------------------------------
##                                 Estimate  Std. Error  t value Pr(>|t|)
## [Excess_return_France].mu       0.030697    0.000658  46.6778  0.0e+00
## [Excess_return_France].omega    0.000006    0.000001  10.2141  0.0e+00
## [Excess_return_France].alpha1   0.279880    0.044123   6.3432  0.0e+00
## [Excess_return_France].beta1    0.719120    0.045727  15.7263  0.0e+00
## [Excess_return_Market].mu       0.018758    0.001109  16.9210  0.0e+00
## [Excess_return_Market].omega    0.000070    0.000014   5.0125  1.0e-06
## [Excess_return_Market].alpha1   0.424399    0.039817  10.6587  0.0e+00
## [Excess_return_Market].beta1    0.574601    0.039892  14.4038  0.0e+00
## [Joint]dcca1                    0.124883    0.028388   4.3991  1.1e-05
## [Joint]dccb1                    0.874085    0.028765  30.3872  0.0e+00
##
## Information Criteria
## ---------------------
##
## Akaike       -5.0850
## Bayes        -5.0633
## Shibata      -5.0850
## Hannan-Quinn -5.0772
##
##
## Elapsed time : 1.7
```

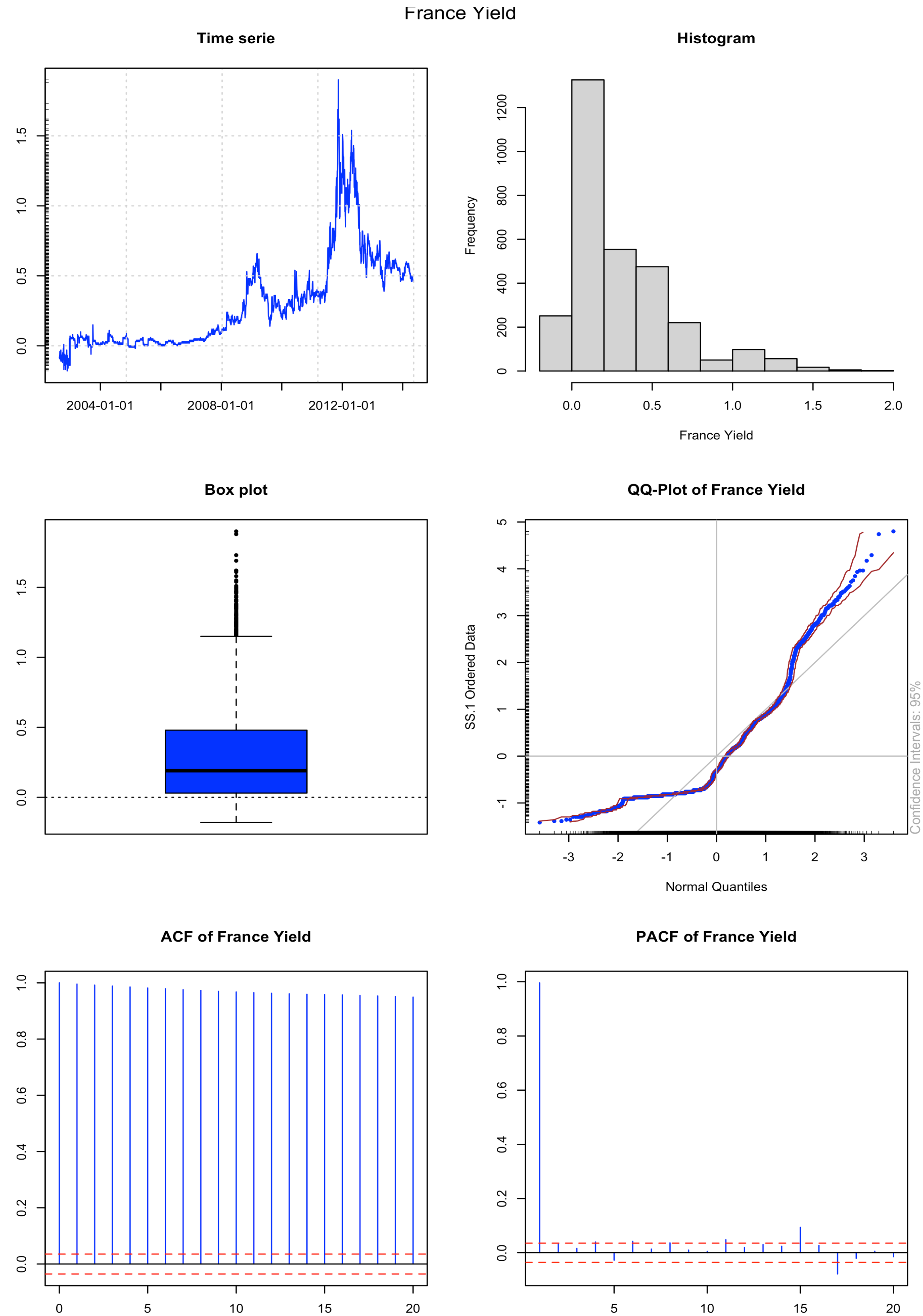
All the parameters are significant, in particular beta1 (representative of the Garch model)

Exercise 4

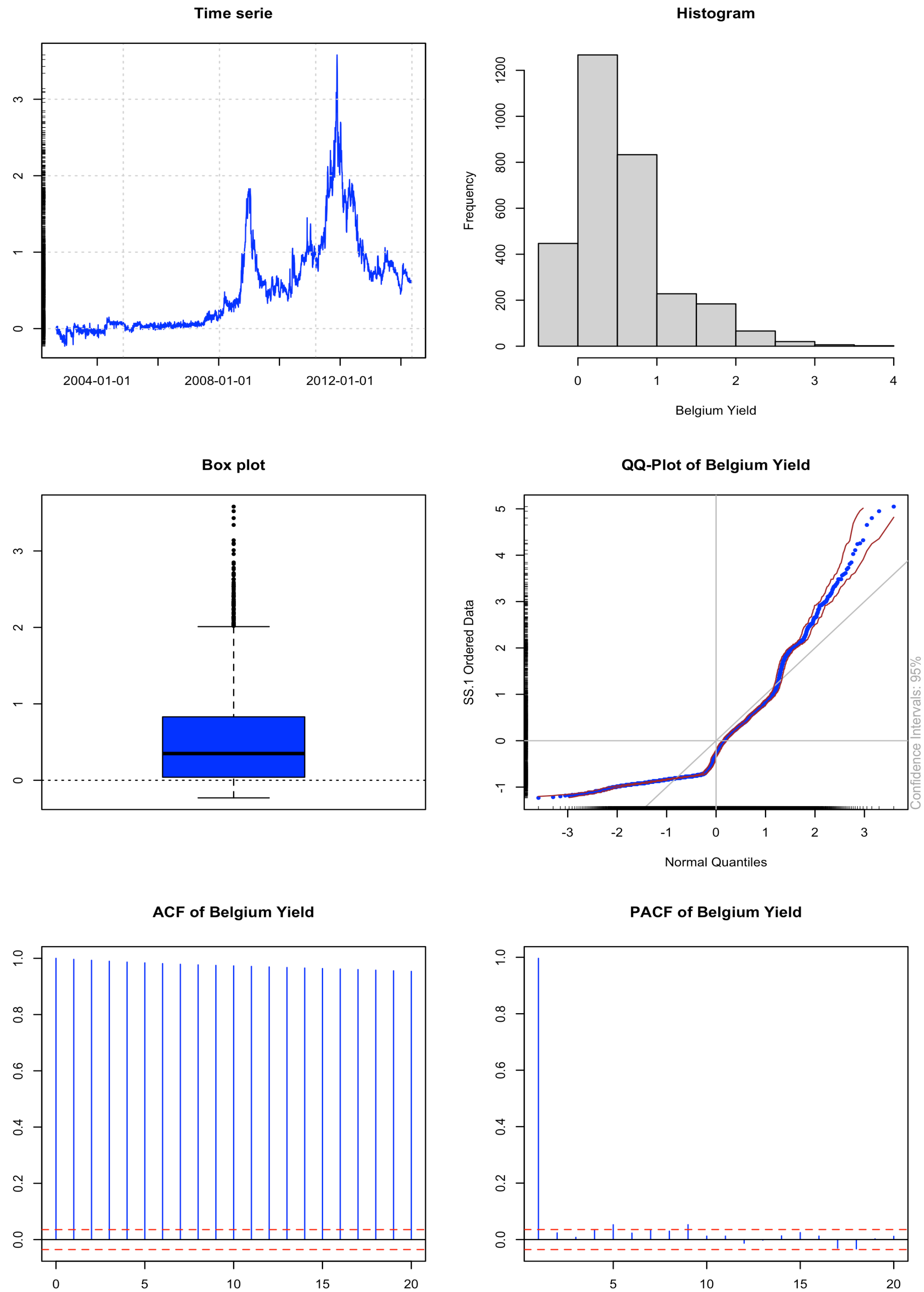
Now, we wish to model the correlation between the spreads of countries i and j using copulas.

