Assignment 2 - ARMA Processes and Seasonal Processes

Christian Glissov, s146996

March 15, 2018

DTU Compute - Institute for Mathematics and Computer science
Time Series Analysis 02417 March 15, 2018
Lasse Engbo Christiansen

Collaborated with Thomas Ørkild, s154433 and Frederik Boe Hüttel, s151689

Technical University of Denmark

Contents

Opgave 2.1

Opgave 2.1.1

A process has been given by:

$$X_t = \varepsilon_t + \varepsilon_{t-1} + \varepsilon_{t-2} + \varepsilon_{t-3} \tag{1}$$

The standard form of the time series is:

$$X_t = (1 + B + B^2 + B^3)\varepsilon_t \tag{2}$$

This is a MA(q) process with $q = 3$, MA(3), where ε_t is a white noise process and for this case it is assumed $\sigma_\varepsilon = 0.1$. The second order moment of the time series is given by:

$$\begin{aligned}
\mu(t) &= \mathbb{E}(X_t) \\
&= \mathbb{E}(\varepsilon_t + \varepsilon_{t-1} + \varepsilon_{t-2} + \varepsilon_{t-3}) \\
&= \mathbb{E}(\varepsilon_t) + \mathbb{E}(\varepsilon_{t-1}) + \mathbb{E}(\varepsilon_{t-2}) + \mathbb{E}(\varepsilon_{t-3}) \\
&= 0
\end{aligned}$$

Since it is assumed that ε_t is white noise, by definition it is given that $\mathbb{E}(\varepsilon_t) = 0$. The autocovariance function can be found by using (5.65) in [1]. From this it is found:

$$\gamma(k) = \begin{cases} (1 + \theta_1^2 + \theta_2^2 + \theta_3^2)\sigma_\varepsilon^2 & k = 0 \\ (\theta_1 + \theta_2^2 + \theta_3^2)\sigma_\varepsilon^2 & |k| = 1 \\ (\theta_2 + \theta_3^2)\sigma_\varepsilon^2 & |k| = 2 \\ \theta_3\sigma_\varepsilon^2 & |k| = 3 \\ 0 & |k| = 4, 5, \ldots \end{cases}$$

Where $\theta_1 = 1$, $\theta_2 = 1$ and $\theta_3 = 1$ in this case and $\sigma_\varepsilon^2 = 0.01$. This gives the auto-covariance of each lag:

$$\gamma(k) = \begin{cases} 0.04 & k = 0 \\ 0.03 & |k| = 1 \\ 0.02 & |k| = 2 \\ 0.01 & |k| = 3 \\ 0 & |k| = 4, 5, \ldots \end{cases}$$

The variance is then given by:

$$\mathrm{Cov}(X_t, X_t) = \mathrm{Var}(X_t) = \gamma(0) = 0.04 \tag{3}$$

The auto-correlation is then found by simply normalizing the auto-covariance with $\gamma(0)$

$$\rho(k) = \begin{cases} 1.00 & k = 0 \\ 0.75 & |k| = 1 \\ 0.50 & |k| = 2 \\ 0.25 & |k| = 3 \\ 0 & |k| = 4, 5, ... \end{cases} \tag{4}$$

Opgave 2.1.2

By Theorem 5.8 (i) in [1], a MA(q) process is always stationary. Therefore the process, X_t, in eq. 1 is stationary. To check if the process is invertible theorem 5.8 (ii) is used, by using the transfer function and solving $\theta(z^{-1}) = 0$ the unit roots can be found:

$$\theta(z^{-1}) = 1 + z^{-1} + z^{-2} + z^{-3}, \quad z \in \{1, i, -i\}$$

For the process to be invertible $|z| < 1$ must be satisfied. However $|z| = 1$, therefore the process is not invertible, it is not within the unit circle. This means the process can't be inverted into an AR process, this means past innovations can't be observed and the model can't be used for forecasting the dependent variable.

Opgave 2.1.3

Simulating 10 realisations with 200 observations each is simulated using the MA(3) process, X_t, with the white noise, $\sigma_\varepsilon = 0.1$ and $\mathbb{E}(\varepsilon_t) = 0$, in *R*. For all simulations in this report a seed of 100 was set, if one wishes to replicate the results. The simulations can be seen in figure (1)

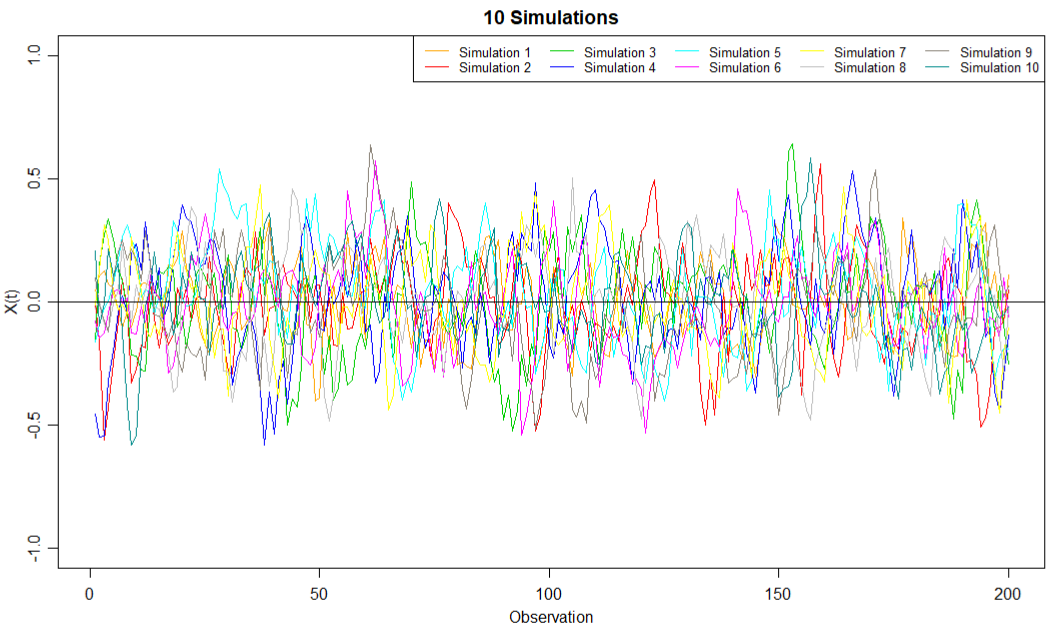

Figure 1: Simulation of 10 realisations of the process in eq. 1

One may notice each realisation fluctuate around the black line at 0, this is because of $\mathbb{E}(X_t) = 0$. Finally it can be seen that the process is also normally distributed due to the sum of normally distributed white noise, which is expected:

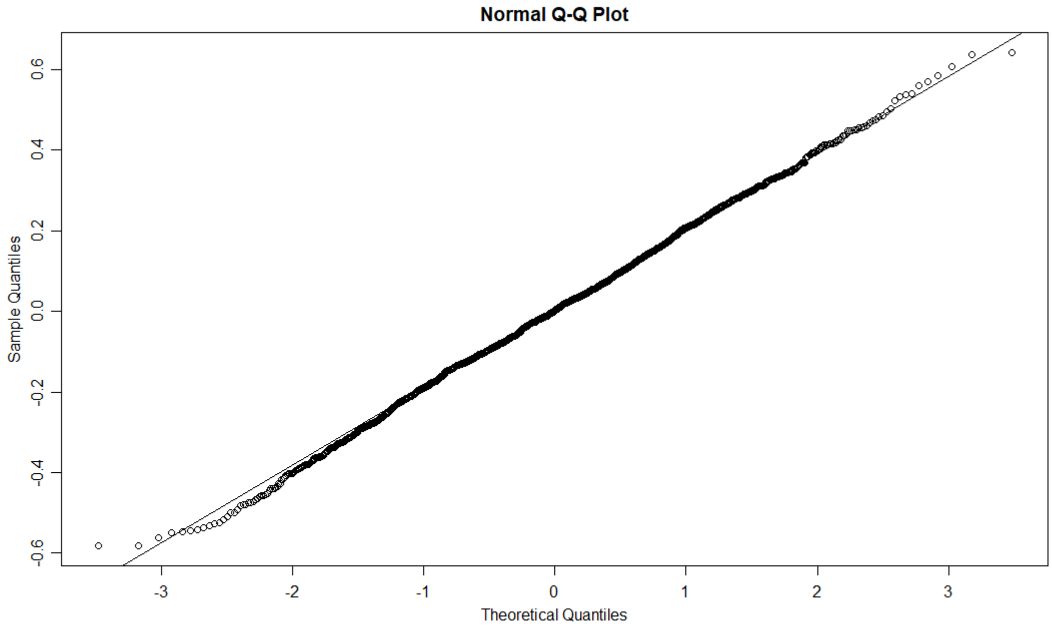

Figure 2: The simulated processes is normally distributed.

