Time Series - Assignment 1

Damian Baeza

14 February 2018

Question 1

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

```
require(dplyr)
```

```
## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

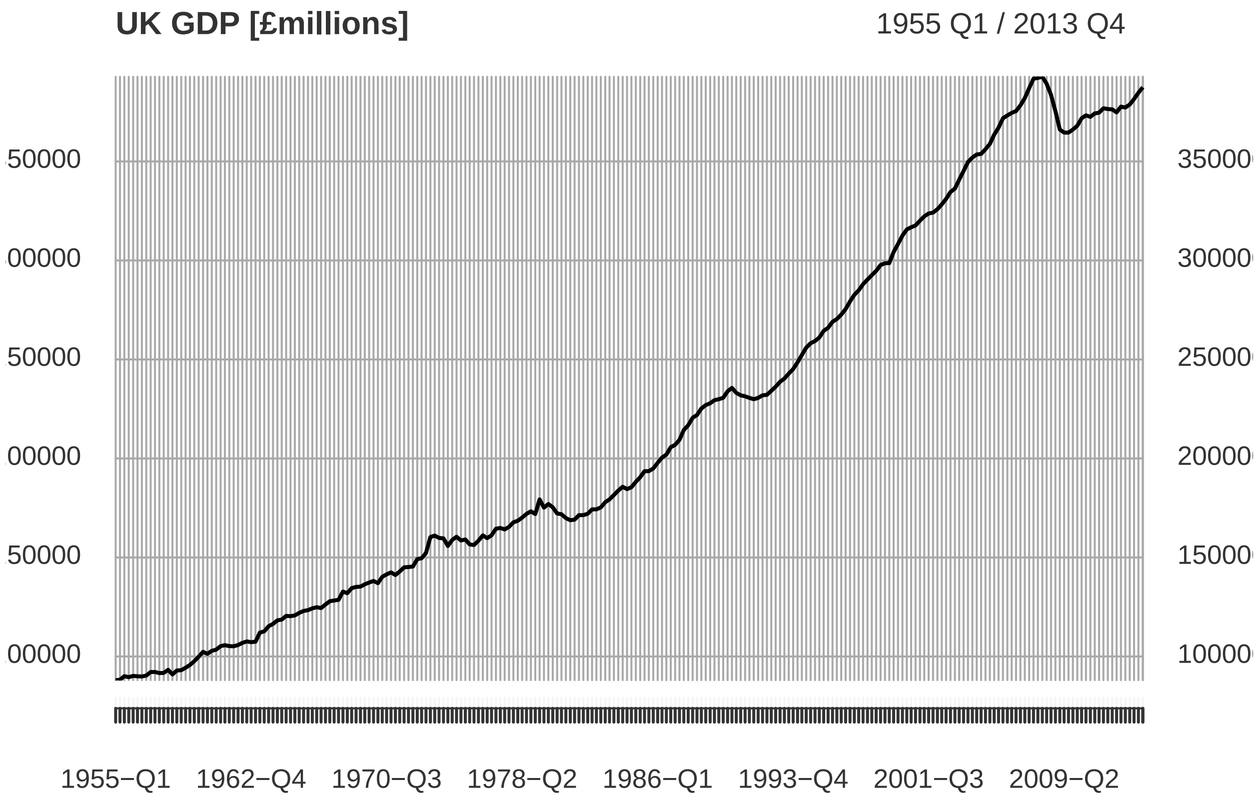
```
require(zoo)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
require(xts)
```

```
## Loading required package: xts

##
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
load("UKGDP.RData")

yq <- paste(UKGDP$V1, UKGDP$V2)
df.yq <- cbind(UKGDP, as.yearqtr(yq))
df.xts <- xts(df.yq$V3, order.by = df.yq$`as.yearqtr(yq)`)

