Exercise 9

Philipp Drebes

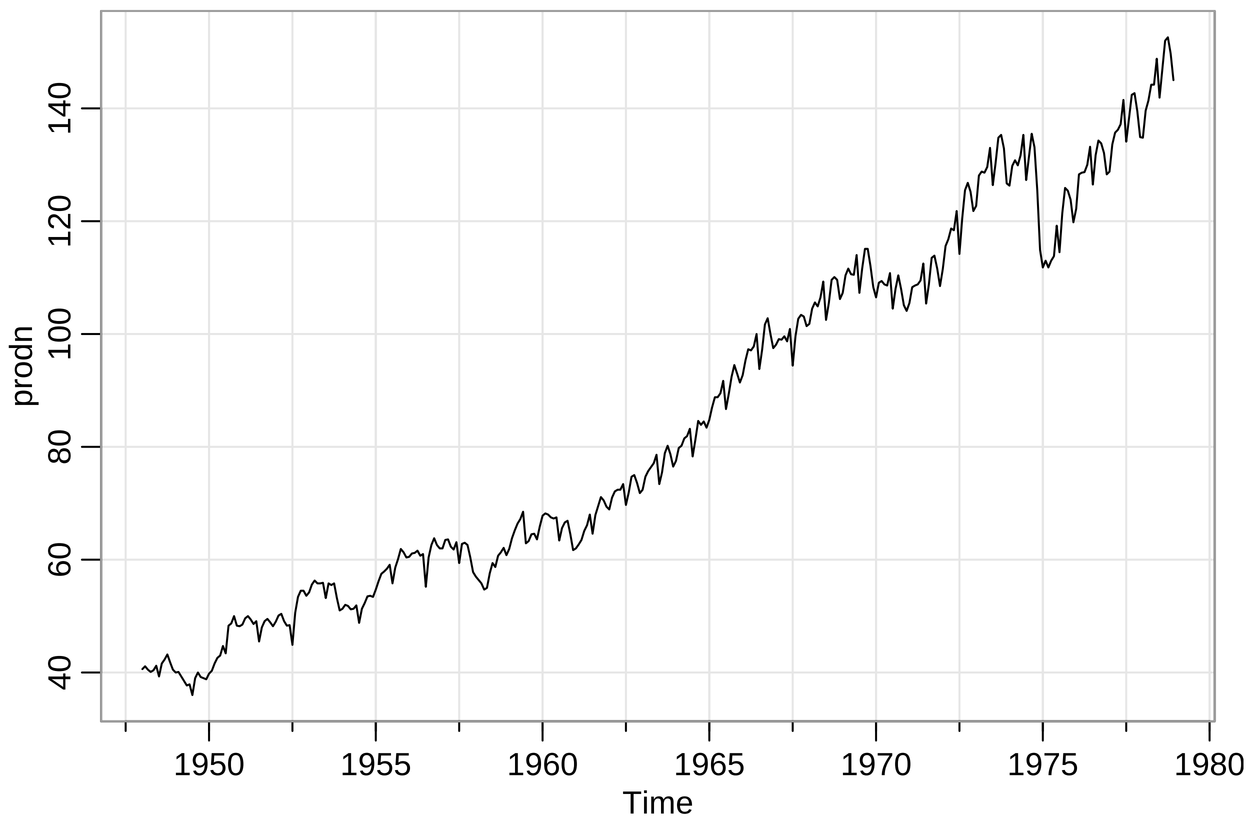
27.04.2023

Exercise 9.1

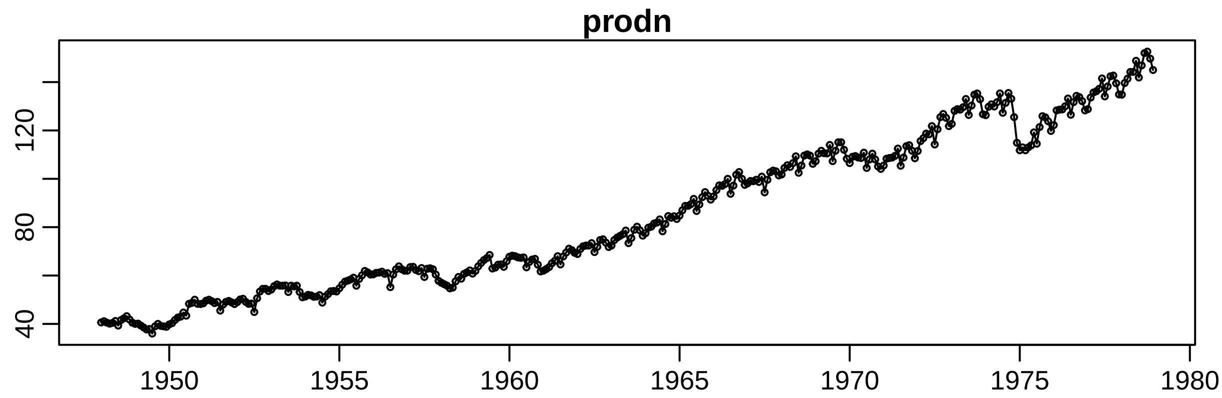
In this exercise, we look at the time series `prodn`, which is available in the package `astsa`. It contains monthly data about the Federal Reserve Board Production Index from 1948-1978, in total the time series contains data for n = 372 months.

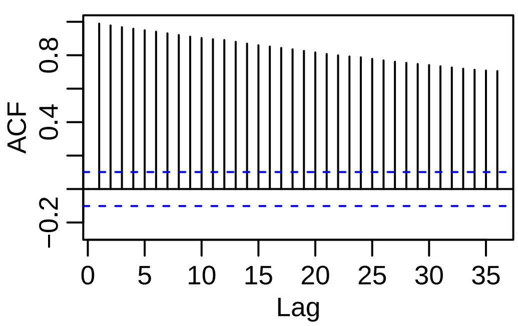
a) Plot the time series. What kind of non-stationarity is evident?

