732A62 Lab 3

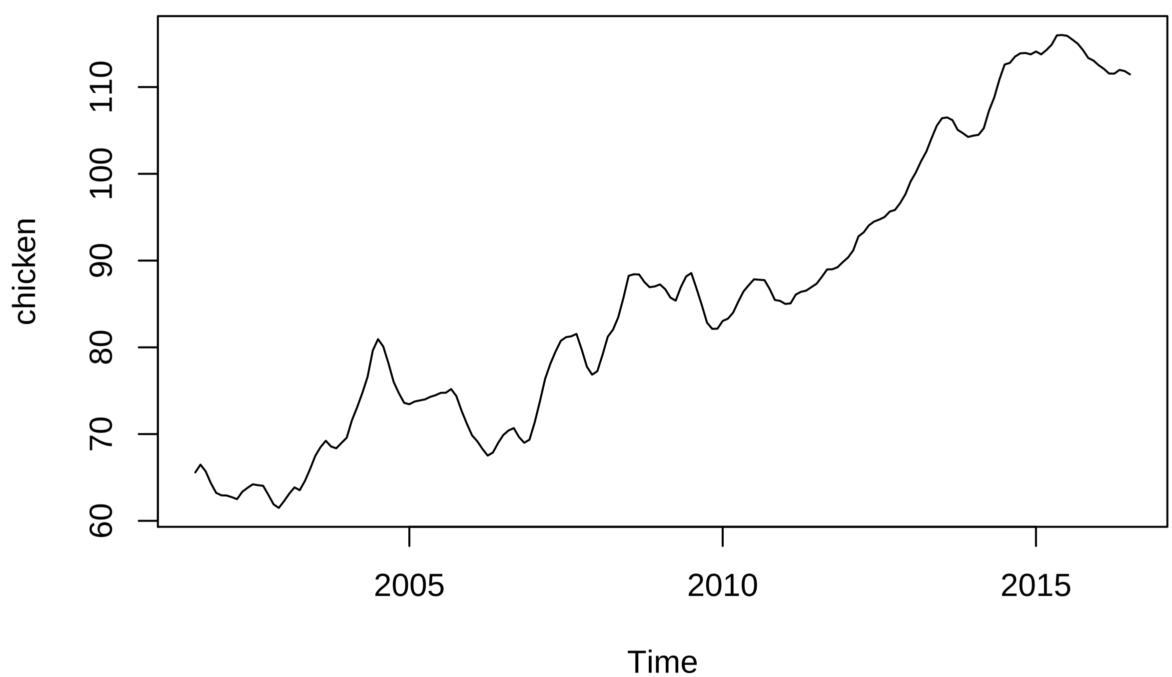
Emil K Svensson & Rasmus Holm

2017-10-11

Assignment 1

1)

```r
library(astsa)
library(TSA)
library(forecast)
library(fGarch)

plot(chicken)
```

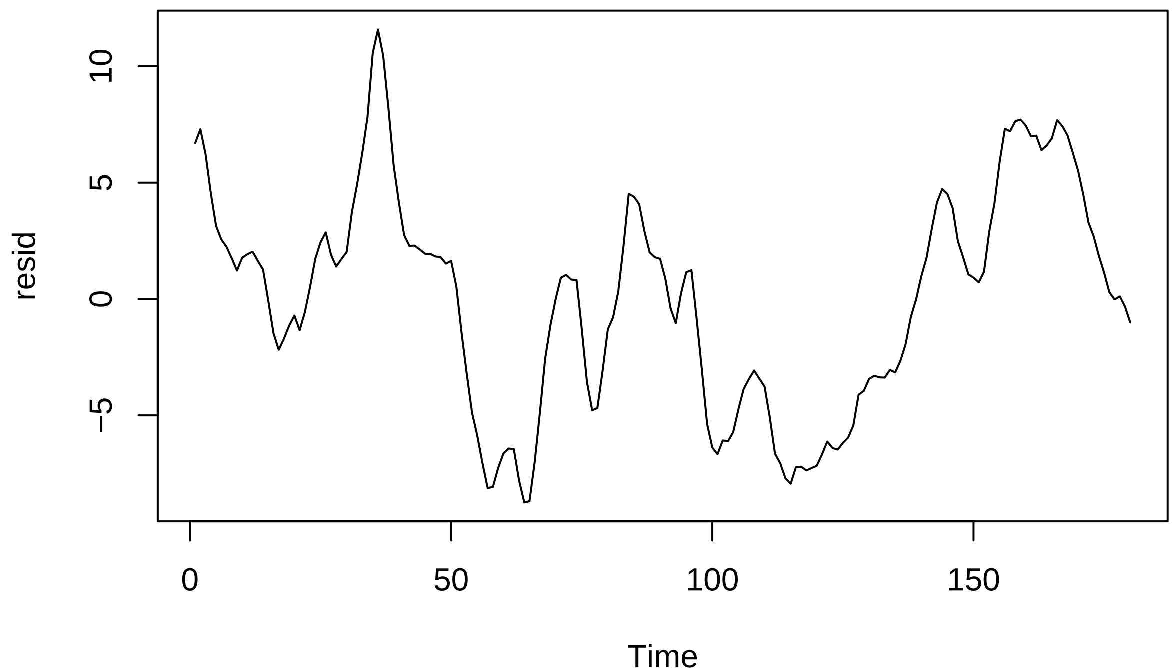


It looks like a linear, potentially quadratic, trend.

2)

```r
lm_data <- data.frame(chicken=chicken, time=1:length(chicken))
lm_fit <- lm(chicken ~ time, lm_data)
```

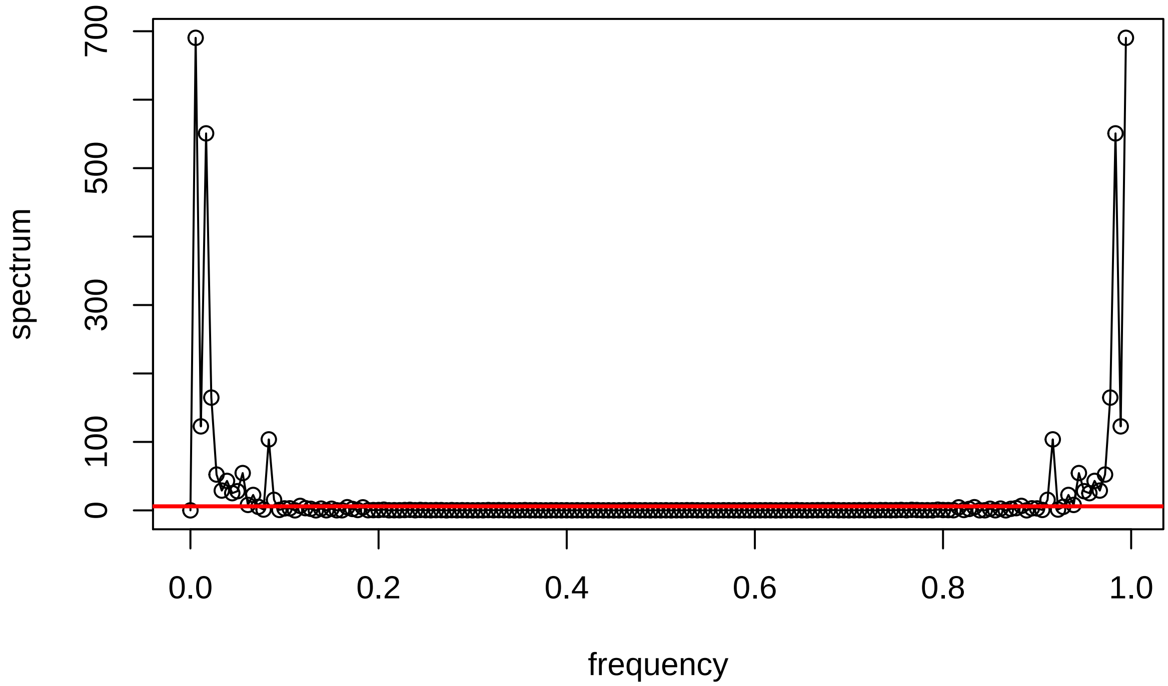
```
z <- resid(lm_fit)
plot(z, type="l", ylab="resid", xlab="Time")
```



The residuals do not look stationary. The data is definitely correlated.