By using R to count all values between 0.2 and -0.2, which was the theoretical standard deviation found in eq. 3, then 1371 out of the 2000 total simulations (approx. 68%) of the simulations lies within 1 standard deviation away from $\mathbb{E}(X_t) = 0$. Given that the observations are normally distributed it is reasonable to assume the average variance will be around 0.04, like was found in eq. 3.

Opgave 2.1.4

The ACF of each simulation can be seen in figure (3)

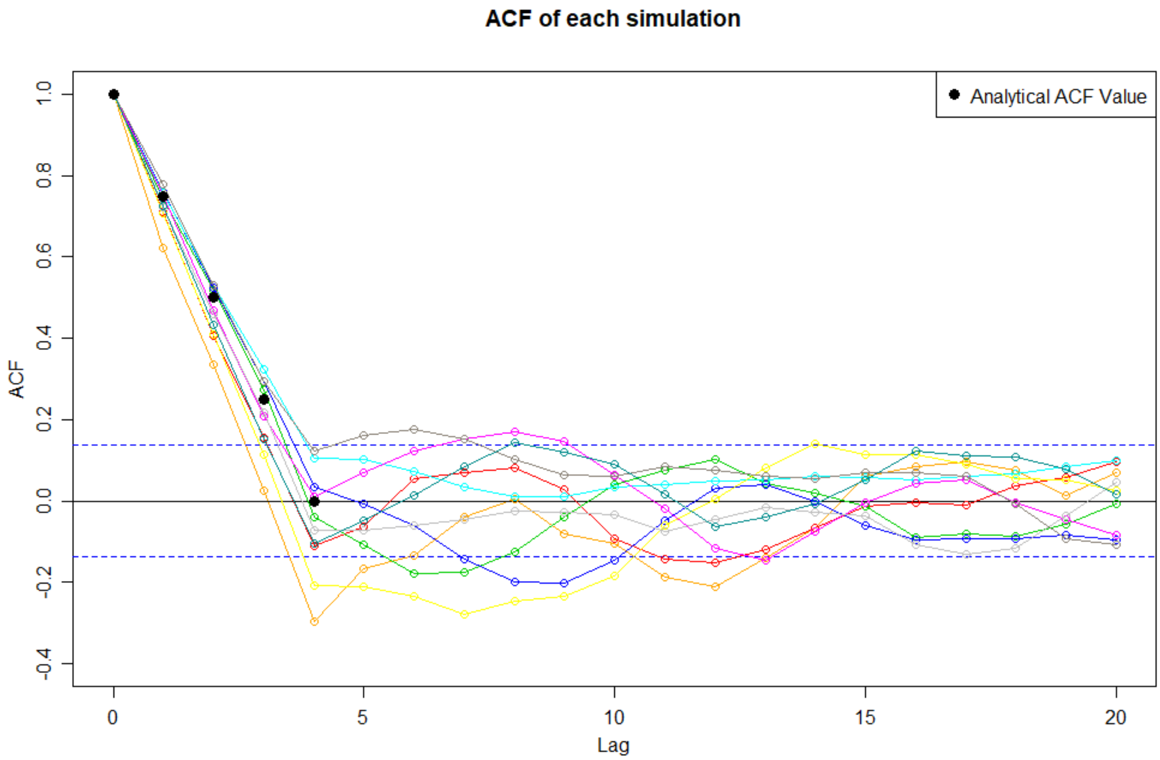

Figure 3: ACF of the simulation of the 10 realisations.

Some auto-correlation can be seen for the first 3 lags, the remaining lags seems to be noise. The analytical auto-correlation found in eq. 4, can be seen at the black points. For a lag greater than 3, $\rho(k) = 0$. It's seen that the simulations of the realisations are close to the theoretical value as expected. It seems like the auto-correlation $\rho(k) = 0$ for $k > 3$ in accordance with table 6.1, page 155 in [1].

Opgave 2.1.5

In figure (4), the PACF can be seen for X_t:

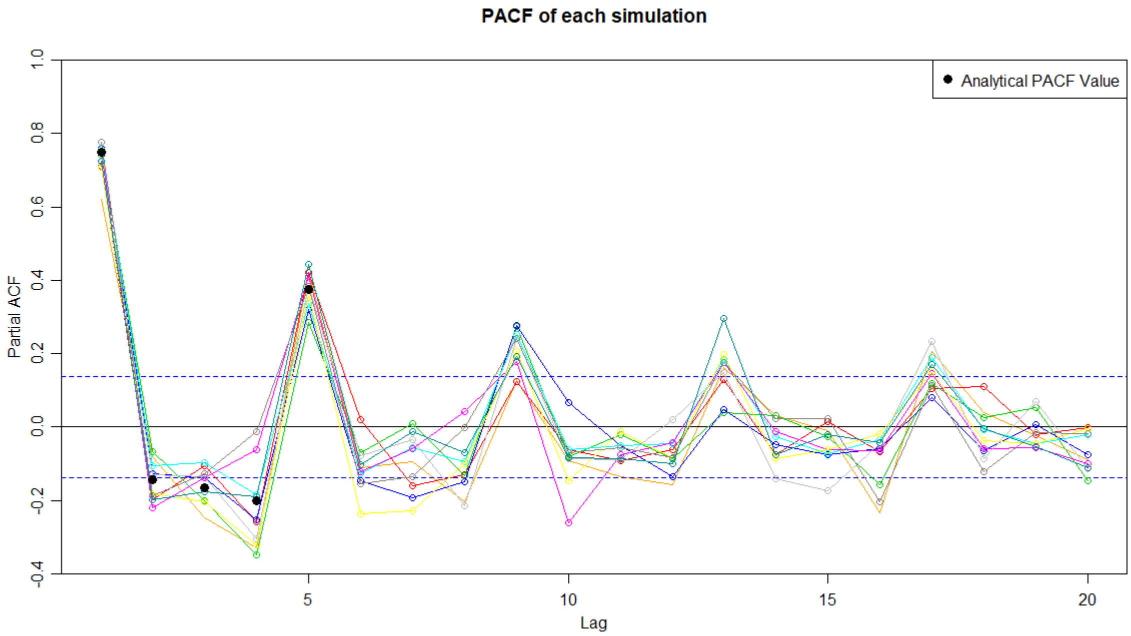

Figure 4: PACF of the simulation of the 10 realisations.

Once again in accordance with table 6.1, page 155 in [1], the PACF is dominated by damped exponential and sine functions, this is what one would expect from a MA(3) process. The numerical values of the PACF is close to the analytical values of the first 5 lags. The theoretical values was found by using eq. (5.91), page 124 in [1] based on the Yule-Walker equations, and is given by:

Table 1: PACF values for the first 5 lags

ϕ_{11}	0.750
ϕ_{22}	-0.143
ϕ_{33}	-0.167
ϕ_{44}	-0.200
ϕ_{55}	0.375

Look in appendix to see calculation.

Opgave 2.1.6

Once again, looking into the variance of each simulation of the realisations:

Realisation i	Variance
1	0.02461
2	0.03758
3	0.04354
4	0.04921
5	0.04269
6	0.03657
7	0.03790
8	0.03874
9	0.04504
10	0.03408

The average sample variance of the above values is approximately 0.039. This is very close to the analytical variance of 0.04. It is also the expected variance, the process consists of 4 white noise terms, each having a variance of 0.01, and due to the coefficients are all 1, then the variance is constant within the process and the sum of the variance is 0.04.

