hw3

20989977 Zhang Mingtao

2024/4/20

1.

```
library(aTSA)
```

```
##
## 载入程辑包：'aTSA'
```

```
## The following object is masked from 'package:graphics':
##
##     identify
```

```
library(tseries)
```

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

```
##
## 载入程辑包：'tseries'
```

```
## The following objects are masked from 'package:aTSA':
##
##     adf.test, kpss.test, pp.test
```

```
library(TSA)
```

```
##
## 载入程辑包：'TSA'
```

```
## The following objects are masked from 'package:stats':
##
##     acf, arima
```

```
## The following object is masked from 'package:utils':
##
##     tar
```

```
library(forecast)
```

```
## Registered S3 methods overwritten by 'forecast':
##   method       from
##   fitted.Arima TSA
##   plot.Arima   TSA
```

```
##
## 载入程辑包：'forecast'
```

```
## The following object is masked from 'package:aTSA':
##
##     forecast
```

```
library(fGarch)
```

```
## NOTE: Packages 'fBasics', 'timeDate', and 'timeSeries' are no longer
## attached to the search() path when 'fGarch' is attached.
##
## If needed attach them yourself in your R script by e.g.,
##         require("timeSeries")
```

```
library(rugarch)
```

```
## 载入需要的程辑包：parallel
```

```
##
## 载入程辑包：'rugarch'
```

```
## The following object is masked from 'package:stats':
##
##     sigma
```

```
######################### 1
df = read.table("C://Users//张铭韬//Desktop//学业//港科大//MSDM5053时间序列//作业//assignment 3//d-sbuxsp0106.txt",header=F)
#  Convert the simple returns into percentage log returns
log_returns_SBUX = log(1 + ts(df$V2))

# Stationarity test
ndiffs(log_returns_SBUX) # d=0
```

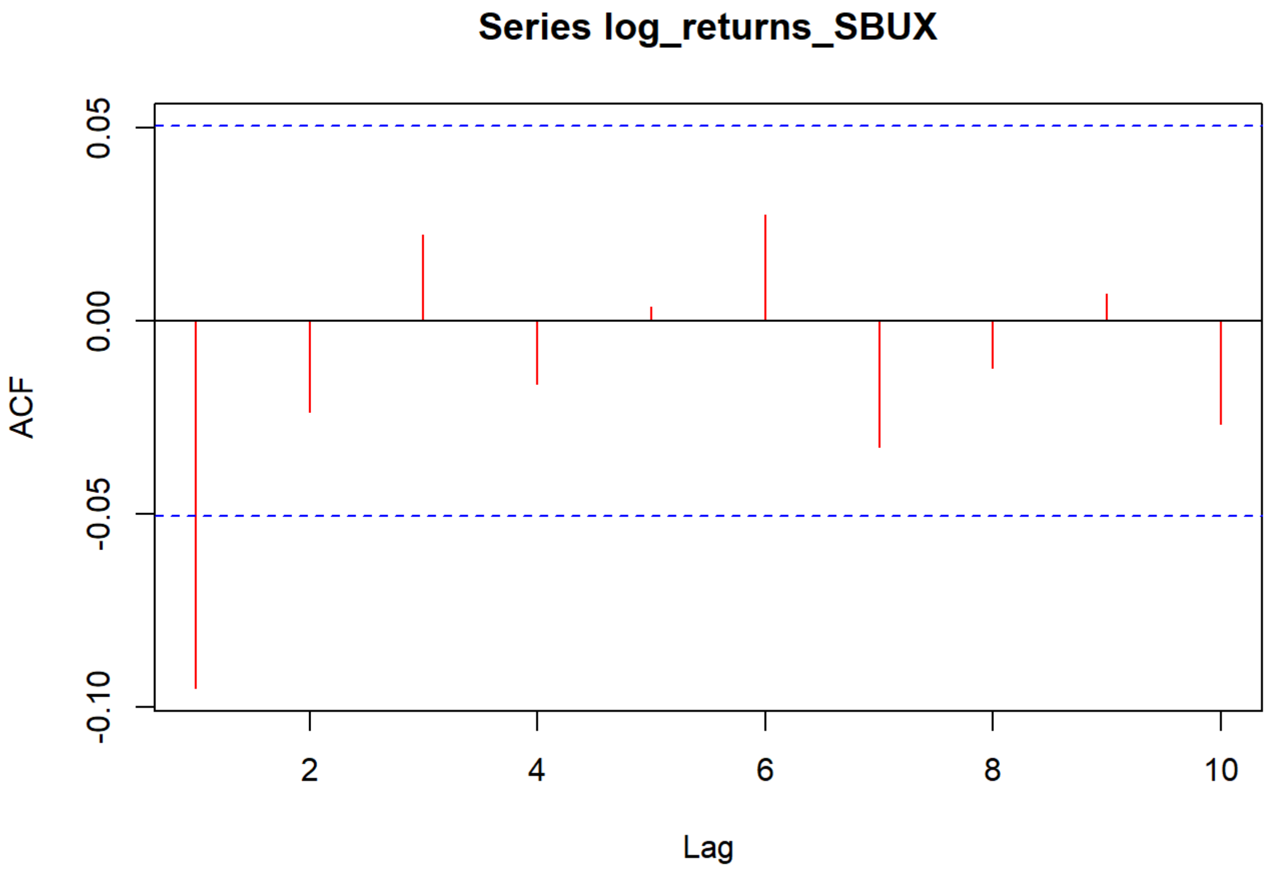
```
## [1] 0
```

```
pp.test(log_returns_SBUX) # p-value < 0.05, reject H0, stationary
```

```
## Warning in pp.test(log_returns_SBUX): p-value smaller than printed p-value
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  log_returns_SBUX
## Dickey-Fuller Z(alpha) = -1624.5, Truncation lag parameter = 7, p-value
## = 0.01
## alternative hypothesis: stationary
```

```
# a
# white noise test
acf(log_returns_SBUX, lag.max = 10, col="red")
```

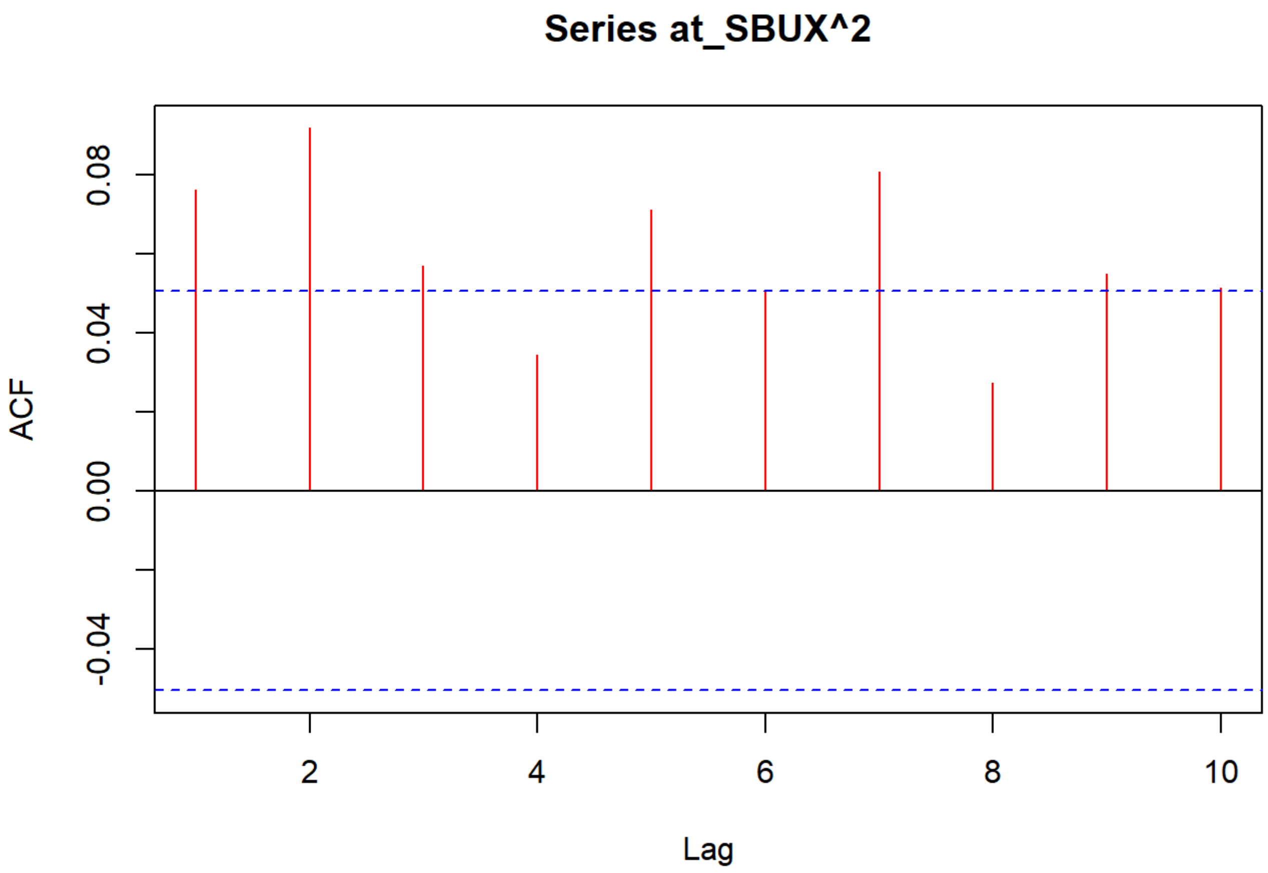



```
Box.test(log_returns_SBUX,lag=10,type="Ljung-Box")
```

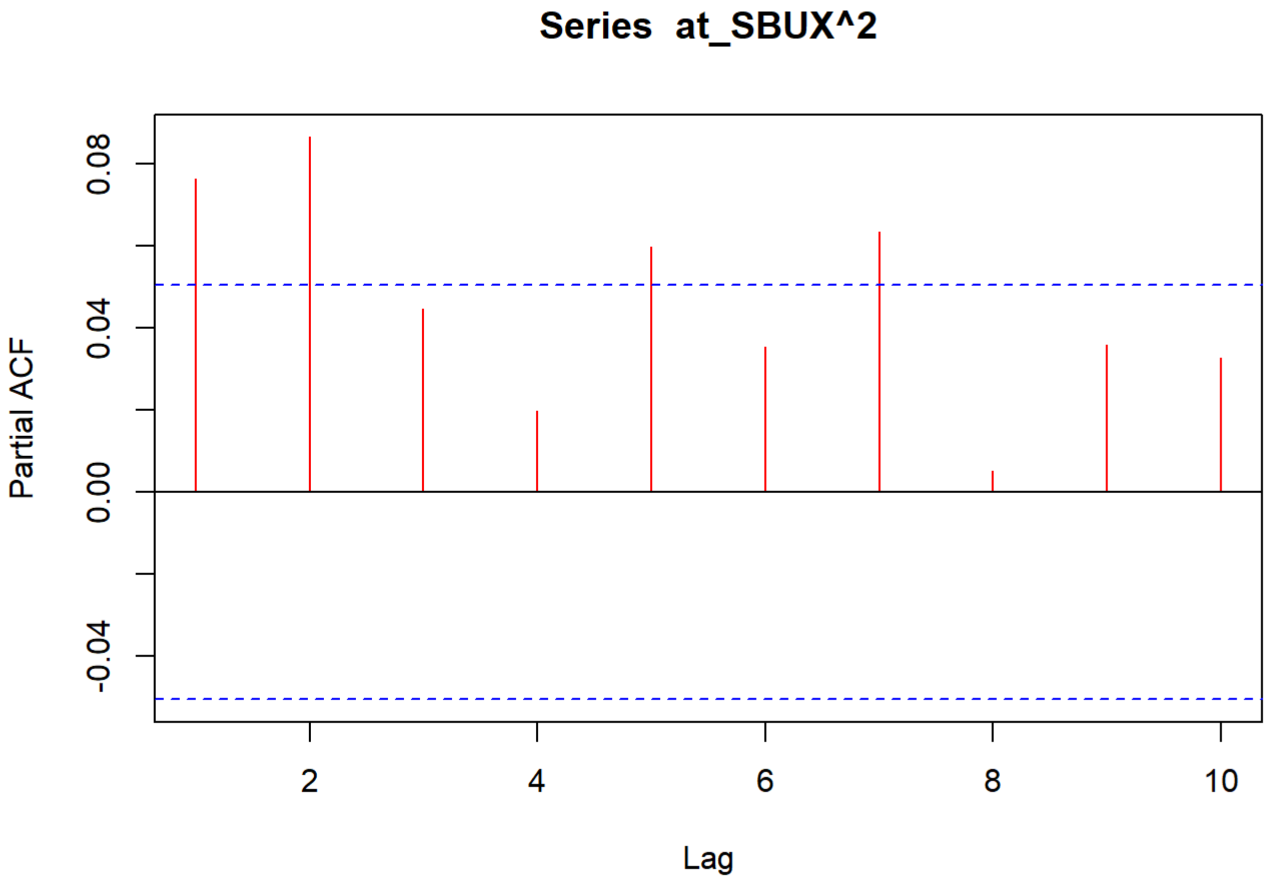
```
##
##  Box-Ljung test
##
## data:  log_returns_SBUX
## X-squared = 19.823, df = 10, p-value = 0.03098
```

```
# p-value < 0.05, there exists serial correlation in the log returns of Starbucks stock
```

```
# b
# ARCH test
at_SBUX=log_returns_SBUX-mean(log_returns_SBUX)
acf(at_SBUX^2, lag.max = 10, col="red")
```




```
pacf(at_SBUX^2, lag.max = 10, col="red")
```




```
Box.test(at_SBUX^2,lag=10,type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  at_SBUX^2
## X-squared = 59.142, df = 10, p-value = 5.265e-09
```

```
# p-value < 0.05, there exists ARCH effect in the log returns of Starbucks stock
```

```
# c
# First fit an ARMA model
auto.arima(log_returns_SBUX)
```

```
## Series: log_returns_SBUX
## ARIMA(3,0,0) with non-zero mean
##
## Coefficients:
##           ar1      ar2     ar3   mean
##       -0.0979  -0.0312  0.0170  8e-04
## s.e.   0.0258   0.0259  0.0258  5e-04
##
## sigma^2 = 0.0004274:  log likelihood = 3709.08
## AIC=-7408.17   AICc=-7408.13   BIC=-7381.58
```

```
est=Arima(log_returns_SBUX, order = c(3, 0, 0))
t = abs(est$coef)/sqrt(diag(est$var.coef))
df_t = length(log_returns_SBUX)-length(est$coef)
pt(t,df_t,lower.tail = F)
```

```
##          ar1          ar2          ar3    intercept
## 7.686944e-05 1.144282e-01 2.548148e-01 5.288355e-02
```

```
# fix ar2 and ar3 and intercept to 0, which are not significant
```

```
est=Arima(log_returns_SBUX, order = c(1, 0, 0))
t = abs(est$coef)/sqrt(diag(est$var.coef))
df_t = length(log_returns_SBUX)-length(est$coef)
pt(t,df_t,lower.tail = F)
```

```
##          ar1    intercept
## 0.0001051113 0.0556441858
```

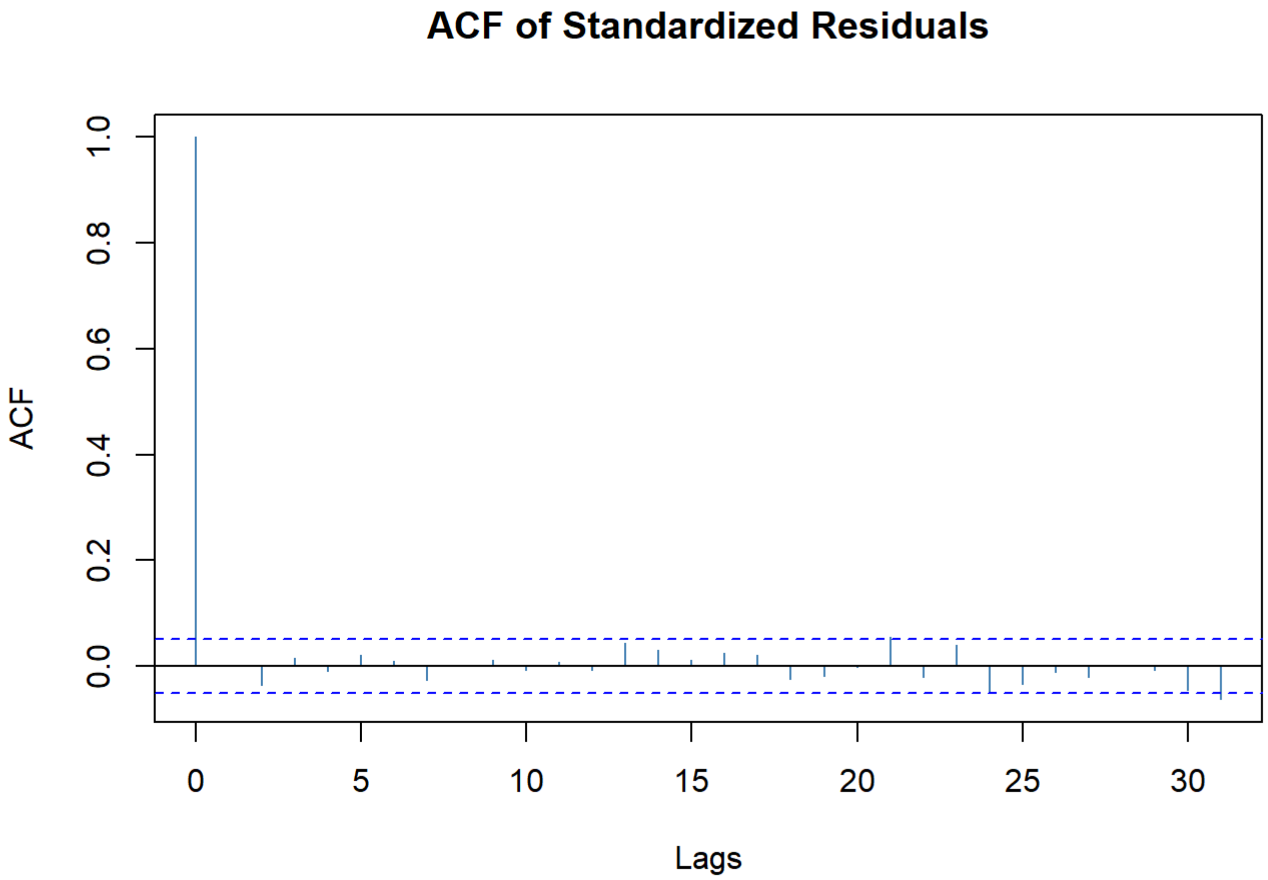
```
# ARMA(1,0)
```

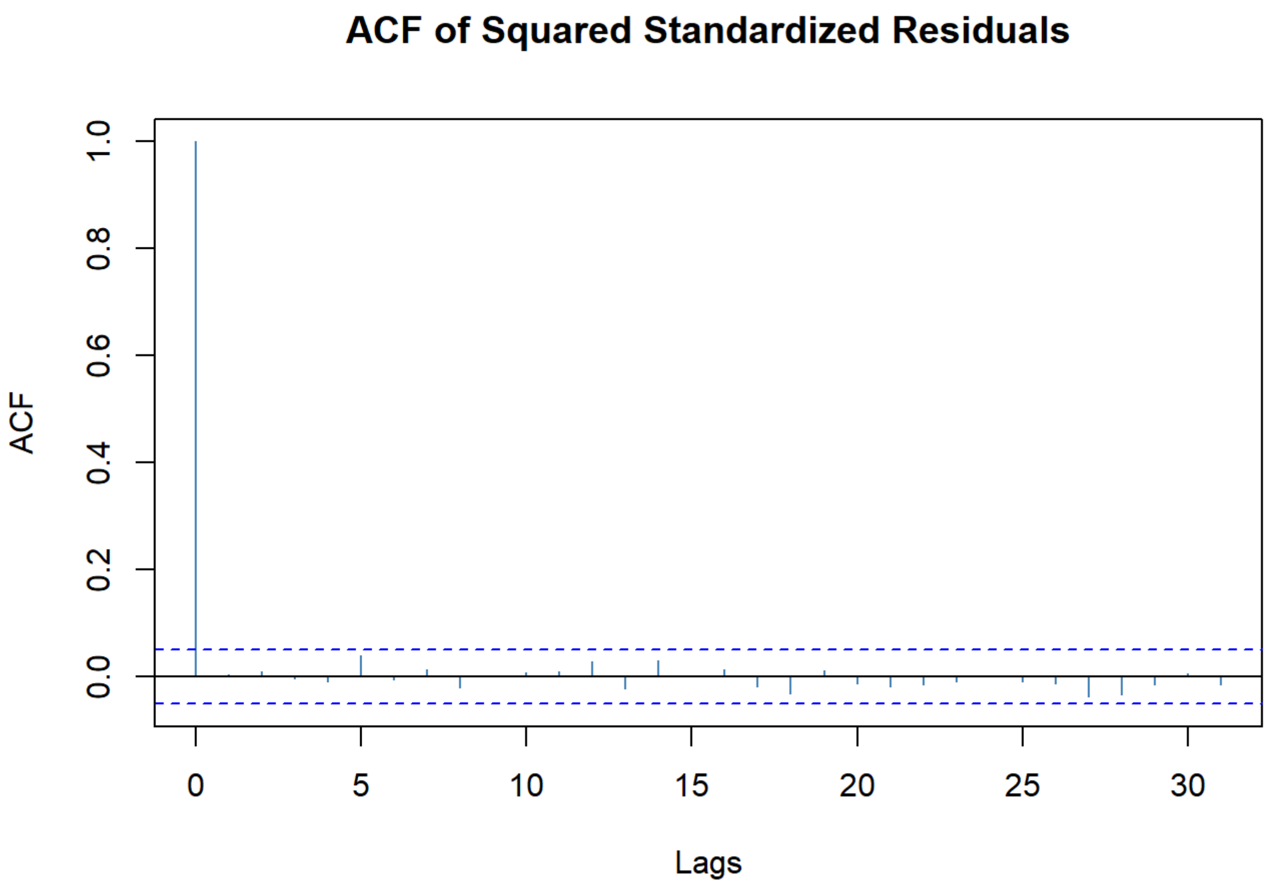
```
m1=garchFit(log_returns_SBUX~arma(1,0)+garch(1,1),data=log_returns_SBUX,trace=F,cond.dist="nor
m")
summary(m1)
```