plot(df.xts, main = "UK GDP [£millions]")
```



As can be observed on the above plot, the gross domestic product of the UK has an trend that varies according to t (the year). This implies that the UK GDP does not have a constant trend, therefore the expected value of X_t (UK GDP) is not constant. The aforementioned indicates that using an ARMA(p,q) model would not be appropiate.

Question 2

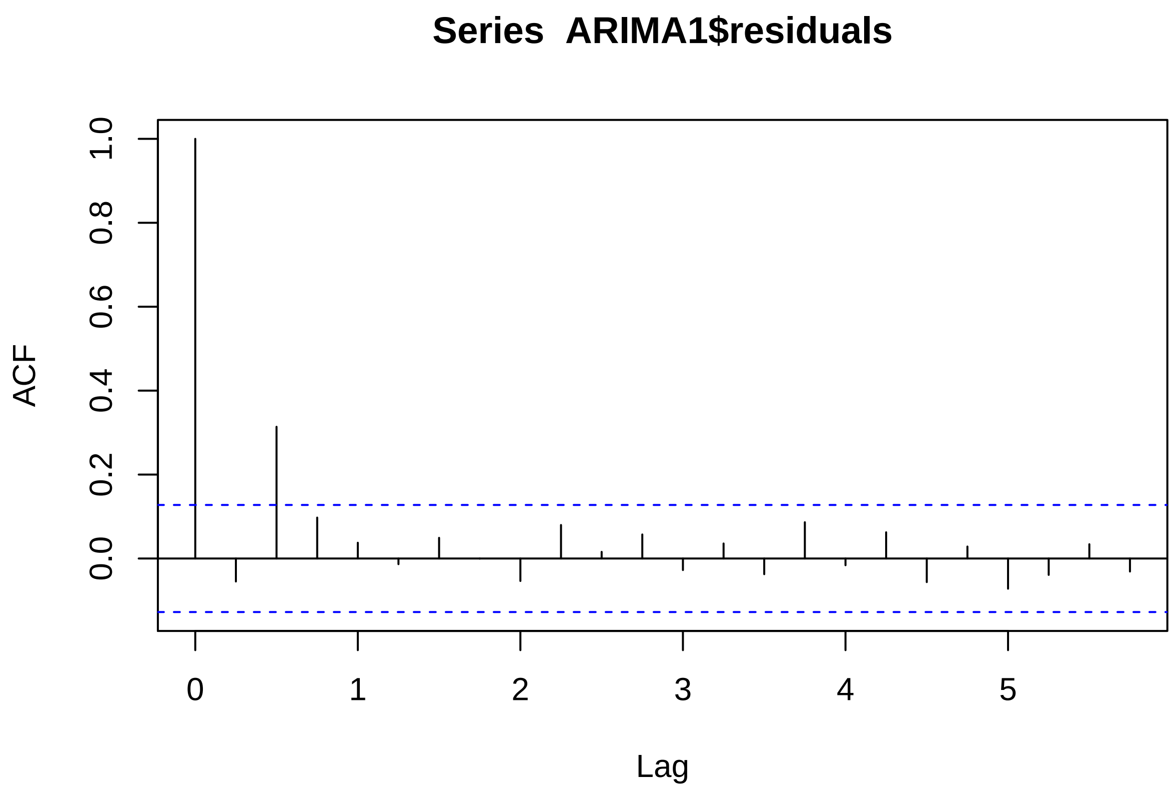
```
ARIMA1.Order <- c(0, 1, 1)
ARIMA2.Order <- c(1, 1, 0)
ARIMA3.Order <- c(1, 1, 1)

ARIMA1 <- arima(x = df.xts, order = ARIMA1.Order)
ARIMA2 <- arima(x = df.xts, order = ARIMA2.Order)
ARIMA3 <- arima(x = df.xts, order = ARIMA3.Order)

AICs <- c(ARIMA1$aic, ARIMA2$aic, ARIMA3$aic)