Summary

The second order moments of the MA(3) model have been found. The ACF and PACF follows the same rules as given in table 6.1 [1], this was discussed in section (2.1.5) and (2.1.4). It can be seen that the numerical result is close to the analytic results which is expected. Finally the variance of each simulation seems to converge towards the theoretical variance an explanation of this was given in section (2.1.6).

Opgave 2.2

The following multiplicative seasonal model has been given:

$$(1 - 0.5B + 0.3B^2)(1 - 0.9B^{12})(Y_t - \mu) = \varepsilon_t$$

The seasonal model can be rewritten as:

$$\begin{aligned}
\varepsilon_t &= (1 - 0.9B^{12} - 0.5B + 0.45B^{13} + 0.3B^2 - 0.27B^14)(Y_t - \mu) \\
&= Y_t - \mu - 0.9B^{12}(Y_t - \mu) - 0.5B(Y_t - \mu) + 0.45B^{13}(Y_t - \mu) \\
&\quad + 0.3B^2(Y_t - \mu) - 0.27B^14(Y_t - \mu) \\
&= Y_t - \mu - 0.9(Y_{t-12} - \mu) - 0.5(Y_{t-1} - \mu) + 0.45(Y_{t-13} - \mu) \\
&\quad + 0.3(Y_{t-2} - \mu) - 0.27(Y_{t-14} - \mu) \qquad (5)
\end{aligned}$$

Since μ is not dependent of time, it is simply ignored by the backshift operator, B. Isolating Y_t the process becomes:

$$Y_t = 0.08\mu + 0.9Y_{t-12} + 0.5Y_{t-1} - 0.45Y_{t-13} - 0.3Y_{t-2} + 0.27Y_{t-14} + \varepsilon_t \tag{6}$$

Where $\mu = 55$. From eq. 6 and slide 31 lecture 5, the prediction values for $t = 2017\text{M}12$ and $t = 2018\text{M}1$ can be found. Both values were estimated in R:

$$\hat{Y}_{2017\text{M}12} = 58.71$$
$$\hat{Y}_{2018\text{M}1} = 58.256$$

The prediction intervals are found by using (5.150) and (5.151), page 137 in [1]. The variance of the prediction error is, where it's given that $\sigma_\varepsilon^2 = 0.5^2$:

$$\begin{aligned}\sigma_1^2 &= \sigma_\varepsilon^2 \\ &= 0.25 \\ \sigma_2^2 &= (1+\phi_1^2)\sigma_\varepsilon^2 \\ &= (1+0.5^2)0.25 \\ &= 0.3125\end{aligned}$$

The values for the 95% prediction interval are:

$$\begin{aligned}\hat{Y}_{2017\text{M}12} \pm u_{\frac{\alpha}{2}}\sqrt{\sigma_1^2} &= 58.71 \pm 1.96\sqrt{0.25} \\ &= [59.69, 57.73] \\ \hat{Y}_{2018\text{M}1} \pm u_{\frac{\alpha}{2}}\sqrt{\sigma_1^2} &= 58.256 \pm 1.96\sqrt{0.3125} \\ &= [59.352, 57.160]\end{aligned}$$

The table containing the relevant values is:

Table 2: The expected values of each date.

Date	Expected	Upper 95%	Lower 95%
2017M12	58.71	59.69	57.73
2018M1	58.256	59.352	57.160

In figure 5 the out-of-sample 2-step predictions and the 95% prediction interval can be seen. One might notice that the prediction follows the down trend when comparing the 2-step predictions with the value of month = 3 and month = 4, this is due to mainly the large negative coefficient of the seasonality of the model such as 0.9 for Y_{t-12}.

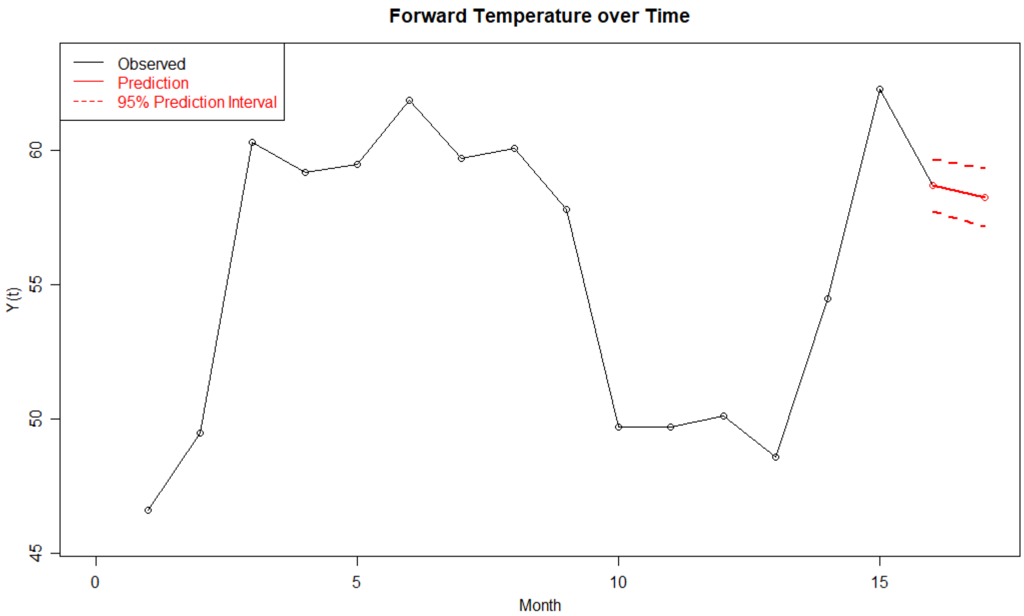

Figure 5: The prediction and the 95% prediction interval.

The prediction interval is not very large and one should trust the short forecast given the previous observations.

To illustrate the stationarity of the process one can look at figure 6. The predictions are stable and the prediction will asymptotic converge towards the estimated expected value $\mu = 55$, this is illustrated in the figure below:

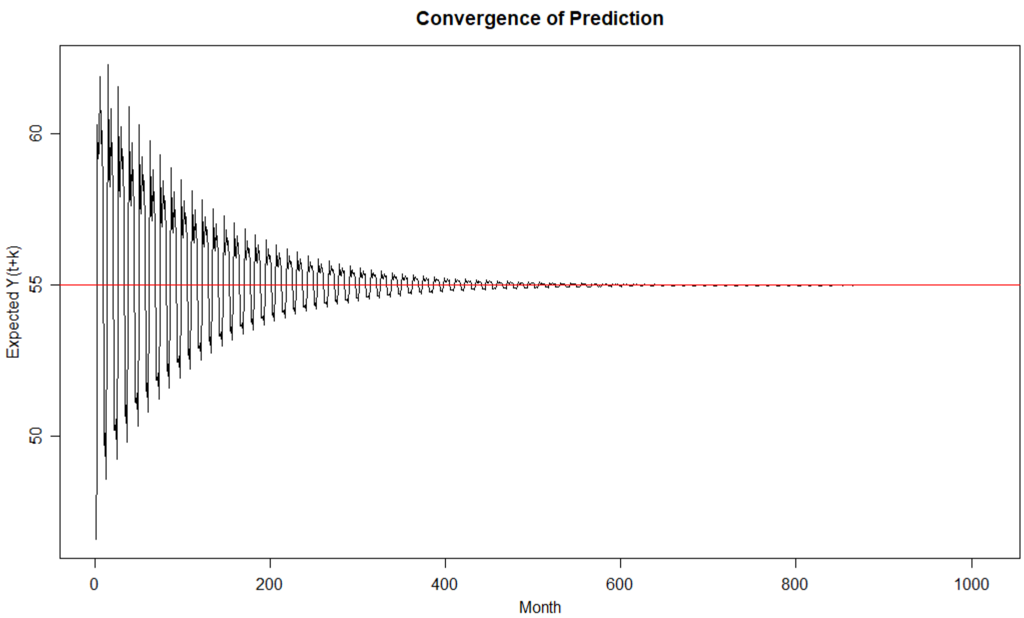

Figure 6: The red line is $\mu = 55$.