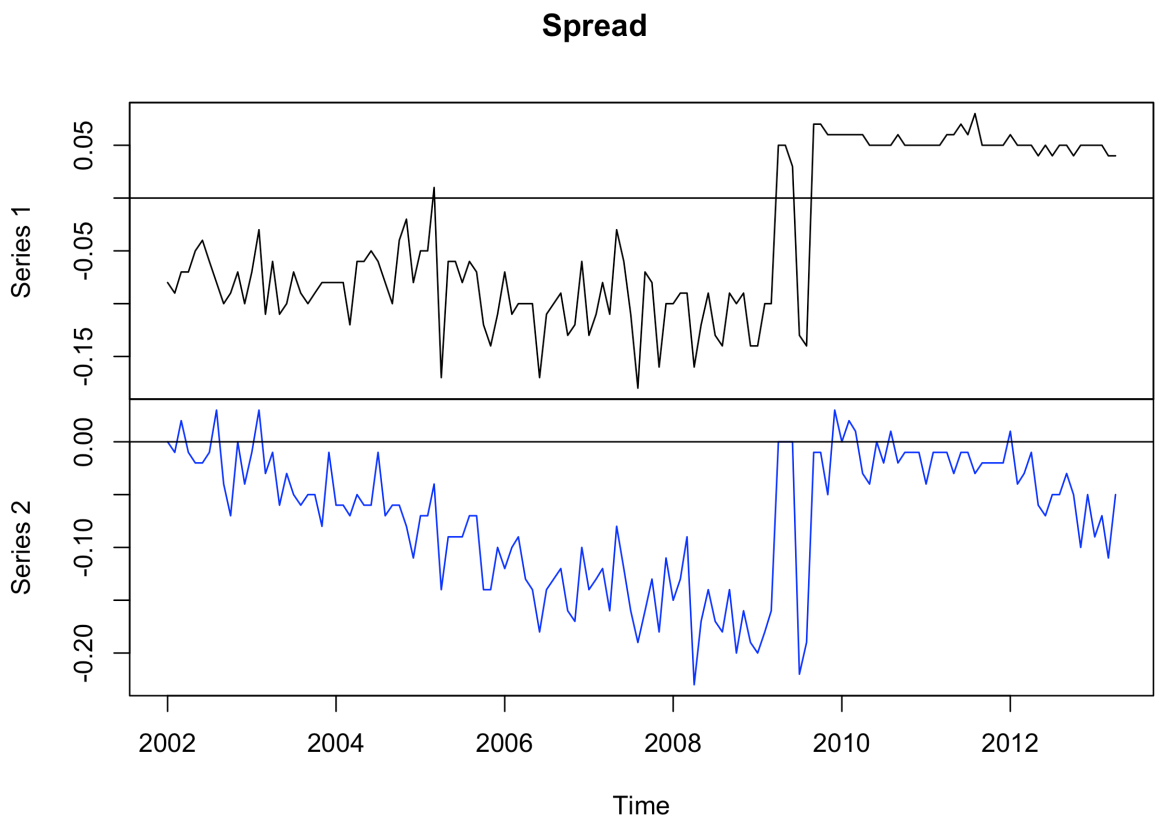
a.