AIC.Probs <- exp((min(AICs) - AICs)/2)
```

(a) Model selection

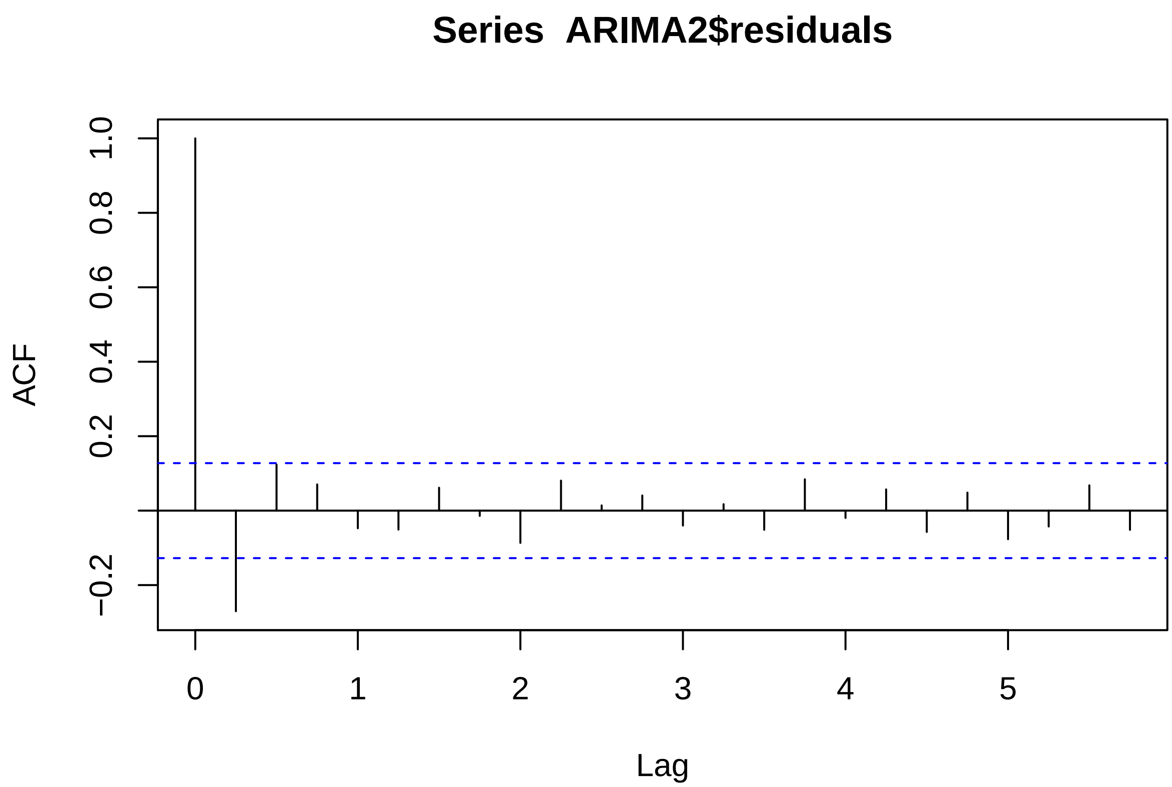
The ARIMA(1,1,1) model presents the lowest AIC value and further to that fact, the ARIMA(0,1,1) and ARIMA(1,1,0) models are $2.261 \cdot 10^{-13}$ and $1.707 \cdot 10^{-5}$ times as probable as the ARIMA(1,1,1) model to minimise the information loss.

The correlograms of the residuals are depicted in the following plots.

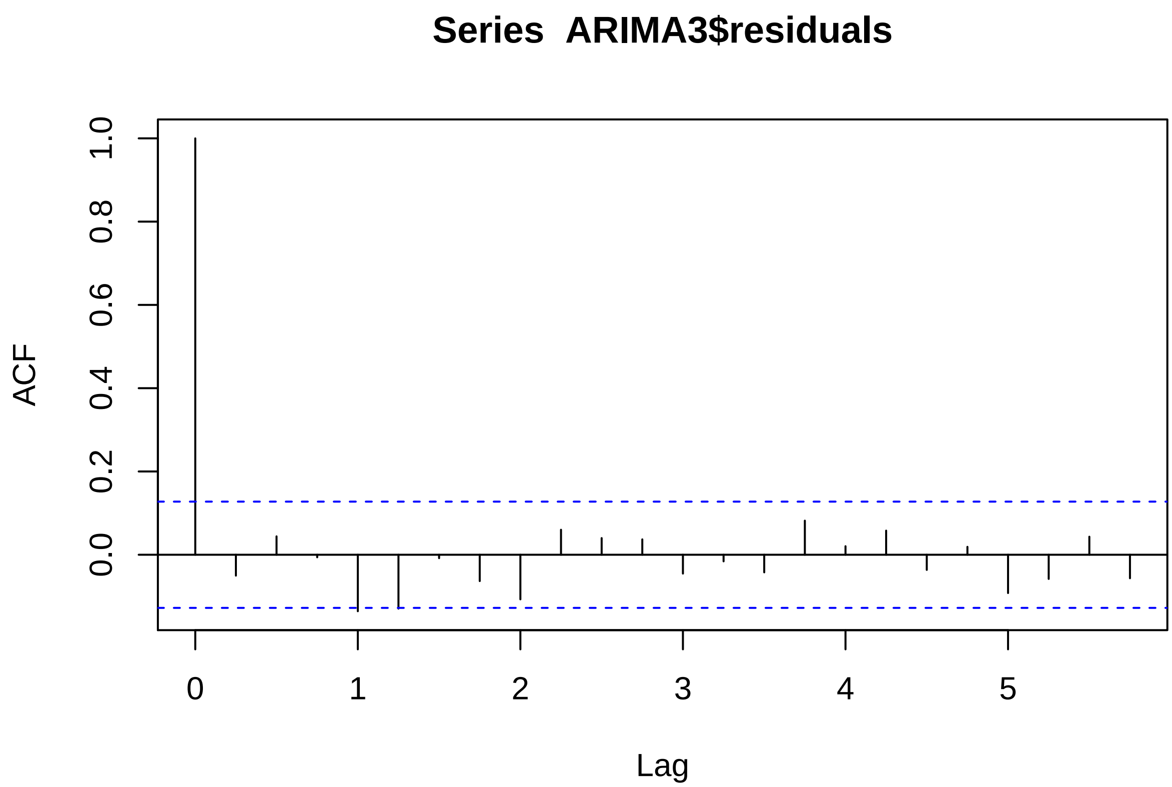
```r
# Correlogram of ARIMA(0,1,1) model
acf(ARIMA1$residuals)
```



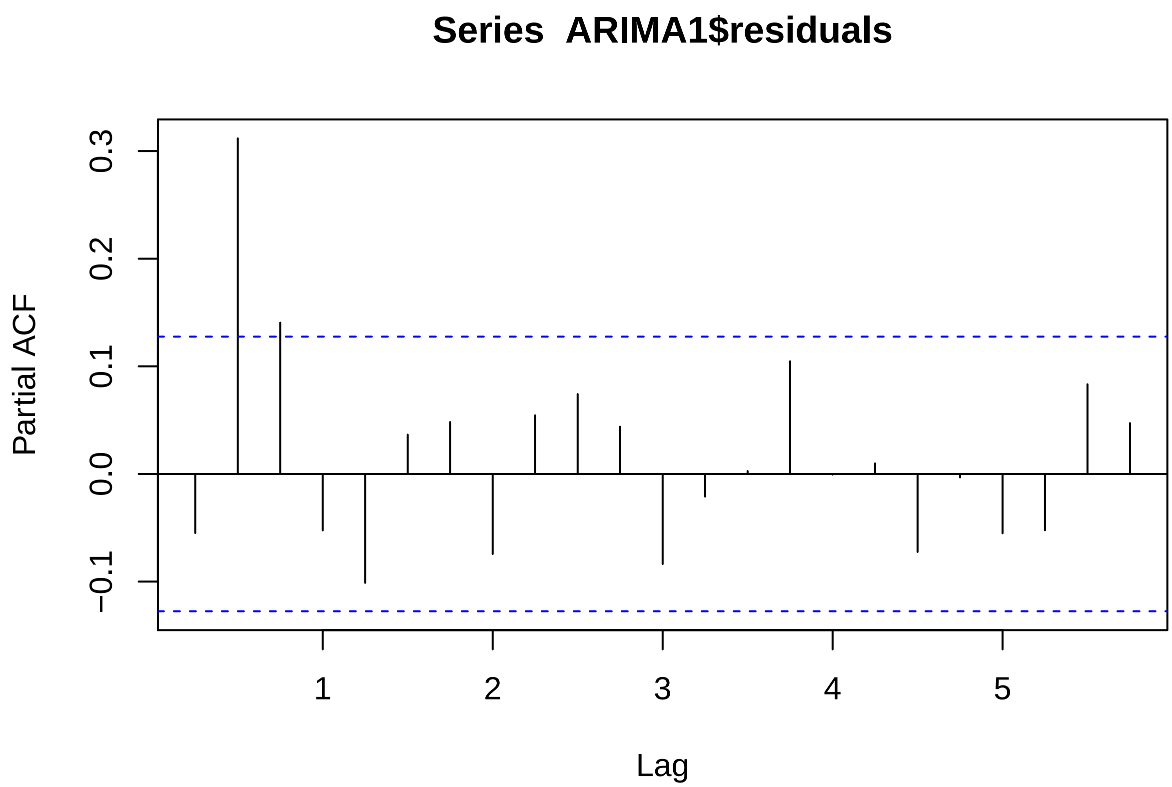