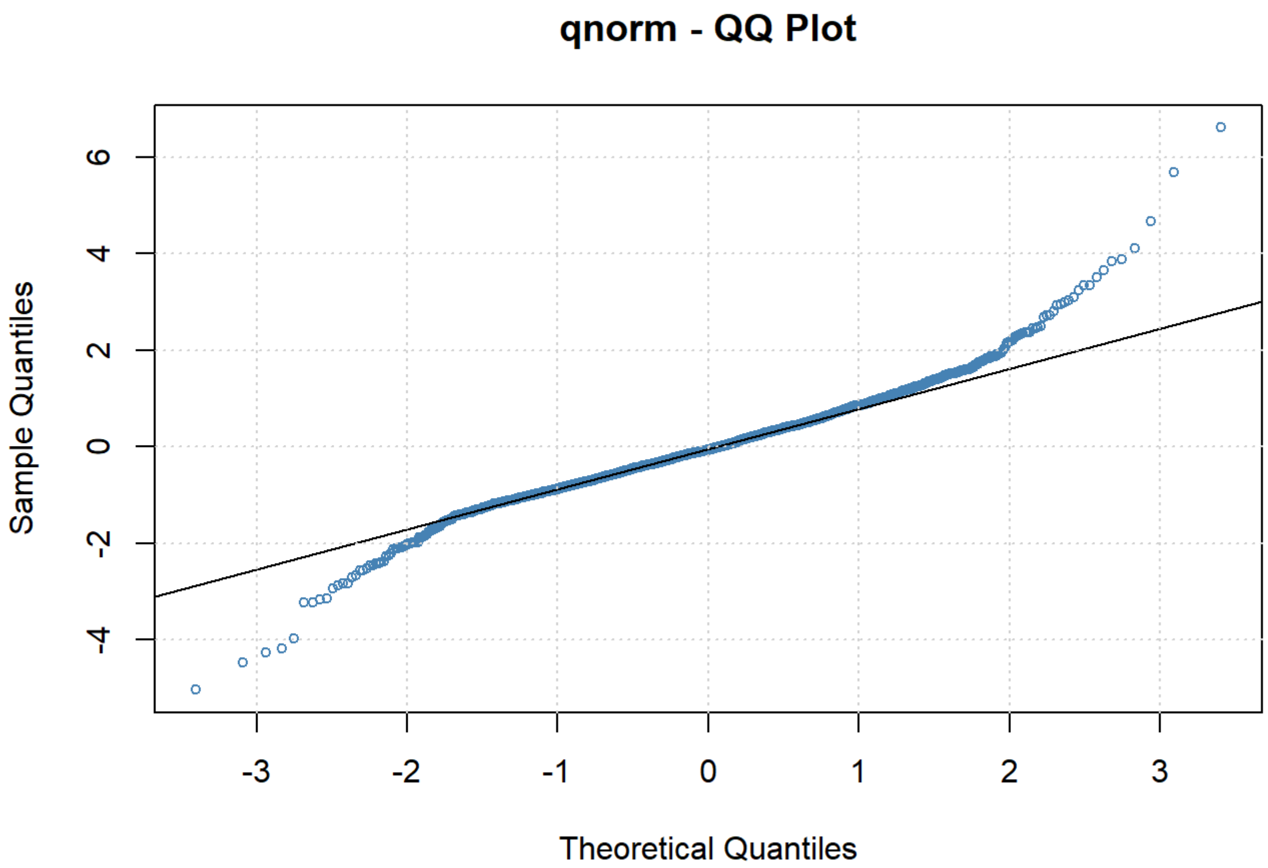
```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = log_returns_SBUX ~ arma(1, 0) + garch(1, 1),
##     data = log_returns_SBUX, cond.dist = "norm", trace = F)
##
## Mean and Variance Equation:
##  data ~ arma(1, 0) + garch(1, 1)
## <environment: 0x00000000289defe8>
##  [data = log_returns_SBUX]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##          mu          ar1        omega       alpha1        beta1
##  1.3257e-03  -7.4751e-02   1.4719e-06   1.8508e-02   9.7755e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## mu      1.326e-03   4.782e-04    2.772  0.00557 **
## ar1    -7.475e-02   2.669e-02   -2.801  0.00509 **
## omega   1.472e-06   6.199e-07    2.375  0.01757 *
## alpha1  1.851e-02   3.714e-03    4.983 6.26e-07 ***
## beta1   9.776e-01   4.105e-03  238.163  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  3790.238    normalized:  2.515089
##
## Description:
##  Sat Apr 20 20:24:21 2024 by user: 张铭韬
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  1000.482  0
##  Shapiro-Wilk Test  R    W      0.9594053 0
##  Ljung-Box Test     R    Q(10)  4.518729  0.920928
##  Ljung-Box Test     R    Q(15)  9.367761  0.857517
##  Ljung-Box Test     R    Q(20)  12.40399  0.9014695
##  Ljung-Box Test     R^2  Q(10)  3.655742  0.9615438
##  Ljung-Box Test     R^2  Q(15)  7.095462  0.9549464
##  Ljung-Box Test     R^2  Q(20)  9.87691   0.9703494
##  LM Arch Test       R    TR^2   4.964507  0.9591539
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -5.023541 -5.005897 -5.023563 -5.016970
```

```
# All coefficients are significant.
# model: ARMA(1,0)-GARCH(1,1)
# r_t = μ_t + a_t
#  μ_t = μ_0 + Φ_1 * r_t-1
# a_t = σ_t *  ε_t
# (σ_t)^2 = α_0 + α_1*(a_t-1)^2 + β_1*(σ_t-1)^2
# where μ_0 = 1.326e-03, Φ_1 = -7.475e-02, α_0 = 1.472e-06, α_1 = 1.851e-02, β_1 = 9.776e-01
```

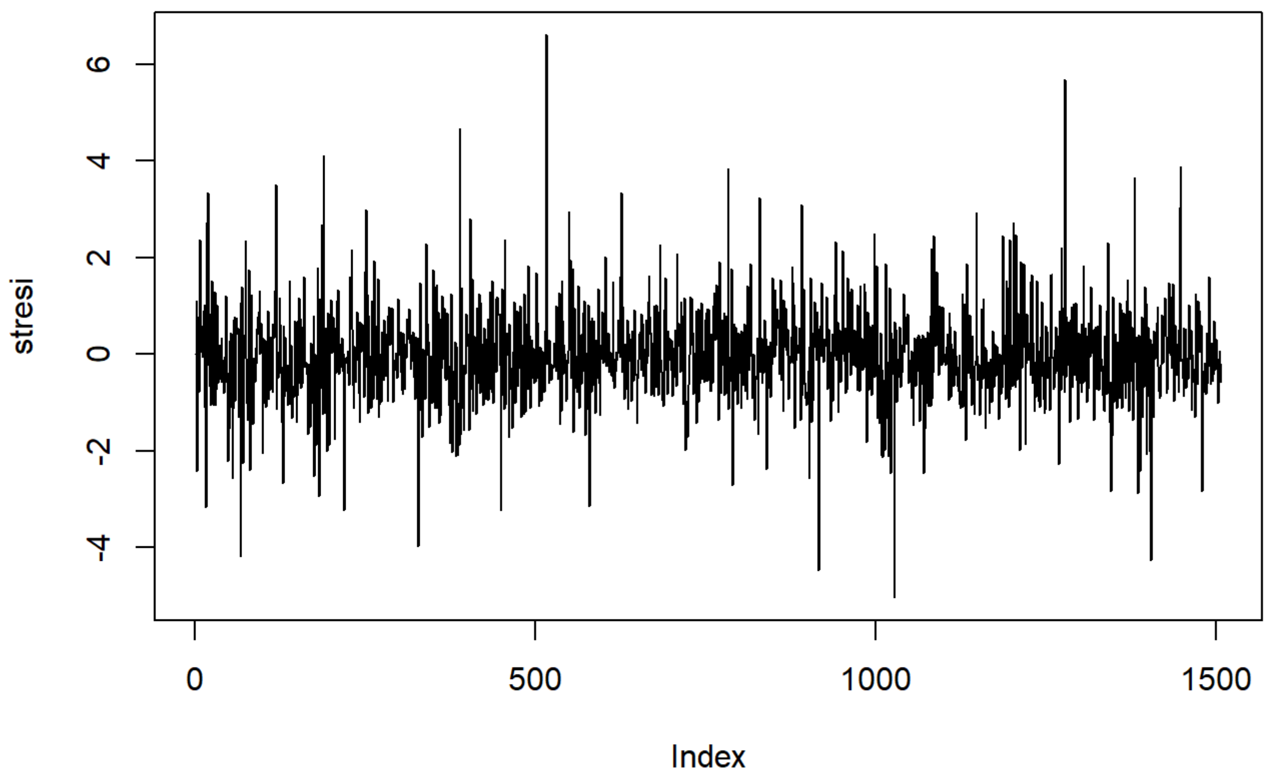
```
plot(m1, which = c(10,11,13))
```


ACF of Standardized Residuals
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
30
Lags


ACF of Squared Standardized Residuals
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
30
Lags




```
stresi=residuals(m1,standardize=T)
plot(stresi,type="l")
```



```
Box.test(stresi,10,type="Ljung-Box",fitdf = 3) # p-value > 0.05, white noise
```

```
##
##  Box-Ljung test
##
## data:  stresi
## X-squared = 4.5187, df = 7, p-value = 0.7185
```

```
Box.test(stresi^2,10,type="Ljung-Box",fitdf = 3) # p-value > 0.05, remains no ARCH effect
```

```
##
##  Box-Ljung test
##
## data:  stresi^2
## X-squared = 3.6557, df = 7, p-value = 0.8185
```

2.

```
######################### 2
log_returns_SP = log(1 + ts(df$V3))
```

```
# Stationarity test
ndiffs(log_returns_SP) # d=0
```

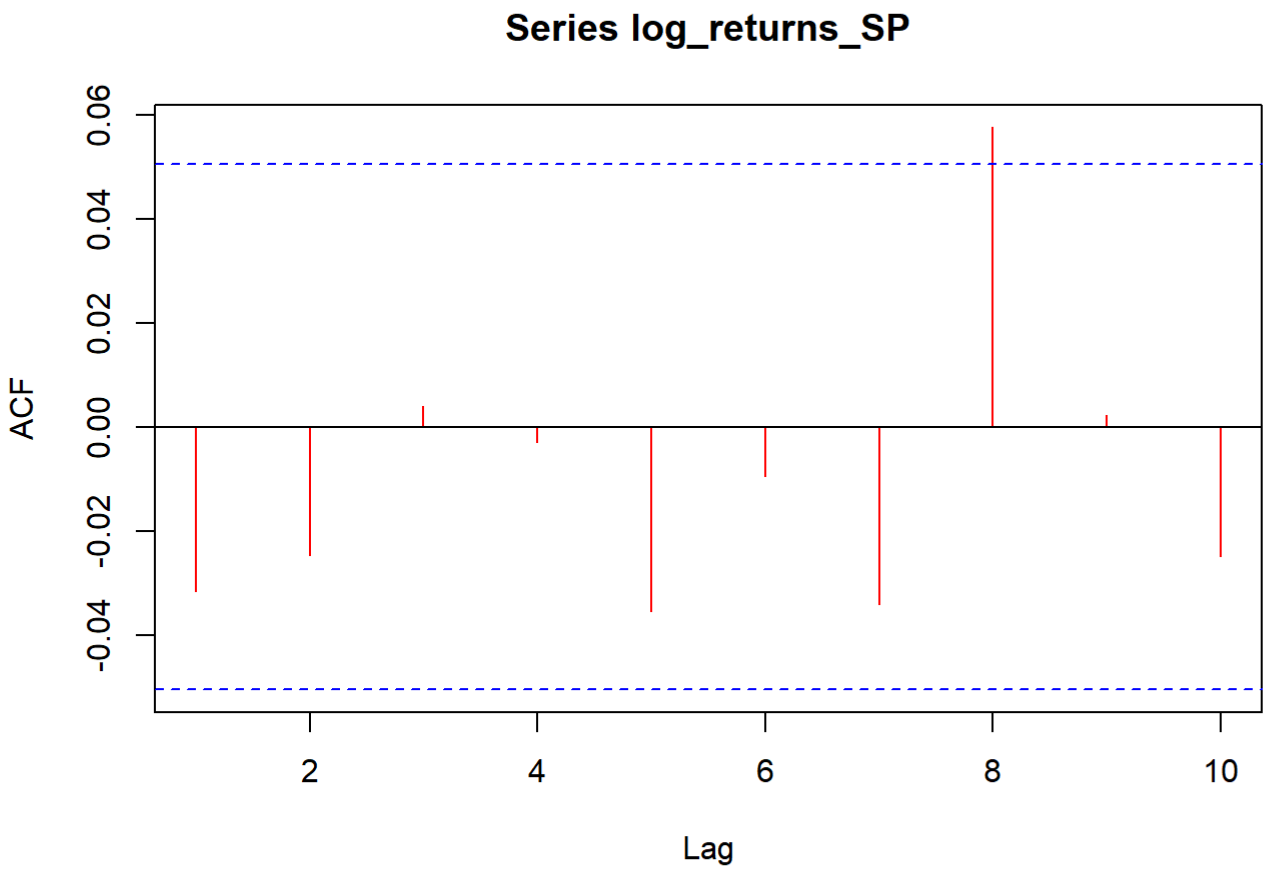
```
## [1] 0
```

```
pp.test(log_returns_SBUX) # p-value < 0.05, reject H0, stationary
```

```
## Warning in pp.test(log_returns_SBUX): p-value smaller than printed p-value
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  log_returns_SBUX
## Dickey-Fuller Z(alpha) = -1624.5, Truncation lag parameter = 7, p-value
## = 0.01
## alternative hypothesis: stationary
```

```
# a
# white noise test
acf(log_returns_SP, lag.max = 10, col="red")
```

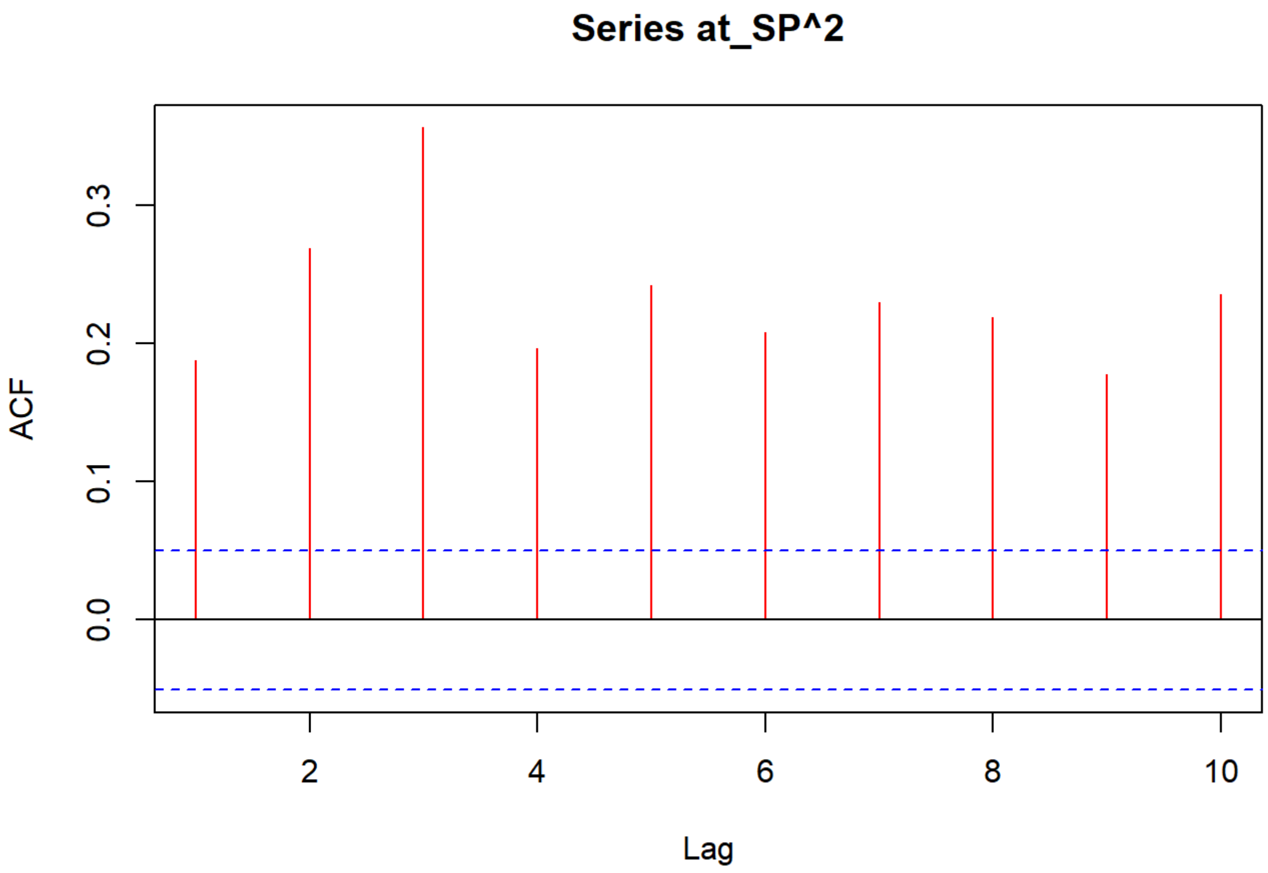



```
Box.test(log_returns_SP,lag=10,type="Ljung-Box")
```

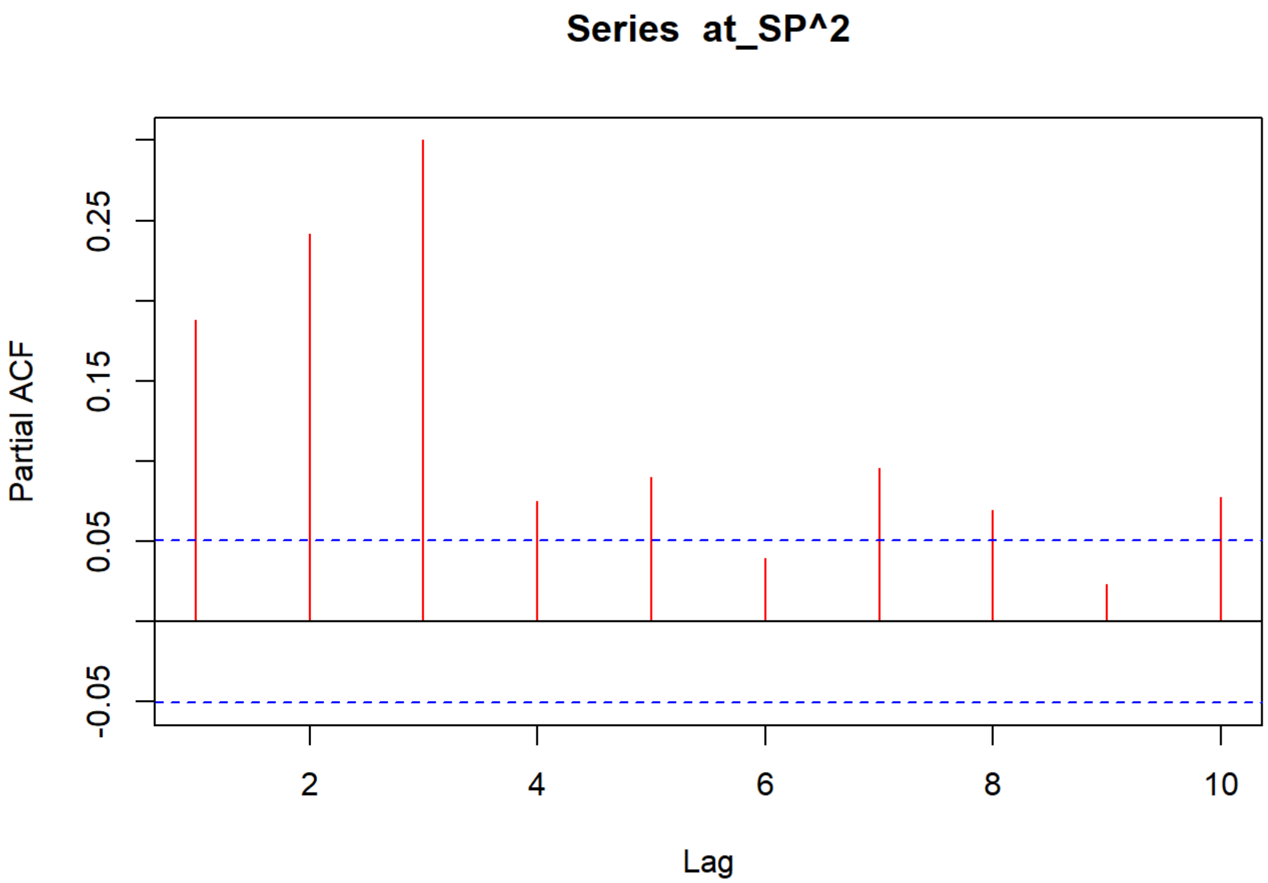
```
##
##  Box-Ljung test
##
## data:  log_returns_SP
## X-squared = 12.253, df = 10, p-value = 0.2685
```

```
# p-value > 0.05, there doesn't exist any serial correlation in the log returns of S&P index
```

```
# b
# ARCH test
at_SP=log_returns_SP-mean(log_returns_SP)
acf(at_SP^2, lag.max = 10, col="red")
```




```
pacf(at_SP^2, lag.max = 10, col="red")
```



```
Box.test(at_SP^2,lag=10,type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  at_SP^2
## X-squared = 850.73, df = 10, p-value < 2.2e-16
```

```
# p-value < 0.05, there exists ARCH effect in the log returns of S&P index
```