Do a preliminary analysis of the spreads based on basics statistics, graphs of the distributions, QQ-plots, box-plot, bivariate scatterplot, graphs of time series, autocorrelation function, pp-plot.

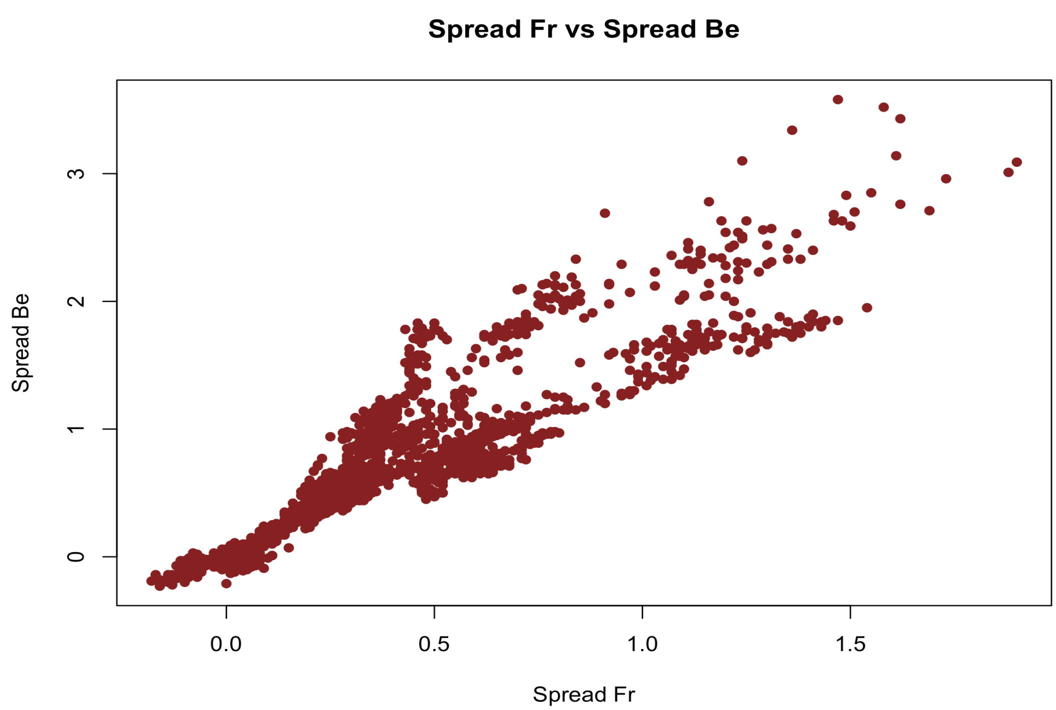

France Yield
Time serie
2004-01-01
2008-01-01
2012-01-01
Histogram
Frequency
France Yield
Box plot
QQ-Plot of France Yield
SS.1 Ordered Data
Normal Quantiles