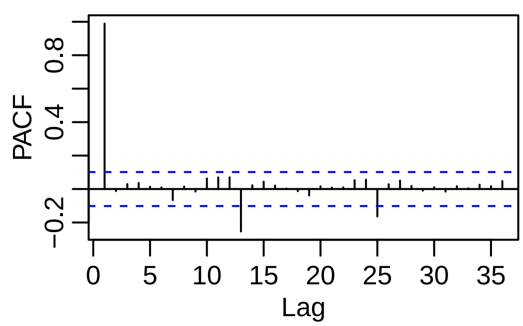
```
library(astsa, quietly = T)
library(forecast, quietly = T)

tsplot(prodn)
```




```
tsdisplay(prodn)
```

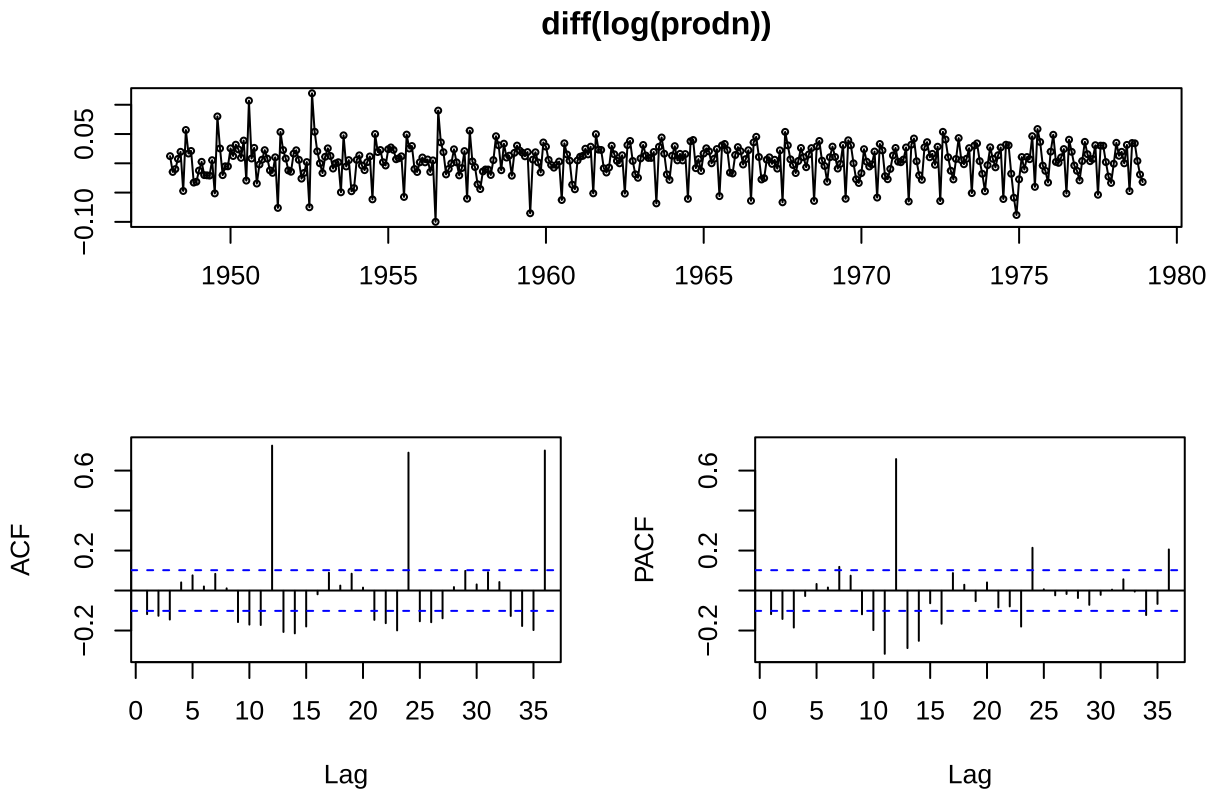




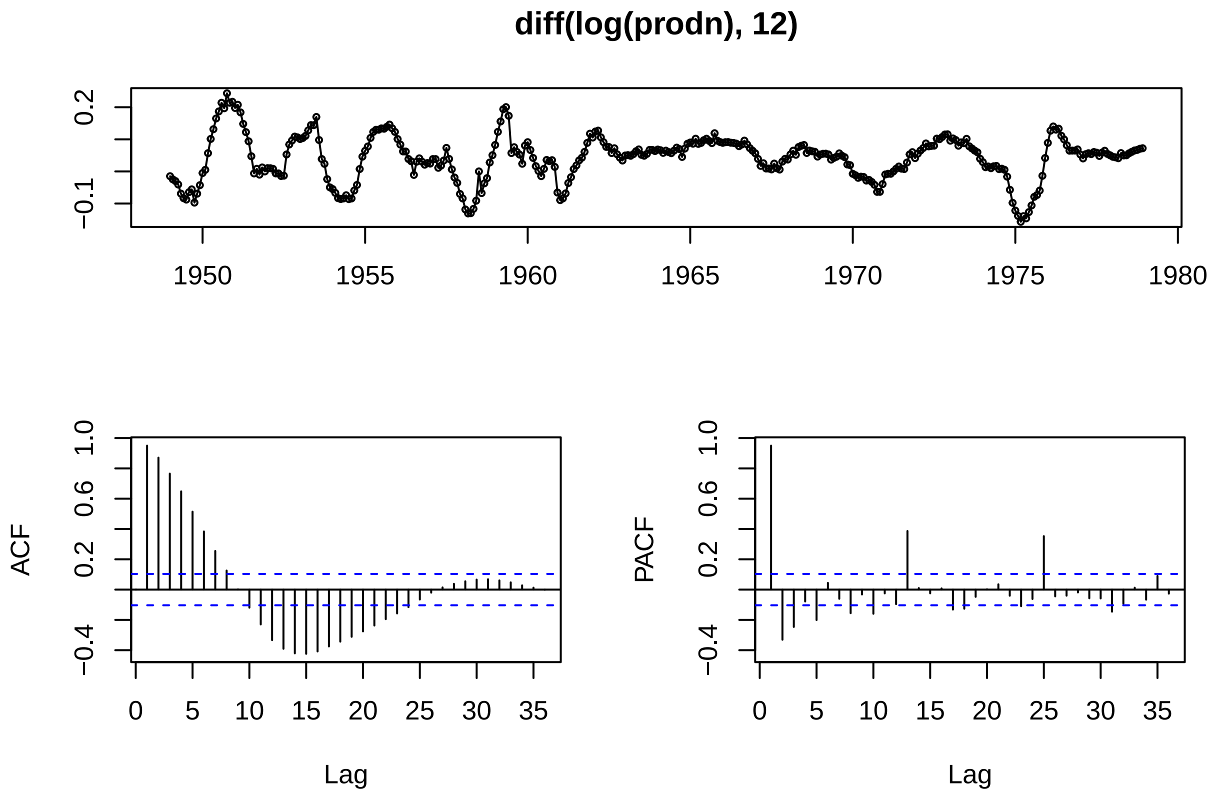


The ACF shows a slow decay, which implies that there is a