3)

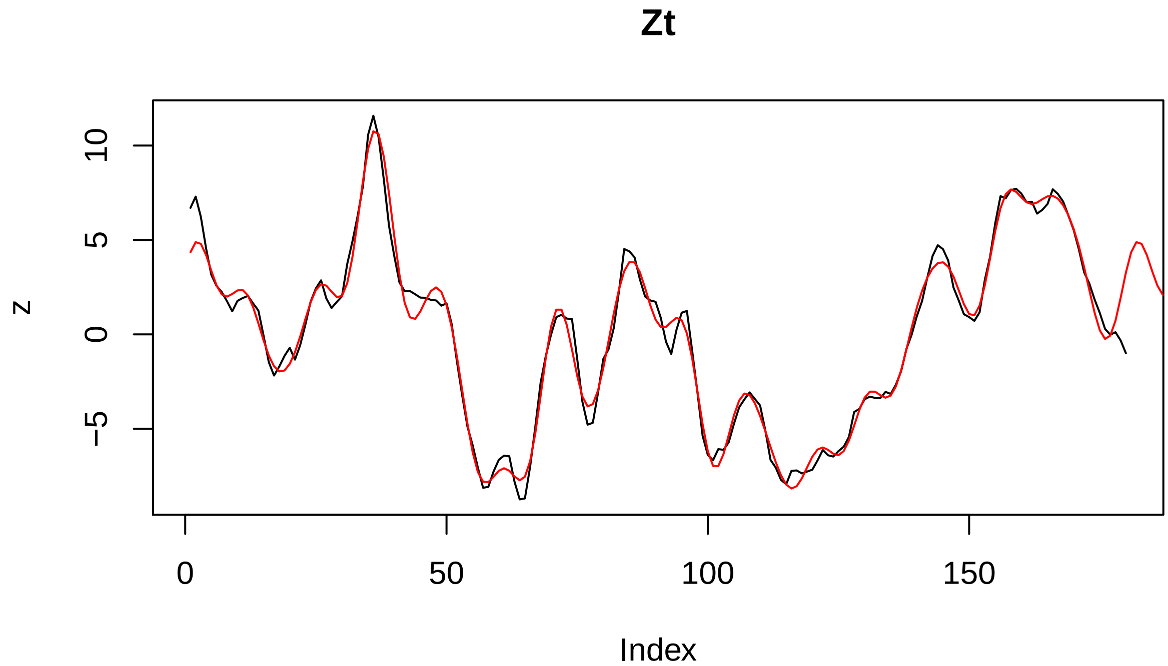
```
denom <- sqrt(length(z)) *
    exp(complex(imaginary=2 * pi * 0:(length(z) - 1) / length(z)))
density <- fft(z) / denom
periodigram <- abs(density)^2

upper <- 2 * mean(periodigram) / qchisq(0.025, 2)
lower <- 2 * mean(periodigram) / qchisq(0.975, 2)

plot(0:(length(chicken) - 1) / length(chicken), periodigram, type="o",
     xlab="frequency", ylab="spectrum")
abline(h=lower, col="red", lwd=2)
```

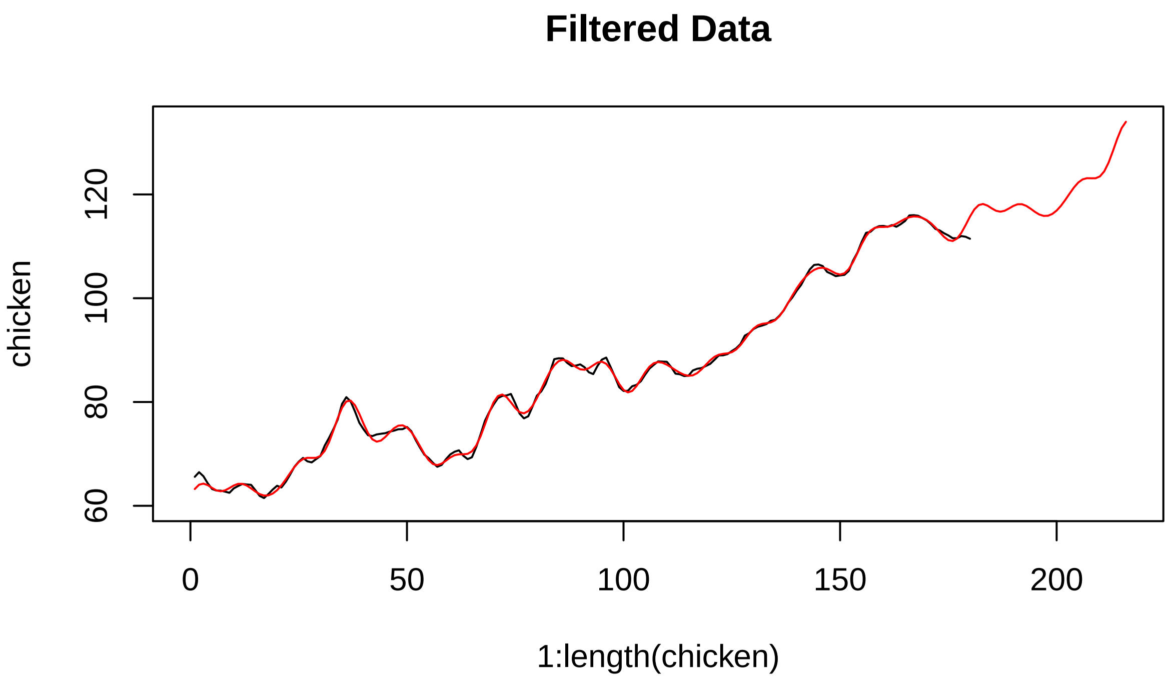


We can see that low and high frequencies are the dominant frequencies. We decided to use the mean as the baseline which sets the lower limit close to zero. This results in that most non-zero frequencies are significant.