```r
# Correlogram of ARIMA(1,1,0) model
acf(ARIMA2$residuals)
```



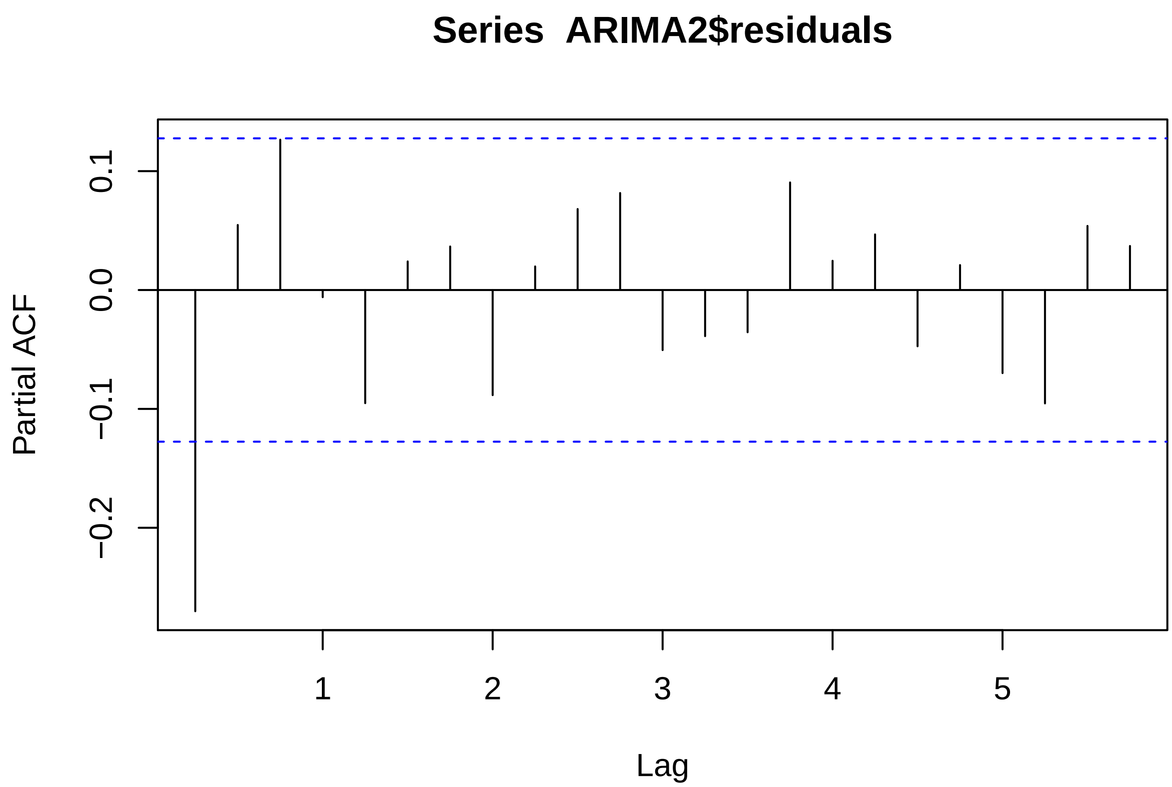