trend. If there was also seasonality, the ACF would show an oscillatory behavior, which it does only very slightly.

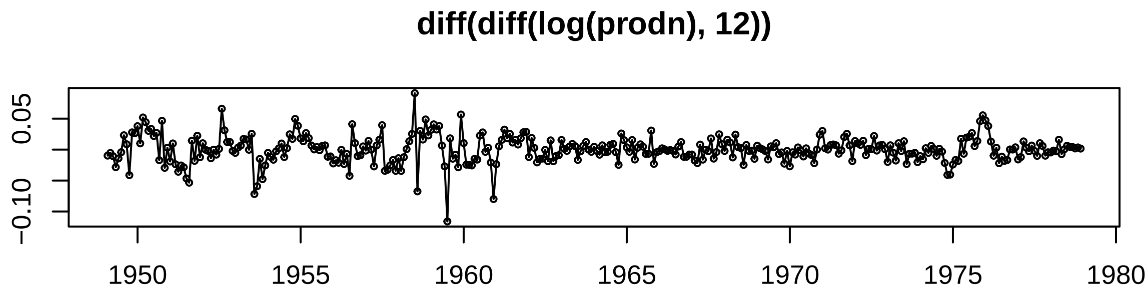
```
tsdisplay(diff(log(prodn)))
```

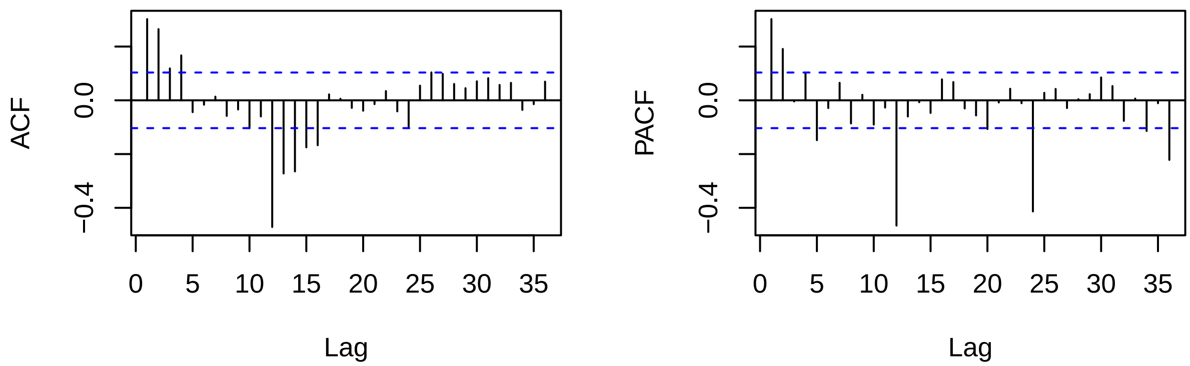



```r
tsdisplay(diff(log(prodn), 12)) # remove seasonality
```