Which also illustrates one of the points of why it's important the process being stationary.

Opgave 2.3

5 different multiplicative models on the form $(p, d, q) \times (P, D, Q)_s$ are given. From definition 5.22, page 132 in [1] a model is seasonal if:

$$\phi(B)\Phi(B^s)\nabla^d\nabla_s^D Y_t = \theta(B)\Theta(B^s)\varepsilon_t$$

Where ε_t is white noise. Before one can simulate the following models, by using the function `arima.sim` in *R*, which has no seasonal module. Therefore one should rewrite the model so it corresponds to the ARIMA(p,d,q) process in definition 5.21, page 130 [1]

$$\phi(B)\nabla^d Y_t = \theta(B)\varepsilon_t$$

Furthermore one should make sure each model corresponds to the model parameterization, given by the `arima` function in *R*. In practice one should simply isolate Y_t in the models. This means the parameters of the AR process will change sign.

A lag of 40 will be used, this contains 3 seasons and should be enough to illustrate tendencies within the ACF and PACF.

1. $(1, 0, 0) \times (0, 0, 0)_{12}$ model with the parameter $\phi_1 = -0.85$

The first model is a non-seasonal AR(1) model, rewriting so it works with the `arima.sim` function in *R*:

$$(1 - 0.85B)Y_t = \varepsilon_t \Leftrightarrow Y_t = 0.85Y_{t-1} + \varepsilon_t \tag{7}$$

Using the transfer function (5.75) in [1] to check stationarity:

$$\begin{aligned} \phi(z^{-1}) &= (1 - 0.85z^{-1}), \\ 0 &= (1 - 0.85z^{-1}), \quad z \in 0.85 \end{aligned}$$

Therefore the AR(1) process is stationary, by theorem (5.9) in [1] the AR process is always invertible. In figure 7 one can see the simulated process and its ACF and PACF.

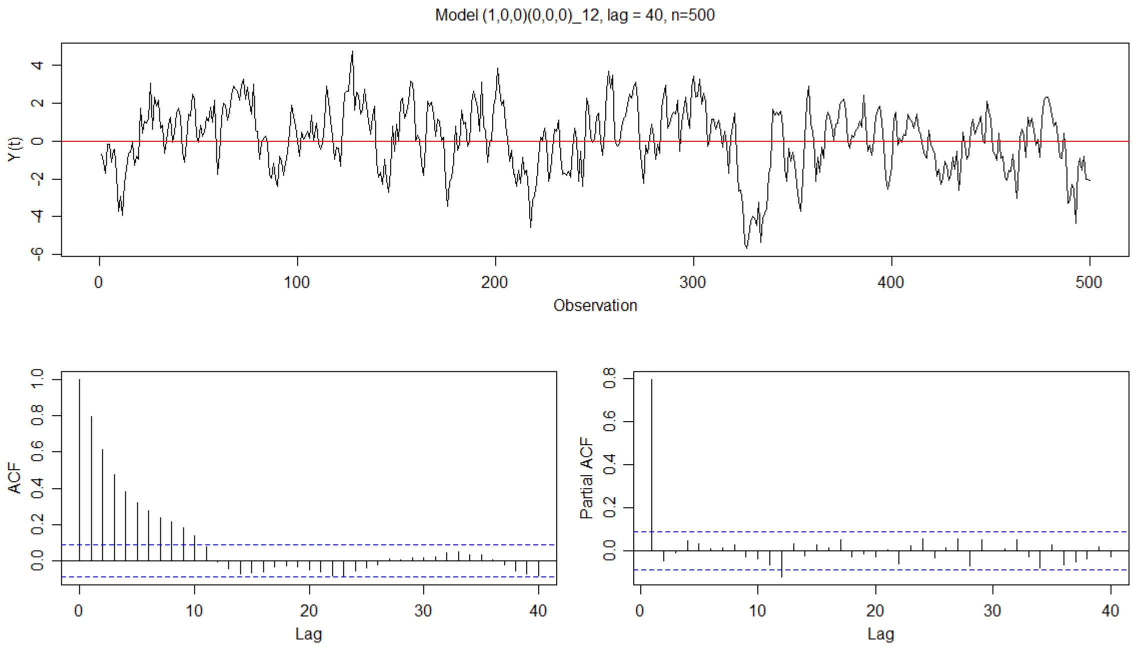

Figure 7: The non-seasonal AR(1) process and its ACF and PACF

Notice the ACF follows table (6.1) in [1], it seems to be damped exponentially. Furthermore the PACF also match, with $\phi_{kk} = 0$ for $k > p$, where $p = 1$ in this case. It can be seen by the plot of the process simulations that it's stationary, it seems to have no trend and fluctuate in a stable fashion around the red line, which is the expected value of the process.

2. $(0,0,0) \times (1,0,0)_{12}$ model with the parameter $\Phi_1 = 0.85$

The second model is a seasonal $\text{AR}(1)_{12}$ model, with a season of $s = 12$.

$$(1 + 0.85B^{12})Y_t = \varepsilon_t \Leftrightarrow Y_t = -0.85Y_{t-1} + \varepsilon_t \tag{8}$$

The results of solving the transfer function is omitted but it can be found by solving the following equation:

$$\phi(z^{-1}) = 1 + 0.85z^{-12}$$

One will notice that some of the roots are near the edge of the unit circle $|z| = 0.987$ however it still satisfies $|z| < 1$ and it is therefore stationary and invertible.

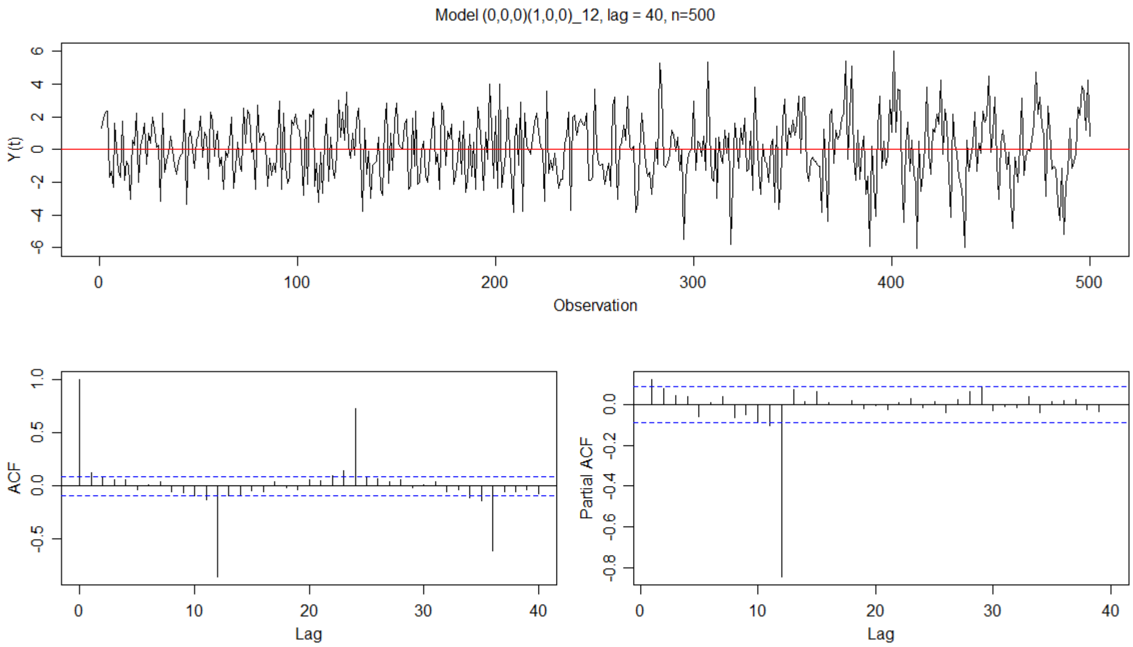

Figure 8: A seasonal $\text{AR}(1)_{12}$, process and its ACF and PACF

Looking at figure 8 the process seems to be stationary. The ACF shows that the auto-correlation is exponentially damped and follows a sine function by changing sign every season for the observed lags $k \in \{12, 24, 36\}$. This is what is expected from table (6.1) in [1].

