4. Computer exercises

Demo exercises

4.1 Examine the following time series

File (.txt)	Variable	Description	Interval	Length
INTEL	Intel_Close Intel_Volume	Intel stock price Intel stock volume	Trading day	$n = 20$
SUNSPOT	Spots	Number of sun spots	1 year	$n = 215$
SALES	Sales	Sales volume of a wholesaler	1 month	$n = 144$

Fit SARIMA processes to the corresponding time series. Use the last fifth of the time series to verify the goodness of fit.

Solution. We begin by accessing the functions of the R-package **forecast**. Note that, you might encounter some problems when installing the corresponding package to a Linux computer.

```
install.packages("forecast")
library(forecast)
# install.packages() run only once
# library() run every time you wish to
# use the functions of the package

INTEL <- read.table("INTEL.txt",header=T)
SUNSPOT <- read.table("SUNSPOT.txt",header=T,row.names=1)
SALES <- read.table("SALES.txt",header=T)

Intel_Close <- ts(INTEL$Intel_Close)
Spots <- ts(SUNSPOT,start=1749)
Sales <- ts(SALES$Sales,frequency=12,start = 1970)
```

Intel_Close

We compute the correlation functions as,

```
acf(Intel_Close,main="ACF")
pacf(Intel_Close,main="PACF")
```

We see from Figure 1 that the PACF seems to cut off after lag 2 and that the ACF decays exponentially. We try to fit an AR(2) model. The fitting of SARIMA models can be done with the function `Arima`. The first argument of the function is the modeled

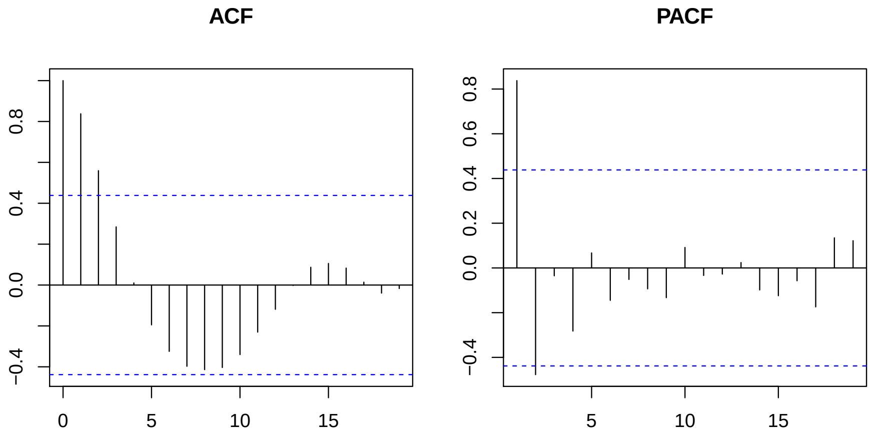

Figure 1: Auto- and parial autocorrelation functions of **Intel_Close**.

time series, `order` sets the degrees (p,h,q), where p, h and q are as in the lecture slides. In addition, it is possible to give argument `seasonal` that sets the degrees (P,H,Q) of the seasonal part.

```
Arima(Intel_Close, order=c(2,0,0))

Series: Intel_Close
ARIMA(2,0,0) with non-zero mean
Coefficients:
         ar1      ar2  intercept
      1.3342  -0.5263    64.3965
s.e.  0.1850   0.2038     1.0501

sigma^2 estimated as 0.8844:  log likelihood=-28.2
AIC=64.39   AICc=67.06   BIC=68.38
```

The estimated model involves a constant term (Intercept). Estimating the model without the constant term can be done with the argument `include.mean=FALSE`. The function `Arima` estimates the model parameters with CSS-ML (conditional sum of squares - maximum likelihood) method as a default. Next, we study the residuals of the fitted AR(2)-model.

```
model <- Arima(Intel_Close, order=c(2,0,0))
acf(model$res,main="ACF")
pacf(model$res,main="PACF")
```

By Figure 2, the residuals do not seem to be correlated. The goal is to find a model such that the residuals resemble white noise. Next, consider the Ljung-Box test for the residuals:

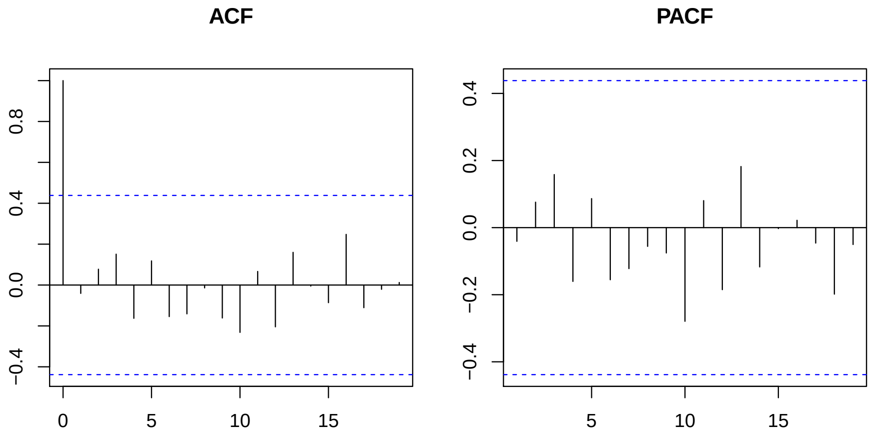

Figure 2: ACF and PACF of the residuals corresponding to the AR(2) model for **Intel_Close**.

```
Box.test(residual,lag=h,fitdf=k,type="Ljung-Box")
```

where h is the lag, and k the number of the estimated parameters of the model. Here, we have that k=2. Note that, the Ljung-Box test is only defined for lags greater than k.

The default test that the `Box.test` function performs is the so called Box-Pierce test. However, in the literature, Ljung-Box test is usually preferred.

```
ljung_box <- c(rep(NA,17))
k <- 2
for(i in 1:17){
  ljung_box[i] <- Box.test(model$res,lag=(i+k),
                           fitdf=k, type="Ljung-Box")$p.value
}
 [1] 0.3786061 0.4713735 0.5906405 0.6158862 0.6485075 0.7648594 0.7350293
 [8] 0.5646156 0.6414046 0.5078315 0.4545944 0.5403243 0.5671877 0.1945652
 [15] 0.1691062 0.2126523 0.2622378
```

The null hypothesis of the Ljung-Box test is that there is no correlation. Hence, by the Ljung-Box test, the AR(2) model is satisfactory with significance level of 5%. The fitted model and the original time series are presented in Figure 3.

```
fit <- fitted(model)
plot(fit,type="b",col="blue",ylim=c(60,68),
     ylab="Price",xlab="Time")
lines(Intel_Close,col="red",type="b")
legend(16,68, legend=c("time series", "fit"),
       col=c("red","blue"),lty=c(1,1),cex=0.8)
```

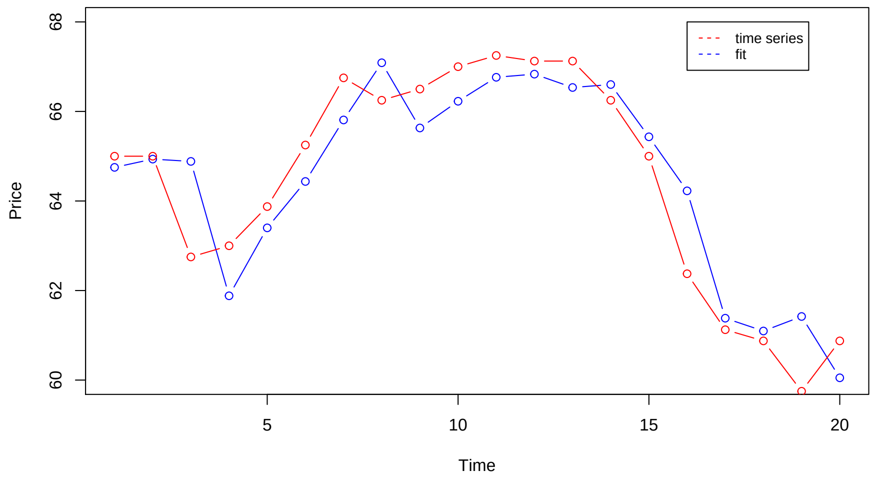

Figure 3: **Intel_Close** time series as red and the fitted AR(2) model as blue.

