Predictive Modeling

Series 13

Exercise 13.1

We consider the autoregressive process

$$X_n = 1.21X_{n-1} - 0.5X_{n-2} - 0.13X_{n-3} + 0.24X_{n-4} + W_n$$

where W_n is a Gaussian white noise process with variance $\sigma^2 = 1$.

a) Write down the characteristic polynomial of the process and compute its zeros.

Hint: You may use the **polyroot()** command but you can also use the computation device of your choice. The zeros of the polynomial $f(x) = 2 - x + 3x^2$ for instance can be computed by

```
polyroot(c(2, -1, 3))
```

b) Is the process stationary?

c) Generate four different times series from the process with $n = 200$ observations each. Plot the time series and compare them. Do the time series look mutually similar? What do they have in common?

Hint: In **R** you can generate time series from a given autoregressive model with the **arima.sim** command. The following code creates a time series with $n = 100$ from an AR(2) model with coefficients $a_1 = 0.2$ and $a_2 = -0.1$

```
my.ts = arima.sim(model = list(ar = c(0.2, -0.1)), n = 100)
```

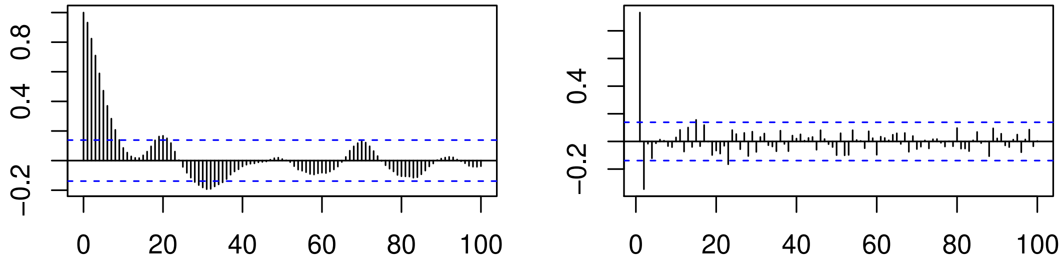
d) Compute the autocorrelation and partial autocorrelation of the process and plot the sequences up to lag 50 next to each other.

Hint: The exact (partial) autocorrelation of an AR(p) process can be computed with the **ARMAacf()** command. The following code produces the autocorrelation and the correlogram for an AR(2) with coefficients $a_1 = 0.2$ and $a_2 = -0.1$

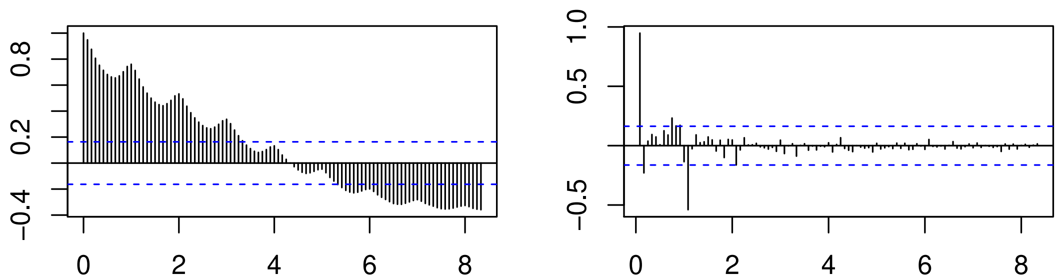
```
acf = ARMAacf(ar = c(0.2, -0.1), pacf = F, lag.max = 50)
plot(acf, type = "h")
```