3. $(1, 0, 0) \times (0, 0, 1)_{12}$ model with the parameter $\phi_1 = -0.8$ and $\Theta_1 = 0.9$

This multiplicative model contains a non-seasonal AR(1) coefficient and a seasonal $\text{MA}(1)_{12}$ coefficient.

$$(1 - 0.8B)Y_t = (1 + 0.9B^{12})\varepsilon_{t-1} \Leftrightarrow Y_t = 0.8Y_{t-1} + 0.9\varepsilon_{t-12} + \varepsilon_t \tag{9}$$

To check if the process is stationary and invertible one can solve the transfer function for the AR part and the MA part, if both satisfy $|z| < 1$ then the ARMA process is stationary and invertible, theorem (5.12) in [1]. Due to the high order of the MA part:

$$\theta(z^{-1}) = 1 + 0.9z^{-12}$$

Invertibility will be omitted in this case. However the eigenvalues for the AR(1) coefficient are given by:

$$\phi(z^{-1}) = 1 - 0.8z^{-1} = 0.8$$

An MA process is always stationary and the process is therefore stationary.

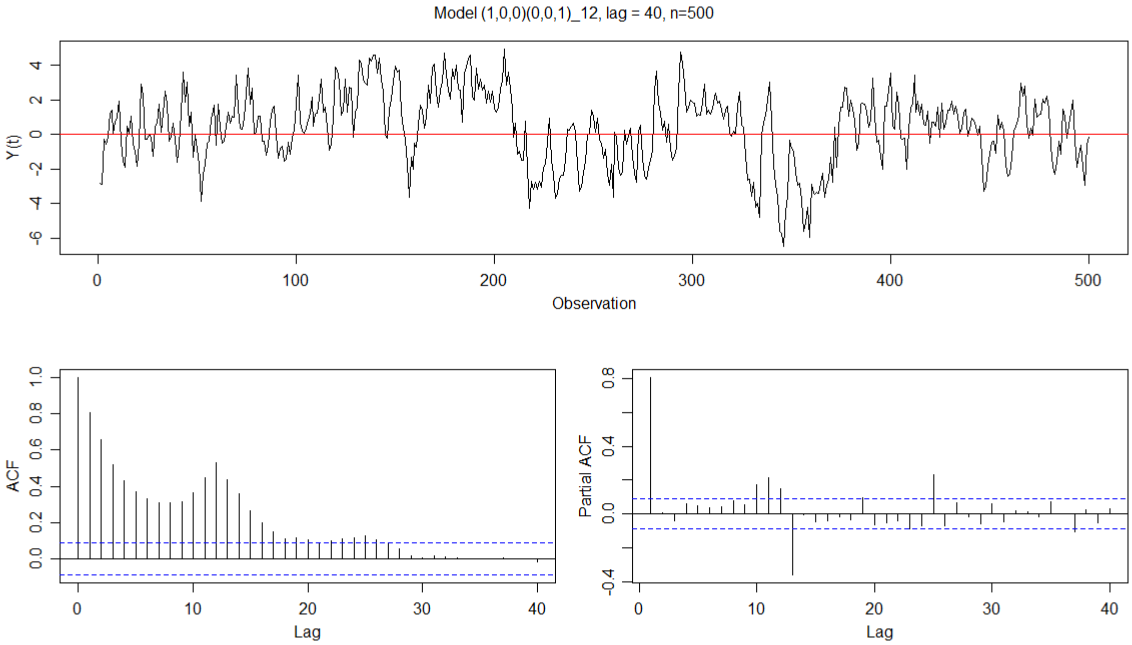

Figure 9: Multiplicative model with a non-seasonal AR(1) and a seasonal $\text{MA}(1)_{12}$ coefficient and its ACF and PACF

The process fluctuates around the expected mean. Once again matching the ACF and PACF with table 6.1, the AR process of the ARMA process will cause the ACF to consist of damped exponential functions after lag $p - q = 0$. Furthermore auto-correlation can be found at lag 12 due to the seasonality of the MA process, when an observation is close to the season the auto-correlation will start increasing, and after lag 12 the auto-correlation will again follow a damped exponential function due to the AR process. For $k > 12$ $\rho(k) = 0$ the future ACF value of the seasons will be 0 because of the MA process.
For the PACF it is dominated by peaks of auto-correlation at each season, the auto-correlation will follow a damped exponential and sine function for each season, as is expected for the process, after each lag (1,12,24,36) at the seasons the auto-correlation will be 0 because $\rho(k) = 0$ for $k > 12$ of the AR process.

4. $(1, 0, 0) \times (1, 0, 0)_{12}$ model with the parameter $\phi_1 = 0.7$ and $\Phi_1 = 0.8$

This is a multiplicative model containing a seasonal and a non-seasonal AR(1) coefficient.

$$(1 + 0.7B)(1 + 0.8B^{12})Y_t = \varepsilon_t \Leftrightarrow Y_t = -0.7Y_{t-1} - 0.8Y_{t-12} - 0.56Y_{t-13} + \varepsilon_t \quad (10)$$

The unit roots can be found by solving

$$\phi(z^{-1}) = (1 + 0.7z^{-1})(1 + 0.8z^{-12})$$

This is omitted, instead the process can be seen that it is stationary in figure 10. It can also be seen that there are periodic patterns in the simulation due to the seasonality.

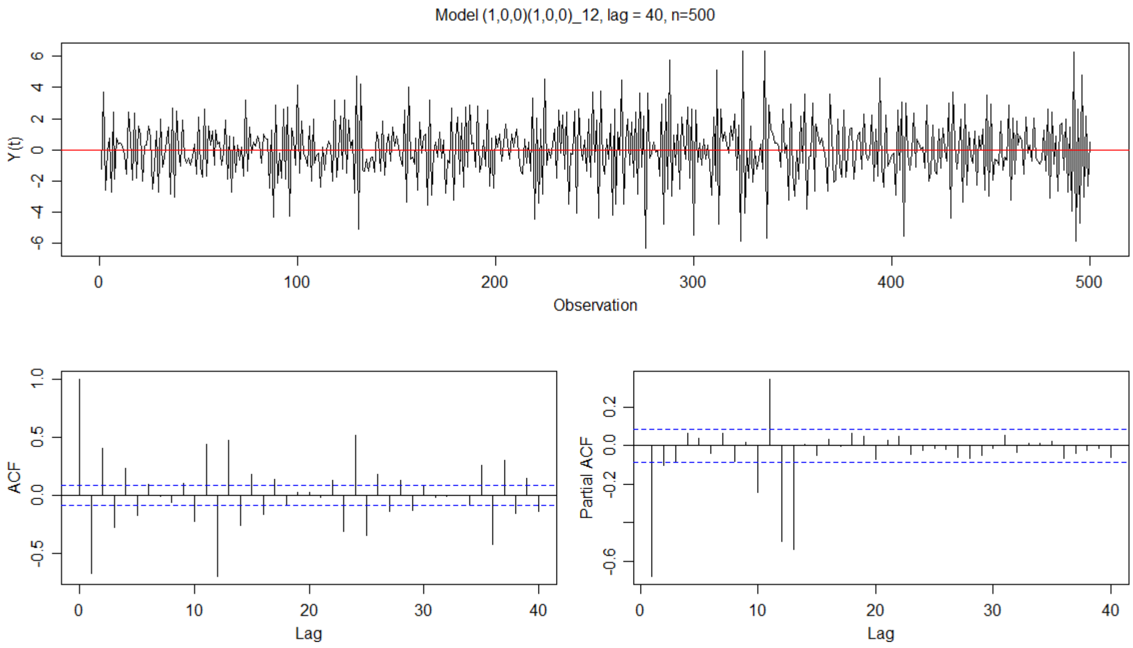

Figure 10: Multiplicative model with a non-seasonal AR(1) and a seasonal $\text{AR}(1)_{12}$ coefficient and its ACF and PACF

Matching the ACF and PACF with table 6.1 in [1], one can see that the ACF follows the expected behavior. It consists of exponential damped and sin functions, where the seasonality of the AR(1) can be seen at every 12th and 13th lag, which will then slowly decrease.