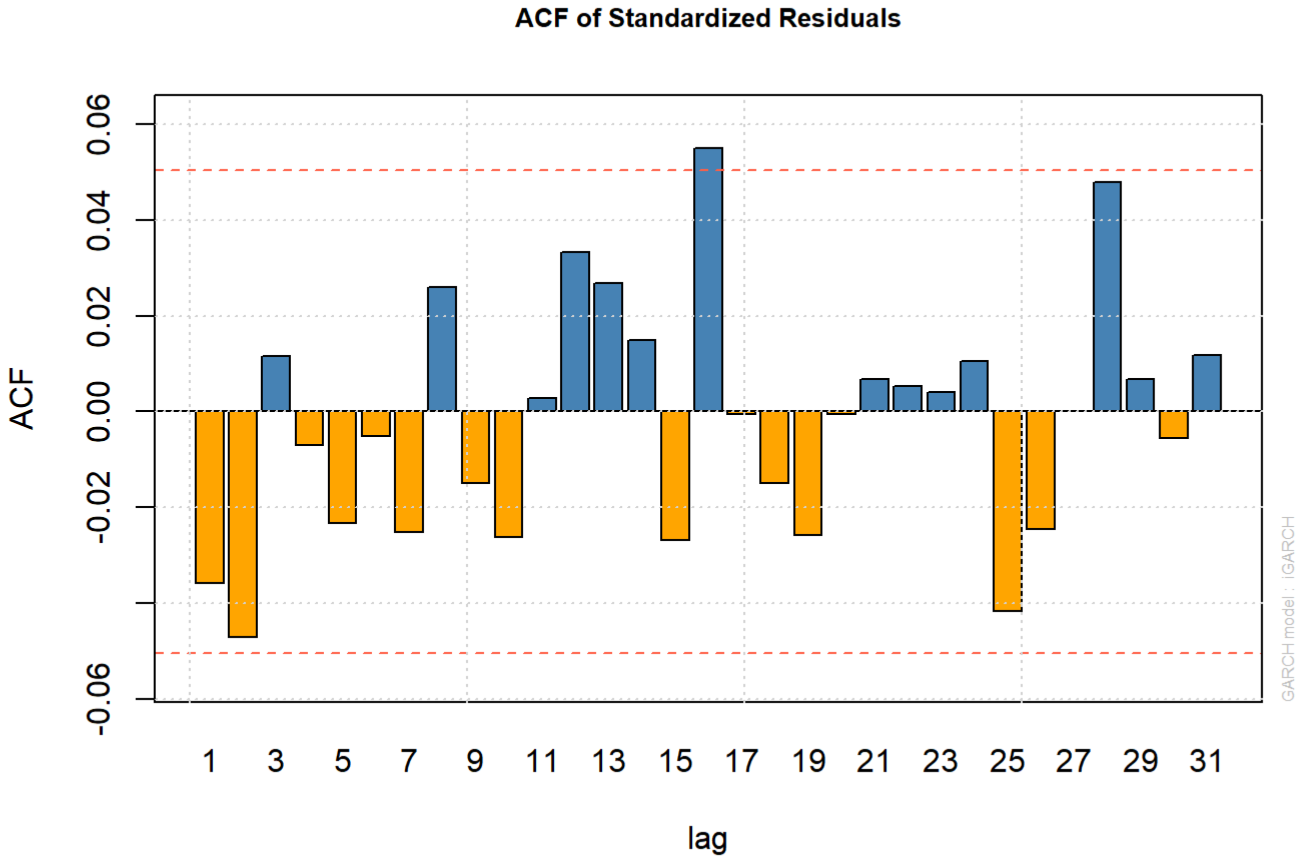
```
# c
spec2=ugarchspec(variance.model=list(model="iGARCH",garchOrder = c(1, 1)),
                 mean.model=list(armaOrder=c(0,0),include.mean = TRUE),
                 distribution.model = "std")

m2=ugarchfit(spec=spec2,data=log_returns_SP)
m2  ### see output
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : iGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : std
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000392    0.000200  1.95543 0.050533
## omega   0.000000    0.000001  0.50636 0.612601
## alpha1  0.062736    0.008813  7.11842 0.000000
## beta1   0.937264          NA       NA       NA
## shape  14.138557    4.762705  2.96860 0.002992
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000392    0.000205  1.90814 0.056372
## omega   0.000000    0.000002  0.14958 0.881099
## alpha1  0.062736    0.047877  1.31035 0.190079
## beta1   0.937264          NA       NA       NA
## shape  14.138557    5.908177  2.39305 0.016709
##
## LogLikelihood : 4937.928
##
## Information Criteria
## ------------------------------------
##
## Akaike       -6.5480
## Bayes        -6.5339
## Shibata      -6.5480
## Hannan-Quinn -6.5428
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                      1.934  0.1644
## Lag[2*(p+q)+(p+q)-1][2]     3.612  0.0959
## Lag[4*(p+q)+(p+q)-1][5]     4.934  0.1585
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.8765  0.3492
## Lag[2*(p+q)+(p+q)-1][5]    3.5079  0.3221
## Lag[4*(p+q)+(p+q)-1][9]    4.5246  0.5024
## d.o.f=2
##
```

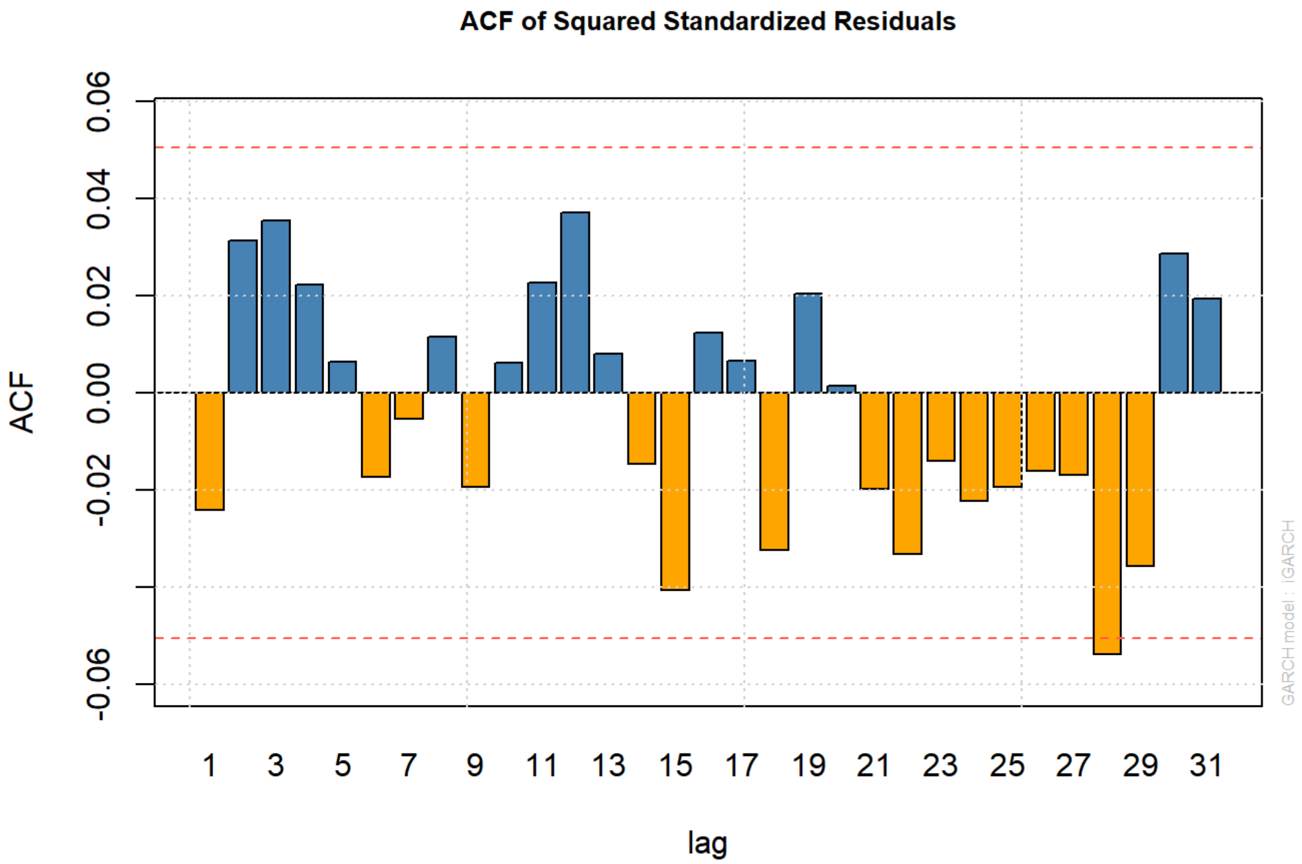
```
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]     1.894 0.500 2.000  0.1687
## ARCH Lag[5]     2.528 1.440 1.667  0.3660
## ARCH Lag[7]     2.857 2.315 1.543  0.5411
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  63.385
## Individual Statistics:
## mu      0.18810
## omega  44.40429
## alpha1  0.07953
## shape   0.24987
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:         1.07 1.24 1.6
## Individual Statistic:    0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value     prob sig
## Sign Bias           0.1384 0.889951
## Negative Sign Bias  0.7240 0.469162
## Positive Sign Bias  2.7901 0.005336 ***
## Joint Effect       13.4413 0.003773 ***
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     33.64      0.02027
## 2    30     34.63      0.21709
## 3    40     45.62      0.21605
## 4    50     46.52      0.57437
##
##
## Elapsed time : 0.1326442
```

```
# Coefficients of mu and omega are not significant.
# model: iGARCH(1,1)
# r_t = μ_t + a_t
# μ_t = μ_0
# a_t = σ_t * ε_t
# (σ_t)^2 = α_0(=0) + α_1*(a_t-1)^2 + β_1*(σ_t-1)^2, where α_1 + β_1 = 1
# where μ_0 = 0.000392(not significant, can be seen as 0), α_0 = 0, α_1 = 0.062736, β_1 = 0.937264
```

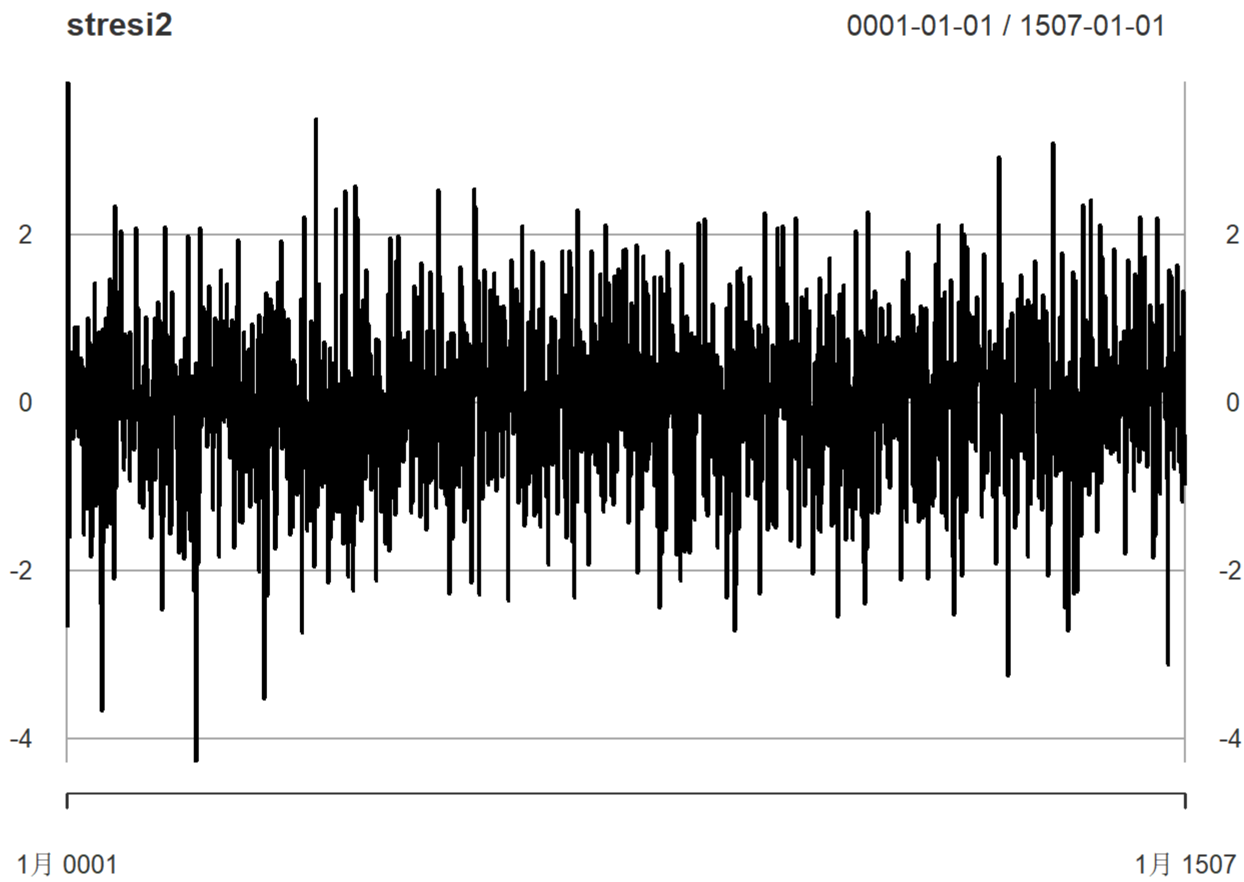
```
plot(m2, which = 10)
```




```
plot(m2, which = 11)
```



```
stresi2=residuals(m2,standardize=T)
plot(stresi2,type="l")
```




```
Box.test(stresi2,10,type="Ljung-Box",fitdf = 1) # p-value > 0.05, white noise
```

```
##
##  Box-Ljung test
##
## data:  stresi2
## X-squared = 9.7928, df = 9, p-value = 0.3675
```

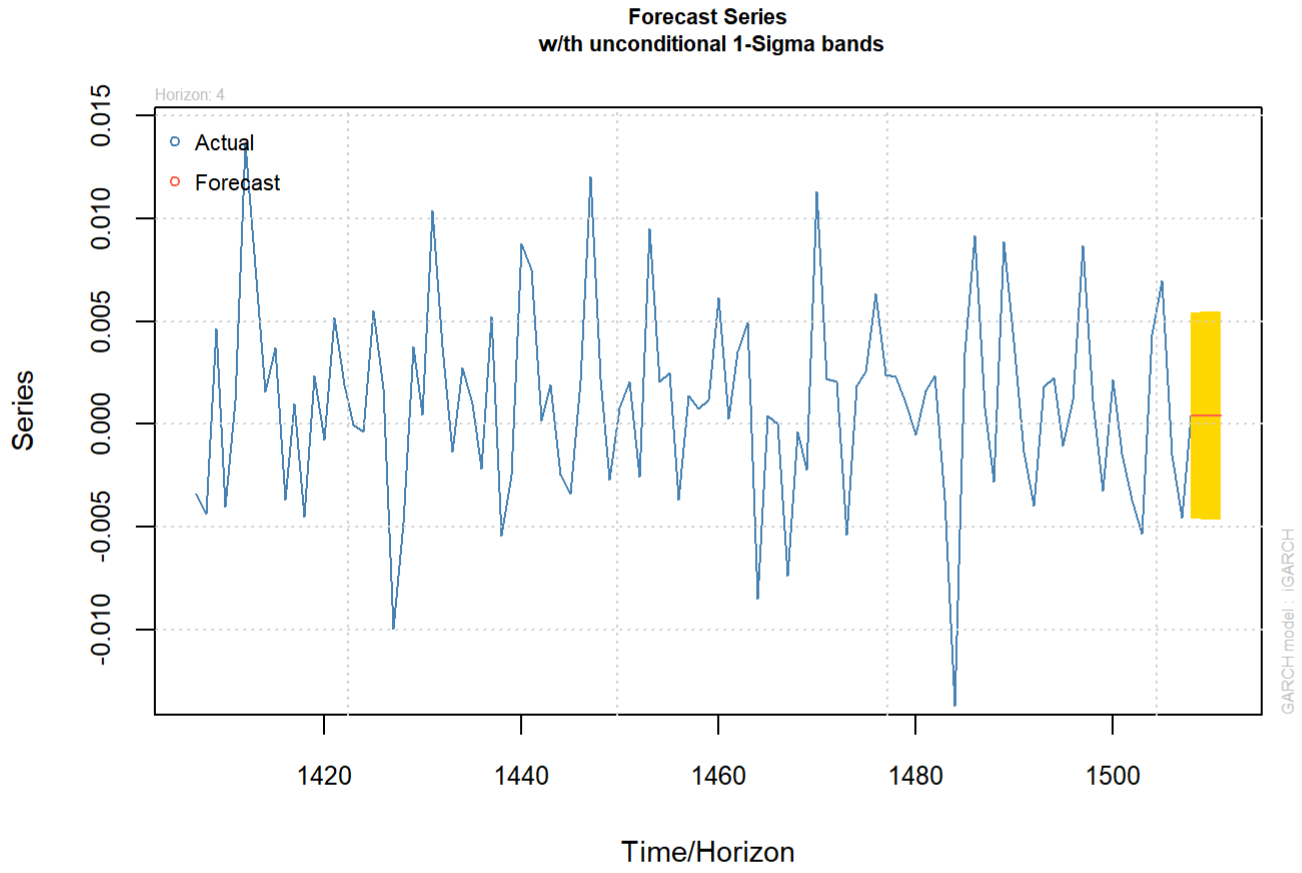
```
Box.test(stresi2^2,10,type="Ljung-Box",fitdf = 1) # p-value > 0.05, remains no ARCH effect
```

```
##
##  Box-Ljung test
##
## data:  stresi2^2
## X-squared = 6.3956, df = 9, p-value = 0.6998
```

```
# d
forecast = ugarchforecast(m2, n.ahead = 4, data=log_returns_SP)
```

```
## Warning in `setfixed<-`(`*tmp*`, value = as.list(pars)): Unrecognized Parameter
## in Fixed Values: beta1...Ignored
```

```
plot(forecast, which = 1)
```




```
U=forecast@forecast$seriesFor+1.96*forecast@forecast$sigmaFor
L=forecast@forecast$seriesFor-1.96*forecast@forecast$sigmaFor
```

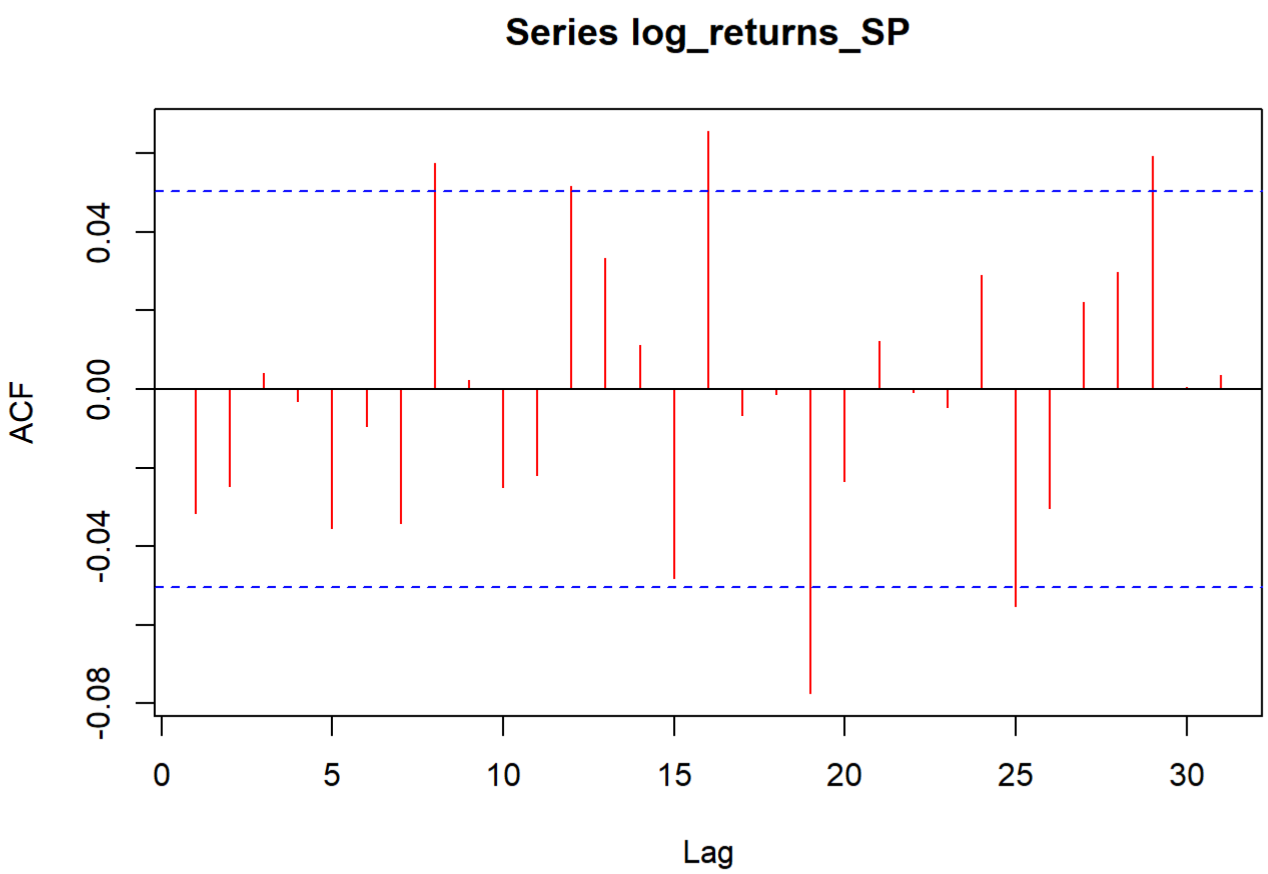
```
forecast
```

```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: iGARCH
## Horizon: 4
## Roll Steps: 0
## Out of Sample: 0
##
## 0-roll forecast [T0=1507-01-01]:
##        Series    Sigma
## T+1 0.0003915 0.004992
## T+2 0.0003915 0.005020
## T+3 0.0003915 0.005048
## T+4 0.0003915 0.005076
```

```
c(L[1],U[1])
```

```
## [1] -0.009393318  0.010176378
```

```
#!! the result looks not right, I think we should take the arma model into account:
# Let's check more lags:
acf(log_returns_SP, col="red")
```




```
for( i in c(6,12,18,24,30) ){
  print(Box.test(log_returns_SP,lag=i,type="Ljung-Box"))
}
```

```
##
##  Box-Ljung test
##
## data:  log_returns_SP
## X-squared = 4.5064, df = 6, p-value = 0.6085
##
##
##  Box-Ljung test
##
## data:  log_returns_SP
## X-squared = 17.026, df = 12, p-value = 0.1486
##
##
##  Box-Ljung test
##
## data:  log_returns_SP
## X-squared = 29.074, df = 18, p-value = 0.04748
##
##
##  Box-Ljung test
##
## data:  log_returns_SP
## X-squared = 40.629, df = 24, p-value = 0.01829
##
##
##  Box-Ljung test
##
## data:  log_returns_SP
## X-squared = 54.25, df = 30, p-value = 0.004311
```

```
# we can see there may exist some serial correlation.
# do the ARMA model:
auto.arima(log_returns_SP)
```

```
## Series: log_returns_SP
## ARIMA(3,0,3) with zero mean
##
## Coefficients:
##           ar1      ar2      ar3     ma1     ma2     ma3
##       -0.1226  -0.6539  -0.5076  0.0891  0.6522  0.5265
## s.e.   1.5233   0.6681   1.4062  1.4791  0.6989  1.3856
##
## sigma^2 = 0.0001156:  log likelihood = 4695.46
## AIC=-9376.91   AICc=-9376.84   BIC=-9339.69
```

```
spec2=ugarchspec(variance.model=list(model="iGARCH",garchOrder = c(1, 1)),
                 mean.model=list(armaOrder=c(3,3),include.mean = TRUE),
                 distribution.model = "std")
```

```
m2=ugarchfit(spec=spec2,data=log_returns_SP)
```

```
m2  ### see output
```