According to Figure 3, the fitted model and the original time series are pretty close to each other. Next, we estimate an AR(2) model by using the first 16 observations and study how well the model predicts the last four observations.

```
model_ver <- Arima(Intel_Close[1:16],order=c(2,0,0))
prediction <- forecast(model_ver,h=4,level=FALSE)$mean
#level=FALSE, omits confidence intervals
plot(Intel_Close,col="red",type="b",ylim=c(60,68),
     ylab="Price",xlab="Time")
lines(prediction,col="blue",type="b")
legend(16,68, legend=c("time series", "prediction"),
       col=c("red", "blue"), lty=c(1,1), cex=0.8)
```

Even though the previous diagnostics imply that the AR(2) model is satisfactory, we can see from Figure 4 that the first 16 observations do not provide a good prediction for the last four observations. This is explained by the shortness of the original time series. When predicting the future behavior of time series, one has to be careful, especially when dealing with short time series.

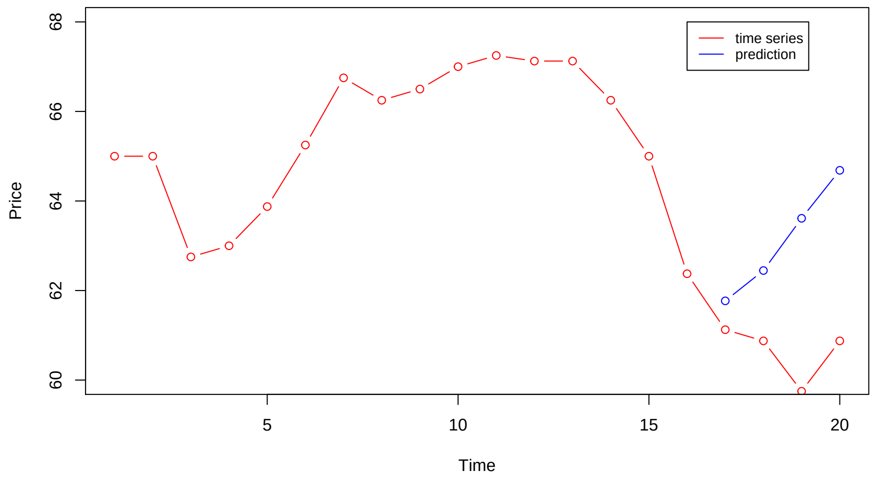

Figure 4: **Intel_Close** time series as red and the four-step prediction given by AR(2) as blue.

Spots

```
acf(Spots,lag.max=50)
pacf(Spots,lag.max=50)
```

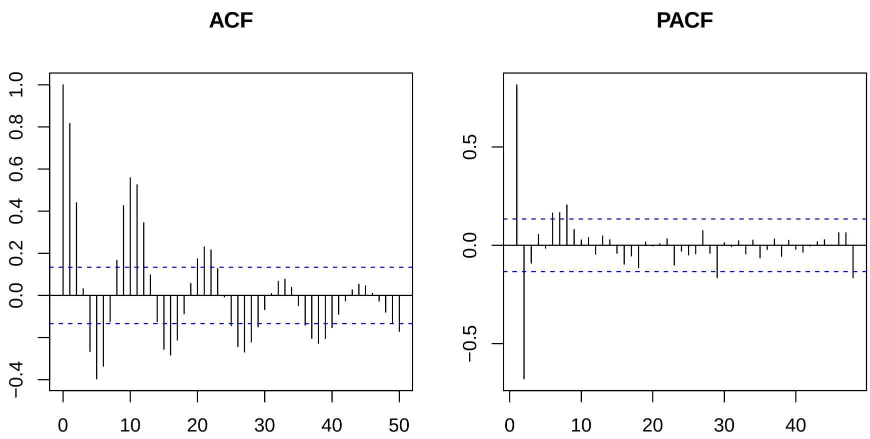

Figure 5: Auto- and partial autocorrelation functions of the **Spots** time series.