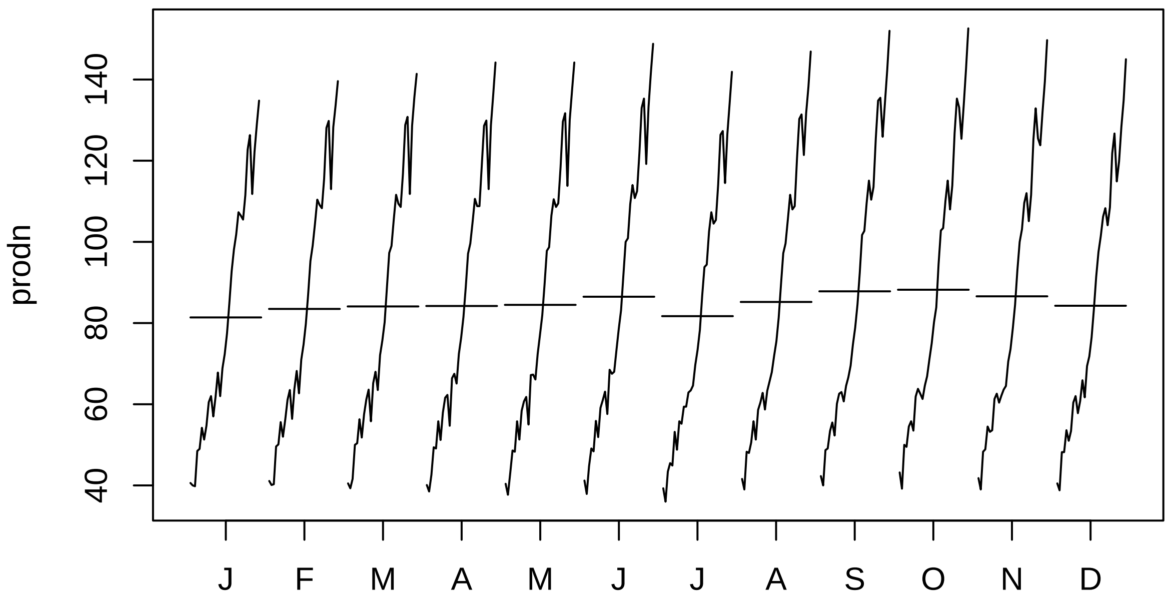



```
tsdisplay(diff(diff(log(prodn), 12))) # remove trend
```

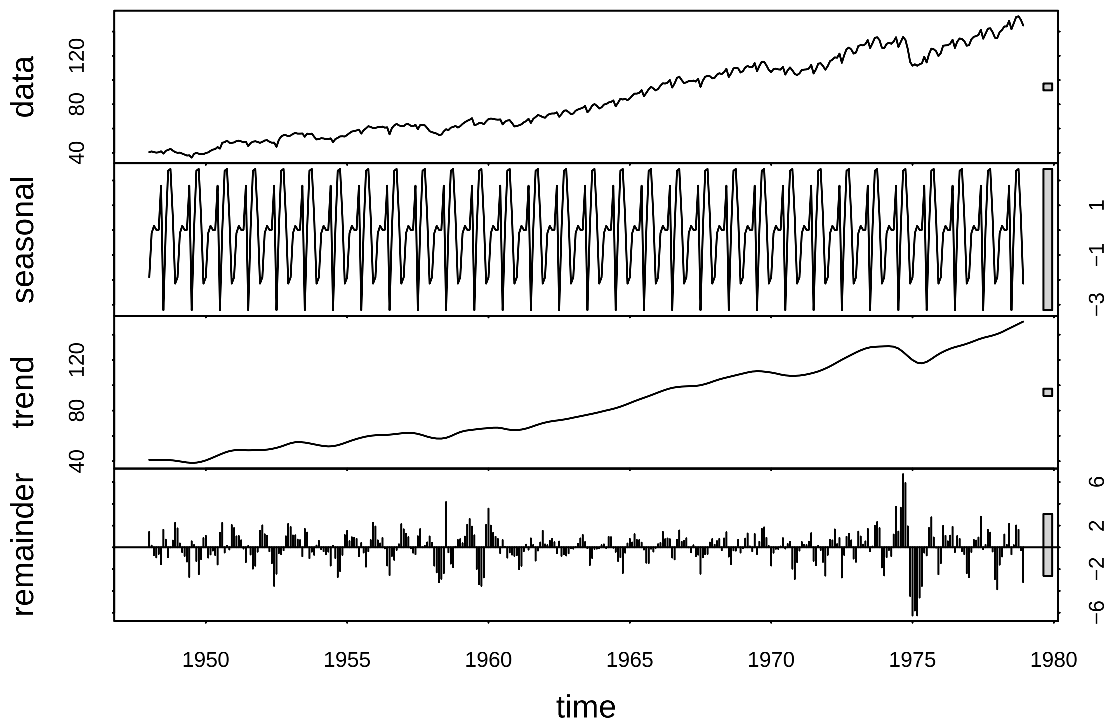





```r
monthplot(prodn)
```




```
plot(stl(prodn, s.window = 'periodic'))
```



b) How can the time series be made stationary?

By applying a SARIMA model.