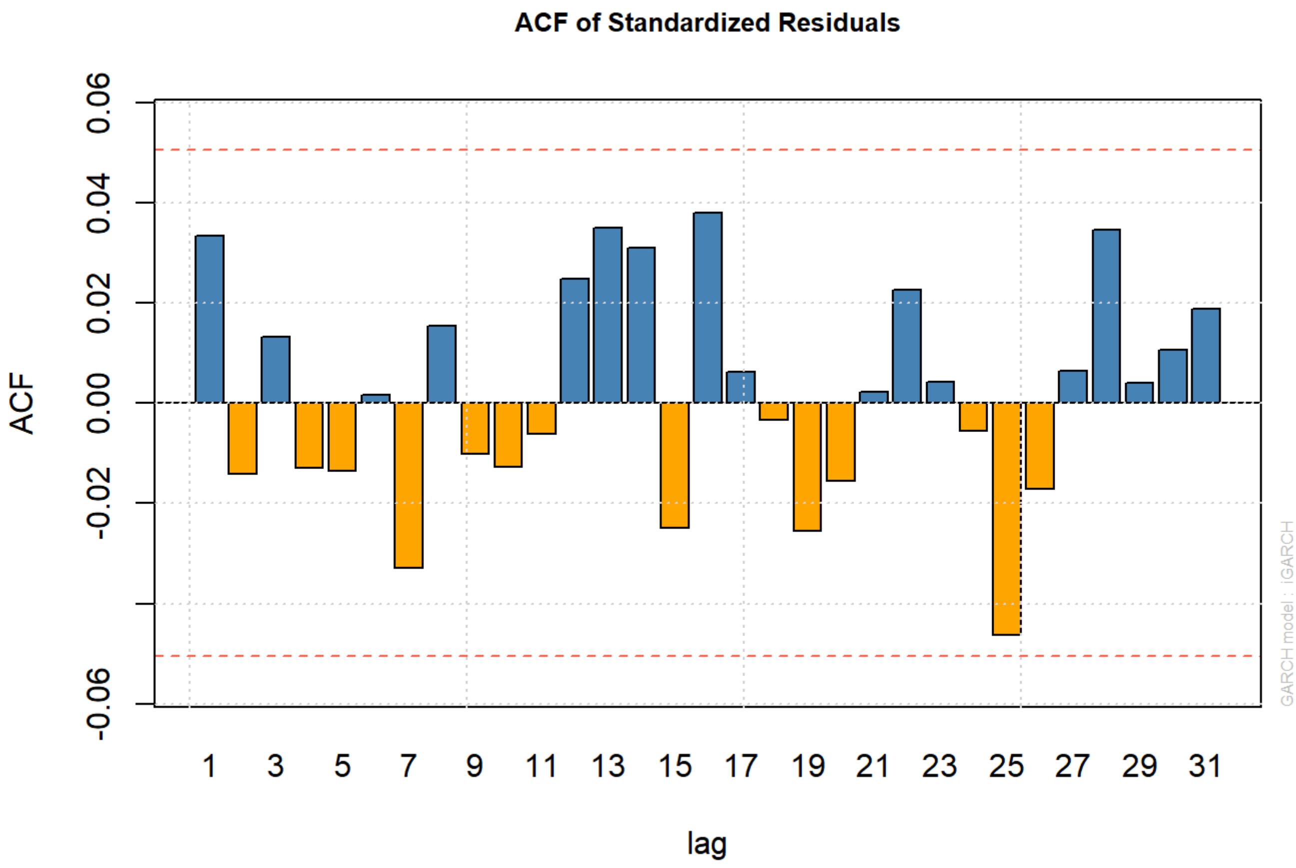
```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : iGARCH(1,1)
## Mean Model   : ARFIMA(3,0,3)
## Distribution : std
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error     t value Pr(>|t|)
## mu      0.000501    0.000019    26.56997 0.000000
## ar1     0.522999    0.000043 12194.27027 0.000000
## ar2    -1.028071    0.000104 -9879.92527 0.000000
## ar3     0.418220    0.000036 11662.16157 0.000000
## ma1    -0.601726    0.000071 -8496.27044 0.000000
## ma2     1.063333    0.000092 11590.13804 0.000000
## ma3    -0.492953    0.000050 -9899.89653 0.000000
## omega   0.000000    0.000000     0.78583 0.431966
## alpha1  0.061252    0.008404     7.28809 0.000000
## beta1   0.938748          NA          NA       NA
## shape  15.650549    6.705100     2.33413 0.019589
##
## Robust Standard Errors:
##         Estimate  Std. Error     t value Pr(>|t|)
## mu      0.000501    0.000011  4.3969e+01  0.00000
## ar1     0.522999    0.000693  7.5447e+02  0.00000
## ar2    -1.028071    0.000696 -1.4774e+03  0.00000
## ar3     0.418220    0.000694  6.0260e+02  0.00000
## ma1    -0.601726    0.001218 -4.9409e+02  0.00000
## ma2     1.063333    0.000543  1.9576e+03  0.00000
## ma3    -0.492953    0.000622 -7.9229e+02  0.00000
## omega   0.000000    0.000005  5.3337e-02  0.95746
## alpha1  0.061252    0.081787  7.4892e-01  0.45391
## beta1   0.938748          NA          NA       NA
## shape  15.650549   17.334404  9.0286e-01  0.36660
##
## LogLikelihood : 4959.552
##
## Information Criteria
## ------------------------------------
##
## Akaike       -6.5687
## Bayes        -6.5335
## Shibata      -6.5688
## Hannan-Quinn -6.5556
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                      1.682  0.1946
## Lag[2*(p+q)+(p+q)-1][17]    5.773  1.0000
```

```
## Lag[4*(p+q)+(p+q)-1][29]    10.165  0.9665
## d.o.f=6
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.3525 0.55270
## Lag[2*(p+q)+(p+q)-1][5]    9.2658 0.01417
## Lag[4*(p+q)+(p+q)-1][9]   10.7215 0.03501
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]     1.661 0.500 2.000  0.1975
## ARCH Lag[5]     2.094 1.440 1.667  0.4507
## ARCH Lag[7]     2.230 2.315 1.543  0.6686
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  70.6724
## Individual Statistics:
## mu      0.03954
## ar1     0.05218
## ar2     0.08153
## ar3     0.02922
## ma1     0.07909
## ma2     0.02719
## ma3     0.08685
## omega  48.23162
## alpha1  0.09283
## shape   0.25533
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          2.29 2.54 3.05
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value      prob sig
## Sign Bias           0.3446 0.7304233
## Negative Sign Bias  1.4282 0.1534485
## Positive Sign Bias  2.8949 0.0038472 ***
## Joint Effect       17.1169 0.0006687 ***
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     29.97      0.05213
## 2    30     41.16      0.06675
## 3    40     46.90      0.18029
## 4    50     66.09      0.05214
##
```

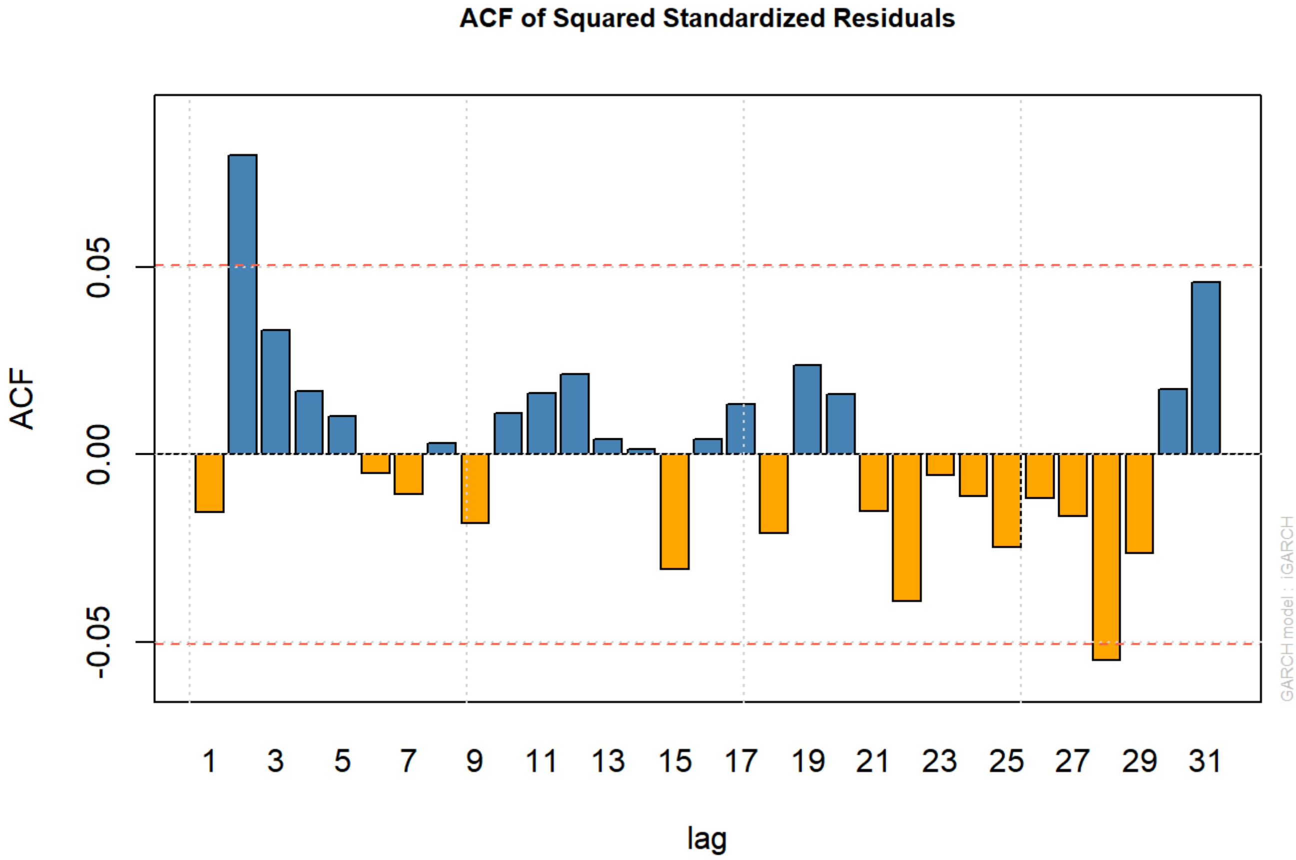
```
##
## Elapsed time : 1.078113
```

```
# Coefficients of mu and omega are not significant.
# model: ARMA(3,3)-GARCH(1,1)
# r_t = μ_t + a_t
#  μ_t = μ_0 + ϕ_1 * r_t-1 + ϕ_2 * r_t-2 + ϕ_3 * r_t-3 - θ_1 * a_t-1 - θ_2 * a_t-2 - θ_3
* a_t-3
# a_t = σ_t * ε_t
# (σ_t)^2 = α_0(=0) + α_1*(a_t-1)^2 + β_1*(σ_t-1)^2, where α_1 + β_1 = 1
# where μ_0 = 0.000501, ϕ_1 = 0.522999, ϕ_2 = -1.028071, ϕ_3 = 0.418220, θ_1 = -0.601726,
 θ_2 = 1.063333, θ_3 = -0.492953
#        α_0 = 0, α_1 = 0.061252, β_1 = 0.938748
```

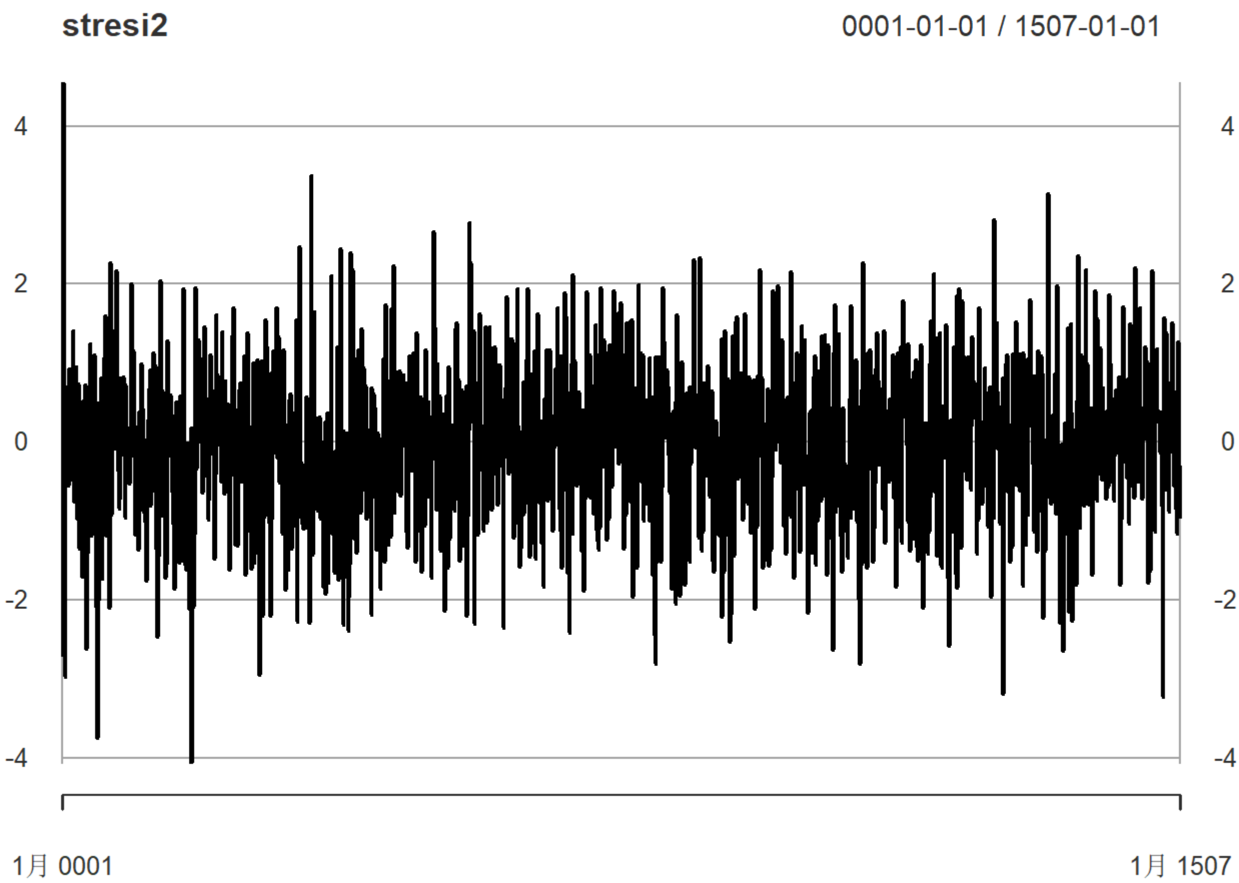
```
plot(m2, which = 10)
```




```
plot(m2, which = 11)
```




```
stresi2=residuals(m2,standardize=T)
plot(stresi2,type="l")
```



```
Box.test(stresi2,20,type="Ljung-Box",fitdf = 7) # p-value > 0.05, white noise
```

```
##
##  Box-Ljung test
##
## data:  stresi2
## X-squared = 14.118, df = 13, p-value = 0.3656
```

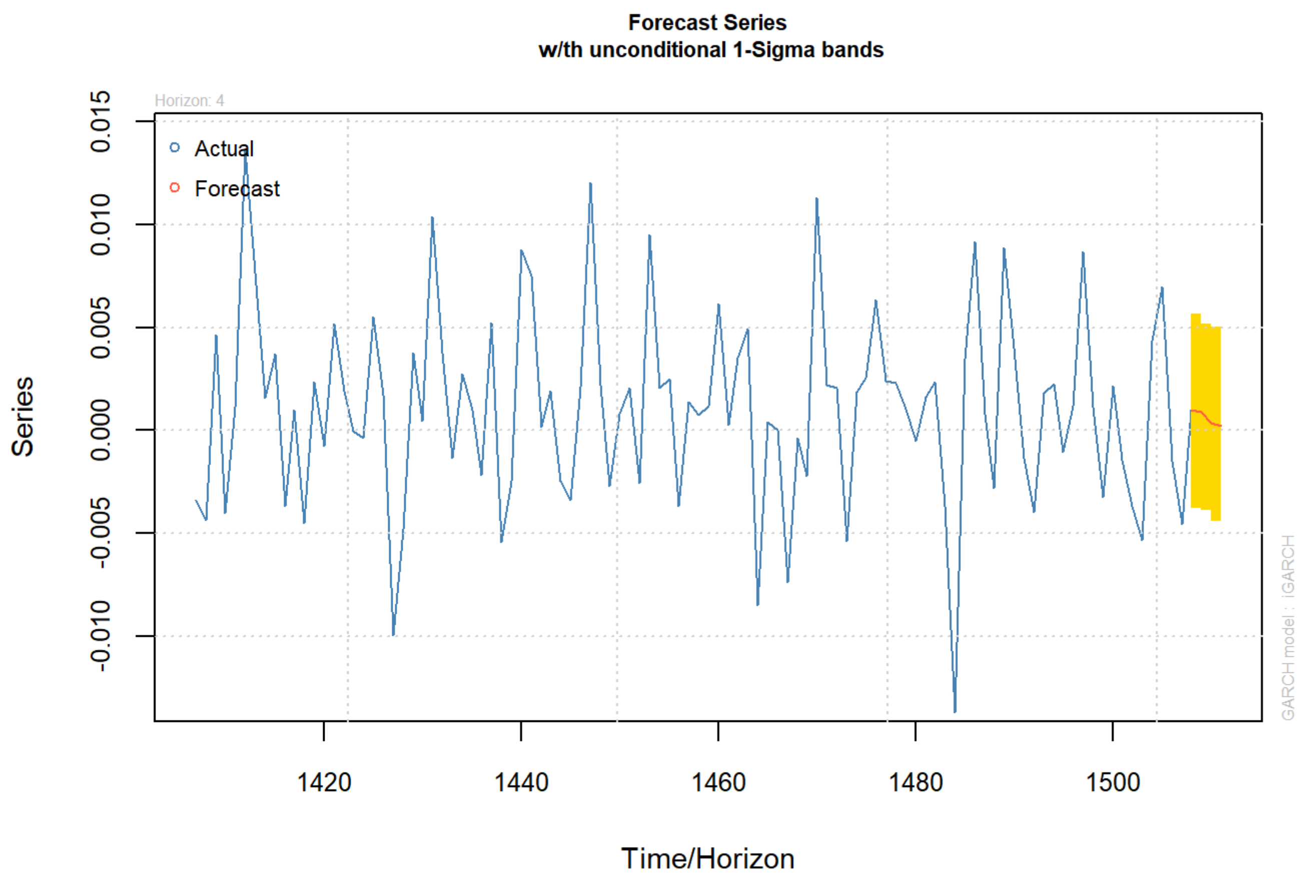
```
Box.test(stresi2^2,20,type="Ljung-Box",fitdf = 7) # p-value > 0.05, remains no ARCH effect
```

```
##
##  Box-Ljung test
##
## data:  stresi2^2
## X-squared = 17.973, df = 13, p-value = 0.1585
```

```
# d
forecast = ugarchforecast(m2, n.ahead = 4, data=log_returns_SP)
```

```
## Warning in `setfixed<-`(`*tmp*`, value = as.list(pars)): Unrecognized Parameter
## in Fixed Values: beta1...Ignored
```

```
plot(forecast, which = 1)
```



```
# this result looks more correct
```

```
U=forecast@forecast$seriesFor+1.96*forecast@forecast$sigmaFor
L=forecast@forecast$seriesFor-1.96*forecast@forecast$sigmaFor

forecast
```

```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: iGARCH
## Horizon: 4
## Roll Steps: 0
## Out of Sample: 0
##
## 0-roll forecast [T0=1507-01-01]:
##        Series    Sigma
## T+1 0.0009730 0.004727
## T+2 0.0008990 0.004753
## T+3 0.0003775 0.004779
## T+4 0.0002246 0.004804
```

```
c(L[1],U[1])
```

```
## [1] -0.00829238  0.01023845
```

3.