5. $(2,0,0) \times (1,0,0)_{12}$ model with the parameter $\phi_1 = 0.6$, $\phi_2 = -0.3$ and $\Phi_1 = 0.8$

This is a multiplicative model with a non-seasonal AR(2) coefficient and a seasonal AR(1) coefficient.

$$\begin{aligned}(1{+}0.6B - 0.3B^2)(1 + 0.8B^{12})Y_t = \varepsilon_t \Leftrightarrow \\ Y_t = -0.6Y_{t-1} + 0.3Y_{t-2} - 0.8Y_{t-12} - 0.48Y_{t-13} + 0.24Y_{t-14} + \varepsilon_t\end{aligned} \tag{11}$$

The unit roots can be found by solving

$$\phi(z^{-1}) = (1 + 0.6z^{-1} - 0.3z^{-2})(1 + 0.8z^{-12})$$

This is omitted, instead the process can be seen that it is stationary in figure 11. It can once again be seen that there are periodic patterns in the simulation due to the seasonality, this time more pronounced due to the slower decreasing auto-correlation of the AR(2) coefficient.

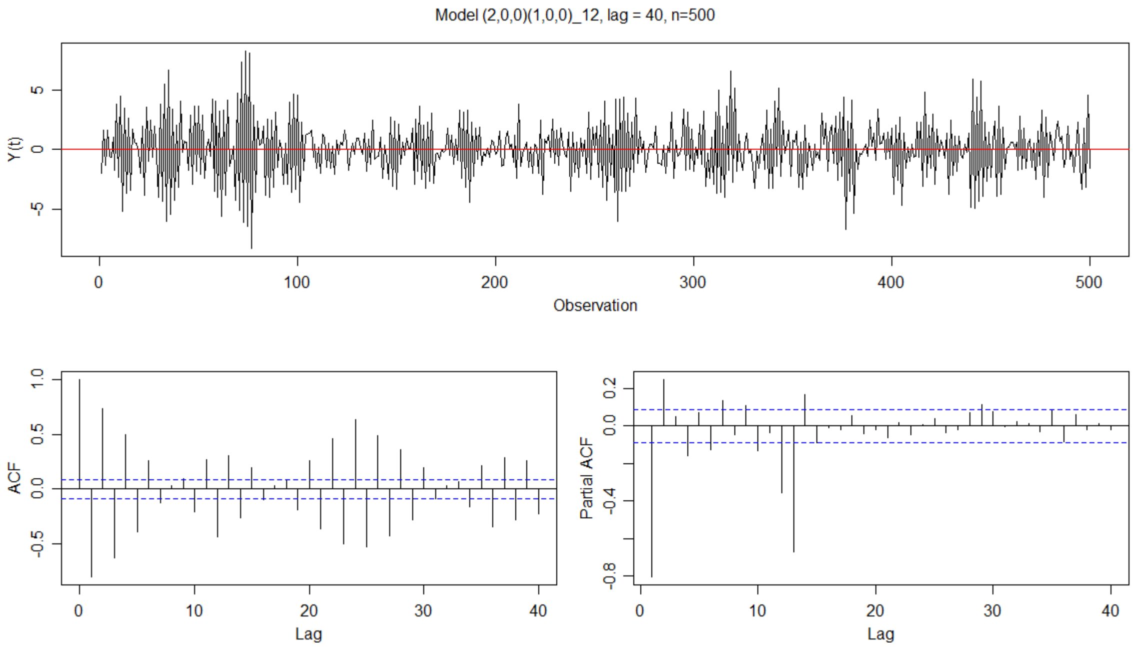

Figure 11: Multiplicative model with a non-seasonal AR(2) and a seasonal $\text{AR}(1)_{12}$ coefficient and its ACF and PACF

The ACF consist of damped exponential and sine functions, with peaks varying in size every 12th, 13th and 14th lag. This matches what is expected according to table 6.1 in [1] and eq. (11). For the PACF the auto-correlation would be expected different to 0 at lags greater than 2 due to AR(2) process. For every 12th, 13th and 14th lag an auto-correlation different to 0 would be expected, however it will follow a damped exponential and sine function.

6. $(0, 0, 1) \times (0, 0, 1)_{12}$ model with the parameter $\theta_1 = -0.4$ and $\Theta_1 = 0.8$
This is a multiplicative model with a non-seasonal MA(1) coefficient and a seasonal $\text{MA}(1)_{12}$ coefficient.

$$Y_t = (1 - 0.4B)(1 + 0.8B^{12})\varepsilon_t \Leftrightarrow Y_t = \varepsilon_t + 0.8\varepsilon_{t-12} - 0.4\varepsilon_{t-1} - 0.32\varepsilon_{t-13} \quad (12)$$

From figure 12 the process is clearly stationary. It is tough to see whether there are any periodic patterns caused by the seasonality.

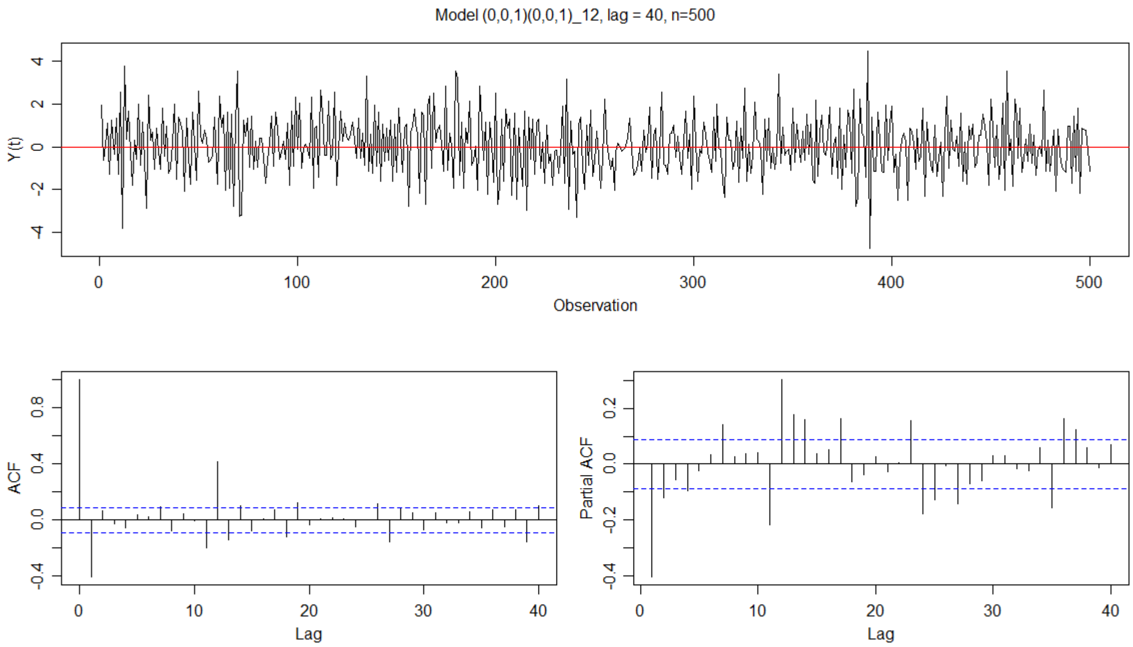

Figure 12: Multiplicative model with a non-seasonal MA(1) and a seasonal $\text{MA}(1)_{12}$ coefficient and its ACF and PACF

Looking at the ACF one would expect $\rho(k) = 0$ for $k > 1$, this is also what is seen. Furthermore due to the seasonality the auto-correlation will have a peak at lag 12 and a slightly larger than 0 auto-correlation at 12 ± 1. For the PACF the auto-correlation will be dominated by damped exponential and sine functions. This will be repeated for every 12th and 13th lag.