According to Figure 5, it seems that PACF cuts off after lag 2, although the sample partial autocorrelations with lags 6-8, 29 and 48 reach over the blue lines indicating statistical significance. We fit an AR(2) model and study the residuals.

```
model2 <- Arima(Spots,order=c(2,0,0))
acf(model2$res)
pacf(model2$res)
```

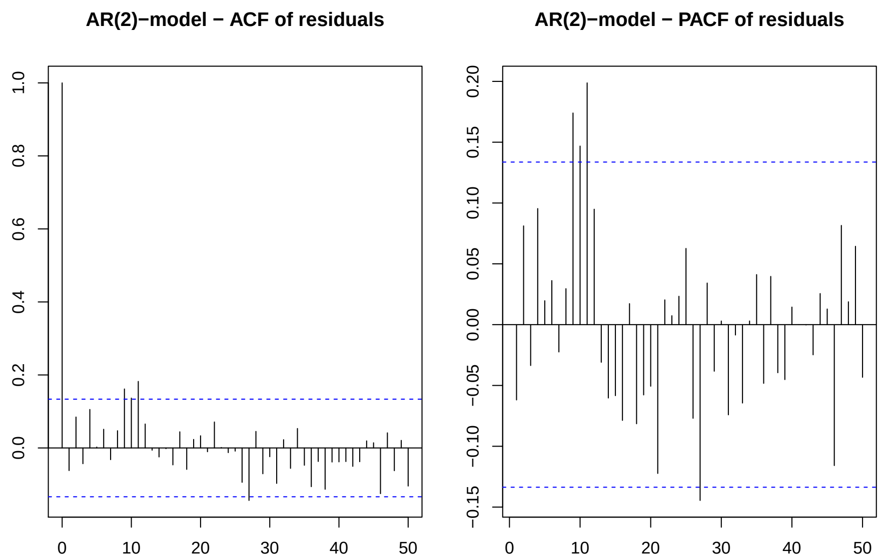

Figure 6: ACF and PACF of the residuals corresponding to the AR(2) model for **Spots**.

We see from Figure 6 that the auto- and partial autocorrelations corresponding to lags 9-11 are significant. We test the residuals with Ljung-Box.

```
k <- 2
spots_bl <- rep(NA,47)
for (i in 1:47)
{
  spots_bl[i]=Box.test(model2$res,lag=(i+k),fitdf=k,
                          type="Ljung-Box")$p.value
}
round(spots_bl,3)

 [1] 0.093 0.072 0.153 0.210 0.298 0.360 0.085 0.033 0.004 0.005 0.008
[12] 0.013 0.020 0.026 0.034 0.039 0.053 0.067 0.090 0.090 0.116 0.146
[23] 0.181 0.149 0.067 0.078 0.077 0.095 0.075 0.092 0.099 0.107 0.118
[34] 0.086 0.099 0.067 0.077 0.088 0.101 0.109 0.122 0.143 0.167 0.103
```

According to Ljung-Box test, the AR(2) model is not satisfactory when modeling the **Spots** time series. The null hypothesis of Ljung-Box is rejected with lags 10-18. Note that the first output of Ljung-Box corresponds to lag $k+1$. Next we try the function `auto.arima`, which automatically finds and fits a SARIMA model to a given time series.

```
model.auto <- auto.arima(Spots)
model.auto
acf(model.auto$res,lag.max=50)
pacf(model.auto$res,lag.max=50)

Series: Spots
ARIMA(3,0,1) with non-zero mean

Coefficients:
         ar1     ar2      ar3     ma1  intercept
      0.6574  0.3494  -0.5433  0.6580    49.1226
s.e.  0.2000  0.2622   0.1308  0.2128     3.4416

sigma^2 estimated as 266.4:  log likelihood=-906.65
AIC=1825.3   AICc=1825.7   BIC=1845.52
```

The algorithm fits the best possible SARIMA model by using AIC, AICc or BIC as a minimizing criterion. Note that, the fitting of SARIMA models is usually not easy. Furthermore, there are no perfect algorithms that always find the best possible models. Therefore, don't trust the function `auto.arima` blindly. The function often chooses unnecessarily complicated models that are only marginally better than some simpler alternatives.

By Figure 7, the auto- and sample autocorrelations corresponding to lags 9,10 and 11 are significant also for the residuals of the ARMA(3,1) fit. In addition, the null hypothesis of Ljung-Box is rejected with lags 10-19 (results below). Thus, ARMA(3,1) model is neither satisfactory.

```
k <- 4
spots_bl <- rep(NA,47)
for (i in 1:47)
{
  spots_bl[i]=Box.test(model.auto$res,lag=(i+k),fitdf=k,
                       type="Ljung-Box")$p.value
}
round(spots_bl,3)

 [1] 0.073 0.144 0.274 0.366 0.055 0.019 0.002 0.002 0.004 0.008 0.013
[12] 0.020 0.025 0.030 0.043 0.057 0.077 0.082 0.108 0.136 0.171 0.117
[23] 0.054 0.065 0.073 0.082 0.065 0.083 0.091 0.103 0.113 0.071 0.082
[34] 0.053 0.060 0.071 0.079 0.084 0.089 0.104 0.124 0.072 0.085 0.093
```