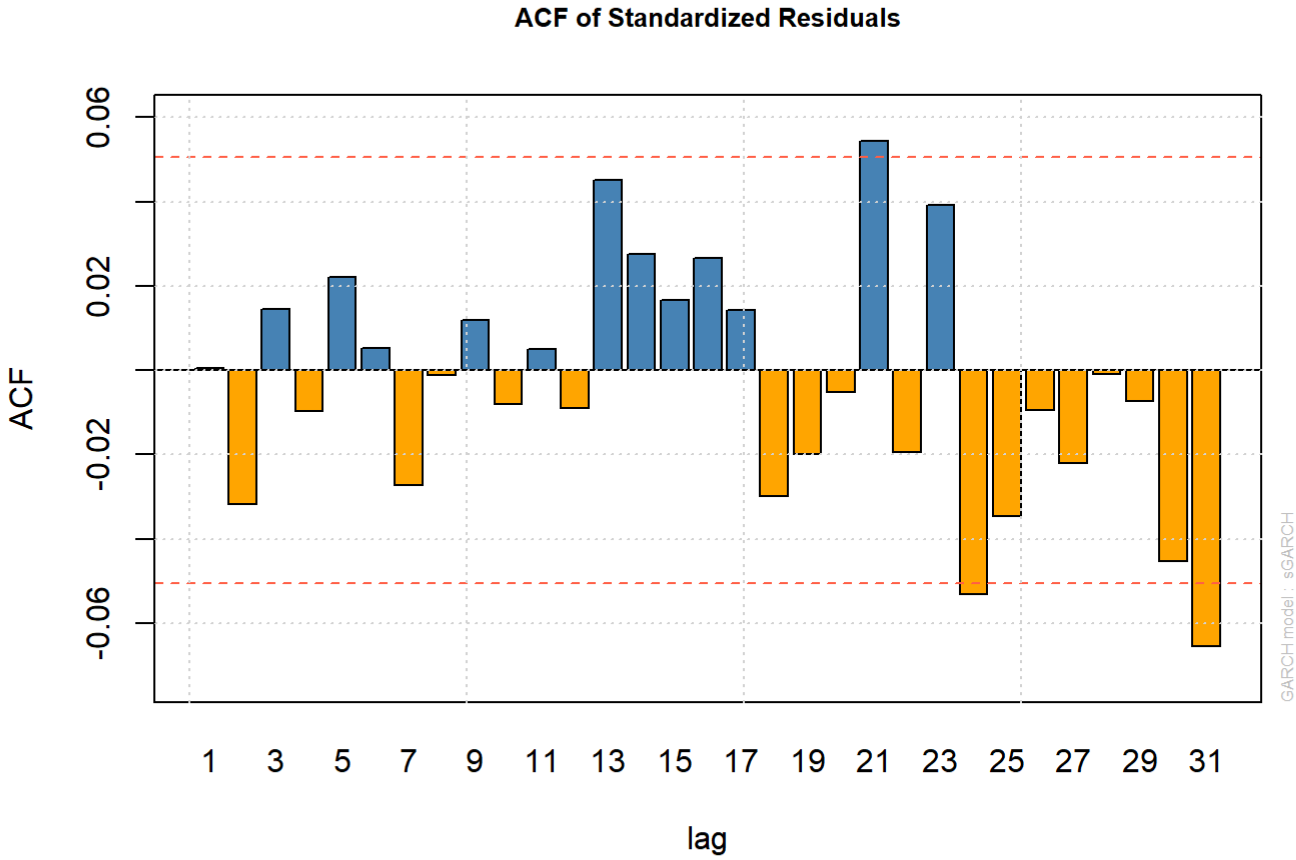
```
######################### 3
# a
# fit an ARMA(1,0)-GARCH(1,1)-M model:
spec3=ugarchspec(variance.model=list(model="sGARCH",garchOrder = c(1, 1)),
                 mean.model=list(armaOrder=c(1,0),include.mean = TRUE,archm=TRUE),
                 distribution.model = "norm")

m3=ugarchfit(spec=spec3,data=log_returns_SBUX)
m3  ### see output
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,1)
## Mean Model   : ARFIMA(1,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error   t value Pr(>|t|)
## mu      0.003756    0.001118    3.3590 0.000782
## ar1    -0.075778    0.026690   -2.8392 0.004522
## archm  -0.138842    0.055757   -2.4901 0.012770
## omega   0.000001    0.000001    1.9885 0.046754
## alpha1  0.018621    0.000912   20.4170 0.000000
## beta1   0.977339    0.000474 2061.4936 0.000000
##
## Robust Standard Errors:
##         Estimate  Std. Error    t value Pr(>|t|)
## mu      0.003756    0.001530    2.45570 0.014061
## ar1    -0.075778    0.029839   -2.53952 0.011101
## archm  -0.138842    0.074038   -1.87528 0.060754
## omega   0.000001    0.000003    0.49446 0.620980
## alpha1  0.018621    0.001523   12.22704 0.000000
## beta1   0.977339    0.000799 1223.27417 0.000000
##
## LogLikelihood : 3788.794
##
## Information Criteria
## ------------------------------------
##
## Akaike       -5.0203
## Bayes        -4.9991
## Shibata      -5.0203
## Hannan-Quinn -5.0124
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                   0.000231  0.9879
## Lag[2*(p+q)+(p+q)-1][2]  0.764245  0.8688
## Lag[4*(p+q)+(p+q)-1][5]  1.616238  0.8174
## d.o.f=1
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.0416  0.8384
## Lag[2*(p+q)+(p+q)-1][5]    0.7976  0.9035
## Lag[4*(p+q)+(p+q)-1][9]    1.8999  0.9166
```

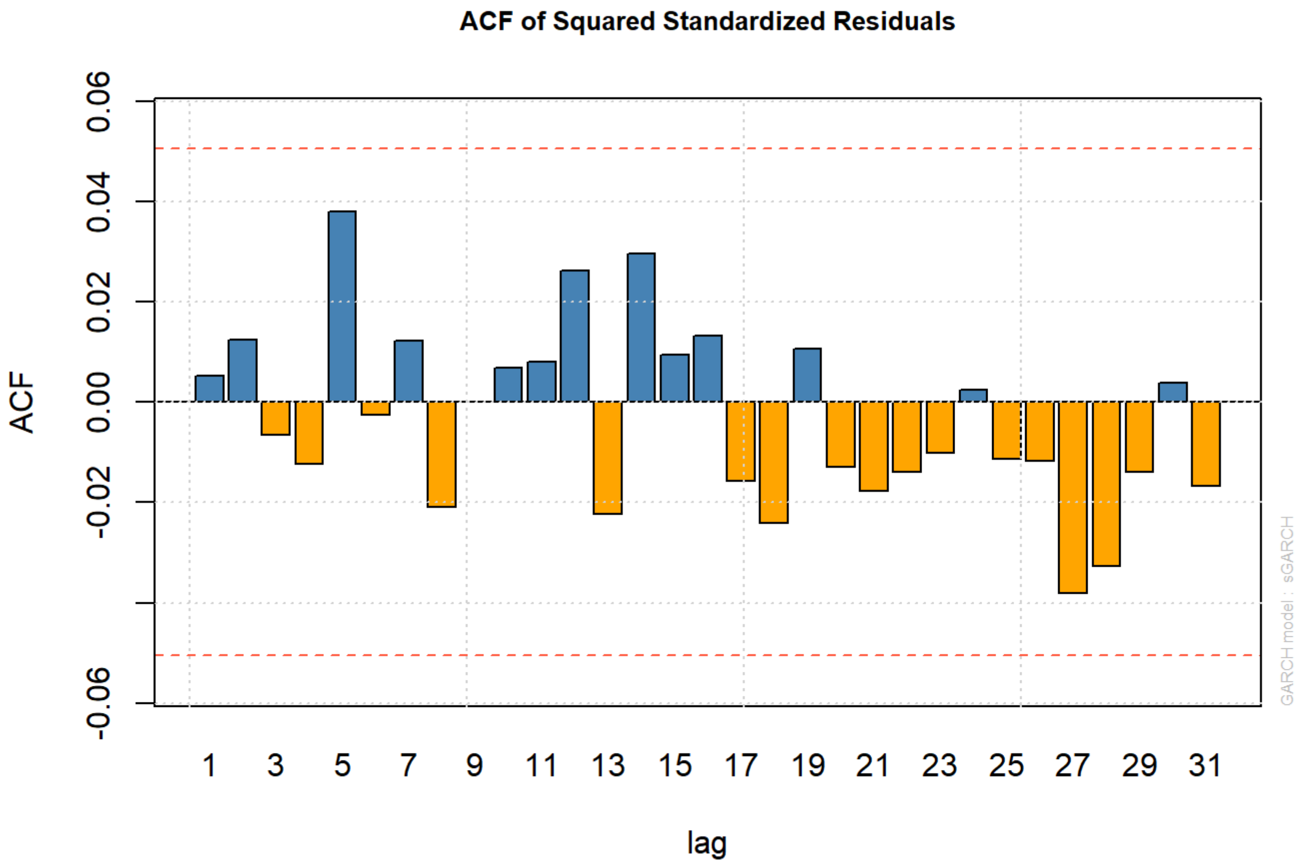
```
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]   0.06398 0.500 2.000  0.8003
## ARCH Lag[5]   1.55951 1.440 1.667  0.5768
## ARCH Lag[7]   1.92388 2.315 1.543  0.7334
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  61.8442
## Individual Statistics:
## mu     0.09723
## ar1    0.45248
## archm  0.08778
## omega  5.79637
## alpha1 0.14605
## beta1  0.11420
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:         1.49 1.68 2.12
## Individual Statistic:    0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias            1.202 0.2296
## Negative Sign Bias   0.122 0.9029
## Positive Sign Bias   1.094 0.2740
## Joint Effect         1.909 0.5915
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     73.66    2.240e-08
## 2    30     82.48    4.989e-07
## 3    40    111.61    6.381e-09
## 4    50    116.92    1.768e-07
##
##
## Elapsed time : 0.5440681
```

```
# model: ARMA(1,0)-GARCH(1,1)-M
# r_t = μ_t + c * (σ_t)^2 + a_t
# μ_t = μ_0 + φ_1 * r_t-1
# a_t = σ_t * ε_t
# (σ_t)^2 = α_0 + α_1*(a_t-1)^2 + β_1*(σ_t-1)^2
# where μ_0 = 0.003756, φ_1 = -0.075778, c = -0.138842, α_0 = 0, α_1 = 0.018621, β_1 = 0.977339
```

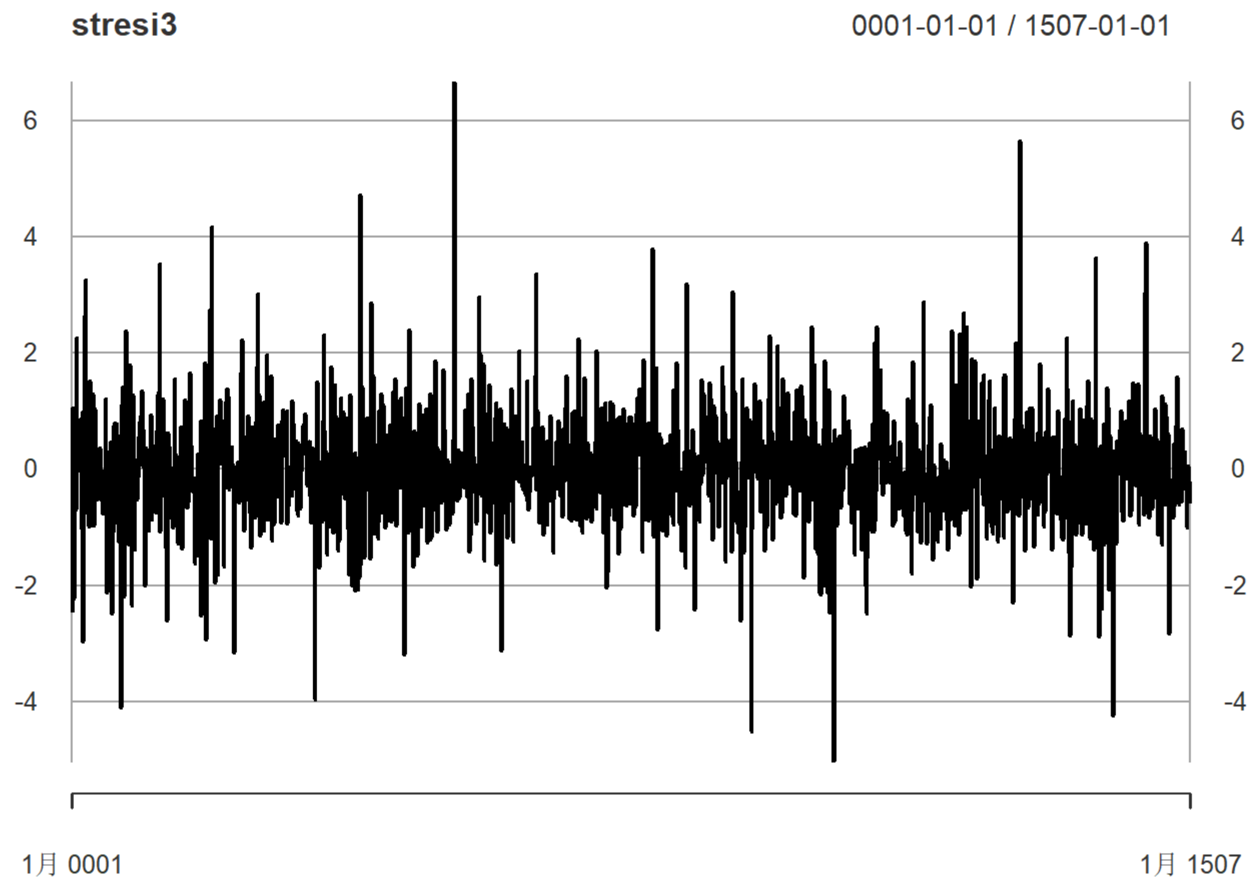
```
plot(m3, which = 10)
```




```
plot(m3, which = 11)
```



```
stresi3=residuals(m3,standardize=T)
plot(stresi3,type="l")
```




```
Box.test(stresi3,10,type="Ljung-Box",fitdf = 4) # p-value > 0.05, white noise
```

```
##
##  Box-Ljung test
##
## data:  stresi3
## X-squared = 4.2112, df = 6, p-value = 0.6481
```

```
Box.test(stresi3^2,10,type="Ljung-Box",fitdf = 4) # p-value > 0.05, remains no ARCH effect
```

```
##
##  Box-Ljung test
##
## data:  stresi3^2
## X-squared = 3.7403, df = 6, p-value = 0.7118
```

```
# b
# c = -0.138842, p-value of t-test is 0.012770 < 0.05, so the ARCH-in-mean parameter is signifi
cant.
```

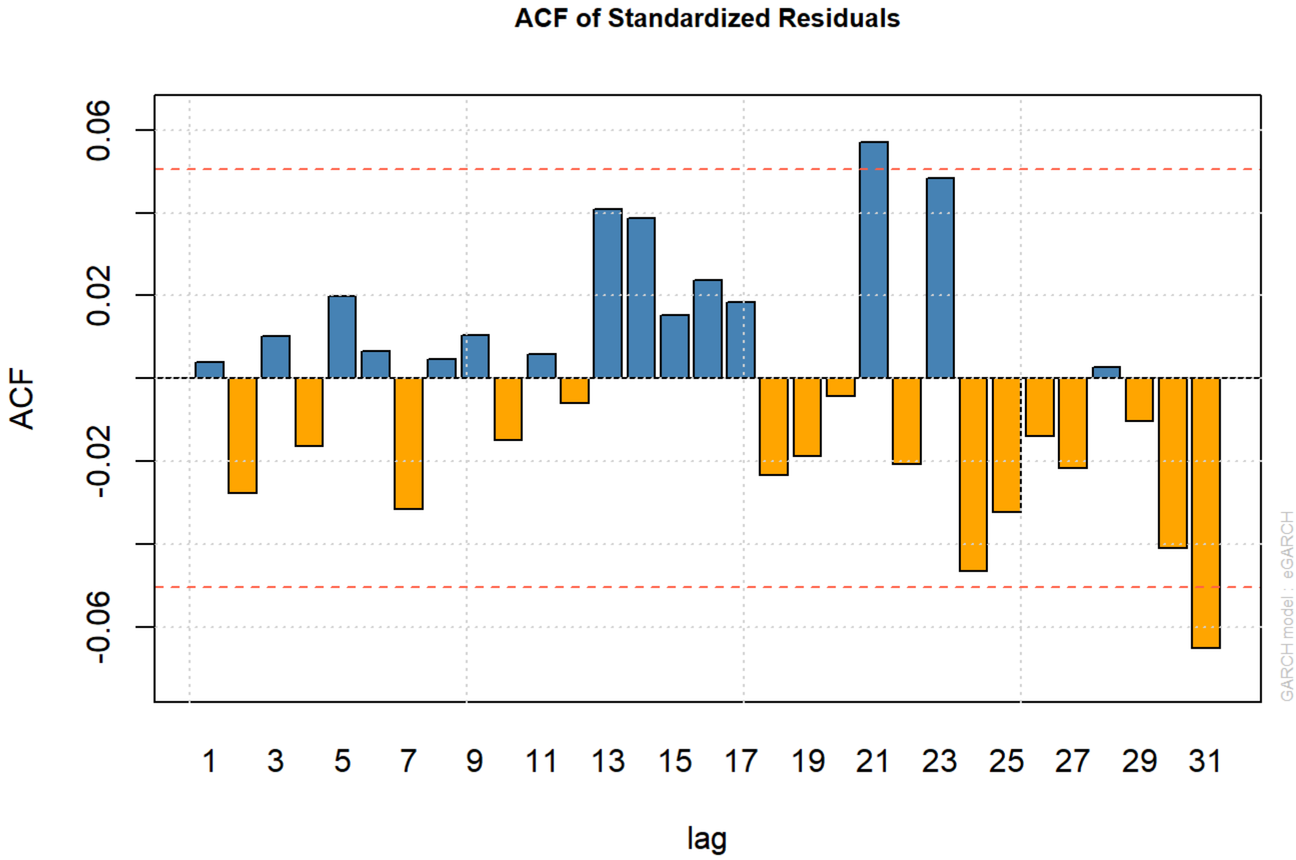
```
# c
# fit an ARMA(1,0)-EGARCH(1,1) model:
spec3_2=ugarchspec(variance.model=list(model="eGARCH",garchOrder = c(1, 1)),
                   mean.model=list(armaOrder=c(1,0),include.mean = TRUE),
                   distribution.model = "norm")

m3_2=ugarchfit(spec=spec3_2,data=log_returns_SBUX)
m3_2  ### see output
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : eGARCH(1,1)
## Mean Model   : ARFIMA(1,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error    t value Pr(>|t|)
## mu      0.000936    0.000362     2.5885 0.009638
## ar1    -0.076938    0.018071    -4.2575 0.000021
## omega  -0.048337    0.001346   -35.9105 0.000000
## alpha1 -0.036930    0.003549   -10.4057 0.000000
## beta1   0.993617    0.000016 60794.0465 0.000000
## gamma1  0.045562    0.010480     4.3477 0.000014
##
## Robust Standard Errors:
##         Estimate  Std. Error    t value Pr(>|t|)
## mu      0.000936    0.000335     2.7940 0.005207
## ar1    -0.076938    0.013708    -5.6127 0.000000
## omega  -0.048337    0.001736   -27.8399 0.000000
## alpha1 -0.036930    0.005382    -6.8618 0.000000
## beta1   0.993617    0.000026 38362.2387 0.000000
## gamma1  0.045562    0.017305     2.6329 0.008466
##
## LogLikelihood : 3807.35
##
## Information Criteria
## ------------------------------------
##
## Akaike       -5.0449
## Bayes        -5.0238
## Shibata      -5.0450
## Hannan-Quinn -5.0370
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.02159  0.8832
## Lag[2*(p+q)+(p+q)-1][2]   0.60771  0.9331
## Lag[4*(p+q)+(p+q)-1][5]   1.33481  0.8849
## d.o.f=1
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                   0.008649  0.9259
## Lag[2*(p+q)+(p+q)-1][5]  0.664211  0.9296
## Lag[4*(p+q)+(p+q)-1][9]  1.798836  0.9276
```

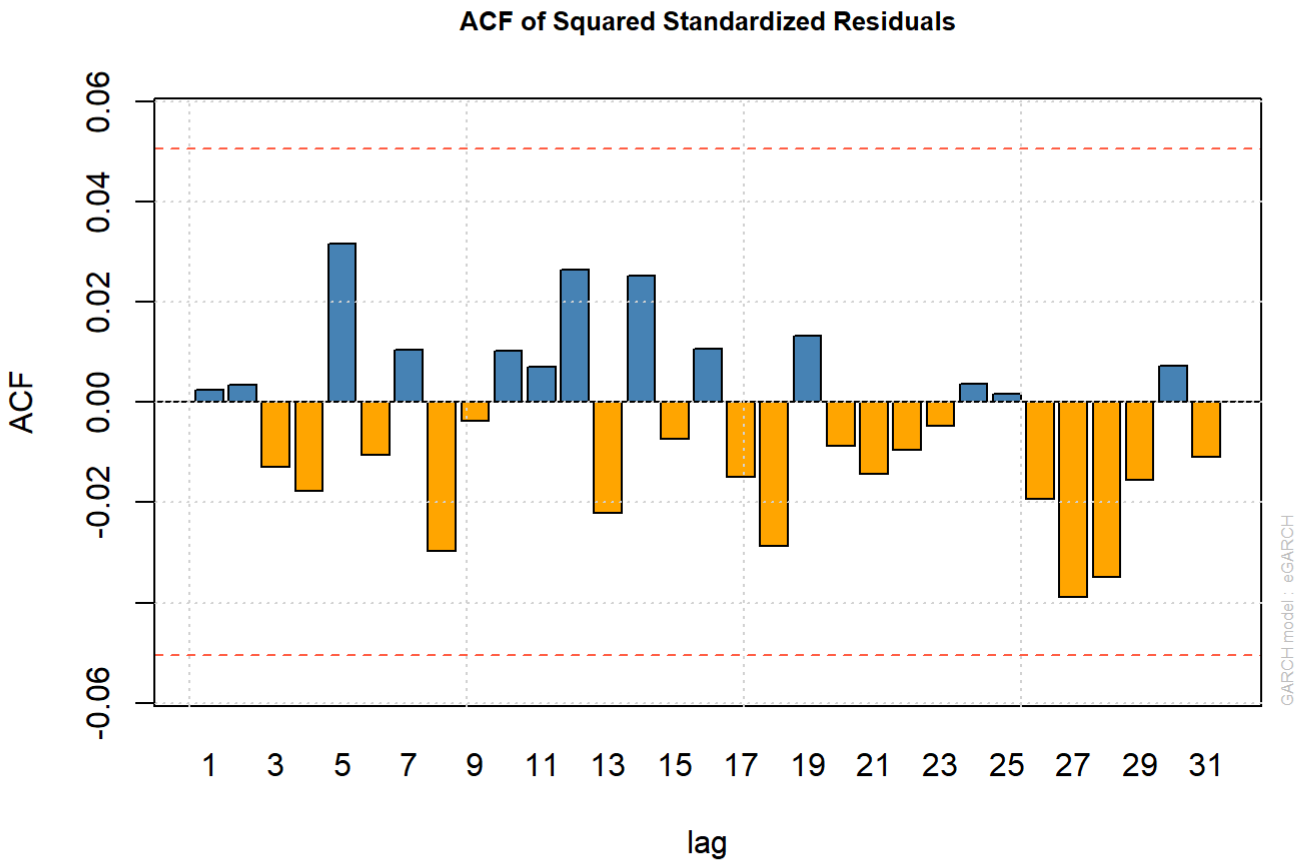
```
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]    0.2501 0.500 2.000  0.6170
## ARCH Lag[5]    1.5258 1.440 1.667  0.5856
## ARCH Lag[7]    1.8877 2.315 1.543  0.7411
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  1.1961
## Individual Statistics:
## mu     0.02588
## ar1    0.53201
## omega  0.08153
## alpha1 0.28379
## beta1  0.07343
## gamma1 0.09373
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.49 1.68 2.12
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias            1.275 0.2025
## Negative Sign Bias   0.858 0.3910
## Positive Sign Bias   1.437 0.1510
## Joint Effect         2.827 0.4191
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     58.69    6.217e-06
## 2    30     77.55    2.634e-06
## 3    40     91.96    3.624e-06
## 4    50     97.94    4.125e-05
##
##
## Elapsed time : 0.3321121
```

```
# model: ARMA(1,0)-EGARCH(1,1)
# r_t = μ_t + a_t
# μ_t = μ_0 + φ_1 * r_t-1
# a_t = σ_t * ε_t
# ln[(σ_t)^2] = α_0 + [α_1*(ε_t-1) + γ_1(|ε_t-1| - E|ε_t-1|)] + β_1*ln[(σ_t-1)^2]
# where μ_0 = 0.000936, φ_1 = -0.076938, α_0 = -0.048337, α_1 = -0.036930, β_1 = 0.993617,
γ_1 = 0.045562
```