4)

```
freq_density <- density
freq_density[periodigram < lower] <- 0

n <- length(z)
ts <- 1:(n + 36)

xs <- rep(0, n + 36)

for (t in ts) {
    xs[t] <- sum(freq_density * exp(complex(imaginary=2 * pi * (0:(n - 1)) / n * t))) / sqrt(n)
}

filtered_data <- predict(lm_fit, data.frame(time=1:length(xs))) + Re(xs)
```

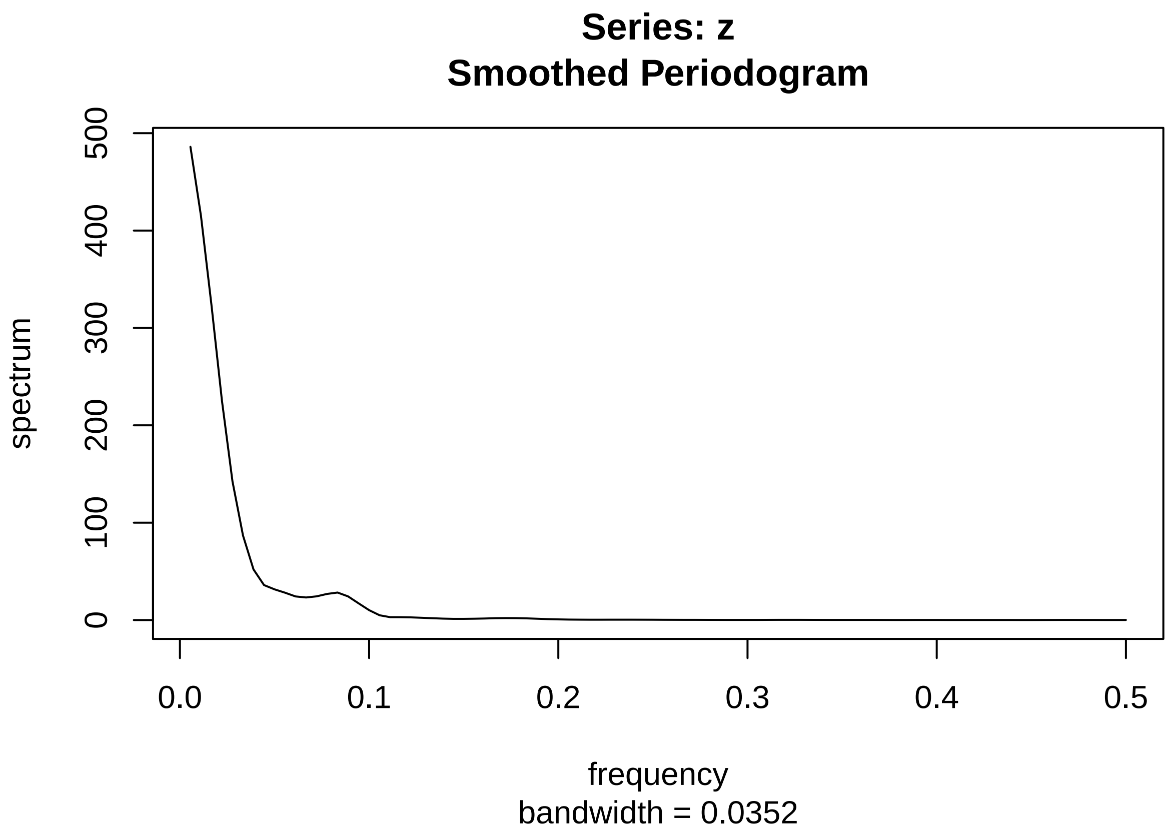




The forecast do look reasonable since it follows the general trend well.

5)

```r
k <- kernel("modified.daniell", c(2,2))
md_dan <- mvspec(z, kernel=k, log="no")
```




```r
Lh <- md_dan$Lh

lower1 <- 2 * Lh * md_dan$spec / qchisq(0.975,2*Lh)
upper1 <- 2 * Lh * md_dan$spec / qchisq(0.025,2*Lh)

# Comparing frequencies

freq_4 <- 0:179/180

freq_4[periodigram > lower]
```

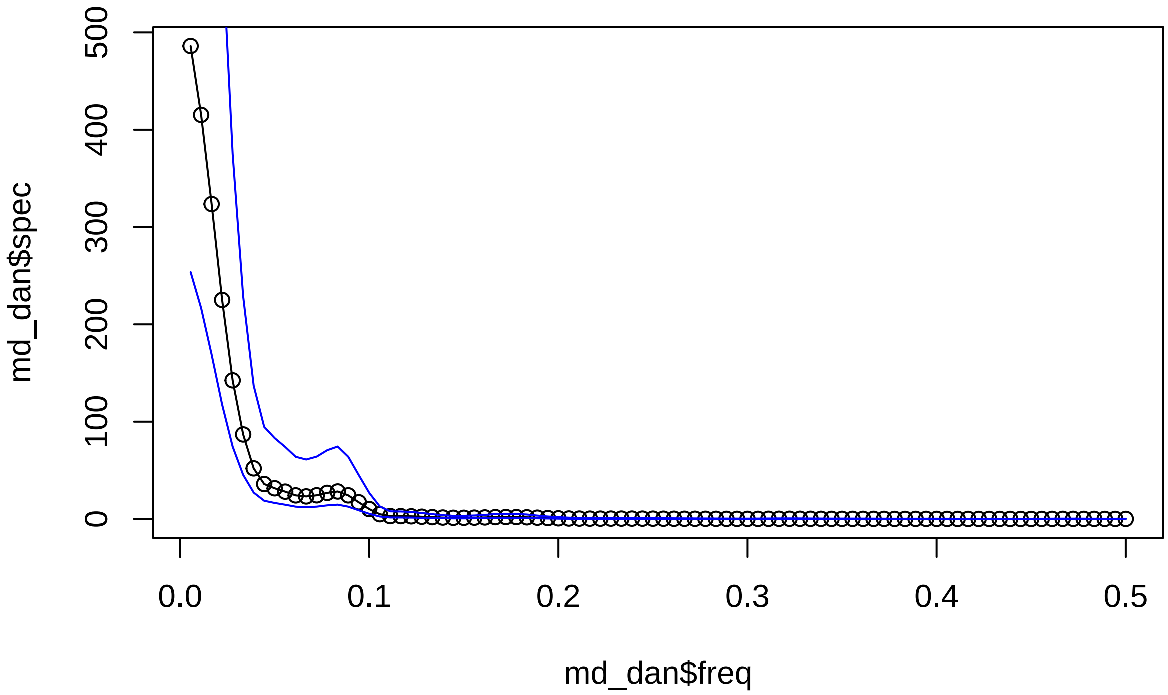
```
##  [1] 0.005555556 0.011111111 0.016666667 0.022222222 0.027777778
##  [6] 0.033333333 0.038888889 0.044444444 0.050000000 0.055555556
## [11] 0.061111111 0.066666667 0.083333333 0.088888889 0.116666667
## [16] 0.883333333 0.911111111 0.916666667 0.933333333 0.938888889
## [21] 0.944444444 0.950000000 0.955555556 0.961111111 0.966666667
## [26] 0.972222222 0.977777778 0.983333333 0.988888889 0.994444444
```

```r
md_dan$freq[md_dan$freq < 0.1]
```

```
##  [1] 0.005555556 0.011111111 0.016666667 0.022222222 0.027777778
##  [6] 0.033333333 0.038888889 0.044444444 0.050000000 0.055555556
```

```
## [11] 0.061111111 0.066666667 0.072222222 0.077777778 0.083333333
## [16] 0.088888889 0.094444444 0.100000000
```