Summary

All the models except the first model are seasonal. Where the first model is simply a non-seasonal AR(1) model. All the models are stationary. For the observations in the processes it can be seen that if seasonality is included, there seems to be periodic patterns, where a higher order of the process will make the periodic patterns more pronounced. Depending on the season chosen, the periodic patterns will either become less frequent or more frequent.

The auto-correlation will be dominated by periodic peaks corresponding to the season chosen, further more the behaviour of the auto-correlation at the peaks will depend of the process used. If an AR(p) process is used the auto-correlation will consists of exponential damped and/or sine functions. For MA(q) processes the correlation will be $\rho(k) = 0$ for $k > q$. If the processes are used in a combination, the auto-correlation will be a mix of the individual behaviours of the processes. This can also be seen in table 6.1 in [1].

References

[1] Henrik Madsen, *Time Series Analysis*, Chapman & Hall/CRC, 2008, Chapter 5-6.

[2] Lasse Engbo Christiansen, slide 5 and 6, Time Series Analysis, 15-03-2018.

Appendix

$$\frac{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 \\ 0.75 & 0.5 \end{bmatrix}\right)}{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 \\ 0.75 & 1 \end{bmatrix}\right)} \qquad -0.1428571429$$

$$\frac{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 & 0.75 \\ 0.75 & 1 & 0.5 \\ 0.5 & 0.75 & 0.25 \end{bmatrix}\right)}{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 & 0.5 \\ 0.75 & 1 & 0.75 \\ 0.5 & 0.75 & 1 \end{bmatrix}\right)} \qquad -0.1666666667$$

$$\frac{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 & 0.5 & 0.75 \\ 0.75 & 1 & 0.75 & 0.5 \\ 0.5 & 0.75 & 1 & 0.25 \\ 0.25 & 0.5 & 0.75 & 0 \end{bmatrix}\right)}{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 & 0.5 & 0.25 \\ 0.75 & 1 & 0.75 & 0.5 \\ 0.5 & 0.75 & 1 & 0.75 \\ 0.25 & 0.5 & 0.75 & 1 \end{bmatrix}\right)} \qquad -0.2000000000$$

$$\frac{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 & 0.5 & 0.25 & 0.75 \\ 0.75 & 1 & 0.75 & 0.5 & 0.5 \\ 0.5 & 0.75 & 1 & 0.75 & 0.25 \\ 0.25 & 0.5 & 0.75 & 1 & 0 \\ 0 & 0.25 & 0.5 & 0.75 & 0 \end{bmatrix}\right)}{\mathit{Determinant}\left(\begin{bmatrix} 1 & 0.75 & 0.5 & 0.25 & 0 \\ 0.75 & 1 & 0.75 & 0.5 & 0.25 \\ 0.5 & 0.75 & 1 & 0.75 & 0.5 \\ 0.25 & 0.5 & 0.75 & 1 & 0.75 \\ 0 & 0.25 & 0.5 & 0.75 & 1 \end{bmatrix}\right)} \qquad 0.3750000000$$