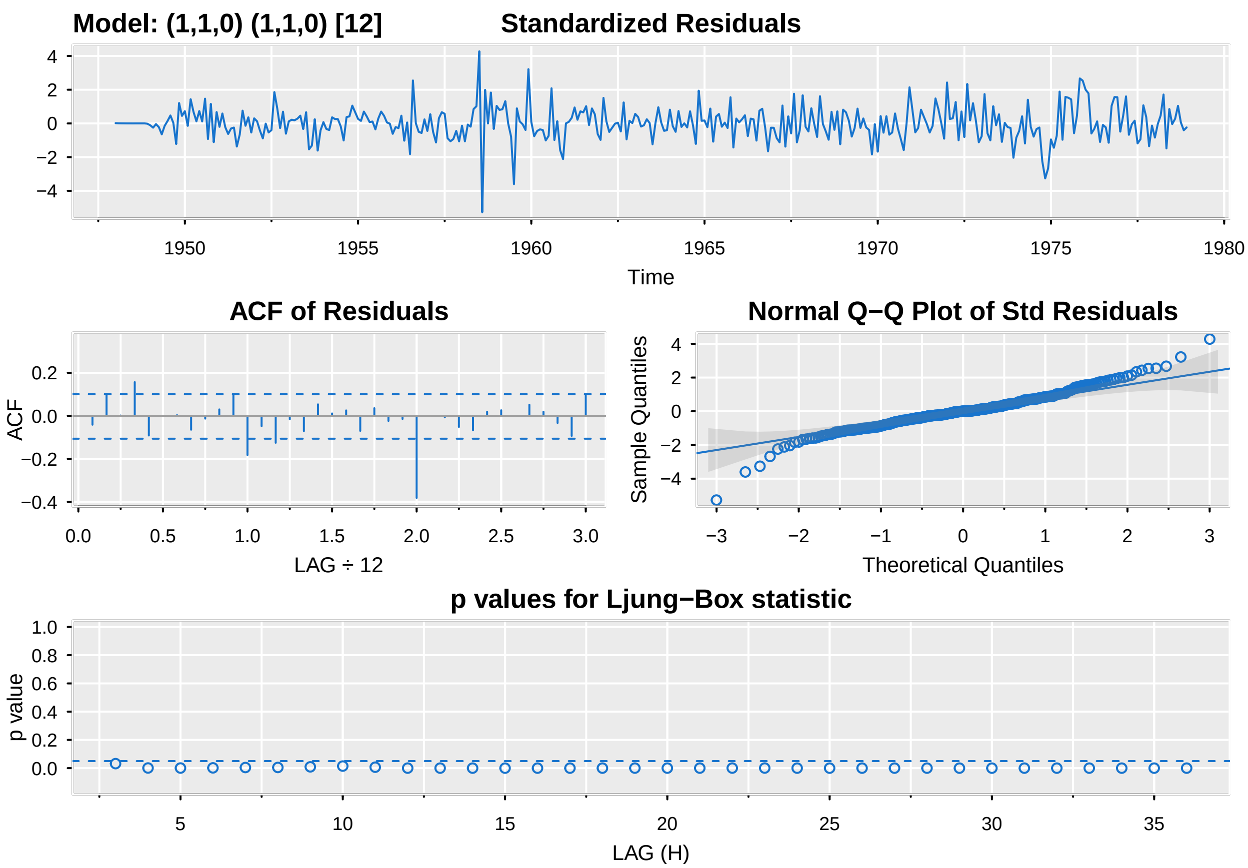
c) Based on your considerations in b), what kind of model would you fit to the original time series `prodn`? Try different fits and choose your favorite.

A SARIMA$(1,1,0)(1,1,0)^{12}$ could be a good fit, because in the PACF of the original time series, there is a sudden drop after lag 1. We also see a trend, therefore $d = 1$ is chosen.

```
fit.110 <- arima(prodn, order = c(1, 1, 0), seasonal = c(1, 1, 0))
fit.110
```

```
##
## Call:
## arima(x = prodn, order = c(1, 1, 0), seasonal = c(1, 1, 0))
##
## Coefficients:
##          ar1     sar1
##       0.3530  -0.4147
## s.e.  0.0493   0.0474
##
## sigma^2 estimated as 1.813:  log likelihood = -617.36,  aic = 1240.72
```

```
sarima(prodn, 1, 1, 0, 1, 1, 0, 12, gg = TRUE, col = 4)
```




```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed,
##     optim.control = list(trace = trc, REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1     sar1
##       0.3530  -0.4147
## s.e.  0.0493   0.0474
##
## sigma^2 estimated as 1.813:  log likelihood = -617.36,  aic = 1240.72
##
## $degrees_of_freedom
## [1] 357
##
## $ttable
##      Estimate     SE t.value p.value
## ar1    0.3530 0.0493  7.1651       0
## sar1  -0.4147 0.0474 -8.7553       0
##
## $AIC
## [1] 3.456045
##
## $AICc
```

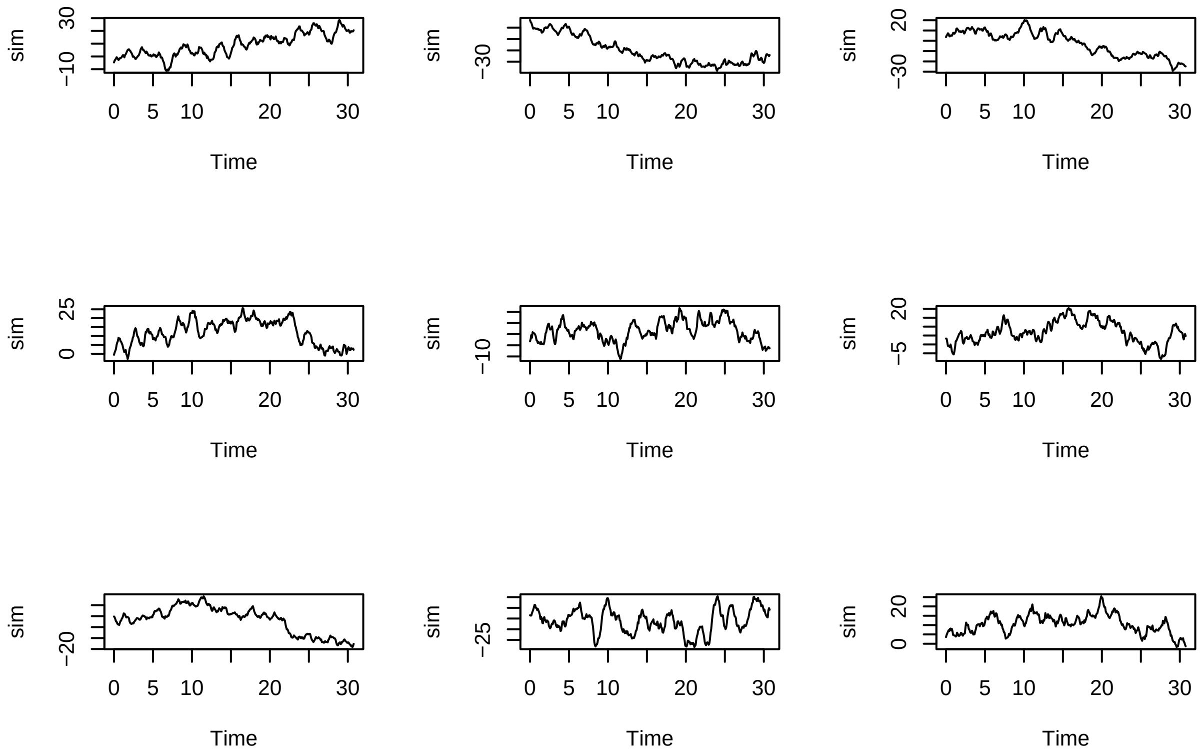
```
## [1] 3.456139
##
## $BIC
## [1] 3.488496
```

The Q-Q plot does not look as good, as there are some points at both end diverging from the line. We also still have some significant values in the ACF.