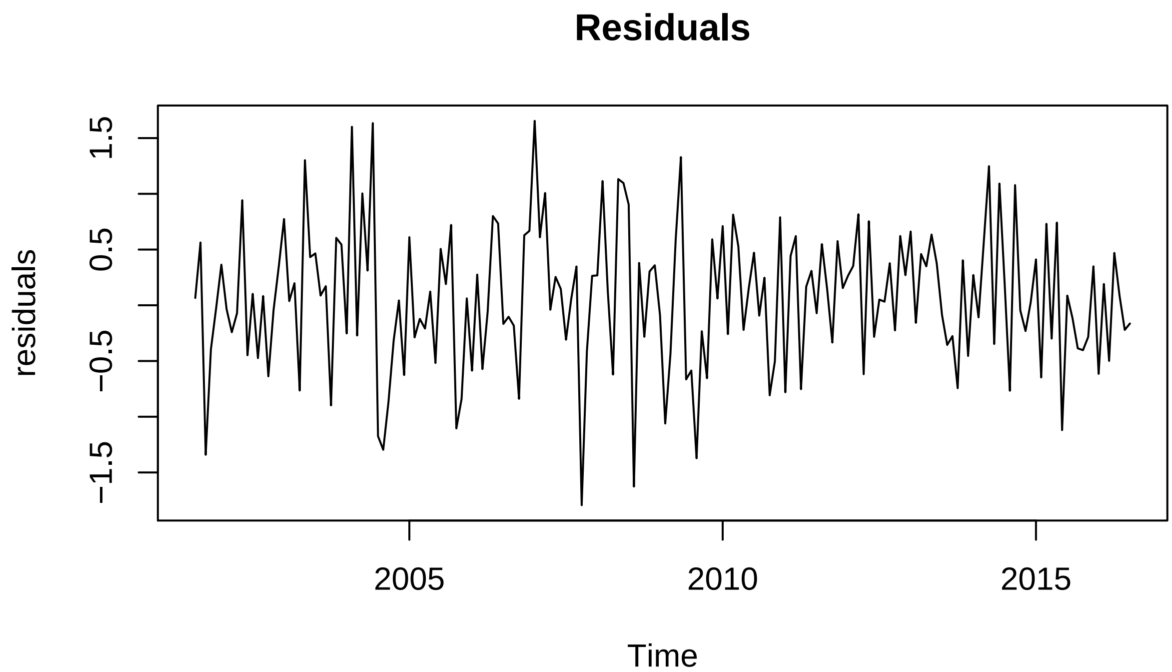
We can see that similar frequencies were found by smoothing the spectrum so the smoothing does seem to help.

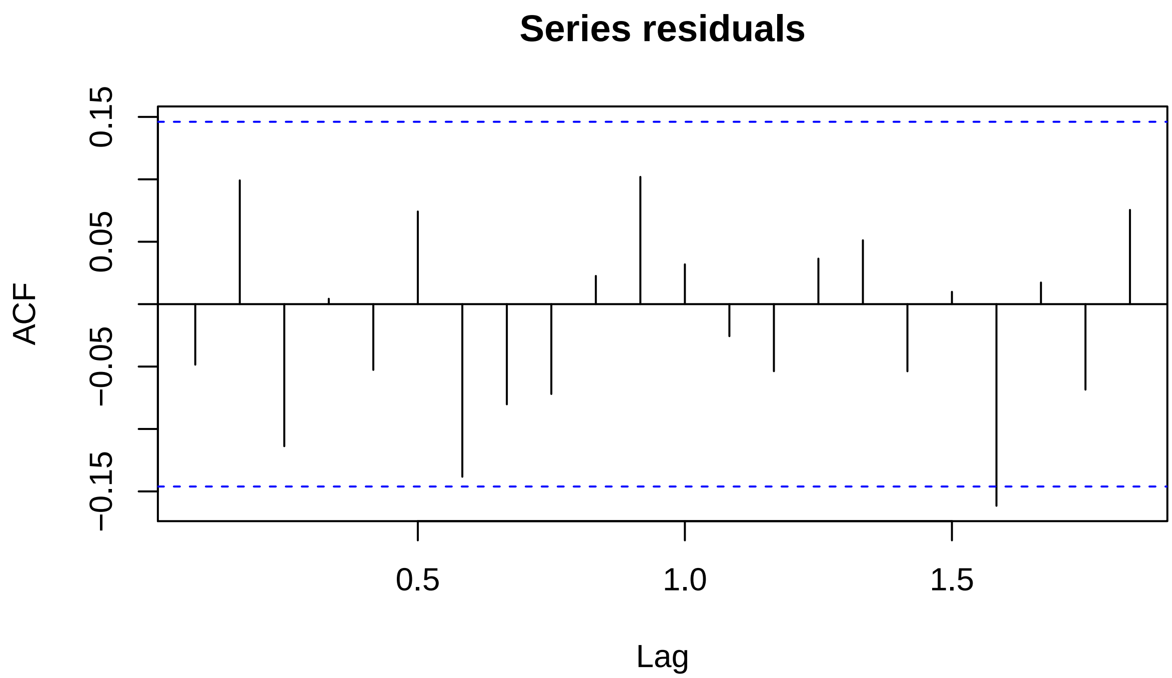


6)

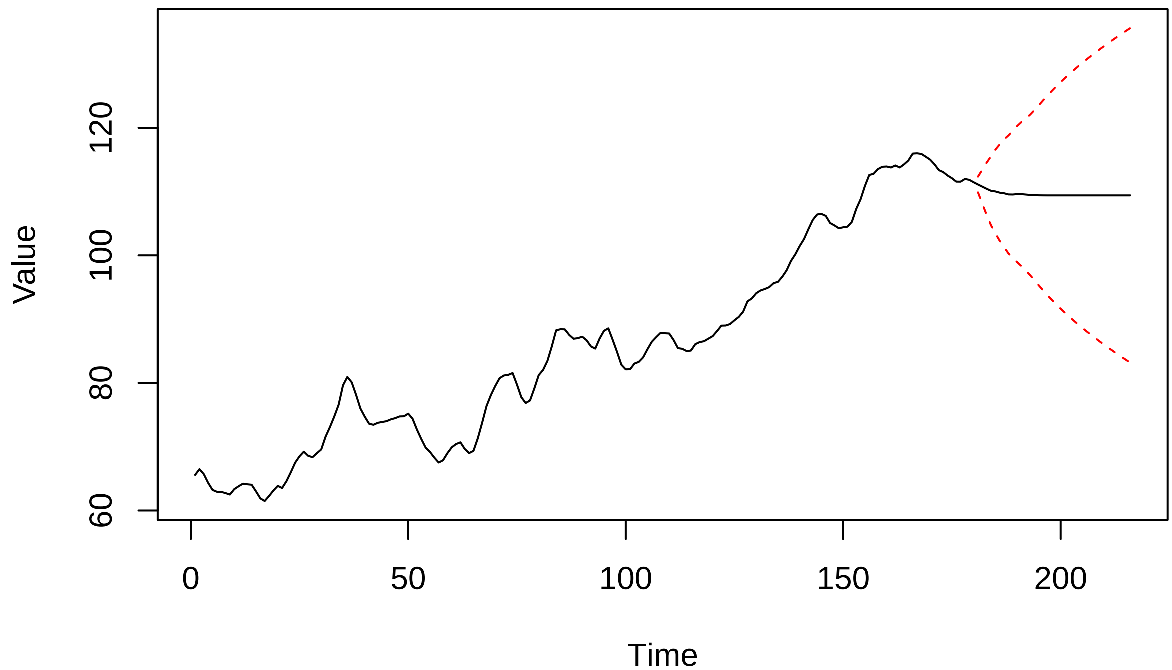
```
fit_plot <- function(model, data) {
    nahead <- 36
    pred <- predict(model, n.ahead=nahead, se.fit=TRUE)
    upper_band <- pred$pred + 1.96 * pred$se
    lower_band <- pred$pred - 1.96 * pred$se

    n <- length(data)

    plot(c(data, pred$pred), type="l",
         ylim=c(min(data), max(upper_band)), ylab="Value", xlab="Time")
    lines(n + 1:nahead, upper_band, lty=2, col="red")
    lines(n + 1:nahead, lower_band, lty=2, col="red")
}
```

```
fit <- arima(chicken, order=c(2, 1, 0), seasonal=list(order=c(0, 0, 1), period=12))
residuals <- residuals(fit)
```