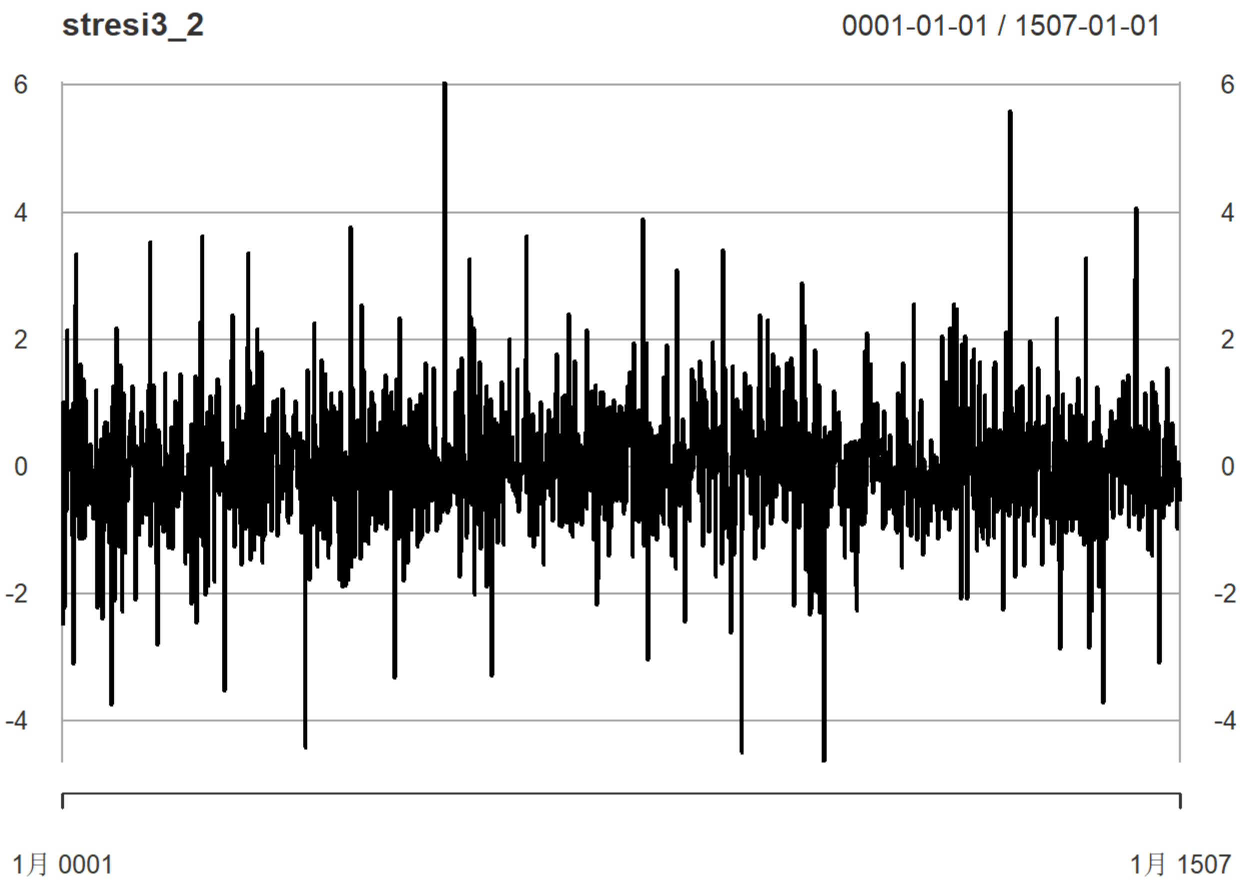
```
plot(m3_2, which = 10)
```




```
plot(m3_2, which = 11)
```



```
stresi3_2=residuals(m3_2,standardize=T)
plot(stresi3_2,type="l")
```




```
Box.test(stresi3_2,10,type="Ljung-Box",fitdf = 4) # p-value > 0.05, white noise
```

```
##
##  Box-Ljung test
##
## data:  stresi3_2
## X-squared = 4.4807, df = 6, p-value = 0.6119
```

```
Box.test(stresi3_2^2,10,type="Ljung-Box",fitdf = 4) # p-value > 0.05, remains no ARCH effect
```

```
##
##  Box-Ljung test
##
## data:  stresi3_2^2
## X-squared = 4.1032, df = 6, p-value = 0.6627
```

```
# d
# α_1 = -0.036930, p-value of t-test is 0.000000 < 0.05, so the leverage parameter is significant.
```

```
# leverage = α_1/γ_1 = -0.036930/0.045562
```

4.

```
######################### 4

df2 = read.table("C://Users//张铭韬//Desktop//学业//港科大//MSDM5053时间序列//作业//assignment3//m-pg5606.txt",header=F)
#  Convert the simple returns into percentage log returns
log_returns_PG = log(1 + ts(df2$V2))
```

```
# a
# Stationarity test
ndiffs(log_returns_PG) # d=0
```

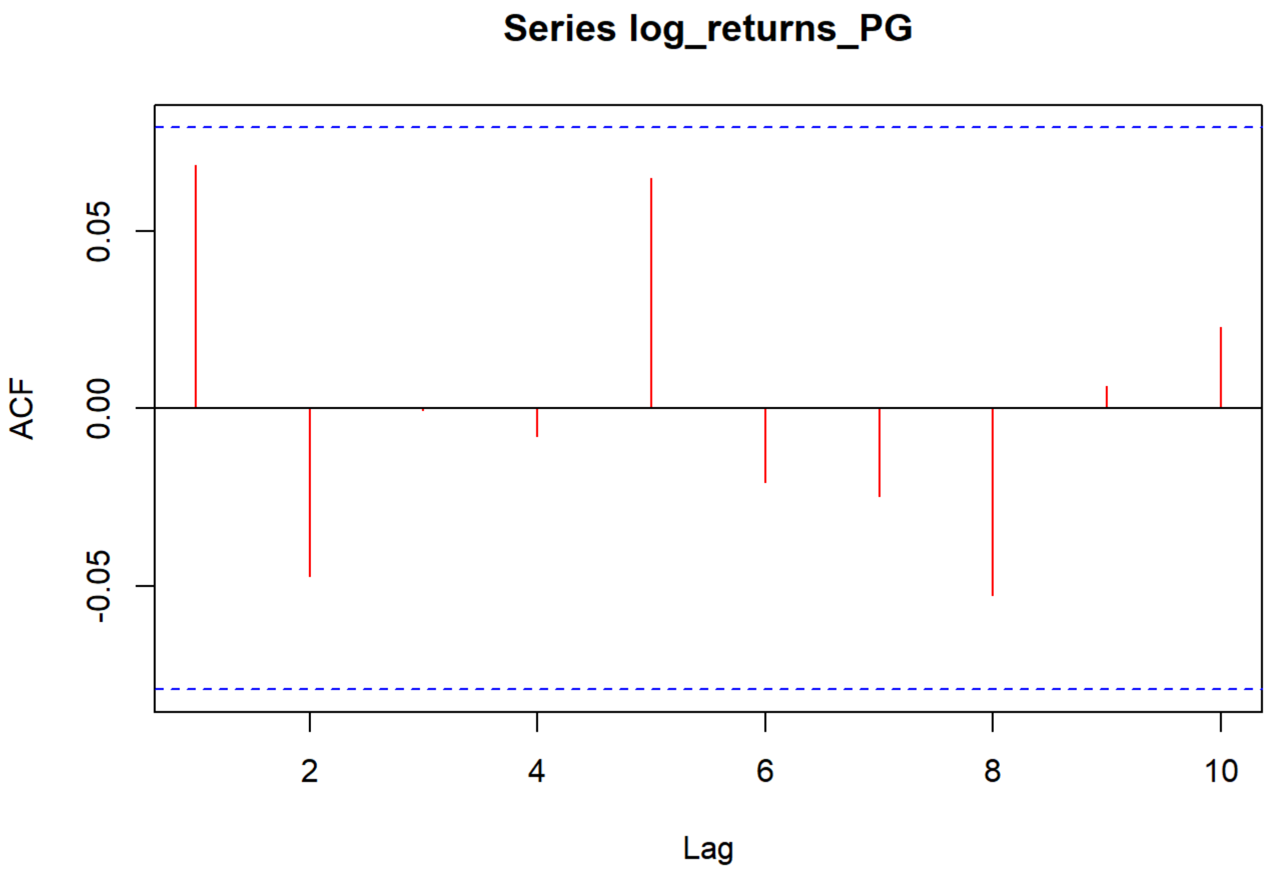
```
## [1] 0
```

```
pp.test(log_returns_PG) # p-value < 0.05, reject H0, stationary
```

```
## Warning in pp.test(log_returns_PG): p-value smaller than printed p-value
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  log_returns_PG
## Dickey-Fuller Z(alpha) = -556.79, Truncation lag parameter = 6, p-value
## = 0.01
## alternative hypothesis: stationary
```

```
# white noise test
acf(log_returns_PG, lag.max = 10, col="red")
```

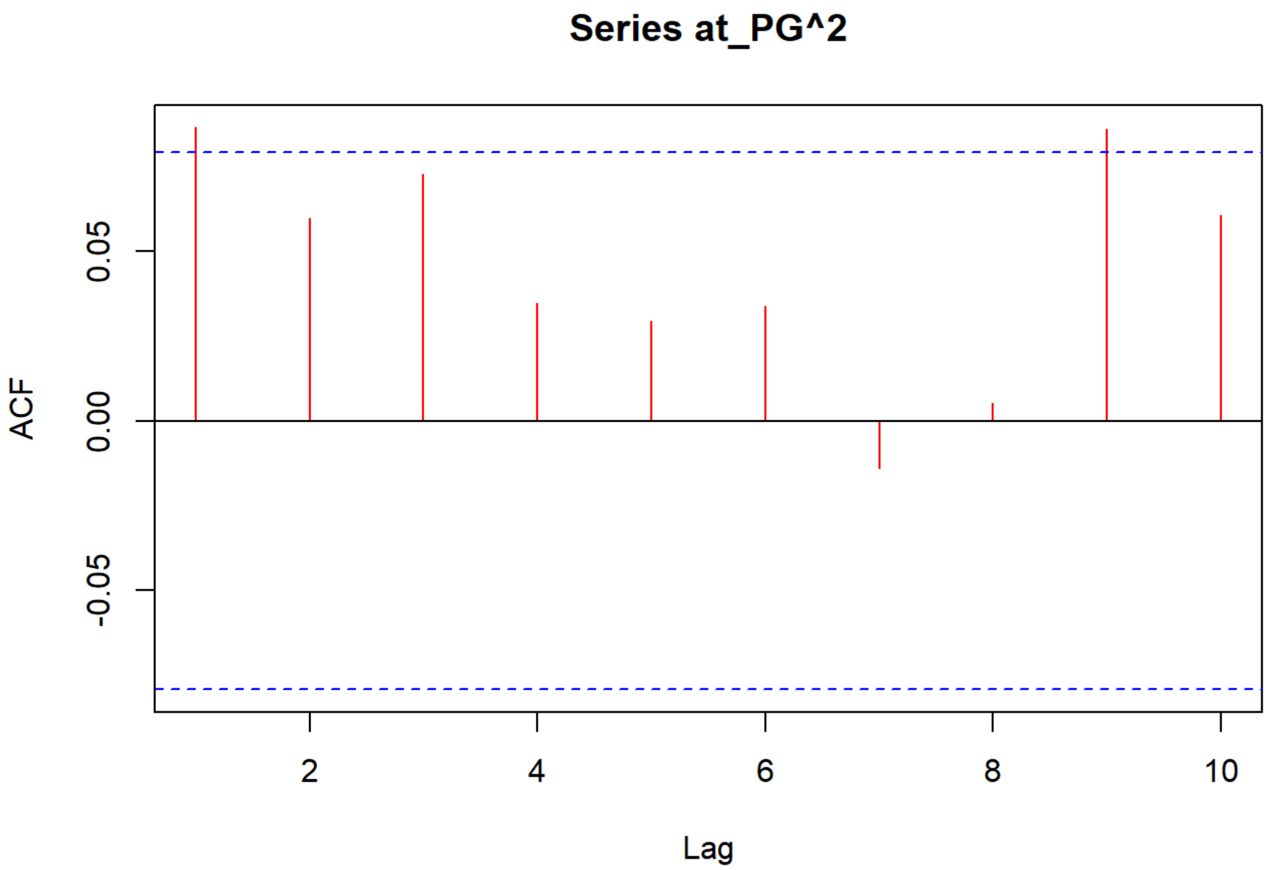



```
Box.test(log_returns_PG,lag=10,type="Ljung-Box")
```

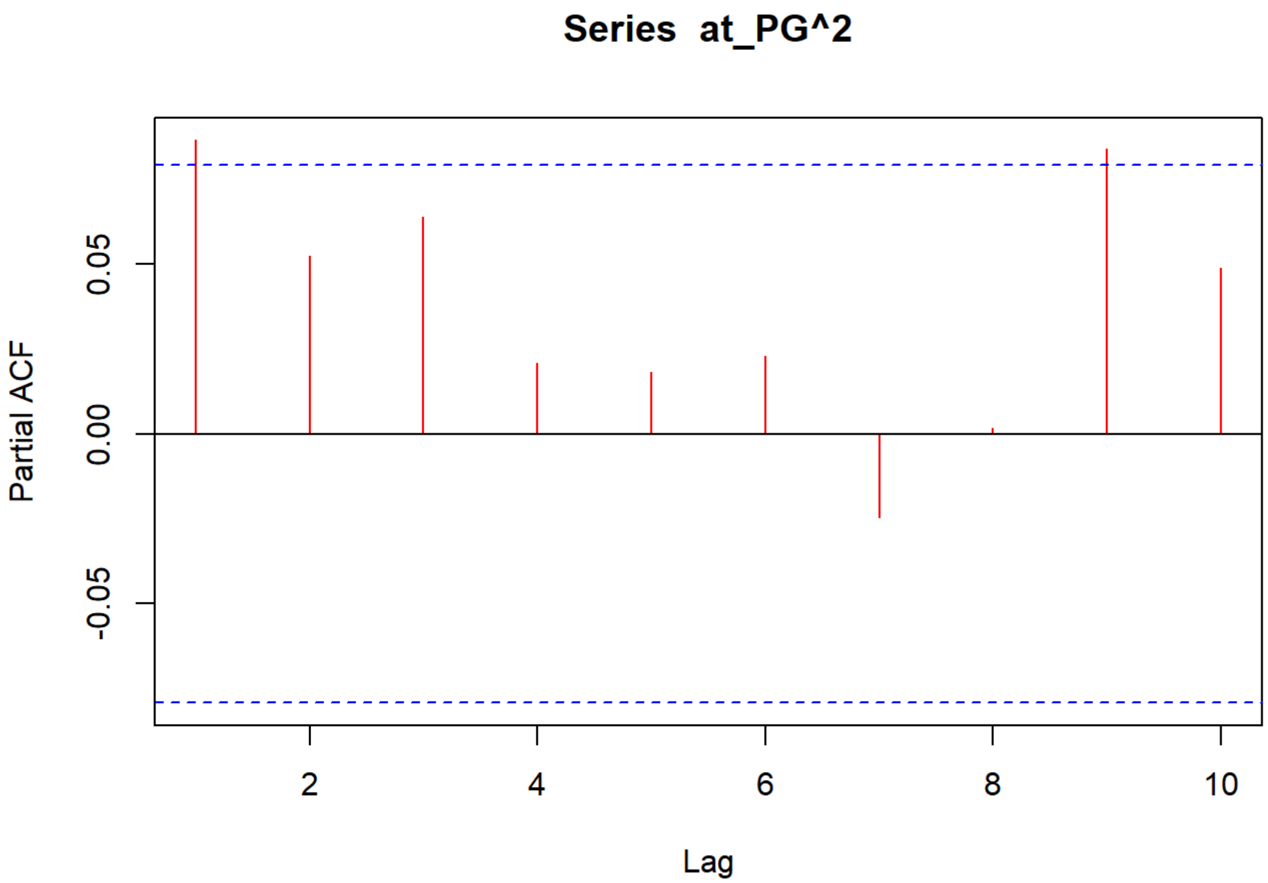
```
##
##  Box-Ljung test
##
## data:  log_returns_PG
## X-squared = 9.65, df = 10, p-value = 0.4717
```

```
# p-value > 0.05, there doesn't exist any serial correlation in the log returns of PG data
```

```
# ARCH test
at_PG=log_returns_PG-mean(log_returns_PG)
acf(at_PG^2, lag.max = 10, col="red")
```




```
pacf(at_PG^2, lag.max = 10, col="red")
```



```
Box.test(at_PG^2,lag=10,type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  at_PG^2
## X-squared = 19.116, df = 10, p-value = 0.03882
```

```
# p-value < 0.05, there exists ARCH effect in the log returns of S&P index
```

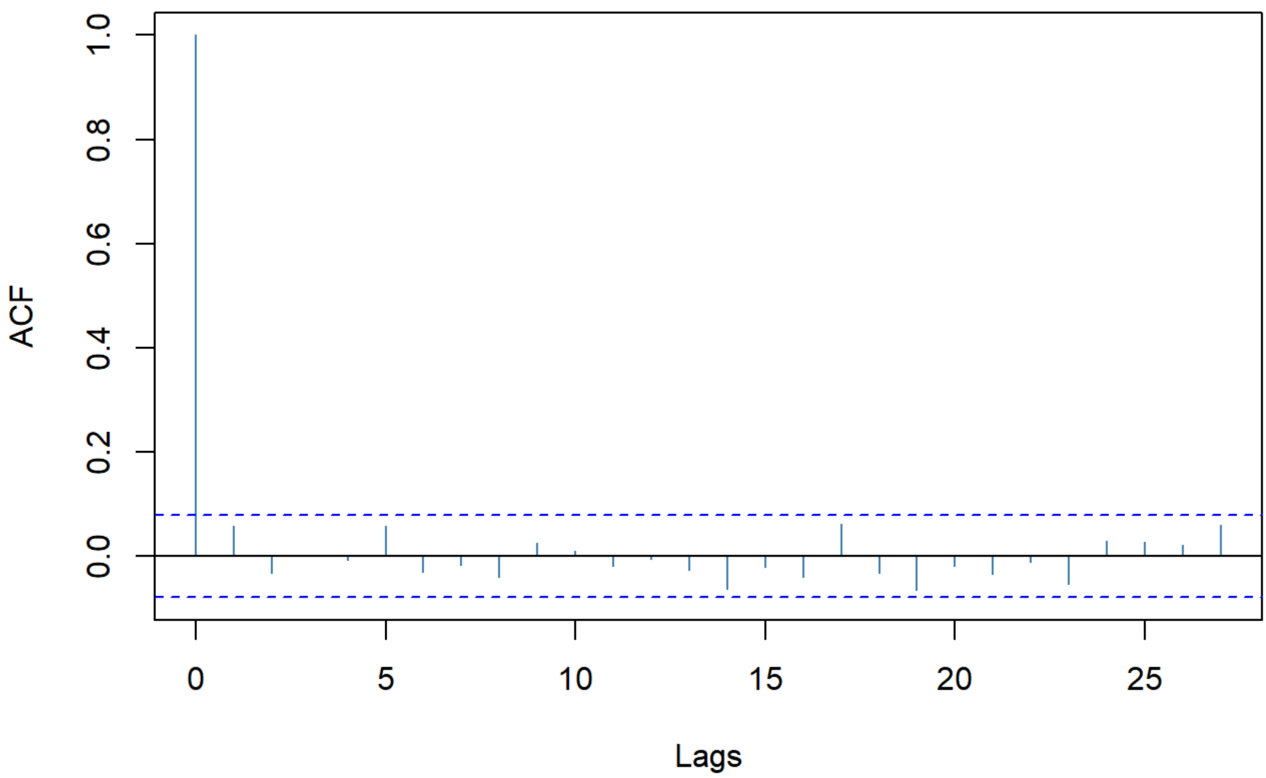
```
# b
m4=garchFit(log_returns_PG~garch(1,1),data=log_returns_PG,trace=F,cond.dist="norm")
summary(m4)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = log_returns_PG ~ garch(1, 1), data = log_returns_PG,
##     cond.dist = "norm", trace = F)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x0000000030834d58>
##  [data = log_returns_PG]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 8.5624e-03  8.5366e-05  9.6309e-02  8.6240e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     8.562e-03   1.580e-03    5.419    6e-08 ***
## omega  8.537e-05   3.921e-05    2.177 0.029450 *
## alpha1 9.631e-02   2.697e-02    3.571 0.000355 ***
## beta1  8.624e-01   3.133e-02   27.526  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  1074.417    normalized:  1.755583
##
## Description:
##  Sat Apr 20 20:24:28 2024 by user: 张铭韬
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  312.9021  0
##  Shapiro-Wilk Test  R    W      0.9616422 1.506883e-11
##  Ljung-Box Test     R    Q(10)  7.001382  0.7253145
##  Ljung-Box Test     R    Q(15)  10.47682  0.7887243
##  Ljung-Box Test     R    Q(20)  17.58848  0.6144965
##  Ljung-Box Test     R^2  Q(10)  4.061319  0.9445384
##  Ljung-Box Test     R^2  Q(15)  5.256527  0.9897226
##  Ljung-Box Test     R^2  Q(20)  6.492114  0.9980483
##  LM Arch Test       R    TR^2   4.234372  0.9788261
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -3.498094 -3.469226 -3.498178 -3.486866
```