Confidence Intervals: 95%
ACF of France Yield
PACF of France Yield

Belgium Yield





Spread
Series 1
Series 2
0.05
-0.05
-0.15
0.00
-0.10
-0.20
2002
2004
2006
2008
2010
2012
Time

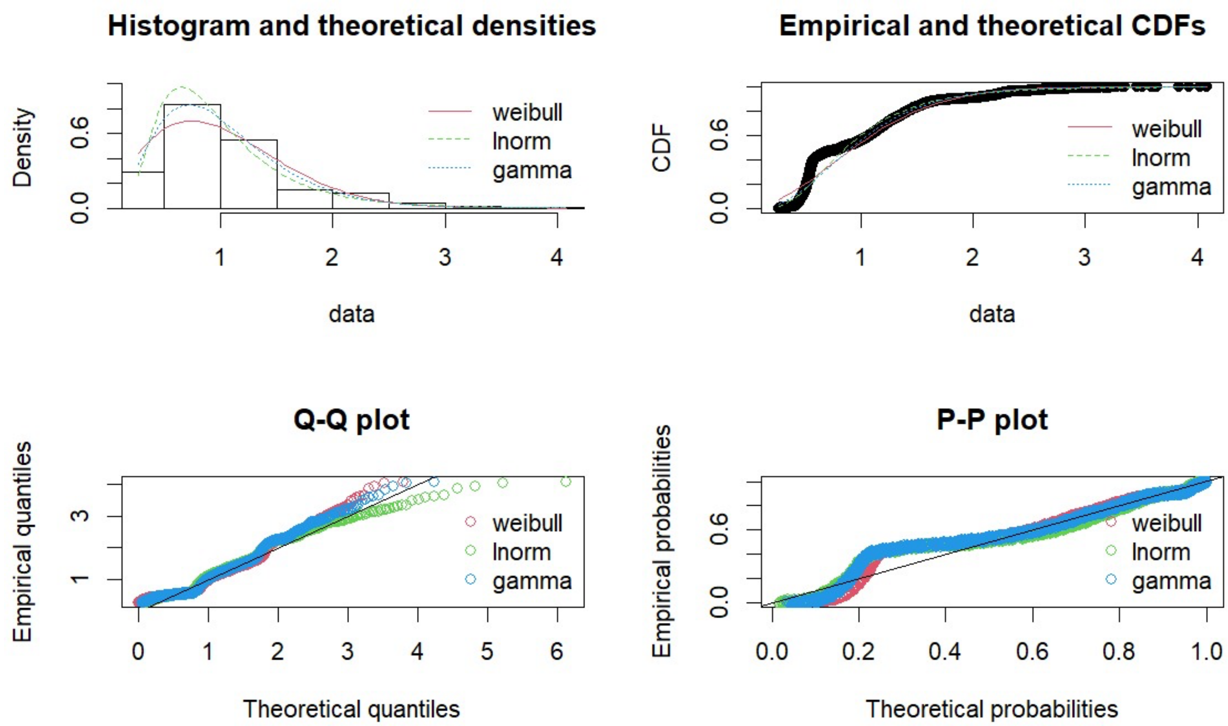


Spread Fr vs Spread Be
Spread Be
Spread Fr
0
1
2
3
0.0
0.5
1.0
1.5

b.

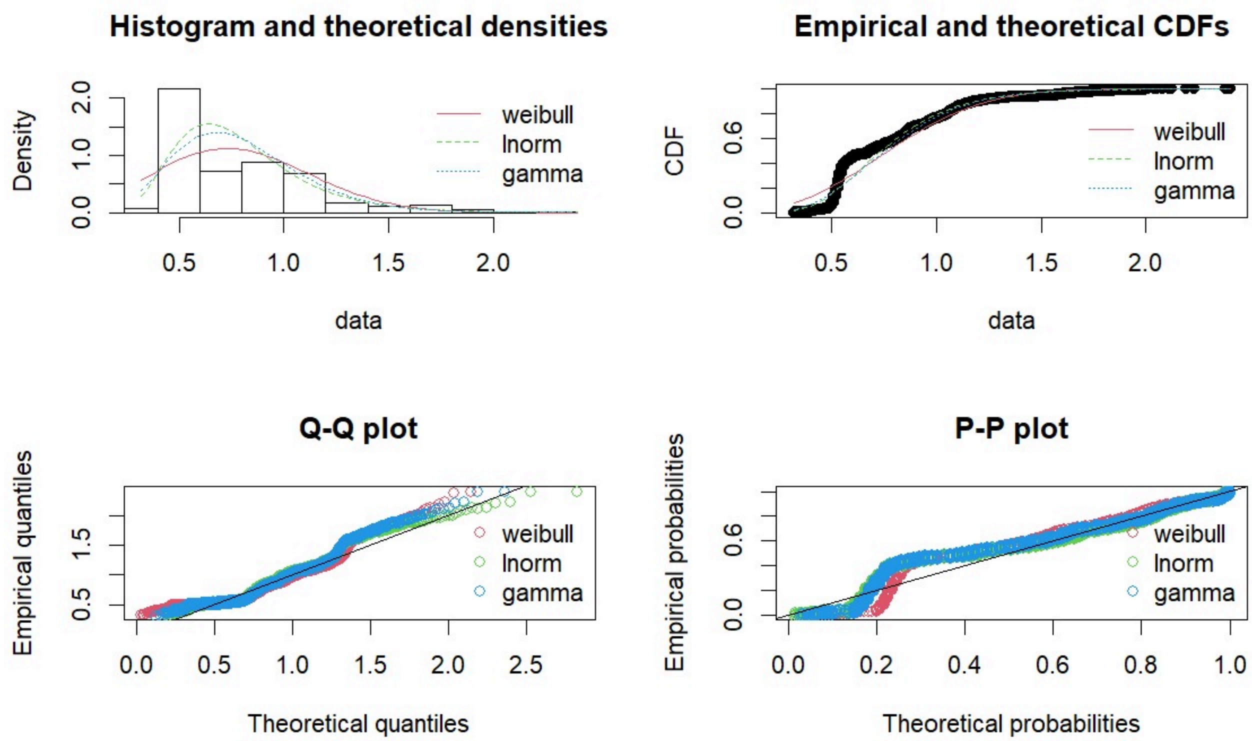
Fit one of the following extreme distribution to the spreads: Weibull, log-norma, Gamma.