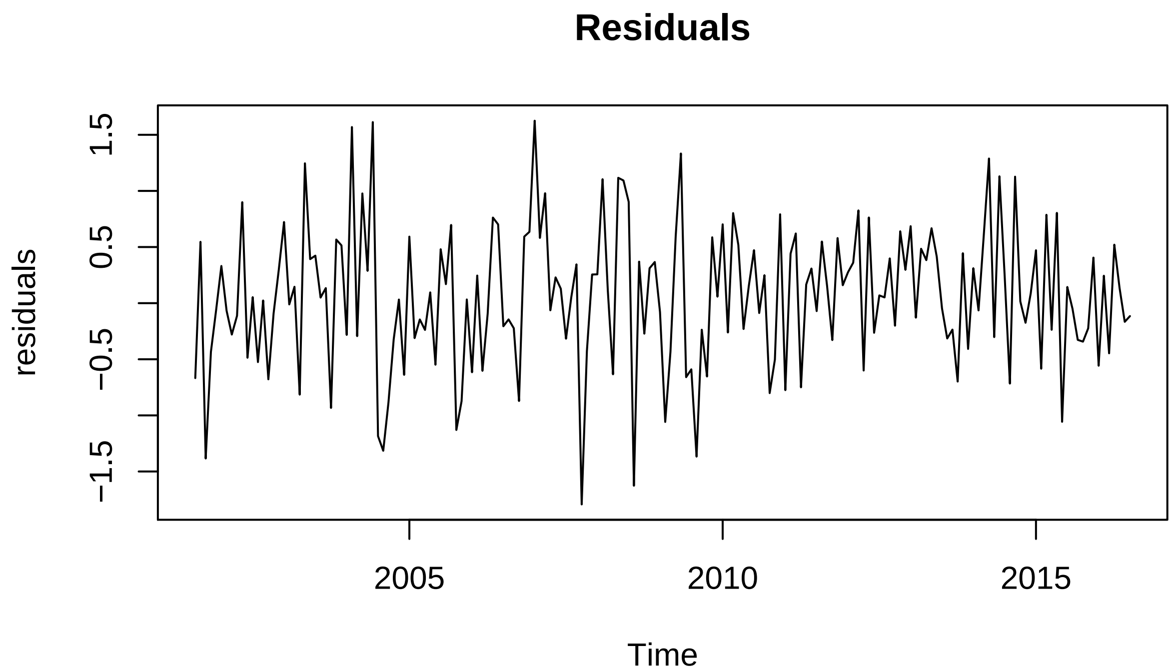
The model seem to fit the data decent with no correlation. However, the variance seem to decrease with time so it may not be completely stationary.

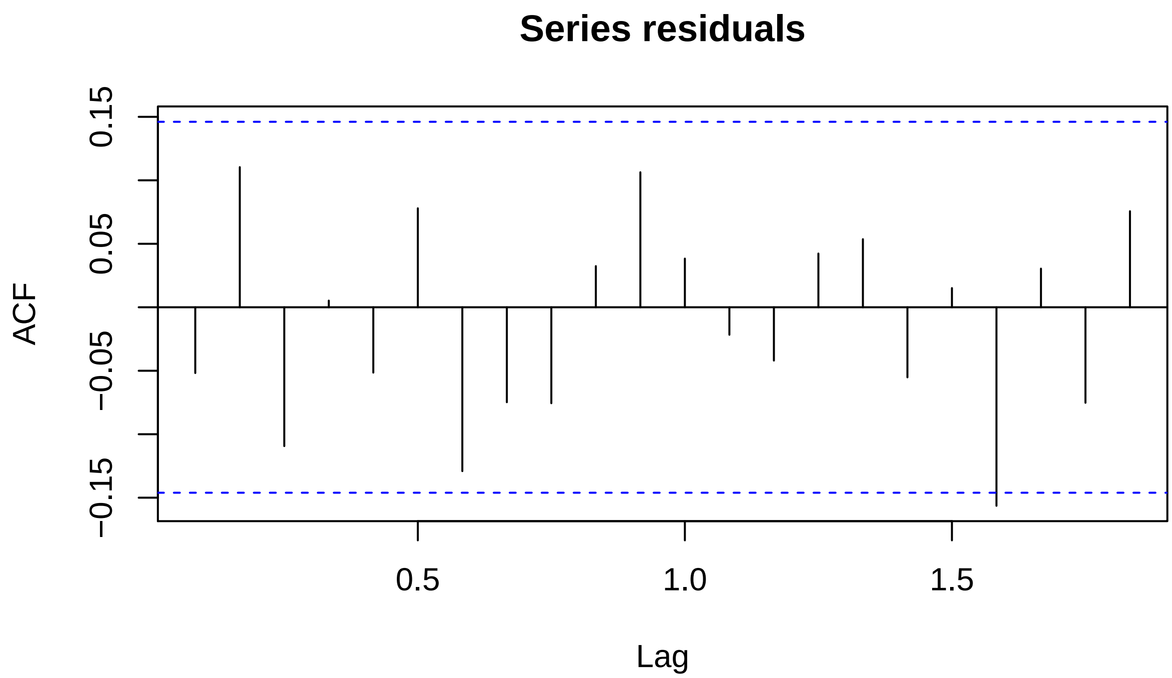


The forecast do not look very good because it does not follow the general trend. We would rather trust the forecast from 1.4.

7)

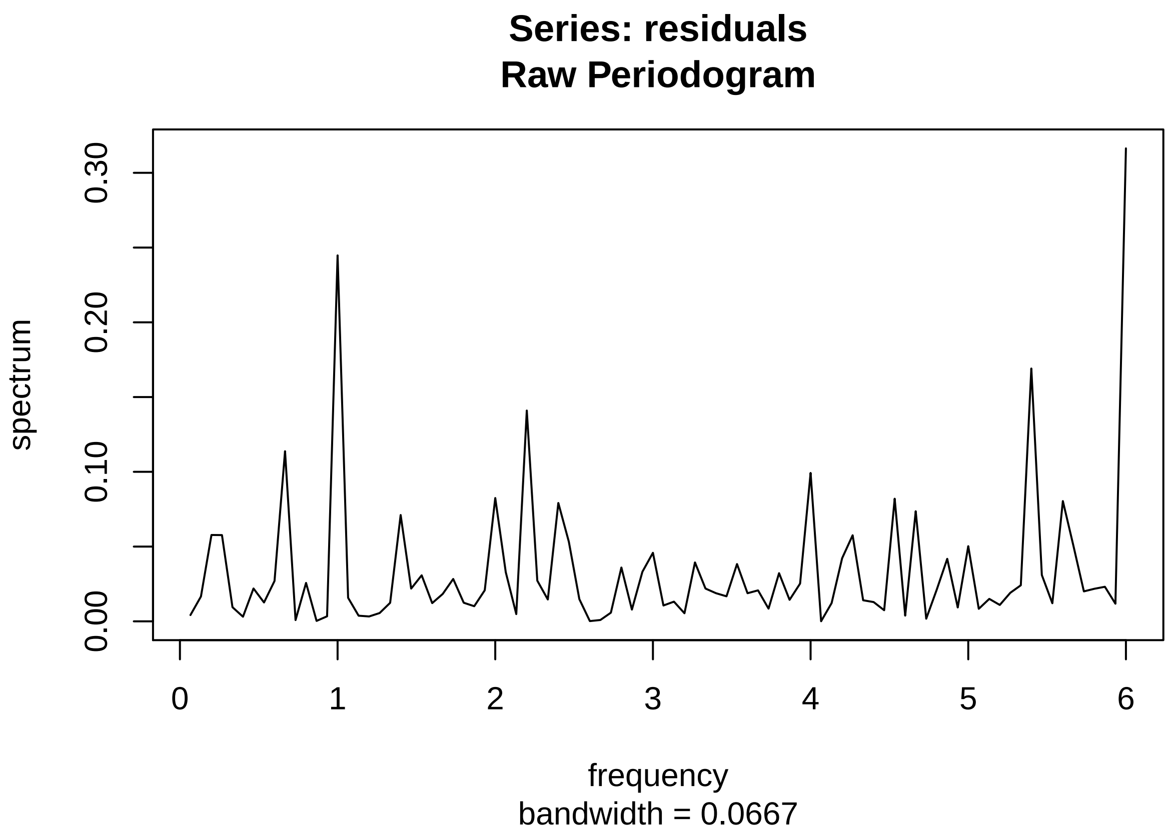
```
fit <- arima(chicken, order=c(3, 0, 0), seasonal=list(order=c(0, 0, 1), period=12))
residuals <- residuals(fit)
```