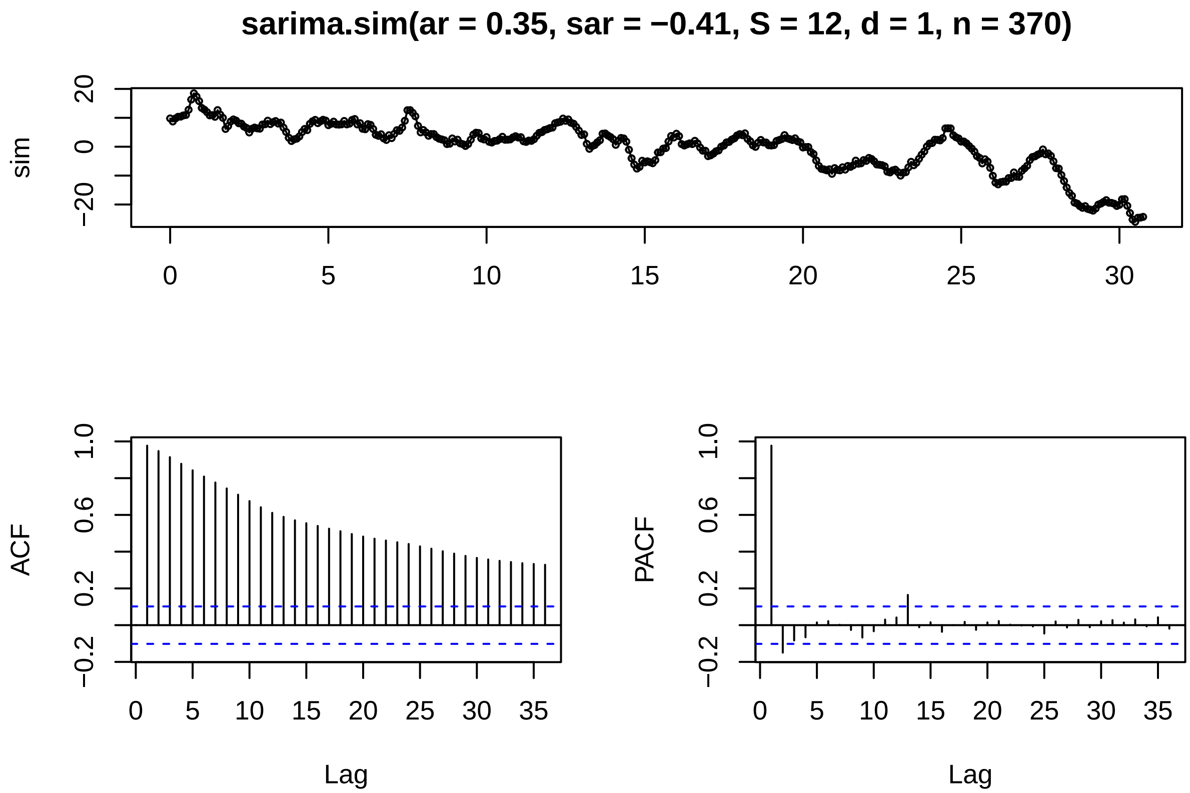
```r
set.seed(3)
par(mfrow = c(3, 3))
summary(fit.110)
```

```
##
## Call:
## arima(x = prodn, order = c(1, 1, 0), seasonal = c(1, 1, 0))
##
## Coefficients:
##          ar1     sar1
##       0.3530  -0.4147
## s.e.  0.0493   0.0474
##
## sigma^2 estimated as 1.813:  log likelihood = -617.36,  aic = 1240.72
##
## Training set error measures:
##                      ME     RMSE       MAE        MPE     MAPE      MASE
## Training set 0.02312018 1.322627 0.9462347 0.03040184 1.206129 0.4810264
##                     ACF1
## Training set -0.04127911
```

```r
replicate(9, plot(sarima.sim(ar = 0.35, sar = -0.41, S = 12, d = 1, n = 370) , ylab = 'sim'))
```