Exercise 13.2

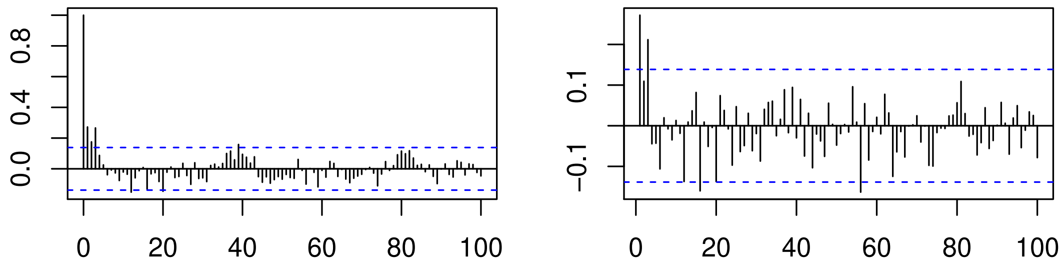
In the following there are three pairs of sample autocorrelation/partial autocorrelation plots each belonging to a time series that is not shown. Decide whether the process is autoregressive or not. If yes, estimate the model order p.



a)



b)



c)

Exercise 13.3

In this section we again consider the monthly global temperature data. Recall that there is an increasing trend in after 1970, which may be due to the *greenhouse effect*. Sceptics may claim that the apparent increasing trend can be dismissed as a transient stochastic phenomenon. For their claim to be consistent with the time series data, it should be possible to model the trend without the use of deterministic functions.

a) Create a time series that contains the annual average of the global temperature and plot the series.

   Hint: Use the `aggregate()` command to do the job. The following code-snippet gives good advice

```
ts.annual = aggregate(ts.monthly, FUN = mean)
```

b) Plot a correlogram and partial correlogram for the mean annual temperature series. Comment on the plots.

c) Fit an AR(4) process to the data and analyze the residuals.

   Hint: The function `ar()` returns a model object that contains the field `resid`. These are the residuals between the model and the data. Examine the residuals with a histogram, a qq-plot, ...

d) Predict the mean annual temperature for the next 100 years with the `predict()` function using the AR(4) model in b). Create a time plot of the mean annual temperature series and add the 100-year forecasts to the plot. **Hint:** This is one way to achieve this (replace the variable names according to your workspace):

```
pred = predict(myModel, n.ahead = 100)
plot(ts.annual, xlim = c(1850, 2217))
lines(pred$pred, col = "red")
```

e) Add a line representing the overall mean global temperature. Comment on the final plot and any potential inadequacies in the fitted model.

   Hint: A constant line can be drawn in **R** with the command `abline(h = ts.annual.mean)`

Exercise 13.4

In this exercise we shall examine measurements of the vertical force acting on a cylinder in a water tank. A total of 320 measurements were taken at intervals of 0.15 seconds.

a) Load the data using

```
load("Daten/force.Rdata")
```

   You will find a time series `force` in your environment. Generate a time series plot.

b) Create a subset of the data containing only the first 280 observations using the `window` command.

c) Inspect the (partial) correlogram and determine the order for an autoregressive model to be estimated. Fit the model to the data and inspect the residuals (cf. Example 13.3).

d) Predict the forces for the next 40 measurements and compare the result to the 40 observations that have been discarded before the modeling step graphically.

   Hint: You will probably find the code in Example 15.1.10 helpful.

Result Checker

Predictive Modeling

Solutions to Series 13

Solution 13.1

a) The characteristic polynomial is given as

$$\Phi(x) = 1 - 1.21x + 0.5x^2 + 0.13x^3 - 0.24x^4$$

The zeros of the polynomial can be computed by

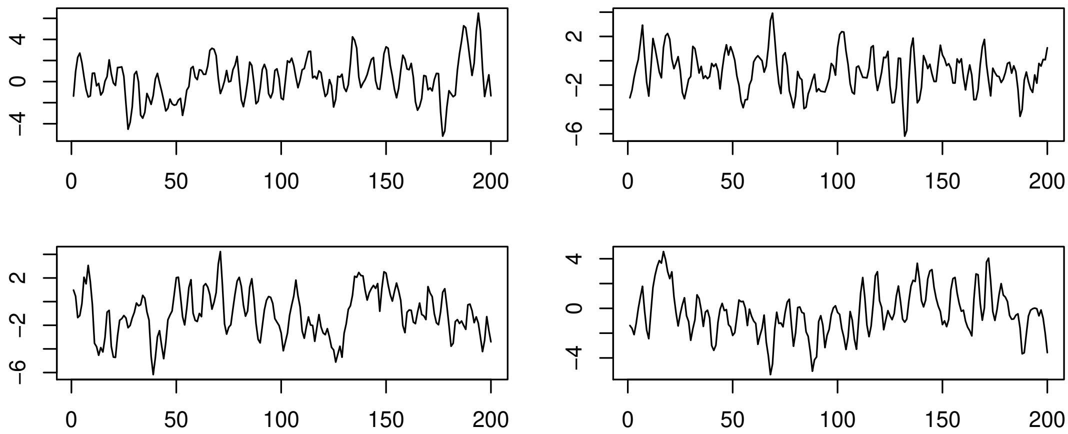
```
zeros = polyroot(c(1, -1.21, 0.5, 0.13, -0.24))
zeros

## [1]  0.714380+1.078574i  1.195442+0.000000i
## [3]  0.714380-1.078574i -2.082536+0.000000i
```

