MA30085, Time Series - Coursework

Student Number 219226406 ; Candidate Number 23720

28/03/2024 noon till 17/04/2024 noon

Description of coursework

For this assignment, you must perform a data analysis task. Data consist of 2 time series randomly selected from the *lts0.Rda* data file (containing 1000 time series). The goal for you is to study carefully both time series, and attempt to model them using the main steps of the Box-Jenkins methodology.

The coursework will be marked based on the statistical reasoning that has led to your conclusions. Thus to carry through the appropriate statistical tasks and reasoning is more important than the conclusions themselves. The highest score achievable is 100 (60 for series 1 and 40 for series 2).

A thorough illustration of this way of proceeding is included in the document prepared for the computational laboratory 3. You can use that document as guidance, but should also feel free to attempt other types of investigation, if necessary.

If you have followed all lectures, studied the related material and followed all computationally laboratories, the time required to analyse both time series turns out to be approximately six to eight hours.

Preparation - How to obtain your data

Creation of random seed

The two time series to be analysed are different for each student. Your data are obtained using an integer seed, generated using your unique student number. More specifically, the integer seed to generate the random numbers used for the work consists of the last five digits of your unique, 9-digits, student number. Representing a student number with letters,

student number = $abcdefghi$,

the seed number is

seed number = $efghi$

For example, if your unique student number is 179238011, your seed number is 38011. Or, if your unique student number is 179200810, your seed number is 810, because 00810 is interpreted by R as 810.

Extraction of time series

The time series for your coursework are extracted randomly from a binary file named *lts0.Rda*. This file **must be** located in the same directory in which you have transferred this R markdown document, *MA30085_coursework.Rmd*.

By running the code in the R chunk below, you will automatically import the two time series, called `tser1` and `tser2`. They are objects of class `ts`. If the operation is successful, you should see both time series displayed in a graphic. Once this is done, you are ready for the analysis.

Type of time series simulated

The **first time series** (`tser1`) is a simulation of an ARIMA process, $\{X_t,\ t \in \mathbb{Z}\}$, described by the difference equation

$$\phi(B)(1-B)^d X_t = \theta(B) Z_t,$$

where $\phi(\lambda)$ is the AR characteristic polynomial of order p, $\theta(\lambda)$ is the MA characteristic polynomial of order q, $\{Z_t,\ t \in \mathbb{Z}\}$ a Gaussian white noise process with mean 0 and standard deviation 1, and where B is the backward shift operator. The parameters p, q and d of the ARIMA(p,d,q) process are integers with the following range:

$$d = 0, 1, 2 \qquad p = 0, 1, 2, 3 \qquad q = 0, 1, 2, 3.$$

The **second time series** (`tser2`), Y_t, has the form

$$Y_t = X_t + m_t,$$

where X_t is an ARIMA(p,d,q) process different from that of the first time series, with

$$d = 0, 1 \qquad p = 0, 1, 2 \qquad q = 0, 1, 2,$$

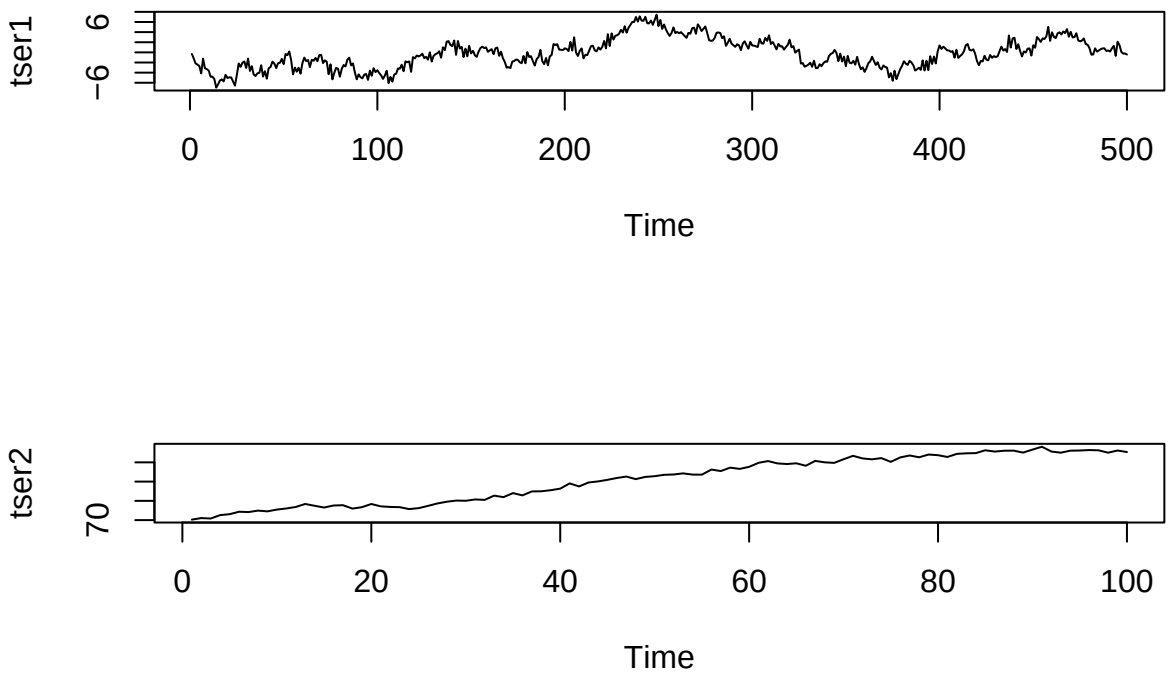
and where m_t is a deterministic polynomial of degree 1 or 2.

R chunk for importing data

Below is the R chunk needed to get your work started, as instructed earlier. Once executed, please add all necessary text, code and graphics (please, make sure the size of the graphics window is big enough for me to check details) under the line **SOLUTION**. The final work should be uploaded on Crowdmark as a PDF document.

```
#************************#
#!!! VERY IMPORTANT !!!
#************************#
# Please, replace the "2" inside set.seed() with your
# unique seed. Failure to do so might result in your work
# being penalised
set.seed(26406)

#************************#
#!!! VERY IMPORTANT !!!
#
#  DON'T MODIFY THE LINES
#  IN THE REMAINING CODE
#
#************************#
# Loading data
load("lts0.Rda")

# Extracting time series
idx1 <- sample(1:500,size=1)
idx2 <- sample(501:1000,size=1)
tser1 <- lts0[[idx1]]
tser2 <- lts0[[idx2]]

# Test you've got the time series in the workspace
```

```r
par(mfrow=c(2,1))
plot(tser1)
plot(tser2)
```




```r
# Back to one plot per window
par(mfrow=c(1,1))
```