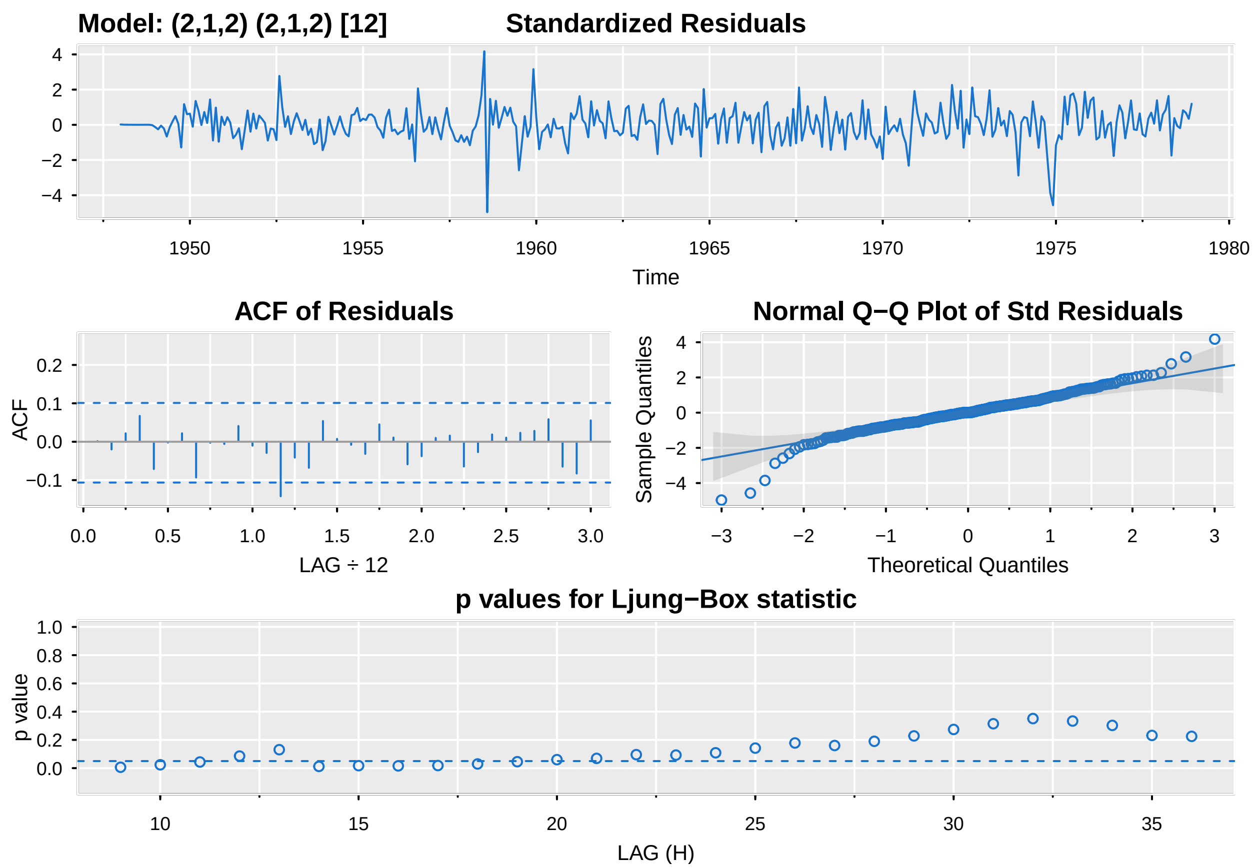

```
tsdisplay(sarima.sim(ar = 0.35, sar = -0.41, S = 12, d = 1, n = 370) , ylab = 'sim')
```



In most of the simulations, the ACF is clearly different to the original. The SARIMA$(1,1,0)(1,1,0)^{12}$ is so far not very promising.

A SARIMA$(2,1,2)(2,1,2)^{12}$ could also be possible, because when we remove trend and seasonality from the original time series, we see drops in the ACF at lag 2 and in the PACF at lag 2.

```
fit.212 <- sarima(prodn, 2, 1, 2, 2, 1, 2, 12, gg = TRUE, col = 4)
```



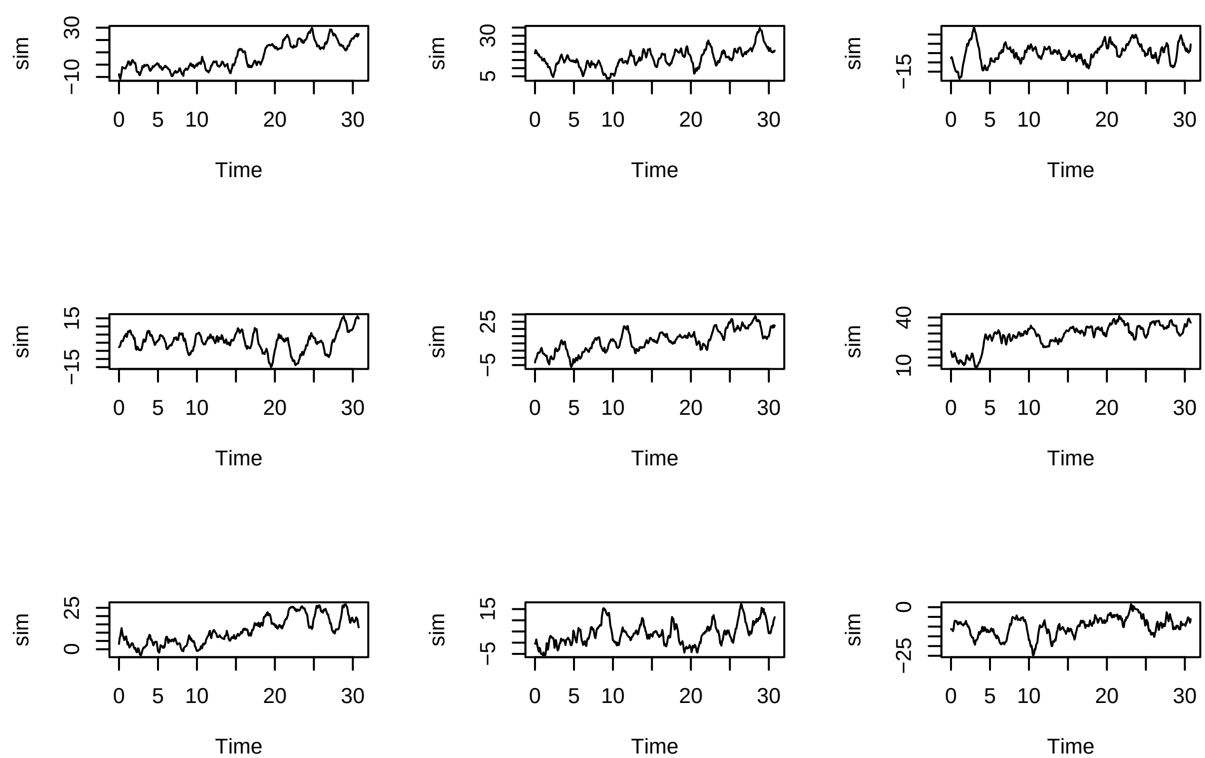
The Q-Q plot seems to a bit better than with the prior model. Also, there are no more significant values in the ACF of the residuals.

```
set.seed(2)
par(mfrow = c(3, 3))
fit.212
```

```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed,
##     optim.control = list(trace = trc, REPORT = 1, reltol = tol))
##
## Coefficients:
##           ar1     ar2     ma1      ma2    sar1     sar2     sma1    sma2
##       -0.2786  0.3396  0.5851  -0.0522  0.3822  -0.2970  -1.1301  0.4891
## s.e.   0.2202  0.1644  0.2331   0.1752  0.1604   0.0748   0.1631  0.1146
##
## sigma^2 estimated as 1.317:  log likelihood = -564.45,  aic = 1146.89
##
## $degrees_of_freedom
## [1] 351
##
## $ttable
##      Estimate     SE t.value p.value
## ar1   -0.2786 0.2202 -1.2649  0.2067
```