Two of the zeros form a complex conjugate pair.

b) The process is stationary because the absolute values of the zeros of the characteristic polynomial are all larger than 1.

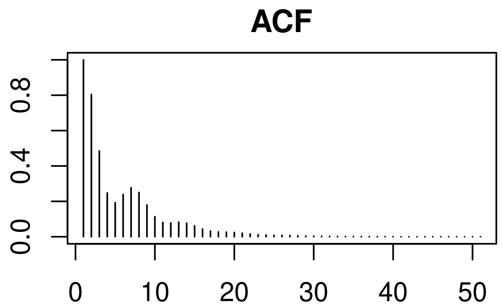
```
abs(zeros)

## [1] 1.293701 1.195442 1.293701 2.082536
```

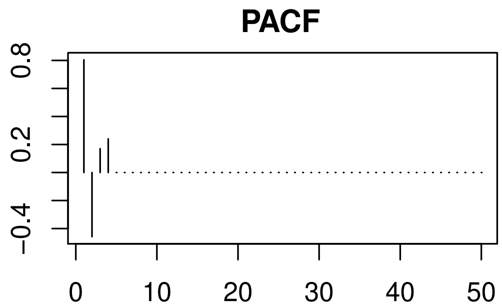
c)
```
for (i in 1:4) {
    my.ts = arima.sim(model = list(ar = c(1.21, -0.5, -0.13,
        0.24)), n = 200)
    plot(my.ts, xlab = "")
}
```



The time series point-wise look very different. There are no trend or seasonal effects whatsoever that occur at fixed times in all sequences. However, the range of the values is similar and there is a structural resemblance of the patterns appearing in the series. The typical distance between the maxima or minima is simimlar between as well as within the signals. The signals are not periodic.

d)
```
acf = ARMAacf(ar = c(1.21, -0.5, -0.13, 0.24), pacf = F,
    lag.max = 50)
plot(acf, type = "h", main = "ACF")

pacf = ARMAacf(ar = c(1.21, -0.5, -0.13, 0.24), pacf = T,
    lag.max = 50)
plot(pacf, type = "h", main = "PACF")
```





The sequences show the typical behaviour of an autoregressive process: The autocorrelation is trailing and decaying exponentially to zero (up to some oscillations). The partial autocorrelation has only some lags that exhibit nonzero values.

Solution 13.2

a) The correlogram is oscillating and decreasing in an (approximately) exponential manner. The partial correlogram shows large values for lags $h = 1, 2$ and is then relatively small. There are large values at higher lags ($h \approx 14$ and $h \approx 22$). So an autoregressive model is plausible but the order is hard to estimate by eyeballing (the plots are generated from an AR(4) data set.)

b) The correloogram shows a peculiar shape. There is a clear seasonal dependence in the correlogram

Solution 13.3

a)
```
load("Daten/GlobalTemperature.Rdata")
AnTemp = aggregate(Global.ts, nfrequency = 1, FUN = mean)
plot(AnTemp)
grid()
```