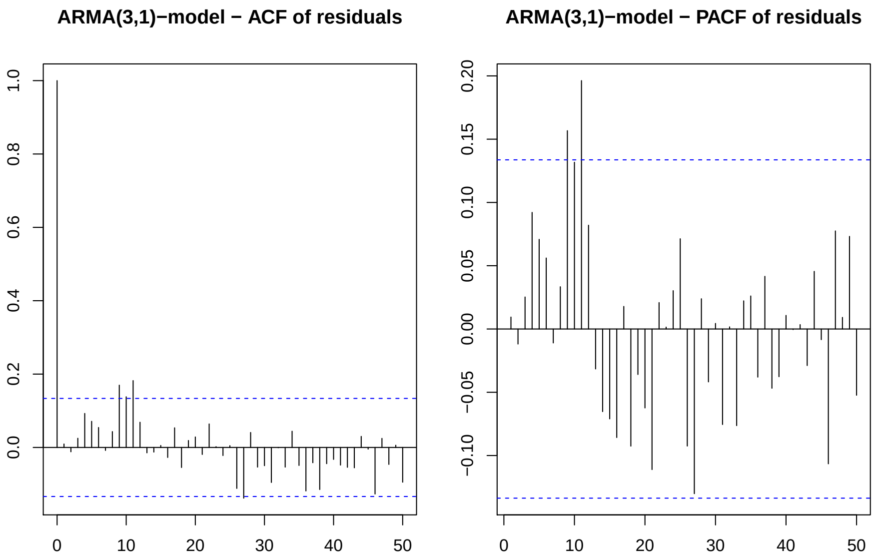

Figure 7: ACF and PACF of the residuals corresponding to the model given by auto.arima().

Comments:

(1) Time series **Spots** turned out to be hard to model with SARIMA processes. Here, the period (around 11 years) of the seasonal component is not constant, which is difficult to model with SARIMA processes.

(2) The corresponding sun spot time series has been widely studied in the literature. According to current understanding, one of the better candidates to describe its behavior is a so called threshold model.

We plot the fitted models together with the original time series.

```
fit.sun <- fitted(model2)
fit.auto <- fitted(model.auto)
plot(fit.sun,type="b",col="blue",
     ylab="Spots",xlab="Time")
lines(Spots,col="red",type="b")
lines(fit.auto,col="green",type="b")
legend("topleft", legend=c("Spots time series", "AR(2)-fit",
                           "ARMA(3,1)-fit"),
       col=c("red","blue","green"),lty=c(1,1),cex=0.8)
```

We can see from Figure 8 that the behaviour of both fitted models is consistent with the original time series. We predict the last 43 observations by using an AR(2) model:

```
model_ver <- Arima(Spots[1:172],order=c(2,0,0))
# Set the first year of the prediction correctly
```