The residuals looks similar to those from the other fit. Uncorrelated but not stationary because of changing variance.

```
mvspec(residuals, log="no")
```

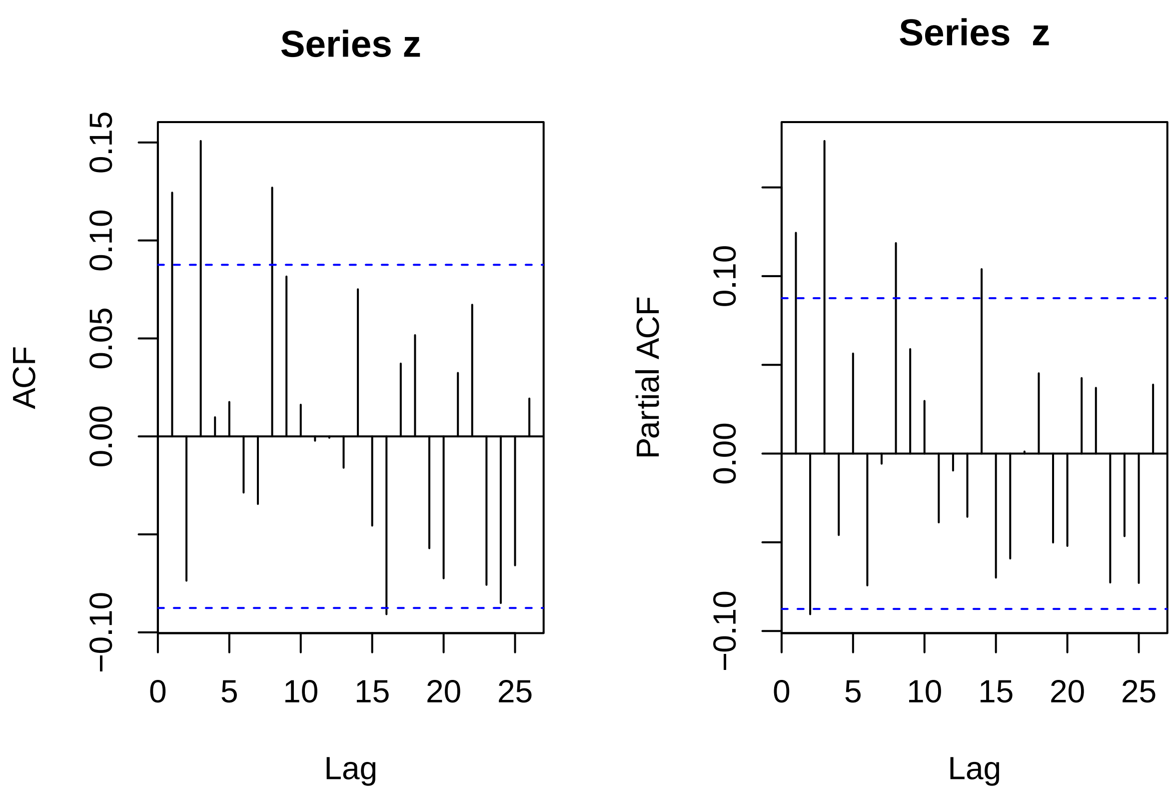


We can see that the spectrum is non-zero for a lot of frequencies and not just low ones. This indicates that the residuals are not stationary.

Assignment 2

1)

```
ld_oil <-diff(log(oil))

z <-ld_oil[1:(52*9 + 33)]

old <- par(mfrow = c(1,2))
acf(z)
pacf(z)
```




```
par(old)

suggested_model <- Arima(z, order = c(3,0,0))

summary(suggested_model)
```

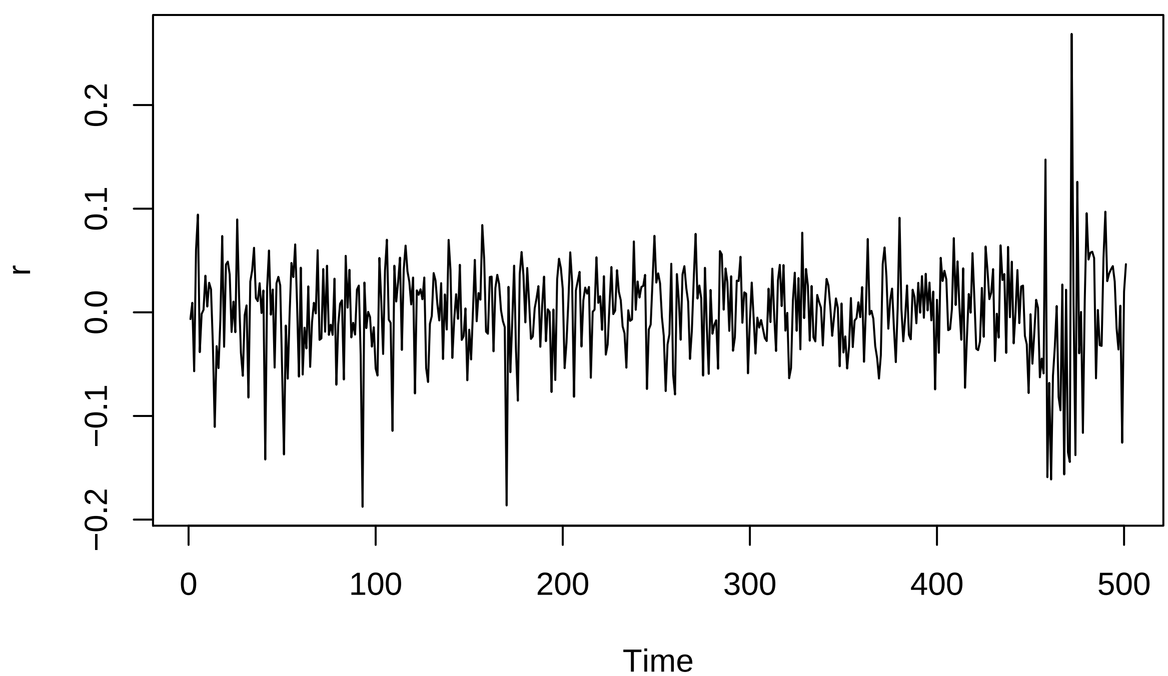
```
## Series: z
## ARIMA(3,0,0) with non-zero mean
##
## Coefficients:
##          ar1      ar2     ar3    mean
##        0.151  -0.1147  0.1777  0.0018
## s.e.   0.044   0.0442  0.0442  0.0026
##
```

```
## sigma^2 estimated as 0.002171:  log likelihood=827.28
## AIC=-1644.55   AICc=-1644.43   BIC=-1623.47
##
## Training set error measures:
##                        ME       RMSE        MAE  MPE MAPE      MASE
## Training set 2.381642e-05 0.04640656 0.03454024 -Inf  Inf 0.7492286
##                     ACF1
## Training set 0.008324494
```