```
# Correlogram of ARIMA(1,1,1) model
acf(ARIMA3$residuals)
```



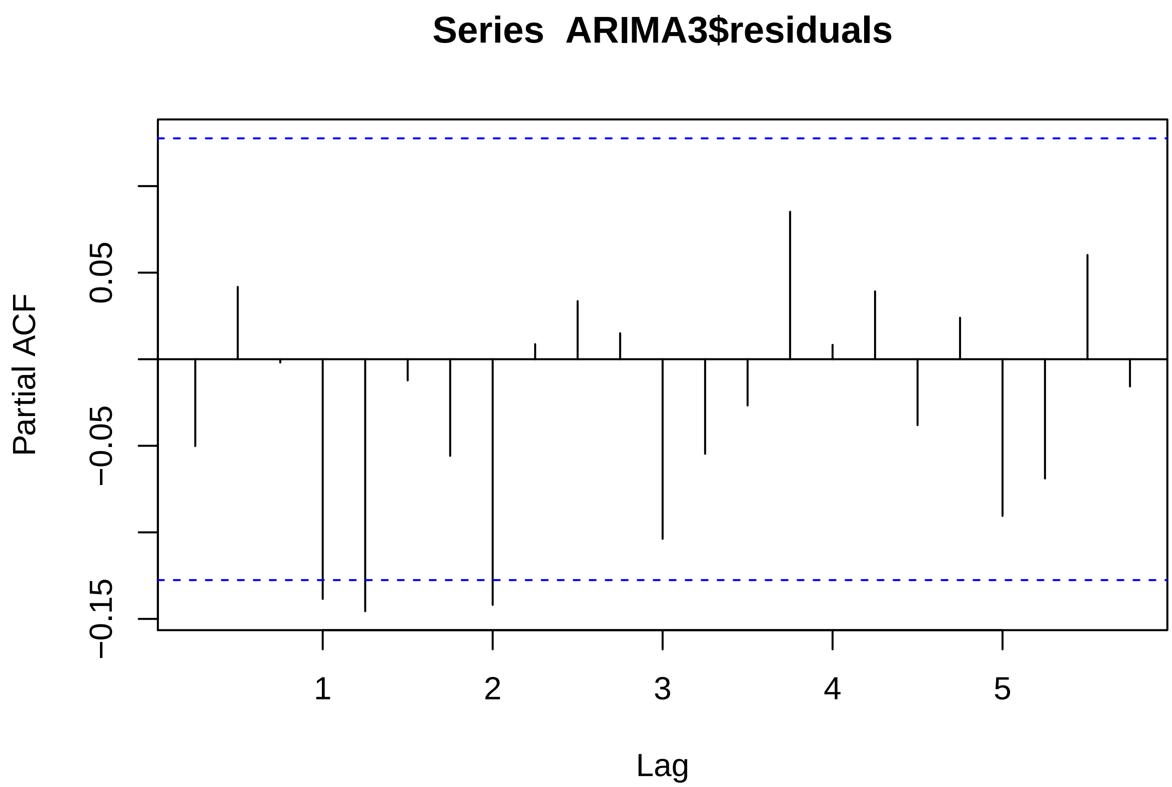