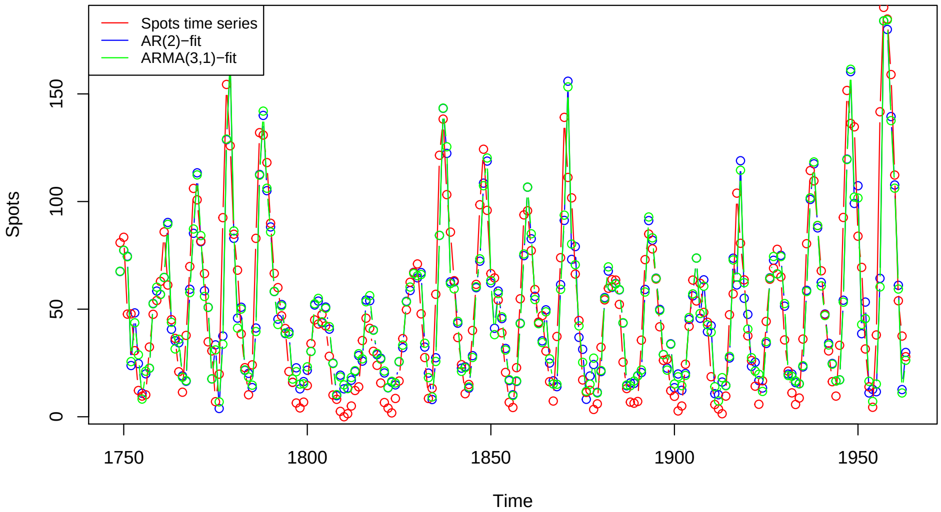

Figure 8: The **Spots** time series as red, fitted AR(2) model as blue and fitted ARMA(3,1) model as green.

```
prediction <- ts(forecast(model_ver,h=43)$mean,start=1921)
plot(Spots,col="red",type="l",ylim=c(0,200))
lines(prediction,col="blue",type="l")
```

From Figure 9 we see that long term predictions can be unreliable. On the other hand, predicting just few next steps works out relatively well. It is no surprise that the model is not suited for long term predictions, since it was labeled as unsatisfactory by the earlier diagnostics.

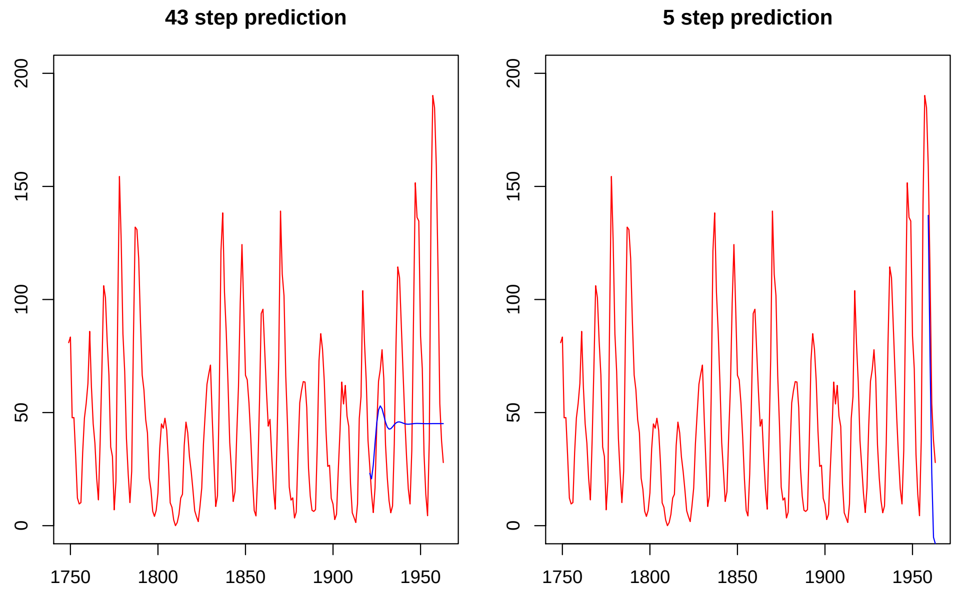

Figure 9: The **Spots** time series as red and the predictions given by the estimated AR(2) models as blue.

Sales

By the previous homework assignment, the time series **Sales** does not look stationary, but $D_{12}D$**Sales** might be. The auto- and partial autocorrelation functions indicate that **Sales** could be modeled with a SARIMA process. We try to find a valid model with the `auto.arima` function, and study the corresponding residuals.