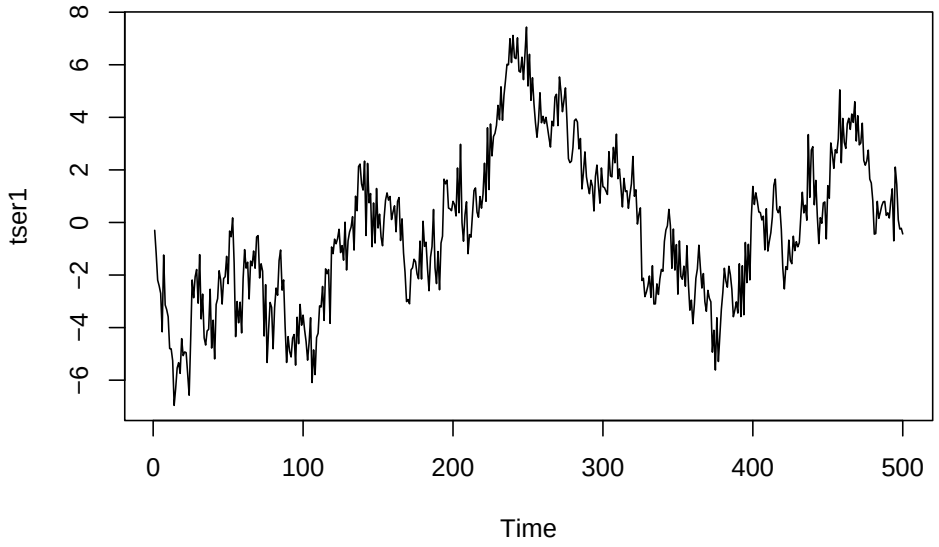
SOLUTION

```r
library(astsa)
library(tseries)
```

Time Series 1

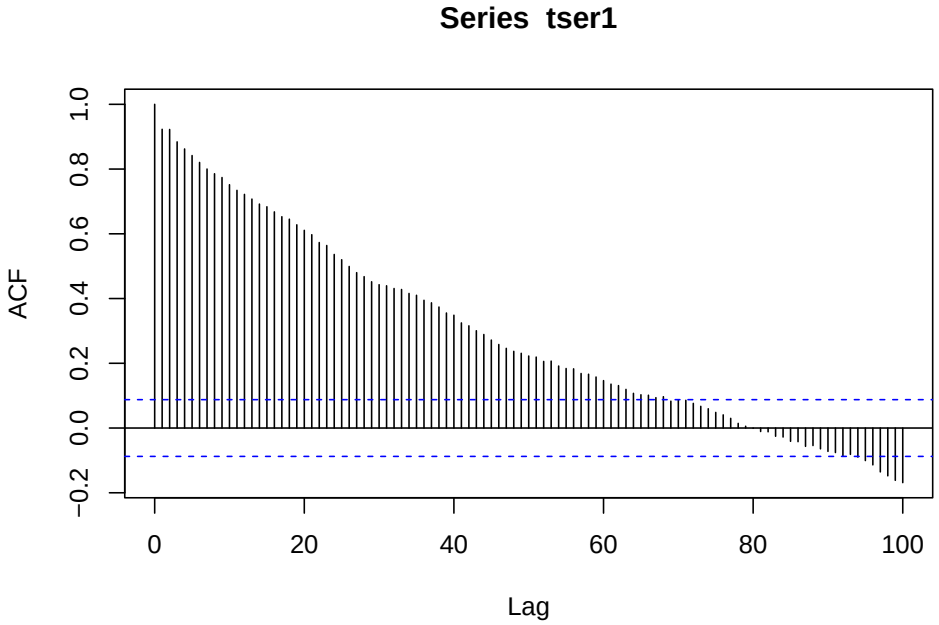
Checking for non-stationarity

```r
plot(tser1)
```



Immediately, we can guess that this is not stationary just by looking at it, lets check the acf:

```
acf(tser1, lag.max = 100)
```



The series does decay, but its not fast enough to exclude the possibility stationary just yet. Performing a Kolmogorov-Smirnov test to look for non-stationarity.

```r
#splitting the observations
x <- tser1[1:250]
y <- tser1[251:500]

ks.test(x,y)
```

```
##
##  Asymptotic two-sample Kolmogorov-Smirnov test
##
## data:  x and y
## D = 0.3, p-value = 3.384e-10
## alternative hypothesis: two-sided
```

The p-value is very small, this suggest that this series is non-stationary. We can confirm this with an adf-test.

```r
adf.test(tser1, k= 10)
```

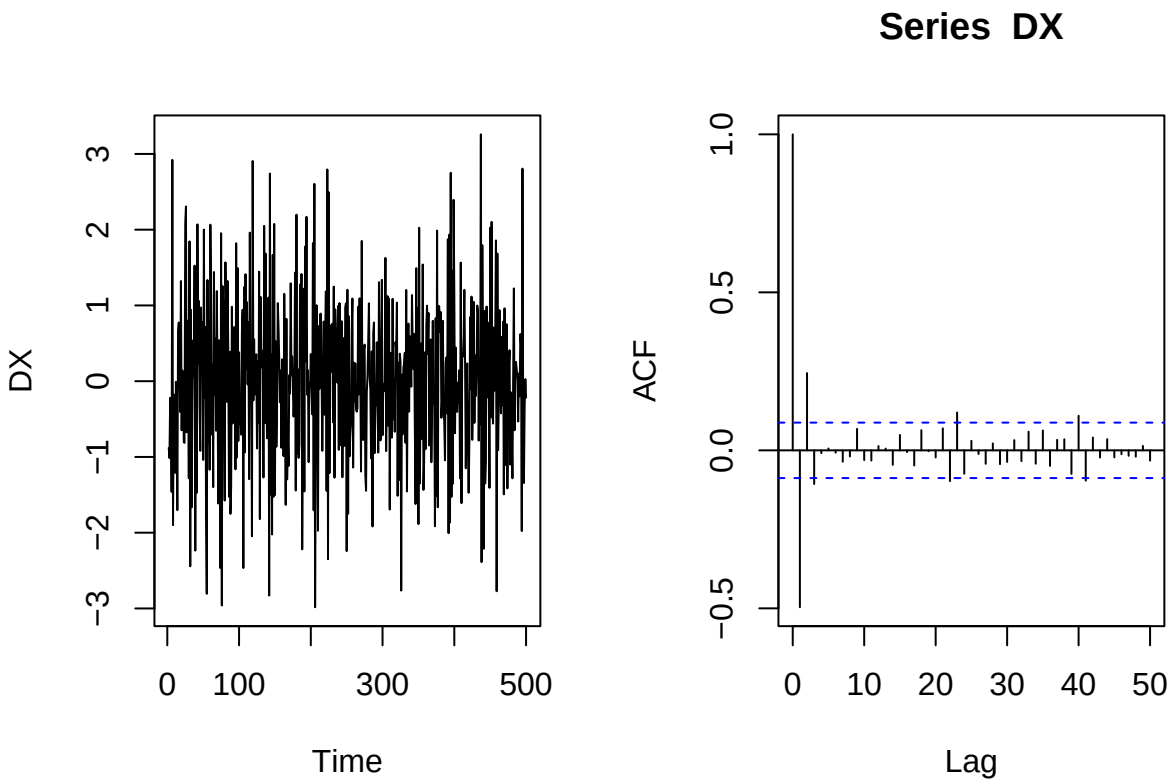
```
##
##  Augmented Dickey-Fuller Test
##
## data:  tser1
## Dickey-Fuller = -2.4588, Lag order = 10, p-value = 0.3841
## alternative hypothesis: stationary
```

The p-value is large, and doesn't really change when we change the value of k. So we can conclude that this series is non-stationary.