France spread:



The lognormal distribution is the one that fits better the data of the french spread.

Belgium spread:



For the belgium spread the gamma distribution is the one that fits better the data.

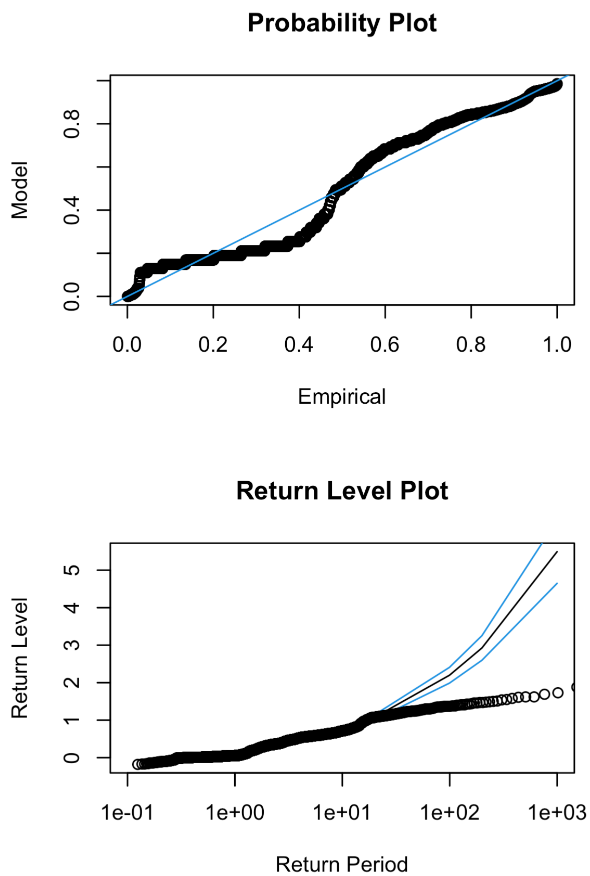
c.

Fit a GEV distribution to the excess return series and plot the QQ-plot, density and CDF.