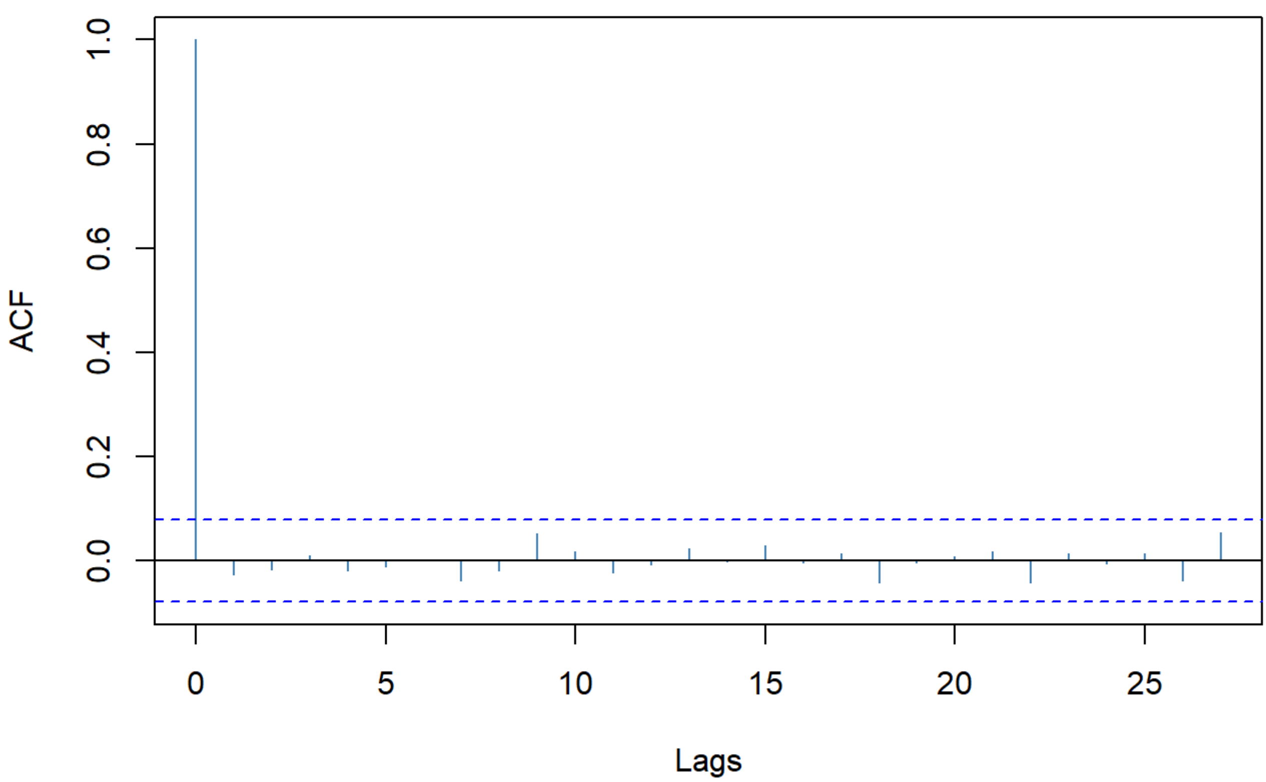
```
# All coefficients are significant.
# model: GARCH(1,1)
# r_t = μ_t + a_t
#  μ_t = μ_0
# a_t = σ_t * ε_t
# (σ_t)^2 = α_0 + α_1*(a_t-1)^2 + β_1*(σ_t-1)^2
# where μ_0 = 8.562e-03, α_0 = 8.537e-05, α_1 = 9.631e-02, β_1 = 8.624e-01
```

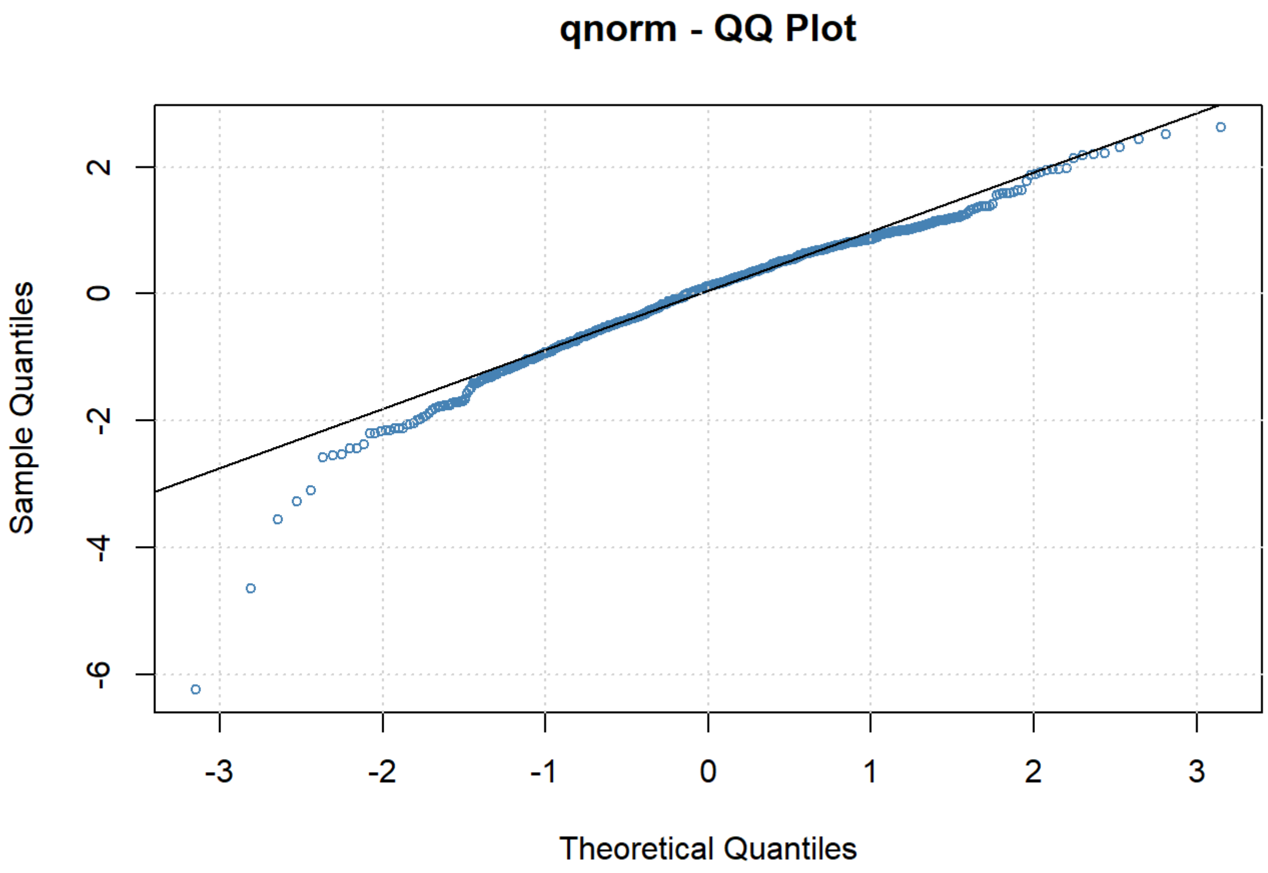
```
plot(m4, which = c(10,11,13))
```

ACF of Standardized Residuals

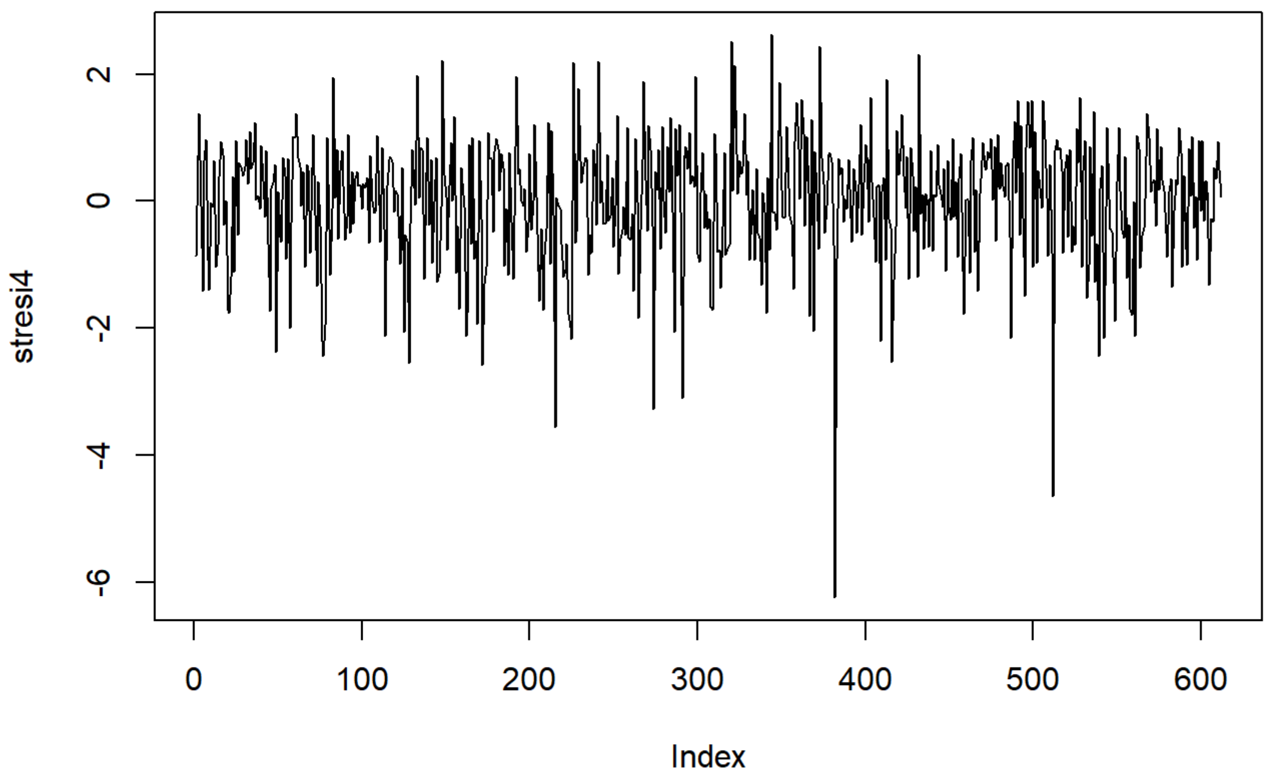


ACF of Squared Standardized Residuals






```
stresi4=residuals(m4,standardize=T)
plot(stresi4,type="l")
```



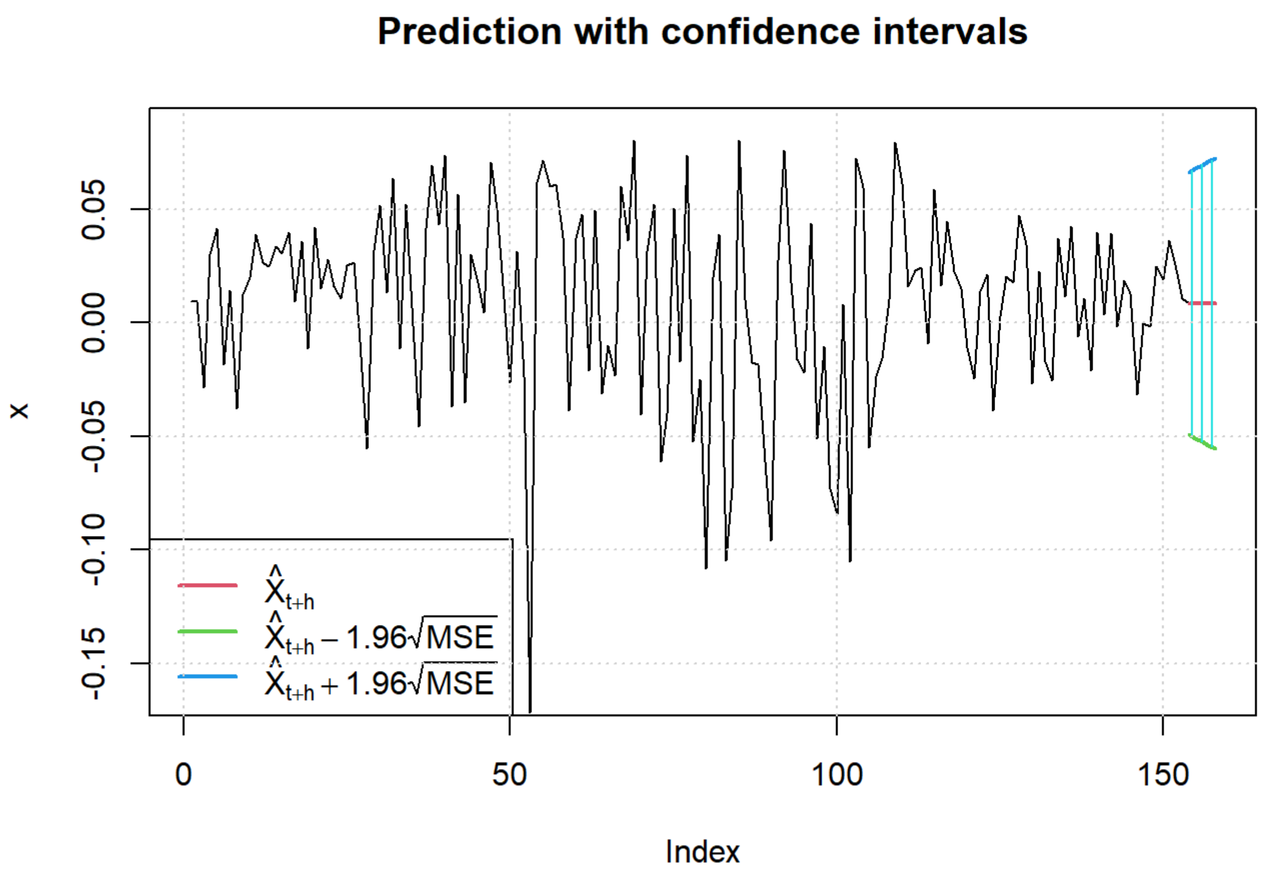
```
Box.test(stresi4,10,type="Ljung-Box",fitdf = 2) # p-value > 0.05, white noise
```

```
##
##  Box-Ljung test
##
## data:  stresi4
## X-squared = 7.0014, df = 8, p-value = 0.5365
```

```
Box.test(stresi4^2,10,type="Ljung-Box",fitdf = 2) # p-value > 0.05, remains no ARCH effect
```

```
##
##  Box-Ljung test
##
## data:  stresi4^2
## X-squared = 4.0613, df = 8, p-value = 0.8515
```

```
# c
predict(m4, n.ahead = 5, trace = FALSE, mse = c("cond","uncond"), plot=TRUE, nx=NULL, crit_val=
NULL, conf=NULL)
```



```
##   meanForecast  meanError standardDeviation lowerInterval upperInterval
## 1  0.008562431 0.02952353        0.02952353   -0.04930263    0.06642749
## 2  0.008562431 0.03034827        0.03034827   -0.05091908    0.06804394
## 3  0.008562431 0.03111843        0.03111843   -0.05242858    0.06955344
## 4  0.008562431 0.03183931        0.03183931   -0.05384148    0.07096634
## 5  0.008562431 0.03251543        0.03251543   -0.05516664    0.07229150
```

```
# 1 step interval:
c(-0.04930263,0.06642749)
```

```
## [1] -0.04930263  0.06642749
```

5.

```
########################## 5
df3 = read.table("C://Users//张铭韬//Desktop//学业//港科大//MSDM5053时间序列//作业//assignment 3//d-exuseu.txt",header=F)
#  Convert the simple returns into percentage log returns
log_returns_df3 = log(1 + ts(df3$V4))
```

```
# a
# Stationarity test
ndiffs(log_returns_df3) # d=1
```

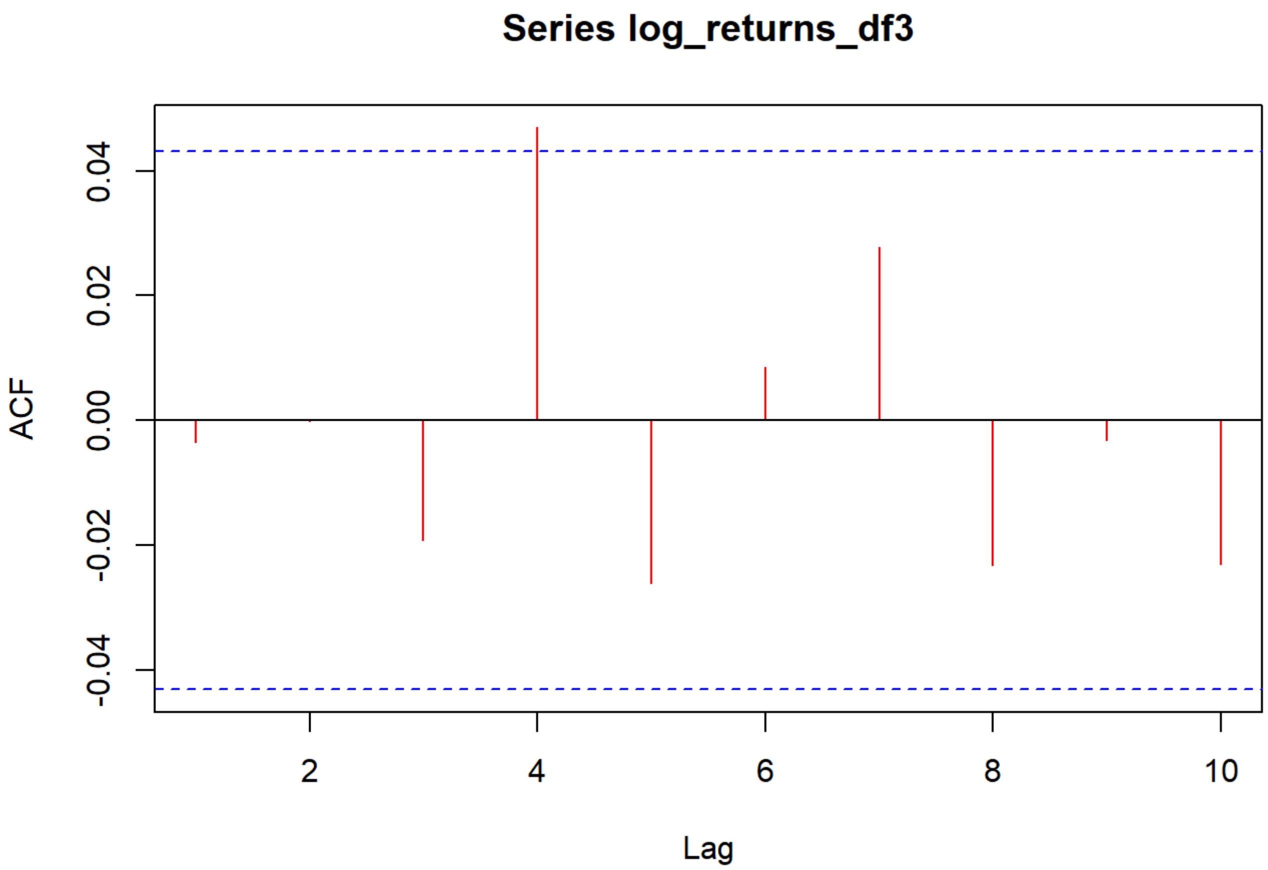
```
## [1] 1
```

```
log_returns_df3=diff(log_returns_df3)
pp.test(log_returns_PG) # p-value < 0.05, reject H0, stationary
```

```
## Warning in pp.test(log_returns_PG): p-value smaller than printed p-value
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  log_returns_PG
## Dickey-Fuller Z(alpha) = -556.79, Truncation lag parameter = 6, p-value
## = 0.01
## alternative hypothesis: stationary
```

```
# white noise test
acf(log_returns_df3, lag.max = 10, col="red")
```

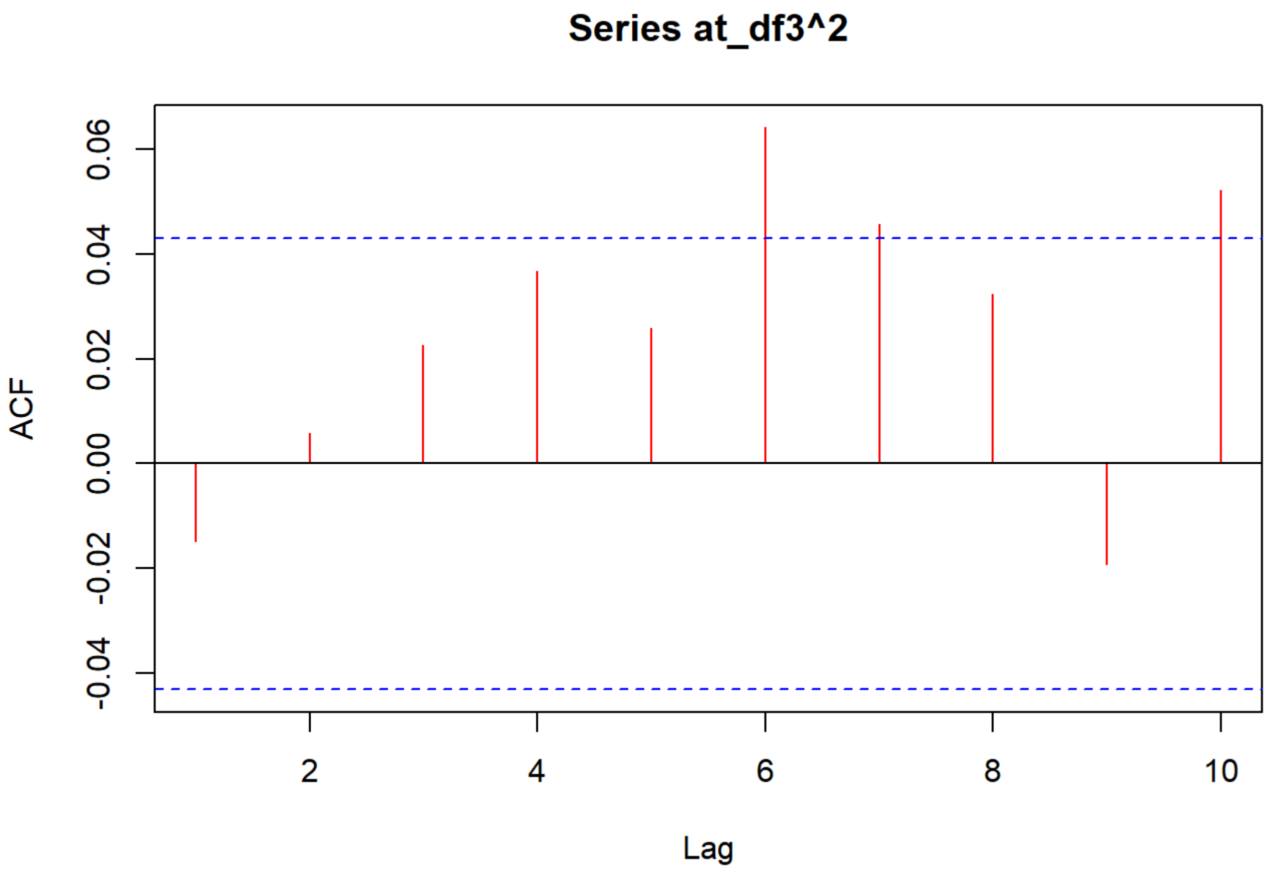



```
Box.test(log_returns_df3,lag=10,type="Ljung-Box")
```