Creating stationarity:

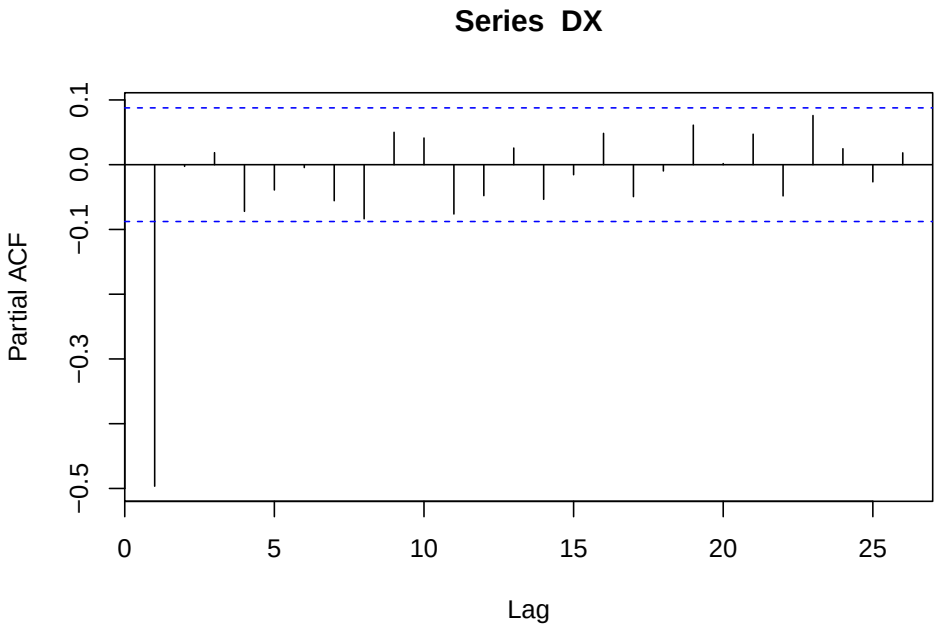
I now what to make the series stationary. I will do this by taking the first difference of the series.

```r
DX <- diff(tser1)

par(mfrow = c(1,2))
plot(DX)
acf(DX, lag.max = 50)
```



This looks better. The acf decreases quite quickly, we have some spikes, but not by much. It appears that the differentiated series is stationary. We can also look at the Partial acf:

```
pacf(DX)
```



All but one spike is inside the Confidence Interval. The one outside the CI is at $\tau = 1$.

Summarising the findings from the acf and pacf: the acf is gradually decreasing and the pacf is truncated at lag 1. Therefore, the appropriate model to describe this time series is ARIMA(1, d, 0), as the first difference is a AR(1) process. Now we want to find the value of d :

Performing an adf test:

```
adf.test(DX)
```

```
## Warning in adf.test(DX): p-value smaller than printed p-value

##
##  Augmented Dickey-Fuller Test
##
## data:  DX
## Dickey-Fuller = -9.8501, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

The p-value is ok, I want to try a different value for the lag-order, to see if anything changes.

```
adf.test(DX, k = 10)
```

```
## Warning in adf.test(DX, k = 10): p-value smaller than printed p-value

##
##  Augmented Dickey-Fuller Test
##
## data:  DX
## Dickey-Fuller = -7.9846, Lag order = 10, p-value = 0.01
## alternative hypothesis: stationary
```

We get the same result. Let's try the Kolmogorov-Smirnov test on the differenced series:

```
x <- DX[1:250]
y <- DX[251:499]
ks.test(x,y)
```

```
##
##  Asymptotic two-sample Kolmogorov-Smirnov test
##
## data:  x and y
## D = 0.079454, p-value = 0.4103
## alternative hypothesis: two-sided
```

The p-value is large for the Kolmogorov-Smirnov test (null hypothesis holds: stationary), but small for the adf test (stationary). So we can conclude that the differenced series is stationary, but the original series is not. Hence $d = 1$ so the model is ARIMA(1,1,0)

Estimation of model parameters:

First I'll try the model we've just worked out :

```
modelA <- arima(tser1, order = c(1,1,0))
print(modelA)
```

```
##
## Call:
## arima(x = tser1, order = c(1, 1, 0))
##
## Coefficients:
##           ar1
##       -0.4962
## s.e.   0.0388
##
## sigma^2 estimated as 0.9514:  log likelihood = -695.77,  aic = 1395.54
```

The estimates are not near zero, hence they are significant. Also $\sigma^2 \approx 1$ and the AIC is large, which are good signs. Here I calculate the 95% Confidence Interval for the parameter α_1:

```
modelA$coef-2*sqrt(diag(modelA$var.coef))
```

```
##       ar1
## -0.573825
```

```
modelA$coef+2*sqrt(diag(modelA$var.coef))
```

```
##        ar1
## -0.4185026
```

$$\alpha_1 \in [-0.573825, -0.4185026]$$

Lets look at some other options for the parameters p and q:

```
modelB<- arima(tser1, order = c(1, 1, 3))
AIC(modelB)
```

```
## [1] 1397.238
```

```
modelC <- arima(tser1, order = c(1,1,1))
AIC(modelC)
```

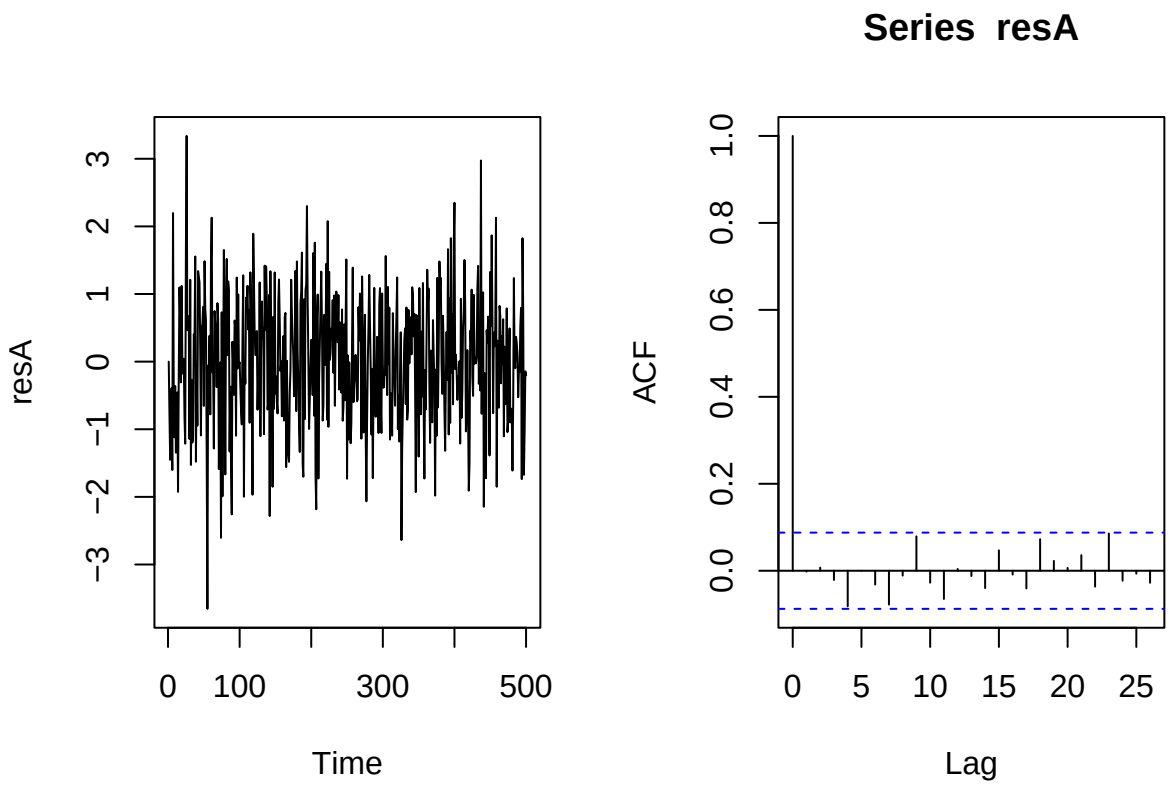
```
## [1] 1397.534
```

We see that our original model ARIMA(1,1,0) has the smallest AIC, therefore it is the best model for the data.

Agreement with data

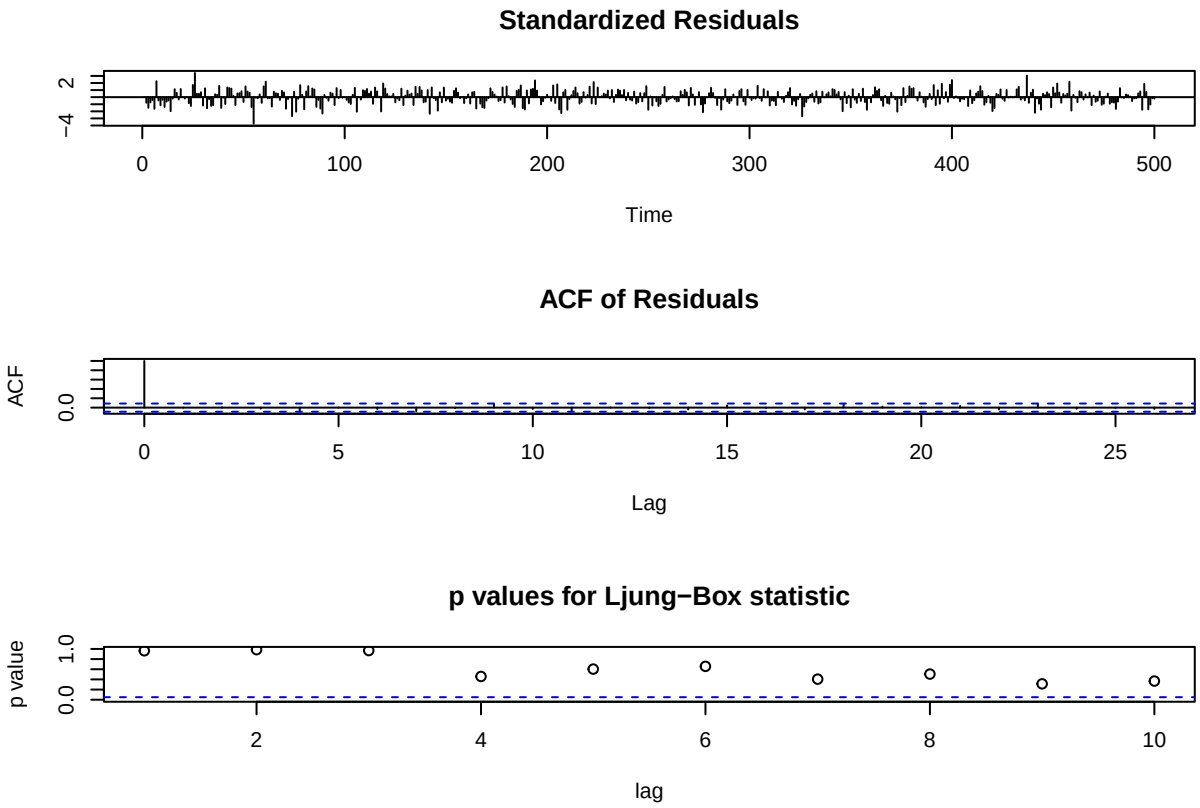
Now we want to look at the residuals of the model, to check how well it fits the data. We want it to look like white noise and the acf to only spike at lag 0.

```r
resA <- ts(modelA$residuals)

par(mfrow = c(1,2))
plot(resA)
acf(resA)
```



This is what we wanted to see, lets also look at a Ljung-Box test:

```r
tsdiag(modelA)
```

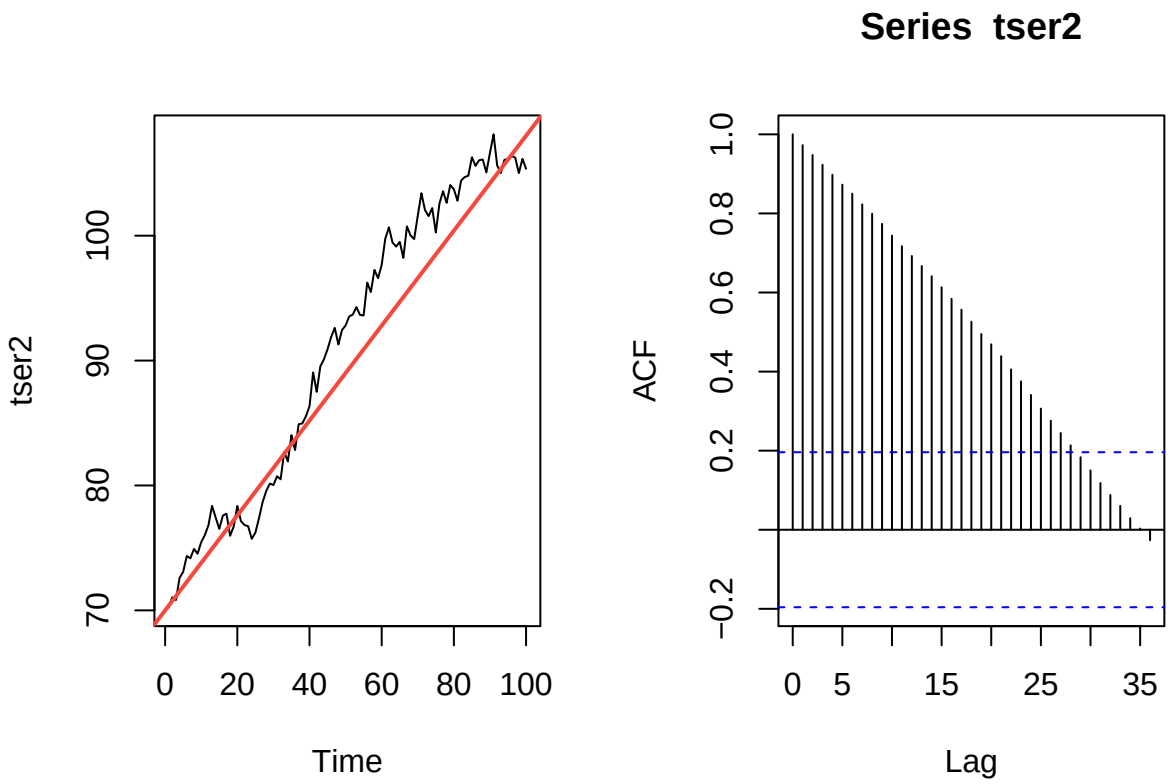


Just looking at the third plot, we have high early p-values. I am satisfied with the model choice of ARIMA(1,1,0).

Time Series 2

Initial observations

```
par(mfrow = c(1,2))
plot(tser2)
abline(a=70,b=0.38,lwd=2,col=2)
acf(tser2, lag.max = 36)
```



Initially, we can state that the time series has a linear trend. We can visualise an upwards sloped line in the original plot. We can also see this linear trend in the acf, we see a slow decrease as lag increases.

We can also see the series is non-stationary using the Kolmogorov-Smirnov test:

```
x2 <- tser2[1:50]
y2 <- tser2[51:100]

ks.test(x2, y2)
```

```
##
##  Exact two-sample Kolmogorov-Smirnov test
##
## data:  x2 and y2
## D = 1, p-value = 2.776e-15
## alternative hypothesis: two-sided
```

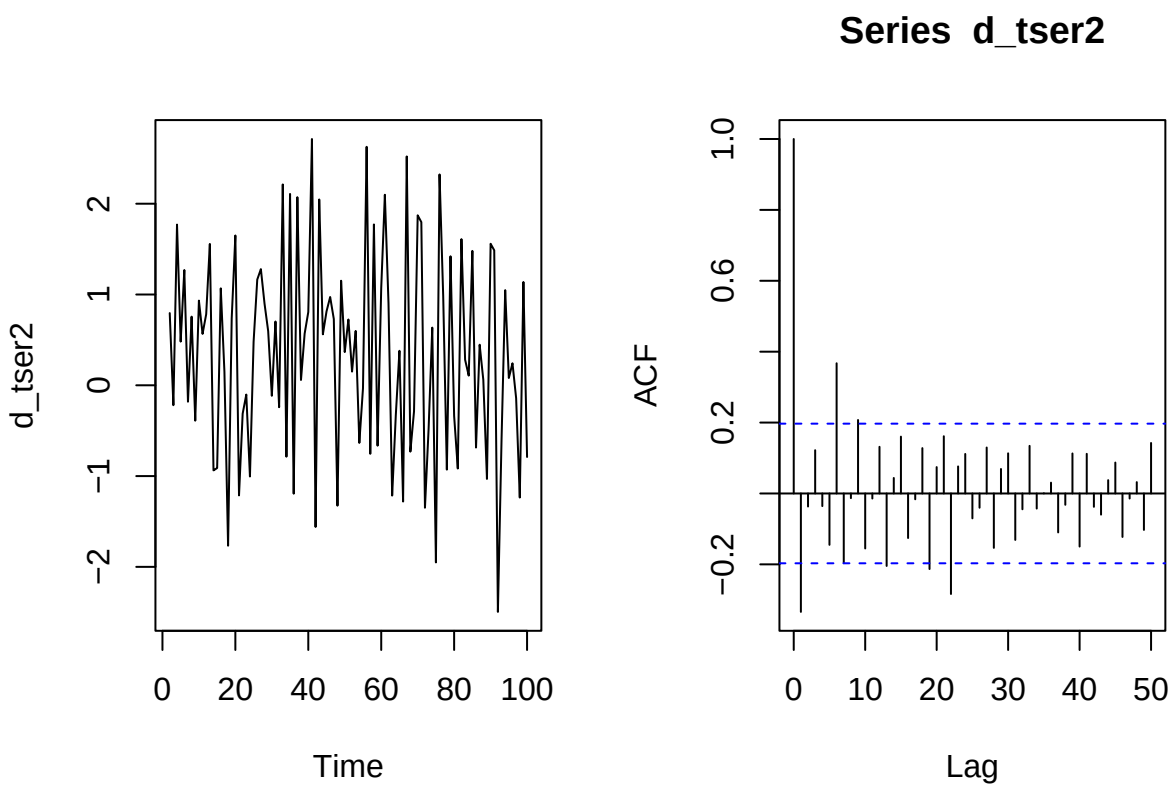
We have a very small p-value, so we can determine that time series 2 is non-stationary.

Trying to removing trend just by taking differences

Starting with taking the first difference:

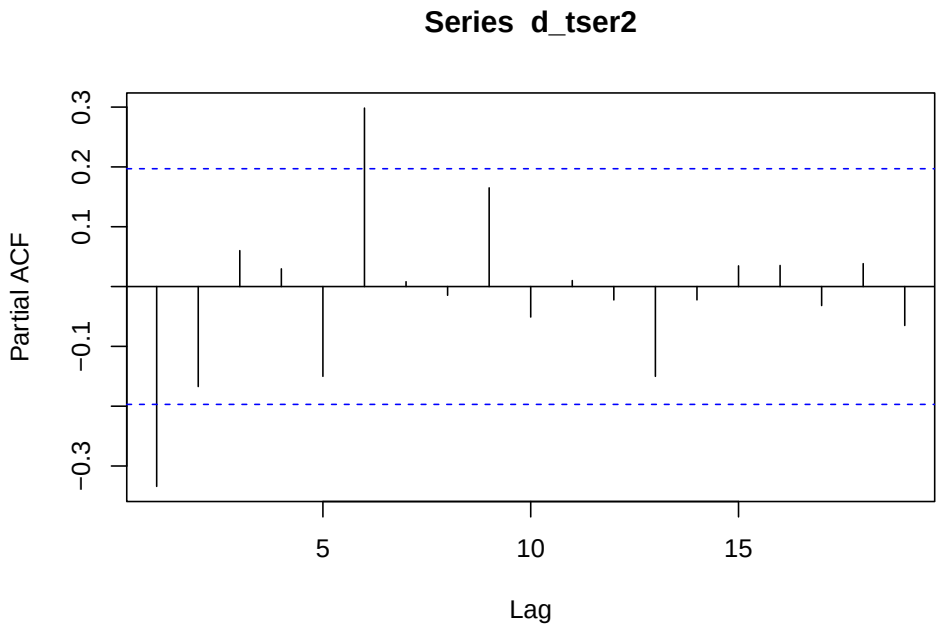
```
d_tser2 <- diff(tser2)