For France:

```
FR_GEV <- gev.fit(Spread_FI$Spread_Fr)

## $conv
## [1] 0
##
## $nllh
## [1] 196
##
## $mle
## [1] 0.11 0.18 0.36
##
## $se
## [1] 0.0037 0.0033 0.0162
```

```
FR_GEV$mle

## [1] 0.11 0.18 0.36
```

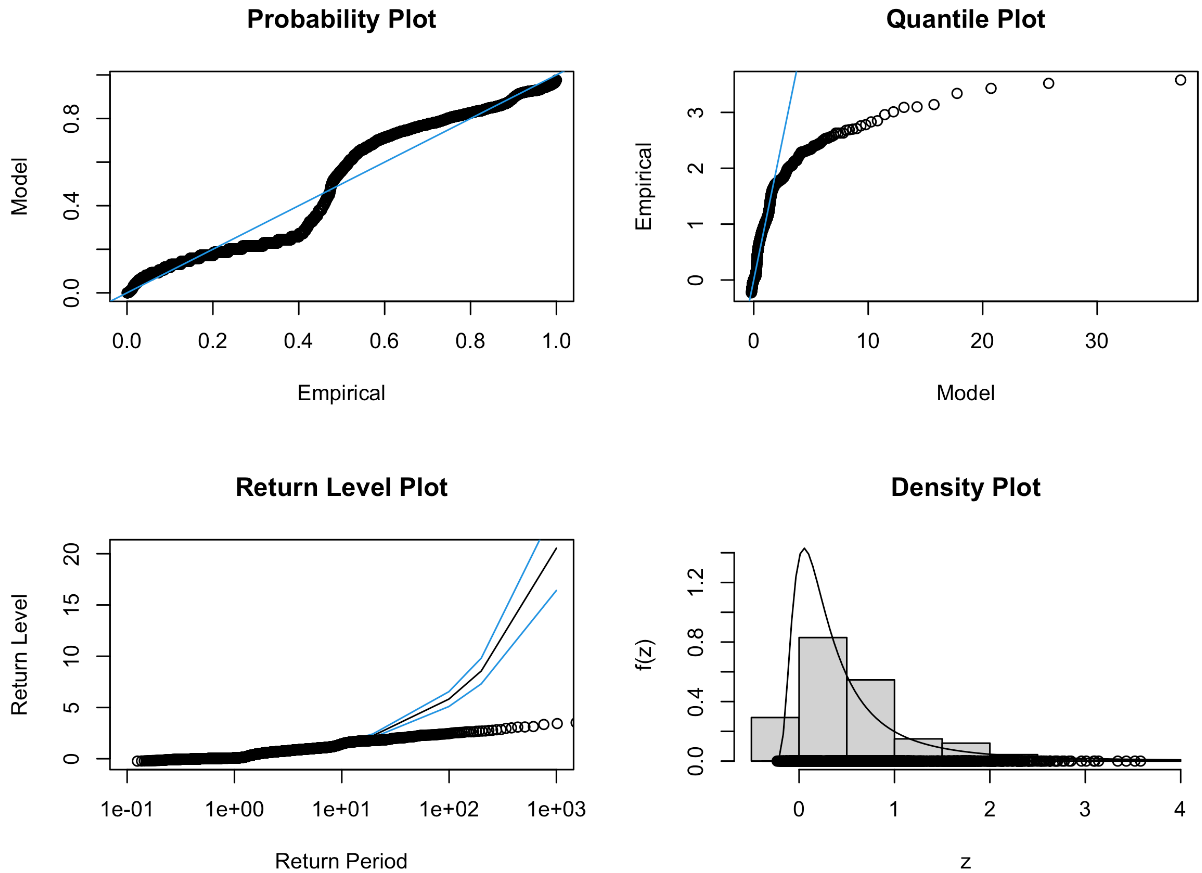


It shows the results for the estimated parameters. The shape parameter is -0.54 ($\xi < 0$). So, a Weibull distribution fits the data with high likelihood.

For Belgium:

```
## $conv
## [1] 0
##
## $nllh
## [1] 1929
##
## $mle
## [1] 0.16 0.29 0.53
##
## $se
## [1] 0.0061 0.0060 0.0191

BE_GEV$mle
```

```
## [1] 0.16 0.29 0.53
```



It shows the results for the estimated parameters. The shape parameter is -0.33 ($\xi < 0$). So, a Weibull distribution fits the data with high likelihood.

d.

Using elliptical copulas, estimate the correlation between the excess returns of your asset and the market excess returns.

```
##                  (France, Belgium) (France, Germany) (Belgium, Germany)
## Pearson rho                   0.93             -0.82              -0.70
## Kendall's tau                 0.77             -0.56              -0.50
## Spearman's rho                0.92             -0.78              -0.73
```

e.

Fit Archimedian copulas (Clayton or Weibull).

```
## Call: fitCopula(Ccop, data = pseudobs)


## Fit based on "maximum pseudo-likelihood" and 3053 2-dimensional
observations.
## Clayton copula, dim. d = 2
##       Estimate Std. Error
## alpha     6.71       0.19
## The maximized loglikelihood is 481
## Optimization converged
## Number of loglikelihood evaluations:
## function gradient
##        3        3
```