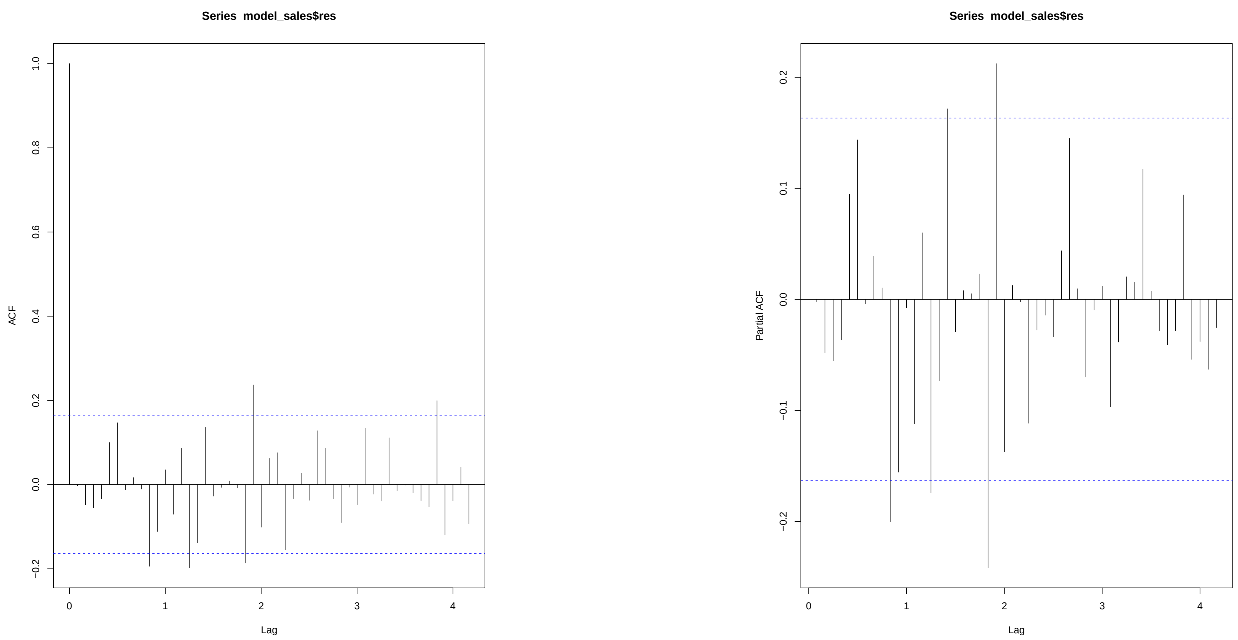

Figure 10: ACF and PACF of the residuals corresponding to the fitted SARIMA(2,1,0)(0,1,1)$_{12}$ for the **Sales** time series

```
Series: Sales
ARIMA(2,1,0)(0,1,1)[12]

Coefficients:
          ar1      ar2     sma1
      -0.7059  -0.4665  -0.4873
s.e.   0.0829   0.0839   0.0957

sigma^2 estimated as 23.67:  log likelihood=-393.6
AIC=795.21   AICc=795.53   BIC=806.71


acf(model_sales$res,lag.max=50)
pacf(model_sales$res,lag.max=50)

# the number of fitted parameters 2+1=3
k <- 3
sales_bl <- rep(NA,47)
for (i in 1:47)
{
  sales_bl[i]=Box.test(model_sales$res,lag=(i+k),fitdf=k,
                       type="Ljung-Box")$p.value
```

```
}
round(sales_bl,3)

[1] 0.326 0.290 0.124 0.216 0.324 0.441 0.109 0.090 0.126 0.144 0.145 0.035 0.020

0.012 0.018 0.027 0.038 0.053 0.016 0.001 0.001

[22] 0.002 0.002 0.001 0.001 0.002 0.002 0.001 0.001 0.002 0.002 0.003 0.003 0.002

0.002 0.003 0.002 0.003 0.004 0.006 0.007 0.009

[43] 0.002 0.001 0.001 0.002 0.001
```

Comments:

The null hypothesis is accepted with lags 4-14 and 21. SARIMA(2,1,0)$(0,1,1)_{12}$ model's result is still not particularly satisfactory according to Ljung-Box test. However, the fitted model is good which can be seen from Figure 11.