par(mfrow = c(1,2))
```

```
plot(d_tser2)
acf(d_tser2, lag.max = 50)
```



Also looking at the partial acf:

```
pacf(d_tser2)
```



Lets perform a Kolmogorov-Smirnov test:

```
x2 <- d_tser2[1:50]
y2 <- d_tser2[51:99]

ks.test(x2, y2)
```

```
##
##  Exact two-sample Kolmogorov-Smirnov test
##
## data:  x2 and y2
## D = 0.23224, p-value = 0.1134
## alternative hypothesis: two-sided
```

We have a larger p-value, suggesting maybe the differenced series is stationary. Lets look further with the 'adf.test'

```
adf.test(d_tser2)
```

```
## Warning in adf.test(d_tser2): p-value smaller than printed p-value

##
##  Augmented Dickey-Fuller Test
##
## data:  d_tser2
## Dickey-Fuller = -5.45, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

Here we have a small p-value for the default lag order 4, but this changes when we increase the lag-order:

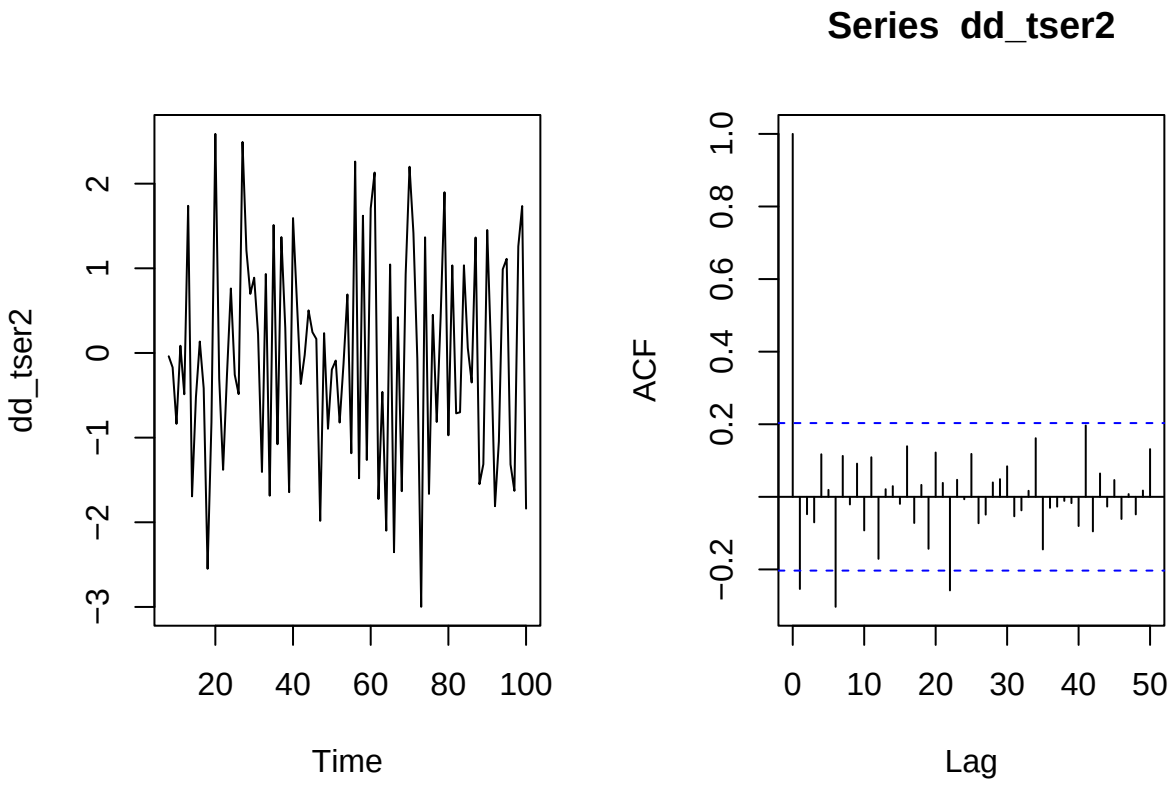
```
adf.test(d_tser2, k = 6)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  d_tser2
## Dickey-Fuller = -3.1223, Lag order = 6, p-value = 0.112
## alternative hypothesis: stationary
```

This has a much bigger p-value, we cannot say the series is stationary.

We will try differencing again, this time including lag = 6 due to significant spikes in the acf and pacf of the first difference series.

```
dd_tser2 <- diff(d_tser2, lag = 6)

par(mfrow = c(1,2))
plot(dd_tser2)
acf(dd_tser2, lag.max = 50)
```



Let's check if we now have stationarity:

```
x2 <- dd_tser2[1:49]
y2 <- dd_tser2[50:98]

ks.test(x2, y2)
```

```
##
##  Exact two-sample Kolmogorov-Smirnov test
##
## data:  x2 and y2
## D = 0.20269, p-value = 0.251
## alternative hypothesis: two-sided
```

The p-value is larger once again. We're getting closer. Checking with an 'adf.test'

```
adf.test(dd_tser2)
```

```
## Warning in adf.test(dd_tser2): p-value smaller than printed p-value
```