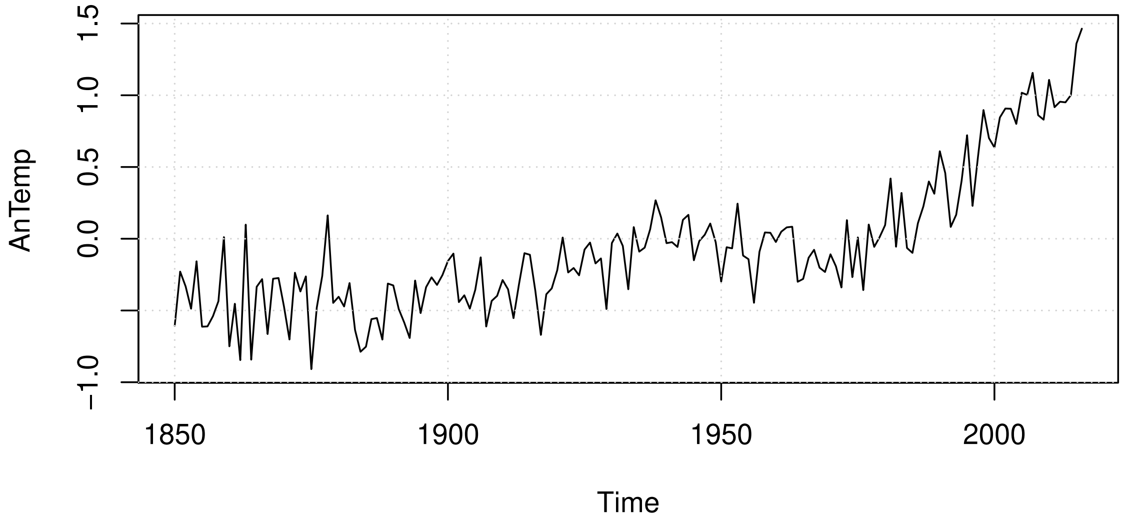
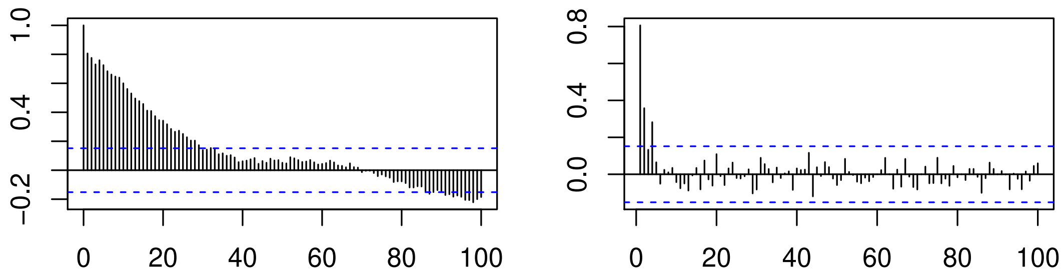
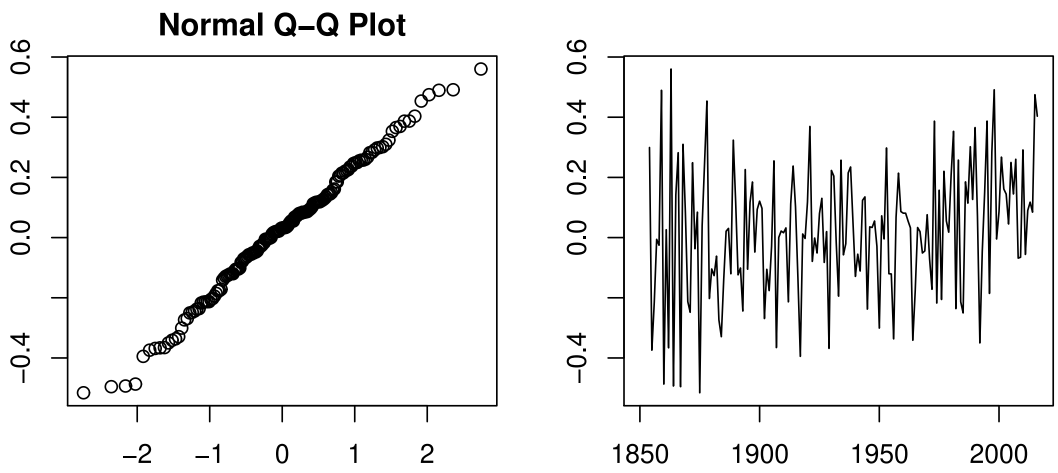