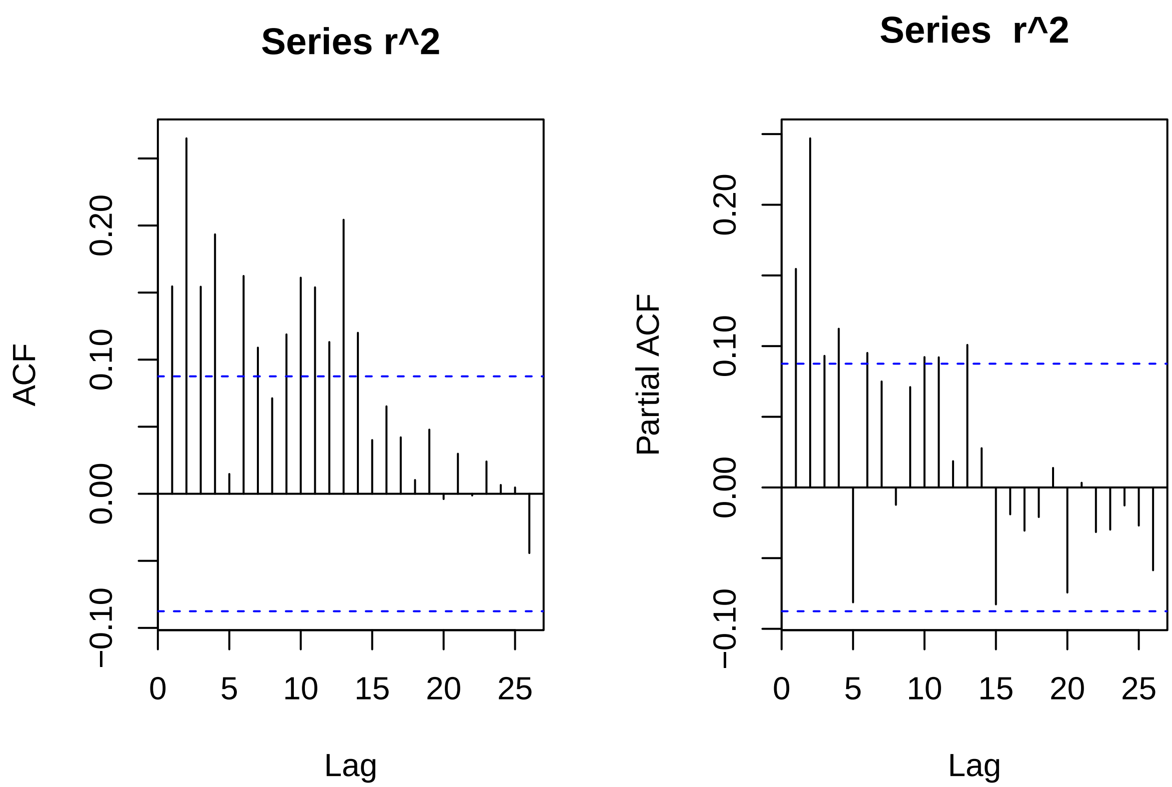
```
r <- resid(suggested_model)
```

2)

```
plot(r)
```




```
old <- par(mfrow = c(1,2))
acf(r^2)
pacf(r^2)
```




```r
par(old)

fit1<- garchFit(~ arma(3,0) + garch(1,1) , data = ld_oil, trace = FALSE)
fit1
```

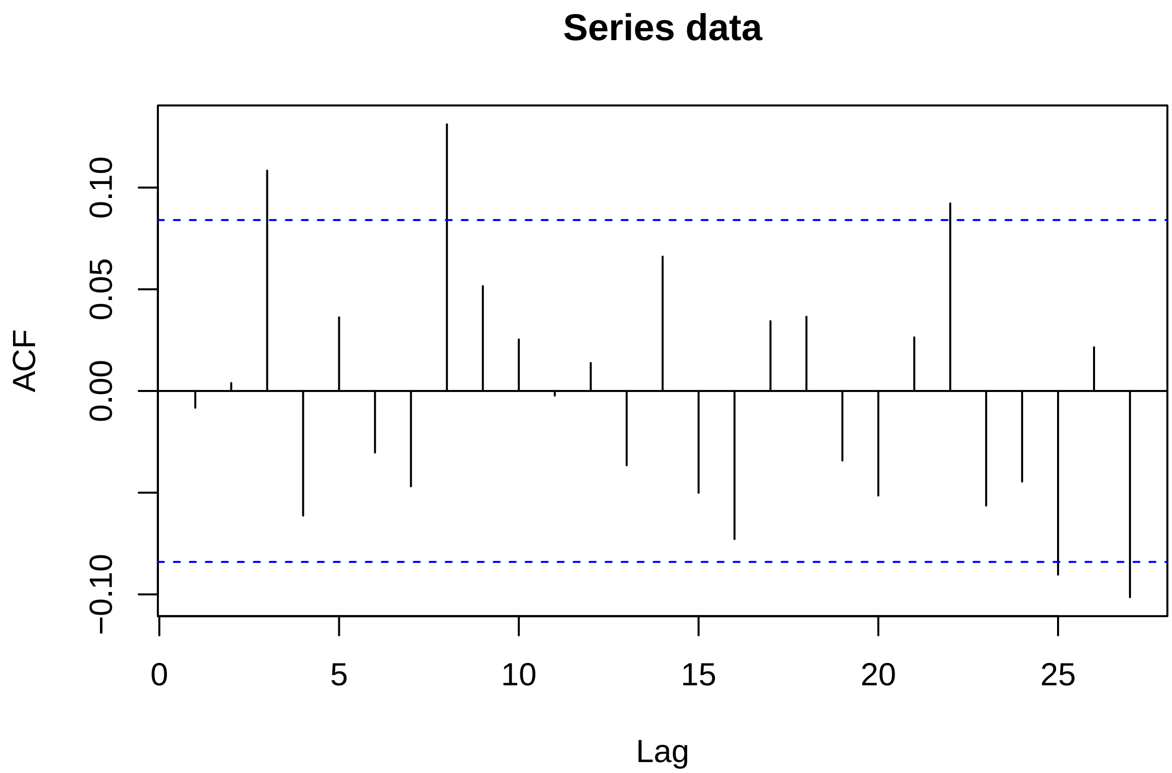
```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~arma(3, 0) + garch(1, 1), data = ld_oil,
##     trace = FALSE)
##
## Mean and Variance Equation:
##  data ~ arma(3, 0) + garch(1, 1)
## <environment: 0xc8adb60>
##  [data = ld_oil]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##          mu          ar1          ar2          ar3        omega
##  0.00239404   0.17510328  -0.12420934   0.07407490   0.00011329
##      alpha1        beta1
##  0.06213801   0.87911973
##
```