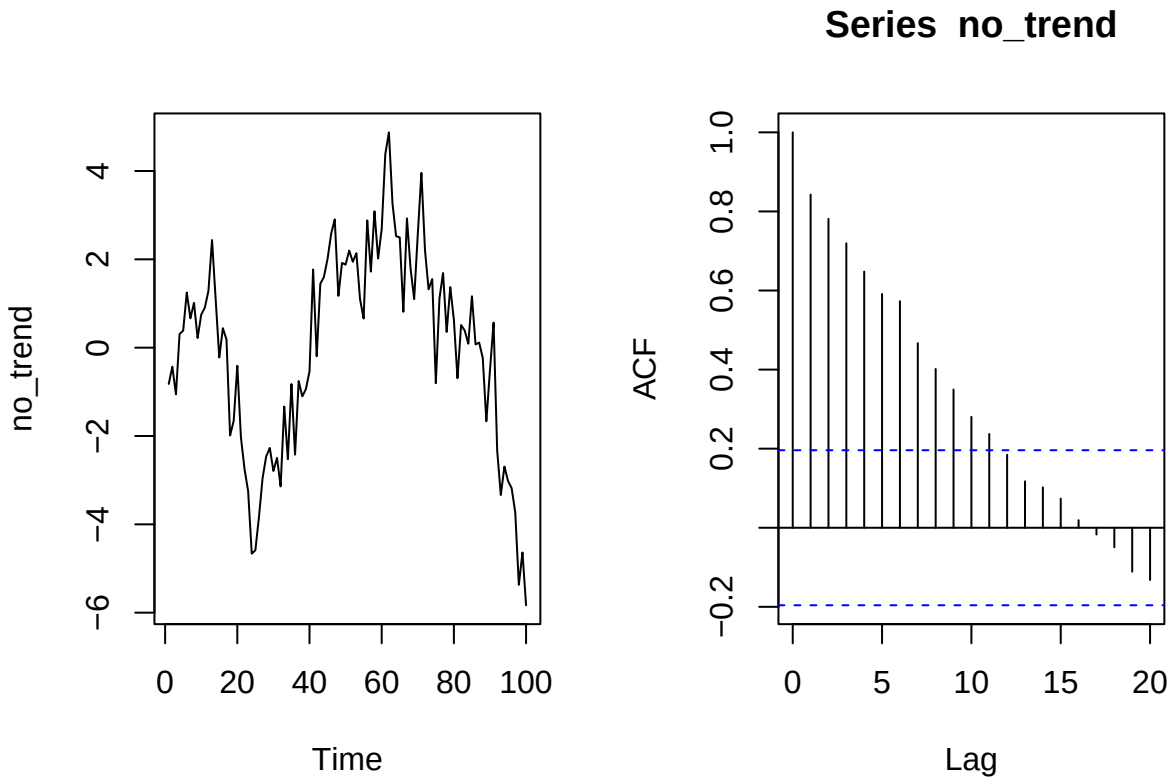
```
##
##  Augmented Dickey-Fuller Test
##
## data:  dd_tser2
## Dickey-Fuller = -4.1797, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

We have a small p-value here. Even when we change the lag-order, k, around.

The p-value is large for the Kolmogorov-Smirnov test (null hypothesis holds: stationary), but small for the adf test (stationary). So when the series transform consists of a first difference at lag 1, followed by a first difference at lag 6, we get stationarity. However, this disagrees with the constraint for Time Series 2 that $d = 0, 1$. Lets try removing the trend using 'lm'

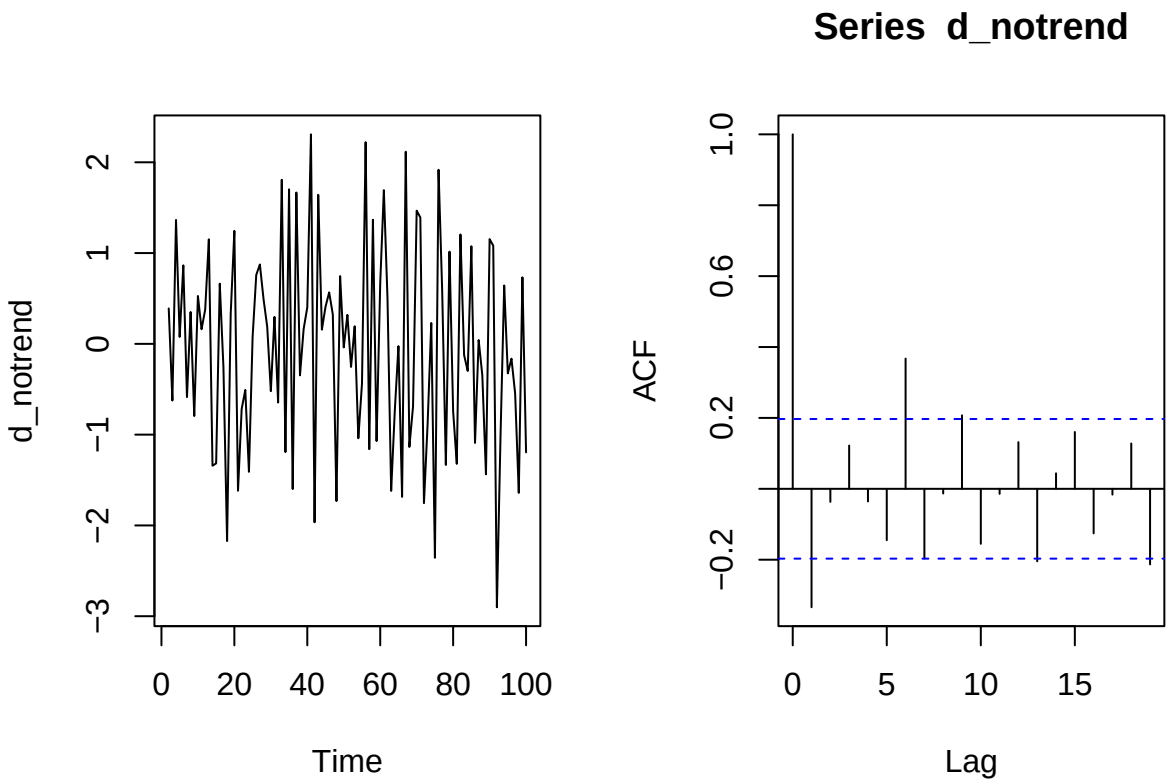
Removing trend with 'lm'

```
time <- 1:100 #length of tser2
lin_model <- lm(tser2~time)
fitted_vals <- predict(lin_model)
no_trend <- tser2 - fitted_vals

par(mfrow = c(1,2))
plot(no_trend)
acf(no_trend)
```



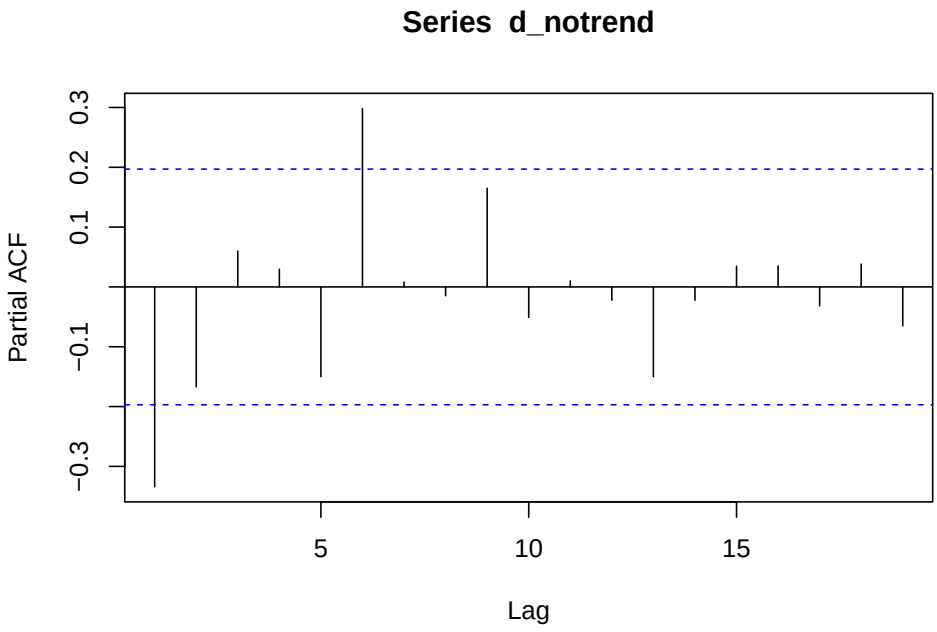
Lets difference this series:

```
d_notrend <- diff(no_trend)

par(mfrow = c(1,2))
plot(d_notrend)
acf(d_notrend)
```



This looks more like White Noise. We have some spikes in the acf at lag 0, 1 and 6. lets look at the pacf:

```
pacf(d_notrend)
```



Again we have large spikes at lag 0,1 and 6.

```
x2 <- d_notrend[1:50]
y2 <- d_notrend[51:99]

ks.test(x2, y2)
```

```
##
##  Exact two-sample Kolmogorov-Smirnov test
##
## data:  x2 and y2
## D = 0.23224, p-value = 0.1134
## alternative hypothesis: two-sided
```

We dont have a small p-value. The series could be stationary.

```
adf.test(d_notrend)
```

```
## Warning in adf.test(d_notrend): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  d_notrend
## Dickey-Fuller = -5.45, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

We get a small p-value here. I can come to the conclusion, that with the trend removed and then the first difference taken at lag 1, we get a stationary series. So we know that $d = 1$

Fitting the model:

Lets try some values for p and q:

```
model2A<- arima(tser2, order = c(1, 1,2 ) )
AIC(model2A)
```

```
## [1] 301.1683
```

```
model2B<- arima(tser2, order = c(2, 1,2 ) )
AIC(model2B)
```

```
## [1] 306.4317
```

```
model2C<- arima(tser2, order = c(2, 1,1 ) )
AIC(model2C)
```

```
## [1] 304.4968
```

Model 2A with 'ARIMA(1,1,2)' has the lowest value for AIC so I would choose this as my preferred model.

Estimation of model parameters

From Model 2A we get the estimates:

```
print(model2A)
```

```
##
## Call:
## arima(x = tser2, order = c(1, 1, 2))
##
## Coefficients:
##          ar1      ma1     ma2
##       0.9861  -1.3634  0.4235
## s.e.  0.0219   0.0929  0.0997
##
## sigma^2 estimated as 1.119:  log likelihood = -146.58,  aic = 301.17
```

Where we can see that the estimates are not near zero, so they are significant. I will again calculate a 95% CI:

```
model2A$coef-2*sqrt(diag(model2A$var.coef))
```

```
##        ar1        ma1        ma2
##  0.9423216 -1.5492615  0.2242071
```

```
model2A$coef+2*sqrt(diag(model2A$var.coef))
```

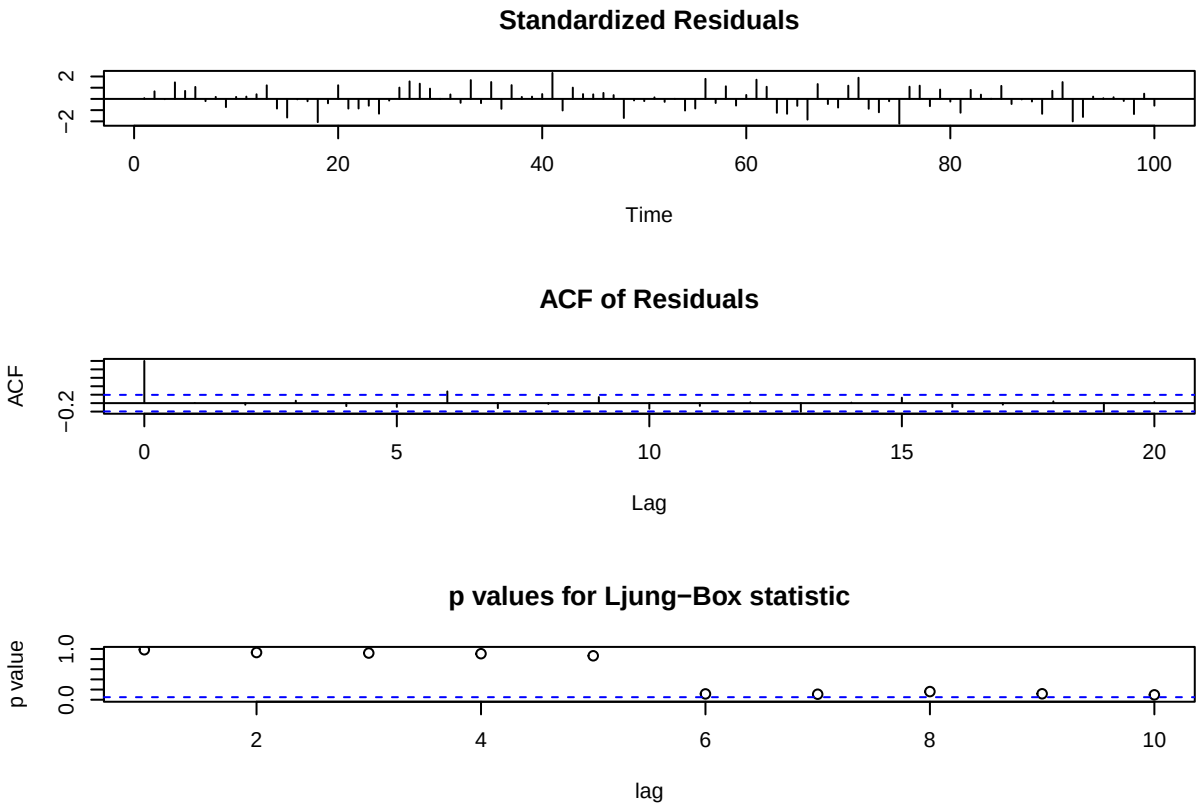
```
##        ar1        ma1        ma2
##  1.0299441 -1.1774792  0.6228188
```

We find:

$$\alpha_1 \in [0.942, 1.03] \quad \beta_1 \in [-1.55, -1.18] \quad \beta_2 \in [0.224, 0.623]$$

Agreement with data

```
tsdiag(model2A)
```



We see high early p-values, and the only high spike in the acf is at lag 0. I am satisfied with the choice of ARIMA(1,1,2)