Figure 13: PACF values

```
rm(list=ls())
library(xtable)
##############################ASSIGNMENT 2###############
########################################################
options("scipen" = -1)
par(mgp=c(2,0.8,0), mar=c(3,3,2,1),las=0, pty="m")

setwd("setWD here")



#######QUESTION 2.1########
set.seed(100)
sims1<-matrix(0, nrow=200, ncol=10)

col1<-c("orange",2:8, "antiquewhite4","darkcyan")
```

```
for(i in 1:10){
  sims1[,i]<-arima.sim(model =
                         list(ma=c(1,1,1), order=c(0,0,3)),
                       n = 200, sd=0.1)
}

par(mgp=c(2,0.8,0), mar=c(3,3,2,1),las=0, pty="m",xpd=F)
par(mfrow=c(1,1))
plot(sims1[,1], col=col1[1], xlab="Observation",
     ylab="X(t)", main="10␣Simulations",
     type='l', ylim=c(-1,1))
for(i in 2:10){
  lines(sims1[,i], col=col1[i])
}
abline(0,0)
legend("topright", inset=c(-0,0),
       c(paste("Simulation", 1:10)),
       ncol=5, col=col1, lty=c(rep(1,10)), cex=0.8)

qqnorm(sims1[,1:10])
qqline(sims1[,1:10])

par(mgp=c(2,0.8,0), mar=c(3,3,3,1),las=0, pty="m",xpd=F)
plot(acf(sims1[,1],plot=FALSE, lag.max=20),col=col1[1],
     type="l", max.mfrow=1, ylim=c(-0.4,1),
     main="ACF␣of␣each␣simulation")
points(0:20,acf(sims1[,1],plot=FALSE,
                lag.max=20)$acf, col=col1[1])
for(i in 2:10){
  lines(0:20,acf(sims1[,i],plot=FALSE,
                 lag.max=20)$acf, col=col1[i])
  points(0:20,acf(sims1[,i],plot=FALSE,
                  lag.max=20)$acf, col=col1[i])
}
points(0, 1, bg=1, pch=21, cex=1.2)
points(1, 0.75, bg=1, pch=21, cex=1.2)
points(2, 0.5, bg=1, pch=21, cex=1.2)
points(3, 0.25, bg=1, pch=21, cex=1.2)
points(4, 0, bg=1, pch=21, cex=1.2)
legend("topright", "Analytical␣ACF␣Value",
       col=1, pt.bg=1, pch=21, pt.cex=1.2)

par(mgp=c(2,0.8,0), mar=c(3,3,3,1),las=0, pty="m",xpd=F)
plot(pacf(sims1[,1],plot=FALSE, lag.max=20),
    type="l", max.mfrow=1,
     ylim=c(-0.4,1), col=col1[1],
     main="PACF␣of␣each␣simulation")
for(i in 2:10){
  lines(1:20,pacf(sims1[,i],plot=FALSE,
```

```
                  lag.max=20)$acf, col=col1[i])
  points(1:20,pacf(sims1[,i],plot=FALSE,
                   lag.max=20)$acf, col=col1[i])
}
points(1, 0.75, bg=1, pch=21, cex=1.2)
points(2, -.1428571429, bg=1, pch=21, cex=1.2)
points(3, -.1666666667, bg=1, pch=21, cex=1.2)
points(4, -.2000000000, bg=1, pch=21, cex=1.2)
points(5, .3750000000, bg=1, pch=21, cex=1.2)
legend("topright", "Analytical PACF Value",
       col=1, pt.bg=1, pch=21, pt.cex=1.2)


for(i in 1:10){
  print(var(sims1[,i]))
}

#find roots of process
polyroot(c(1,1,1,1))

#count of data within 1 standard deviation
length(which(sims1>-0.2 & sims1<0.2))


options(digits=5)
lol<-numeric(10)
for(i in 1:10){
  lol[i]<-var(sims1[,i])
}
data.frame(Realisation=1:10,Variance=lol)
xtable(data.frame(Realisation=1:10,Variance=lol),
       digits=c(0,0,5))

mean(lol)
#######QUESTION 2.2########
dat<- data.frame(Year=c(2016, 2016, 2016, 2016,
                        2017, 2017, 2017, 2017, 2017,
                        2017, 2017, 2017, 2017, 2017, 2017),
                 Month=c(9, 10, 11, 12, 1, 2, 3, 4, 5, 6,
                         7, 8, 9, 10, 11),
                 Temp=c(46.6, 49.5, 60.3, 59.2, 59.5, 61.9,
                        59.7, 60.1, 57.8, 49.7, 49.7, 50.1,
                        48.6, 54.5, 62.3))
dat$Temp <- dat$Temp - 55


plot(dat$Temp)

#s. 108-109
mu<-55
```

```
T<-length(dat$Temp)+1
pred1<-with(dat, 0.9*(Temp[T-12]) + 0.5*(Temp[T-1])-
              0.45*(Temp[T-13])-0.3*(Temp[T-2])+
              0.27*(Temp[T-14])+mu)

T<-T+1
pred2<-with(dat, 0.9*(Temp[T-12]) + 0.5*(pred1-mu)-0.45*
              (Temp[T-13])-0.3*(Temp[T-2])+0.27*(Temp[T-14])+mu)

#pred int:
sigma.err <- 0.5^2

sigma1 <- 1*(0.5^2)
sigma2 <- (1+0.5^2)*0.5^2

pred.int1<-0
pred.int1[1]<- pred1 + qnorm(0.975)*sqrt(sigma1)
pred.int1[2]<- pred1 - qnorm(0.975)*sqrt(sigma1)

pred.int2<-0
pred.int2[1]<- pred2 + qnorm(0.975)*sqrt(sigma2)
pred.int2[2]<- pred2 - qnorm(0.975)*sqrt(sigma2)

plot(c(dat$Temp+55,pred1, pred2), xlim=c(0,17),
     ylim=c(min(dat$Temp)+54, max(dat$Temp)+56),
     ylab="Y(t)", main="Forward Temperature over Time",
     xlab="Month", type='l')
points(1:15,c(dat$Temp+55))
points(16:17,c(pred1, pred2), col=2)
lines(16:17,c(pred1, pred2), col=2, lwd=2)
lines(16:17, c(pred.int1[1], pred.int2[1]), col=2,
    lty=2, lwd=2)
lines(16:17, c(pred.int1[2], pred.int2[2]), col=2,
    lty=2, lwd=2)
legend("topleft", c("Observed", "Prediction",
                    "95% Prediction Interval"),
       lty=c(1,1,2), col=c(1,2,2), text.col = c(1,2,2))
#
T<-length(dat$Temp)+1
mu<-55
pred<-c(dat$Temp+55, numeric(1000))
pred[T]<-with(dat, 0.9*(pred[T-12]) + 0.5*(pred[T-1])
              -0.45*(pred[T-13])-0.3*(pred[T-2])+0.27
              *(pred[T-14])+0.08*mu)
for(i in 17:length(pred)){
  T<-T+1
  pred[T]<-with(dat, 0.9*(pred[T-12]) + 0.5*(pred[T-1])
                -0.45*(pred[T-13])-0.3*(pred[T-2])+0.27
                *(pred[T-14])+
                0.08*mu)
```

```
}
pred <- pred
plot(pred, ylab="Expected␣Y(t+k)",
     main="Convergence␣of␣Prediction", xlab="Month", type='l')
abline(55,0, col=2)
#


#OPGAVE 2.3
#1
sim1<-arima.sim(model = list(ar=0.85,
                             order=c(1,0,0)), n = 500, sd=1)

par( oma = c( 0, 0, 2, 0 ) )
old.par <- par( no.readonly = TRUE )
##  Split the screen into two rows and one column,
#defining screens 1 and 2.
split.screen( figs = c( 2, 1 ) )
##  Split screen 2 into one row and two columns,
#defining screens 6 and 7.
split.screen( figs = c( 1, 1), screen = 1 )
##  Split screen 2 into one row and 2 columns,
#defining screens 3, 4.
split.screen( figs = c( 1, 2 ), screen = 2 )
screen(3)
plot(sim1, ylab="Y(t)", xlab="Observation")
abline(0,0, col=2)
screen(4)
acf(sim1, lag=40, main="")
screen(5)
pacf(sim1, lag=40, main="")
mtext("Model␣(1,0,0)(0,0,0)_12,␣lag␣=␣40,␣n=500", outer = TRUE,
      at=c(1/2), line=-2)
close.screen( all = TRUE )
par( old.par )

#2
sim2<-arima.sim(model = list(ar=c(rep(0,11), -0.85),
                             order=c(12,0,0)), n = 500)

par( oma = c( 0, 0, 2, 0 ) )
old.par <- par( no.readonly = TRUE )
split.screen( figs = c( 2, 1 ) )
split.screen( figs = c( 1, 1), screen = 1 )
split.screen( figs = c( 1, 2 ), screen = 2 )
screen(3)
plot(sim2, ylab="Y(t)", xlab="Observation")
abline(0,0, col=2)
screen(4)
acf(sim2, lag=40, main="")
```

```
screen(5)
pacf(sim2, lag=40, main="")
mtext("Model␣(0,0,0)(1,0,0)_12,␣lag␣=␣40,␣n=500", outer = TRUE,
      at=c(1/2), line=-2)
close.screen( all = TRUE )
par( old.par )

#3
sim3<-arima.sim(model = list(ar=c(0.8), ma=c(rep(0,11),0.9),
                             order=c(1,0,12)), n = 500)

par( oma = c( 0, 0, 2, 0 ) )
old.par <- par( no.readonly = TRUE )
split.screen( figs = c( 2, 1 ) )
split.screen( figs = c( 1, 1), screen = 1 )
split.screen( figs = c( 1, 2 ), screen = 2 )
screen(3)
plot(sim3, ylab="Y(t)", xlab="Observation")
abline(0,0, col=2)
screen(4)
acf(sim3, lag=40, main="")
screen(5)
pacf(sim3, lag=40, main="")
mtext("Model␣(1,0,0)(0,0,1)_12,␣lag␣=␣40,
␣␣␣␣n=500", outer = TRUE, at=c(1/2), line=-2)
close.screen( all = TRUE )
par( old.par )

#4
sim4<-arima.sim(model = list(ar=c(-0.7, rep(0, 10), -0.8, -0.56),
                             order=c(13,0,0)), n = 500)

par( oma = c( 0, 0, 2, 0 ) )
old.par <- par( no.readonly = TRUE )
split.screen( figs = c( 2, 1 ) )
split.screen( figs = c( 1, 1), screen = 1 )
split.screen( figs = c( 1, 2 ), screen = 2 )
screen(3)
plot(sim4, ylab="Y(t)", xlab="Observation")
abline(0,0, col=2)
screen(4)
acf(sim4, lag=40, main="")
screen(5)
pacf(sim4, lag=40, main="")
mtext("Model␣(1,0,0)(1,0,0)_12,␣lag␣=␣40,
␣␣␣␣n=500", outer = TRUE, at=c(1/2), line=-2)
close.screen( all = TRUE )
par( old.par )

#5
```

```
sim5<-arima.sim(model = list(ar=c(-0.6, 0.3, rep(0, 9),
                                  -0.8, -0.48, 0.24),
                             order=c(14,0,0)), n = 500)

par( oma = c( 0, 0, 2, 0 ) )
old.par <- par( no.readonly = TRUE )
split.screen( figs = c( 2, 1 ) )
split.screen( figs = c( 1, 1), screen = 1 )
split.screen( figs = c( 1, 2 ), screen = 2 )
screen(3)
plot(sim5, ylab="Y(t)", xlab="Observation")
abline(0,0, col=2)
screen(4)
acf(sim5, lag=40, main="")
screen(5)
pacf(sim5, lag=40, main="")
mtext("Model␣(2,0,0)(1,0,0)_12,␣lag␣=␣40,
␣␣␣␣n=500", outer = TRUE, at=c(1/2), line=-2)
close.screen( all = TRUE )
par( old.par )

#6
sim6<-arima.sim(model = list(ma=c(-0.4, rep(0, 10), 0.8, -0.32),
                             order=c(0,0,13)), n = 500)

par( oma = c( 0, 0, 2, 0 ) )
old.par <- par( no.readonly = TRUE )
split.screen( figs = c( 2, 1 ) )
split.screen( figs = c( 1, 1), screen = 1 )
split.screen( figs = c( 1, 2 ), screen = 2 )
screen(3)
plot(sim6, ylab="Y(t)", xlab="Observation")
abline(0,0, col=2)
screen(4)
acf(sim6, lag=40, main="")
screen(5)
pacf(sim6, lag=40, main="")
mtext("Model␣(0,0,1)(0,0,1)_12,␣lag␣=␣40,
␣␣␣␣n=500", outer = TRUE, at=c(1/2), line=-2)
close.screen( all = TRUE )
par( old.par )
```