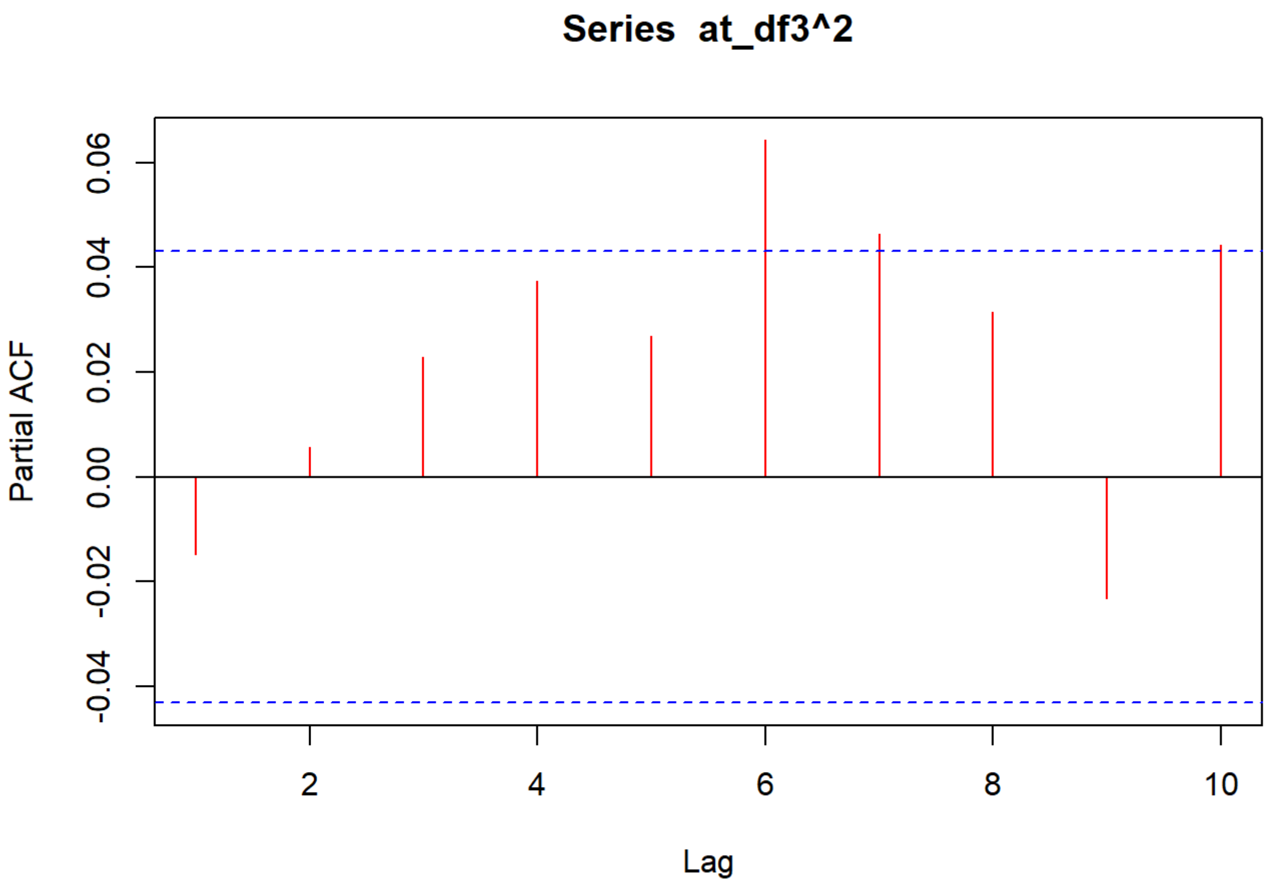
```
##
##  Box-Ljung test
##
## data:  log_returns_df3
## X-squared = 10.762, df = 10, p-value = 0.3764
```

```
# p-value > 0.05, there doesn't exist any serial correlation in the log returns of df3 data
```

```
# b
# ARCH test
at_df3=log_returns_df3-mean(log_returns_df3)
acf(at_df3^2, lag.max = 10, col="red")
```




```
pacf(at_df3^2, lag.max = 10, col="red")
```



```
Box.test(at_df3^2,lag=10,type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  at_df3^2
## X-squared = 27.316, df = 10, p-value = 0.002321
```

```
# p-value < 0.05, there exists ARCH effect in the log returns of df3 data
```

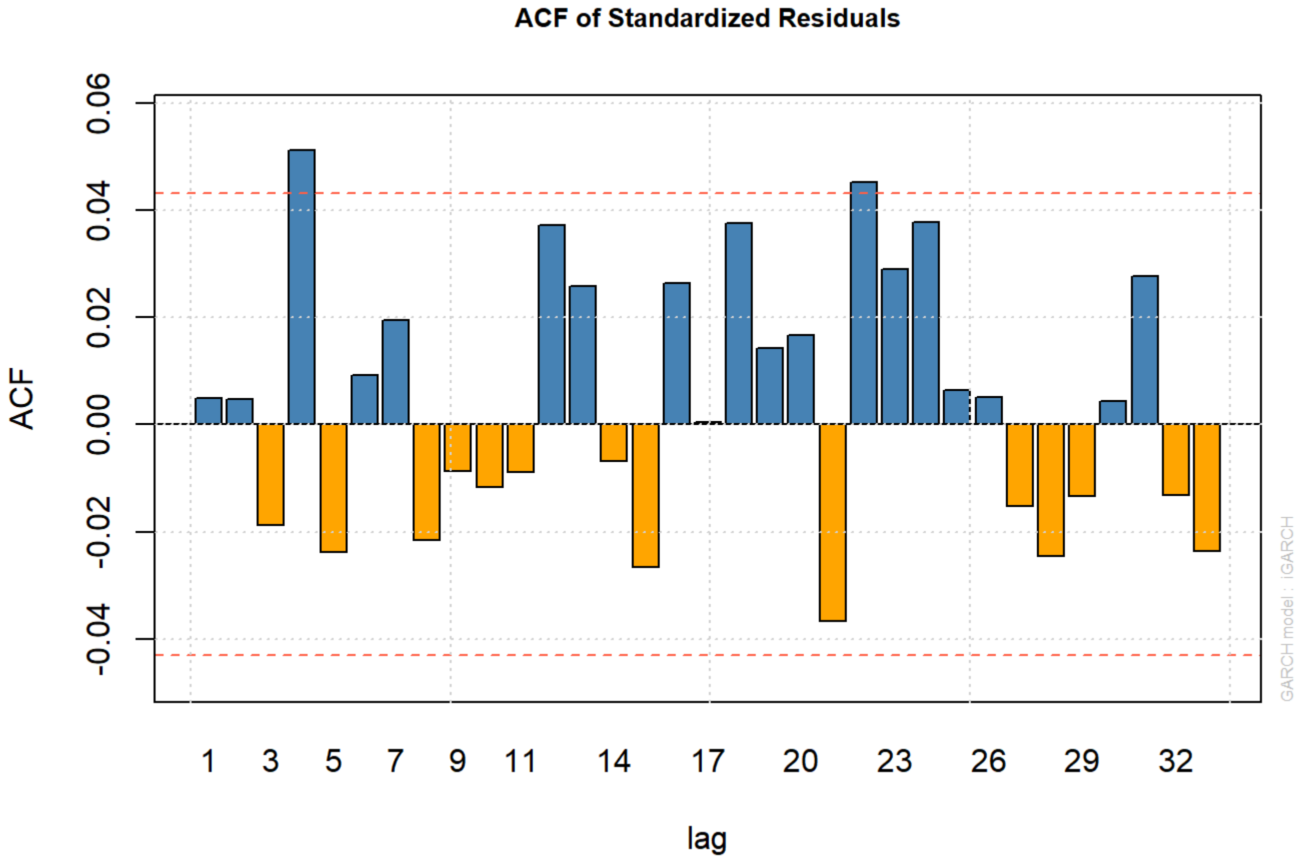
```
# c
spec5=ugarchspec(variance.model=list(model="iGARCH",garchOrder = c(1, 1)),
                 mean.model=list(armaOrder=c(0,0),include.mean = TRUE),
                 distribution.model = "norm")

m5=ugarchfit(spec=spec5,data=log_returns_df3)
m5  ### see output
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : iGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000056    0.000067  0.83829  0.40187
## omega   0.000000    0.000000  0.00932  0.99256
## alpha1  0.017439    0.002283  7.63752  0.00000
## beta1   0.982561          NA       NA       NA
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000056    0.000121 0.465025  0.64191
## omega   0.000000    0.000002 0.000411  0.99967
## alpha1  0.017439    0.111431 0.156502  0.87564
## beta1   0.982561          NA       NA       NA
##
## LogLikelihood : 8991.58
##
## Information Criteria
## ------------------------------------
##
## Akaike       -8.7099
## Bayes        -8.7017
## Shibata      -8.7099
## Hannan-Quinn -8.7069
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.05038  0.8224
## Lag[2*(p+q)+(p+q)-1][2]   0.07362  0.9386
## Lag[4*(p+q)+(p+q)-1][5]   2.94004  0.4181
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                      2.305  0.1289
## Lag[2*(p+q)+(p+q)-1][5]     3.750  0.2869
## Lag[4*(p+q)+(p+q)-1][9]     5.620  0.3442
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
```

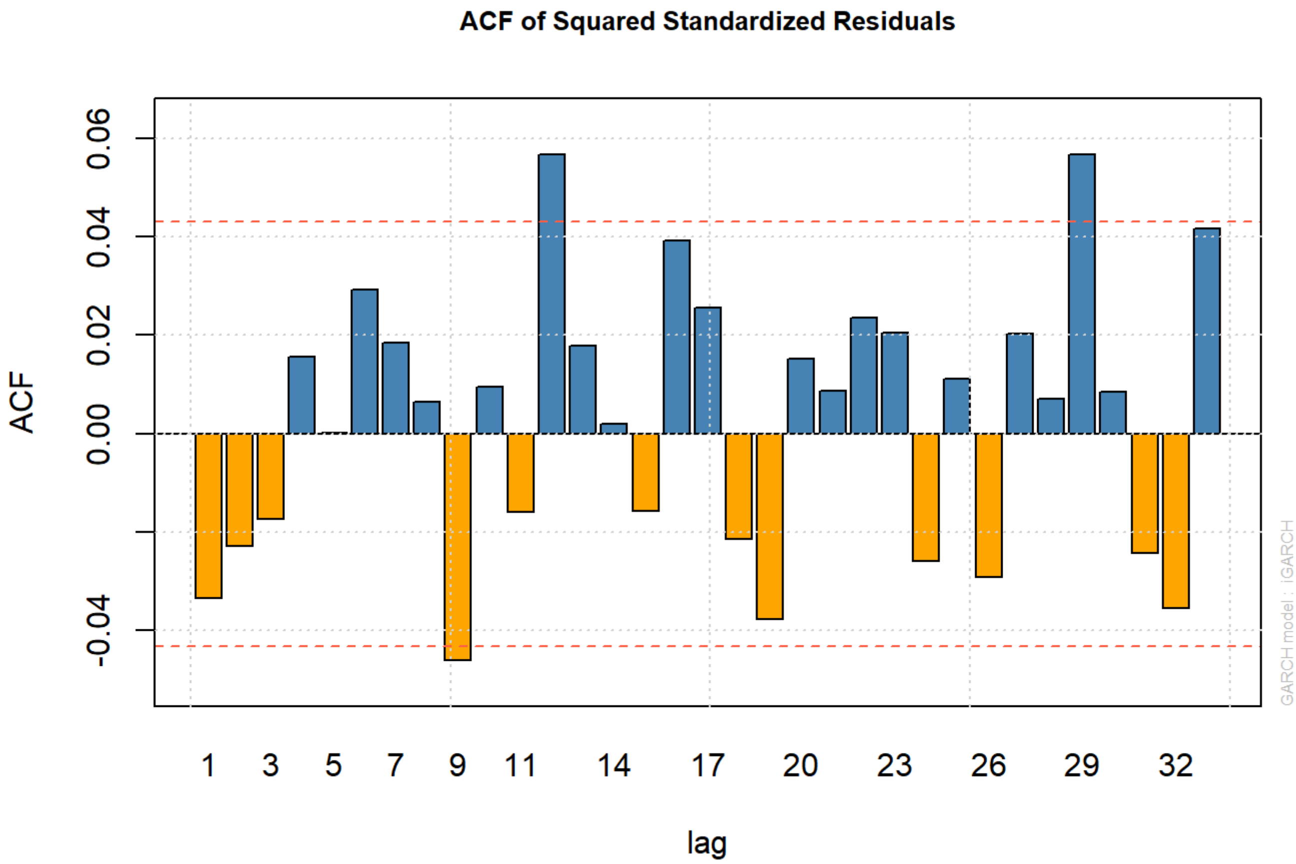
```
##             Statistic Shape Scale P-Value
## ARCH Lag[3]    0.6261 0.500 2.000  0.4288
## ARCH Lag[5]    1.0240 1.440 1.667  0.7258
## ARCH Lag[7]    2.3584 2.315 1.543  0.6418
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  535.067
## Individual Statistics:
## mu       0.4153
## omega  519.2481
## alpha1   0.2650
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          0.846 1.01 1.35
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value     prob sig
## Sign Bias            1.932 0.053468   *
## Negative Sign Bias   2.294 0.021905  **
## Positive Sign Bias   1.214 0.225000
## Joint Effect        11.964 0.007508 ***
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     53.73    3.627e-05
## 2    30     67.08    7.536e-05
## 3    40     61.85    1.135e-02
## 4    50     75.39    9.084e-03
##
##
## Elapsed time : 0.08862495
```

```
# Coefficients of mu and omega are not significant.
# model: iGARCH(1,1)
# r_t = μ_t + a_t
# μ_t = μ_0
# a_t = σ_t * ε_t
# (σ_t)^2 = α_0(=0) + α_1*(a_t-1)^2 + β_1*(σ_t-1)^2, where α_1 + β_1 = 1
# where μ_0 = 0.000056(not significant, can be seen as 0), α_0 = 0, α_1 = 0.017439, β_1 = 0.982561
```

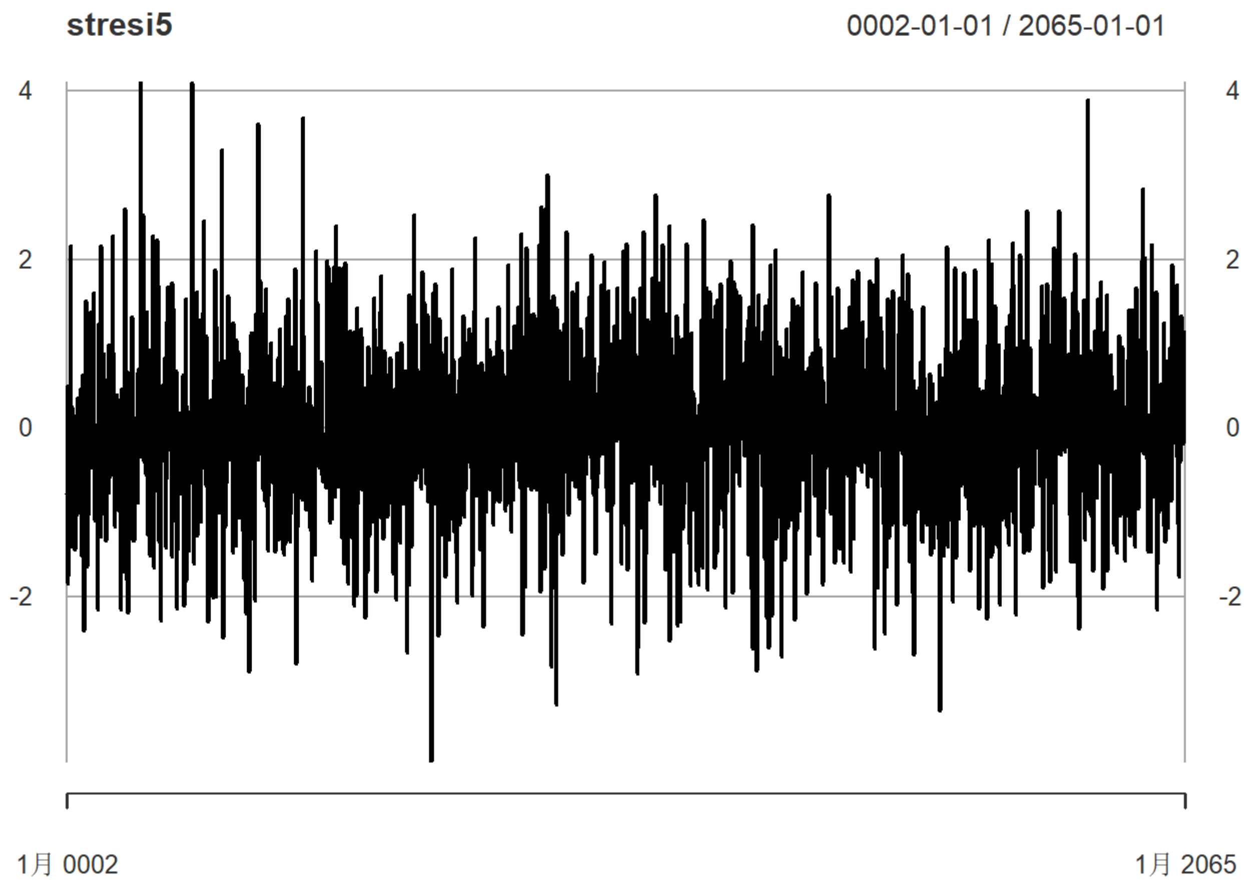
```
plot(m5, which = 10)
```




```
plot(m5, which = 11)
```



```
stresi5=residuals(m5,standardize=T)
plot(stresi5,type="l")
```




```
Box.test(stresi5,10,type="Ljung-Box",fitdf = 1) # p-value > 0.05, white noise
```

```
##
##  Box-Ljung test
##
## data:  stresi5
## X-squared = 9.8259, df = 9, p-value = 0.3648
```

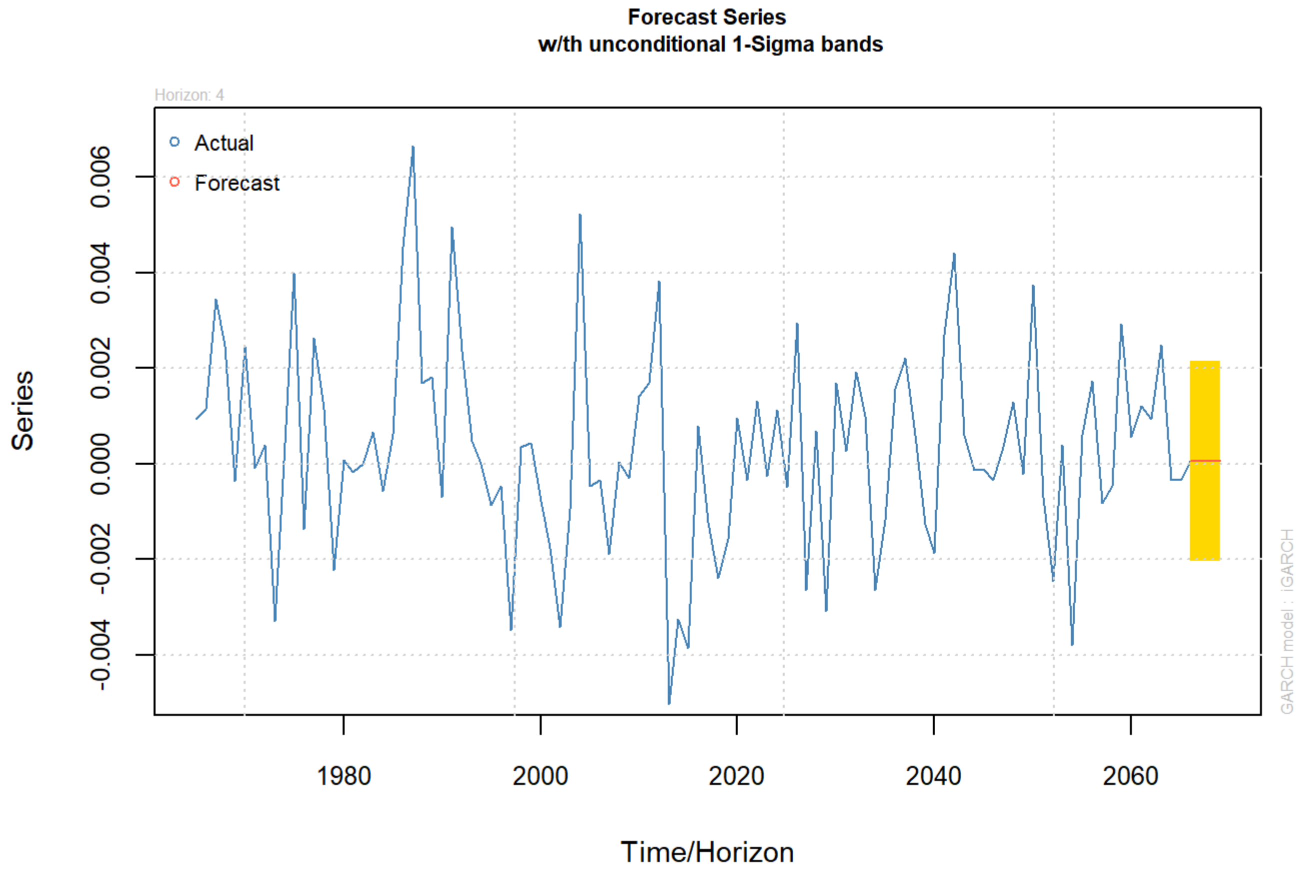
```
Box.test(stresi5^2,10,type="Ljung-Box",fitdf = 1) # p-value > 0.05, remains no ARCH effect
```

```
##
##  Box-Ljung test
##
## data:  stresi5^2
## X-squared = 11.676, df = 9, p-value = 0.2322
```

```
# d
forecast = ugarchforecast(m5, n.ahead = 4, data=log_returns_df3)
```

```
## Warning in `setfixed<-`(`*tmp*`, value = as.list(pars)): Unrecognized Parameter
## in Fixed Values: beta1...Ignored
```

```
plot(forecast, which = 1)
```




```
U=forecast@forecast$seriesFor+1.96*forecast@forecast$sigmaFor
L=forecast@forecast$seriesFor-1.96*forecast@forecast$sigmaFor

forecast
```

```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: iGARCH
## Horizon: 4
## Roll Steps: 0
## Out of Sample: 0
##
## 0-roll forecast [T0=2065-01-01]:
##        Series    Sigma
## T+1 5.604e-05 0.002099
## T+2 5.604e-05 0.002099
## T+3 5.604e-05 0.002100
## T+4 5.604e-05 0.002100
```

```
c(L[1],U[1])
```

```
## [1] -0.004058270  0.004170347
```