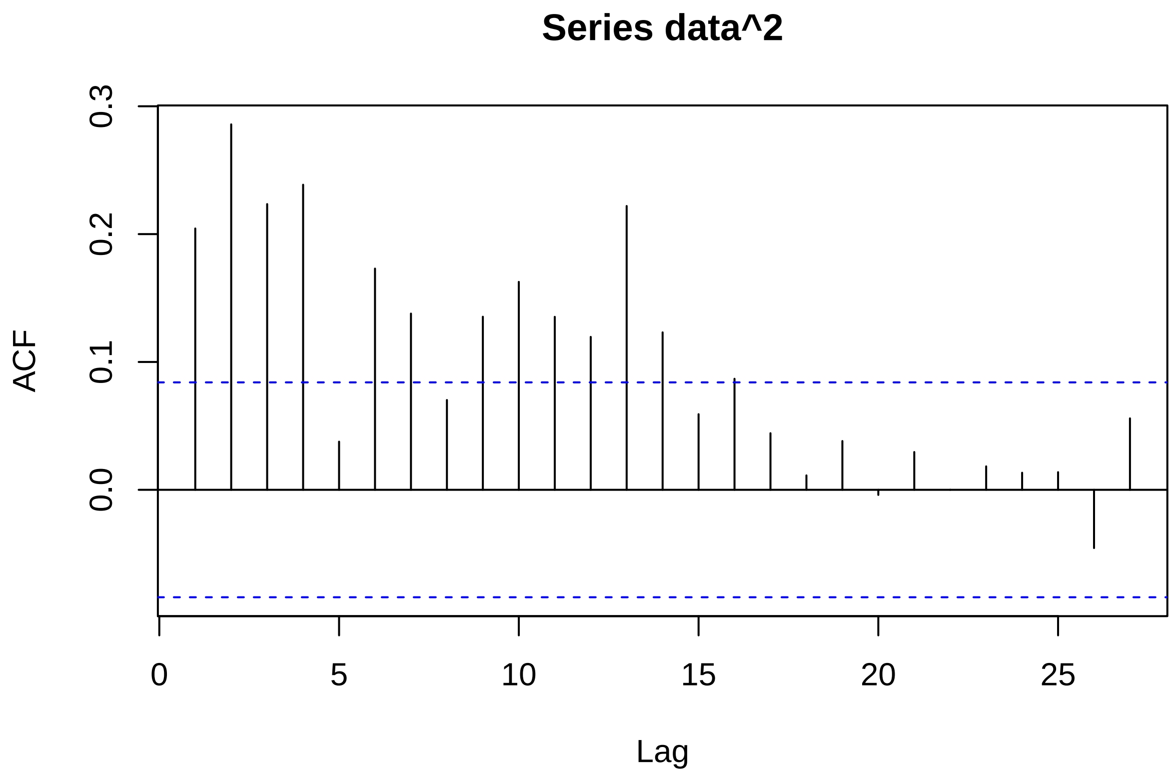
```
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## mu      0.0023940   0.0017860    1.340 0.180109
## ar1     0.1751033   0.0444931    3.936  8.3e-05 ***
## ar2    -0.1242093   0.0449940   -2.761 0.005770 **
## ar3     0.0740749   0.0457586    1.619 0.105486
## omega   0.0001133   0.0000515    2.200 0.027834 *
## alpha1  0.0621380   0.0173666    3.578 0.000346 ***
## beta1   0.8791197   0.0362307   24.265  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  937.9971    normalized:  1.724259
##
## Description:
##  Wed Oct 11 15:51:47 2017 by user: r
```

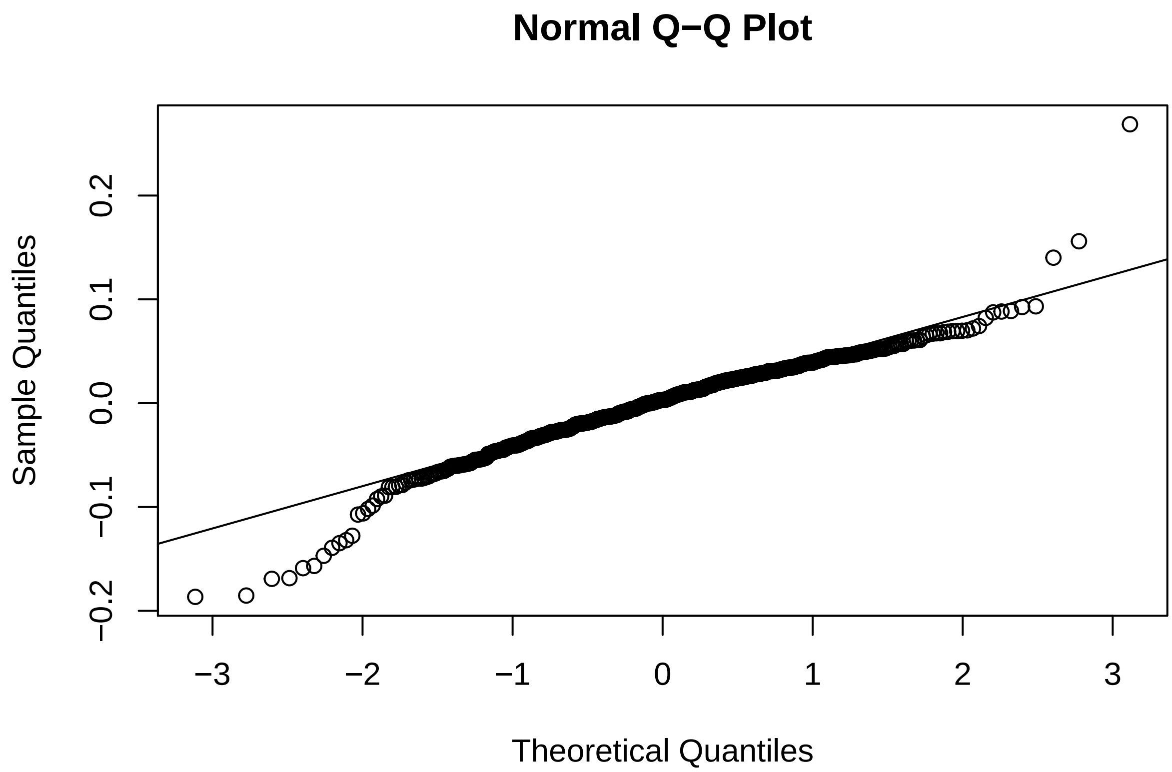
The time series of the residuals seem to have an increasing variance in the end of the residuals.

The ACF of the squared residuals trails of and in the PACF they cuts of after 2 lags. Indicating a GARCH(p,q) An p = 2, q = 0 maybe? ## 3)

```
helper <- function(data){
acf(data)
acf(data^2)
qqnorm(data)
qqline(data)

}



helper(fit1@residuals)
```


Series data
ACF
0.10
0.05
0.00
−0.10
0
5
10
15
20
25
Lag


Series data^2
ACF
0.0
0.1
0.2
0.3
0
5
10
15
20
25
Lag




```
fit1@fit$objective
```

```
## [1] 725.3238
```

4)

5)

6)