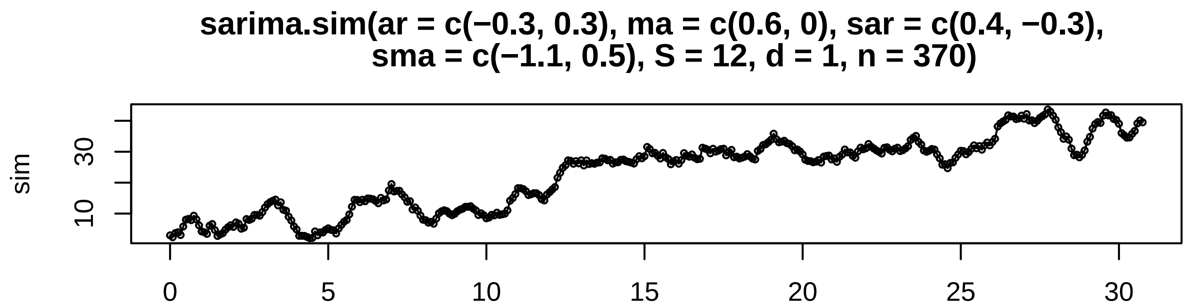
```
## ar2    0.3396 0.1644  2.0659  0.0396
## ma1    0.5851 0.2331  2.5103  0.0125
## ma2   -0.0522 0.1752 -0.2978  0.7660
## sar1   0.3822 0.1604  2.3823  0.0177
## sar2  -0.2970 0.0748 -3.9698  0.0001
## sma1  -1.1301 0.1631 -6.9299  0.0000
## sma2   0.4891 0.1146  4.2676  0.0000
##
## $AIC
## [1] 3.194682
##
## $AICc
## [1] 3.195828
##
## $BIC
## [1] 3.292036
```

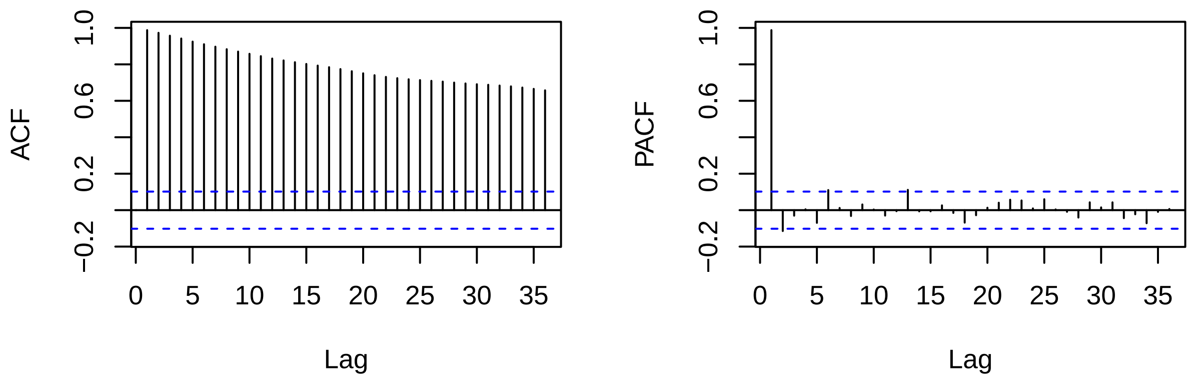
```
# the model is used for the simulations
replicate(9, plot(sarima.sim(ar = c(-0.3, 0.3),
                             ma = c(0.6, -0.1),
                             sar = c(0.4, -0.3),
                             sma = c(-1.1, 0.5),
                             S = 12, d = 1, n = 370) , ylab = 'sim'))
```




```
tsdisplay(sarima.sim(ar = c(-0.3, 0.3),
                     ma = c(0.6, 0),
                     sar = c(0.4, -0.3),
```

```
           sma = c(-1.1, 0.5),
           S = 12, d = 1, n = 370) , ylab = 'sim')
```





The SARIMA$(2,1,2)(2,1,2)^{12}$ appears to be a good fit for the time series, as the residuals only contain white noise behavior. Additionally, the ACF and PACF of the simulated data resemble those of the original.

Some further models, such as SARIMA$(3,1,2)(3,1,2)^{12}$ and SARIMA$(4,1,2)(4,1,2)^{12}$ were tested, as the ACF and PACF of the transformed original data hint at further significant values at those lags. However, creating those models produced NaN values or other errors, which made them not usable.