```
# PACF of ARIMA(0,1,1) model
pacf(ARIMA1$residuals)
```




```r
# PACF of ARIMA(1,1,0) model
pacf(ARIMA2$residuals)
```




```r
# PACF of ARIMA(1,1,1) model
pacf(ARIMA3$residuals)
```




```
#Ljung-Box Test Statistics for lag = 1:50

lb <- function(lag, x, type = "Ljung-Box"){
  Box.test(x = x, lag = lag, type = type)$p.value
}

Lags <- 1:50
n.Lags <- length(Lags)

Model.p.values <- data.frame(ARIMA1 = numeric(n.Lags),
                             ARIMA2 = numeric(n.Lags),
                             ARIMA3 = numeric(n.Lags),
                             lag = Lags)

Model.p.values$ARIMA1 <- sapply(Lags, lb, x = ARIMA1$residuals)
Model.p.values$ARIMA2 <- sapply(Lags, lb, x = ARIMA2$residuals)
Model.p.values$ARIMA3 <- sapply(Lags, lb, x = ARIMA3$residuals)

# ggplot(data = Model.p.values, aes(x = Lags)) +
#   geom_line()
```

From the above correlograms it is possible to visualize that the residuals are uncorrelated for the ARIMA(1,1,1) model. This indicates that the estimated residuals are white noise.

(b) Chosen model

The fitted ARIMA(1,1,1) model can be written as follows:

$$X_t = \alpha_1 X_{t-1} + Z_t - \theta_1 Z_{t-1}$$