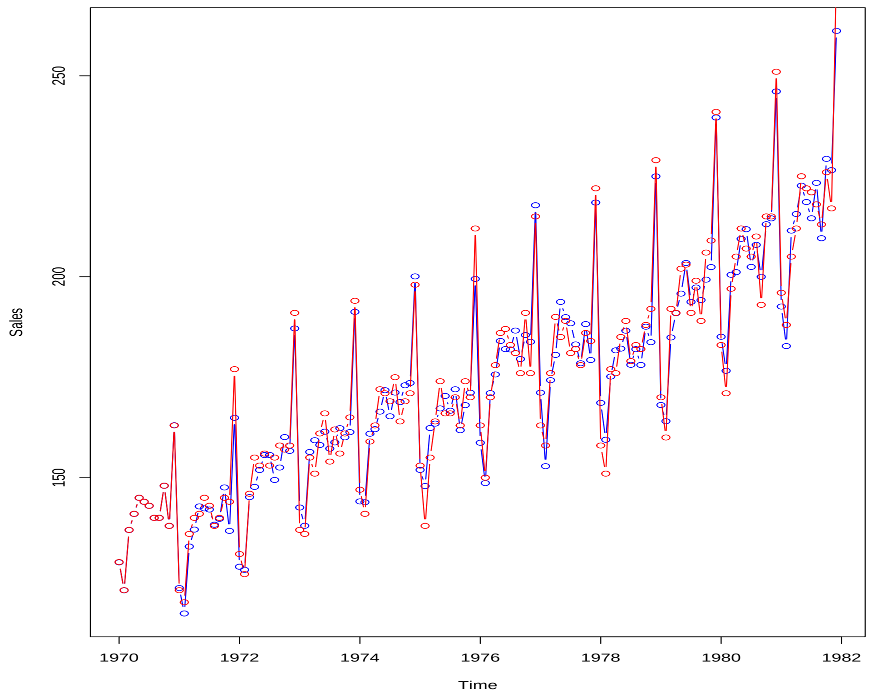

Figure 11: **Sales** as red and the fitted SARIMA(2,1,0)$(1,1,0)_{12}$ process as blue.

Now, how does the 48 time steps (4 years) prediction look like?:

```
fit.sales <- fitted(model_sales2)
plot(fit.sales,type="b",col="blue",
     ylab="Sales",xlab="Time")
lines(Sales,col="red",type="b")

prediction_sales <- forecast(model_sales2,h=48)$mean
plot(Sales,col="red",type="b",ylim=c(100,340),
     xlim=c(1970,1987),ylab="Sales",xlab="Time")
lines(prediction_sales,col="blue",type="b")
```

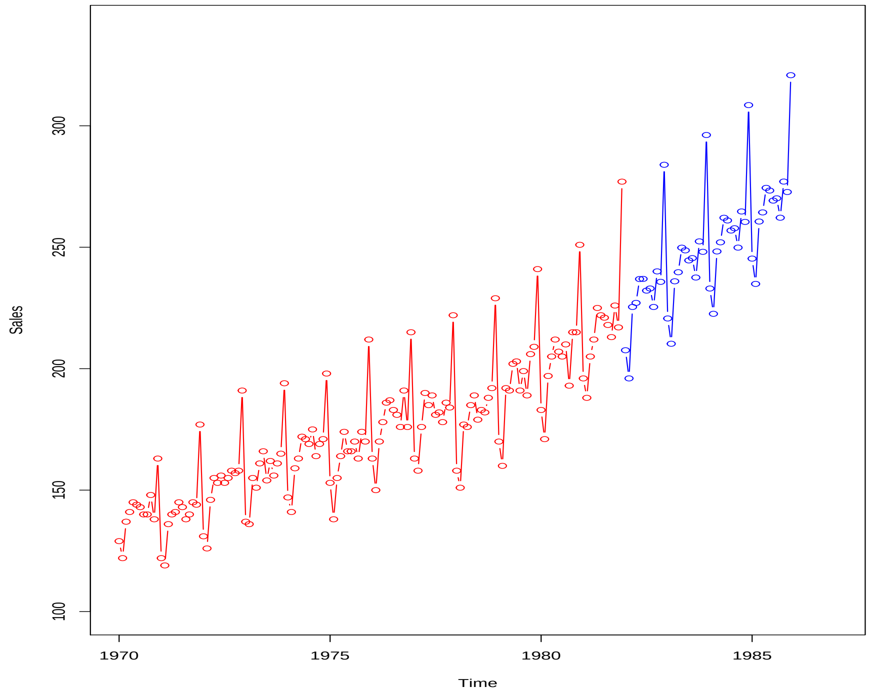

Figure 12: **Sales** as red and the prediction given by the SARIMA(2,1,0)(0,1,1) model as blue.

From Figure 12 we see that the four year prediction seems reasonable, although according to the Ljung-Box test the SARIMA$(2,1,0)(1,1,0)_{12}$ model was not satisfactory.

Homework

4.3 A time series of carbon dioxide measurements from the Mauna Loa volcano is given in the file `MLCO2.txt`. The length of the time series is 216 months. Recall that, we studied this time series during the third computer exercises.

a) Using SARIMA processes, find the best possible model to describe the time series **MLCO2**.

b) Make 2 and 24 time step predictions by using the model chosen in a). Study the goodness of the predictions.