Abbildung 1: Annual average global temperature on the northern hemisphere

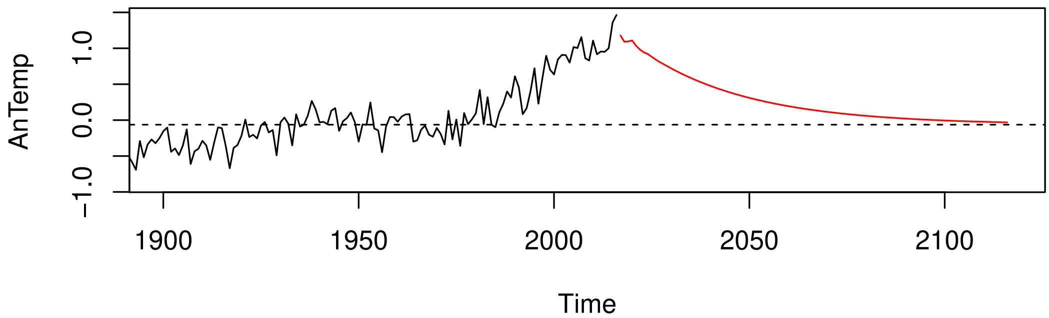
b) 

The shape of the correlogram reveals that the time serie is not stationary.

c)
```
par(mfrow = c(1, 2), mar = c(2, 2, 2, 2))
ar.model = ar(AnTemp, aic = F, order.max = 4)
# plot(AnTemp) lines(AnTemp- ar.model$resid,
# col='red')

qqnorm(ar.model$resid)
plot(ar.model$resid)
```



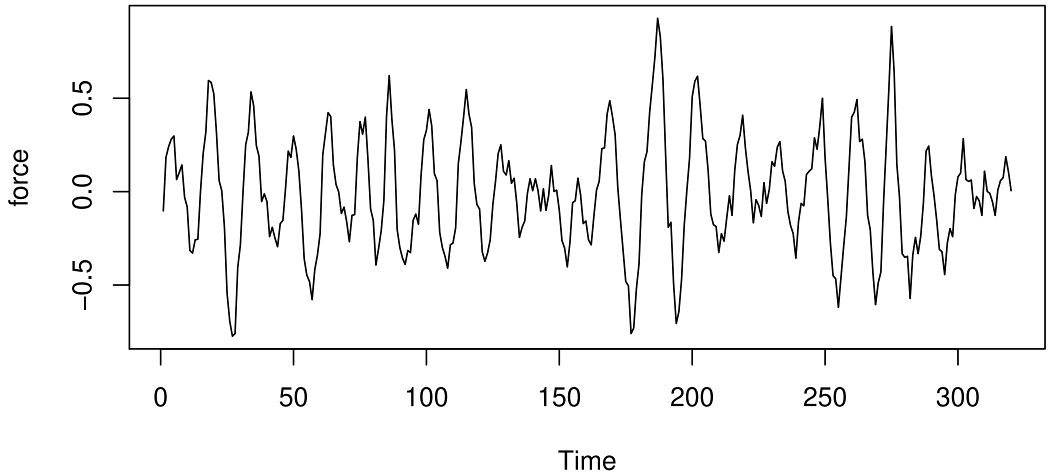
d)
```
AnTempPred = predict(ar.model, n.ahead = 100)$pred
plot(AnTemp, xlim = c(1900, 2117))
lines(AnTempPred, col = "red")
abline(h = mean(AnTemp), lty = 2)
```



e) The time series is not stationary since it exhibits a clear trend after 1970. Modelling this time series with an autoregressive process is therefore conceptually wrong. This can be seen for instance when we keep in mind that AR-processes always converge to zero (or to the mean of the time series) for large lags. The picture above shows that this yields a useless prediction result.

Solution 13.4

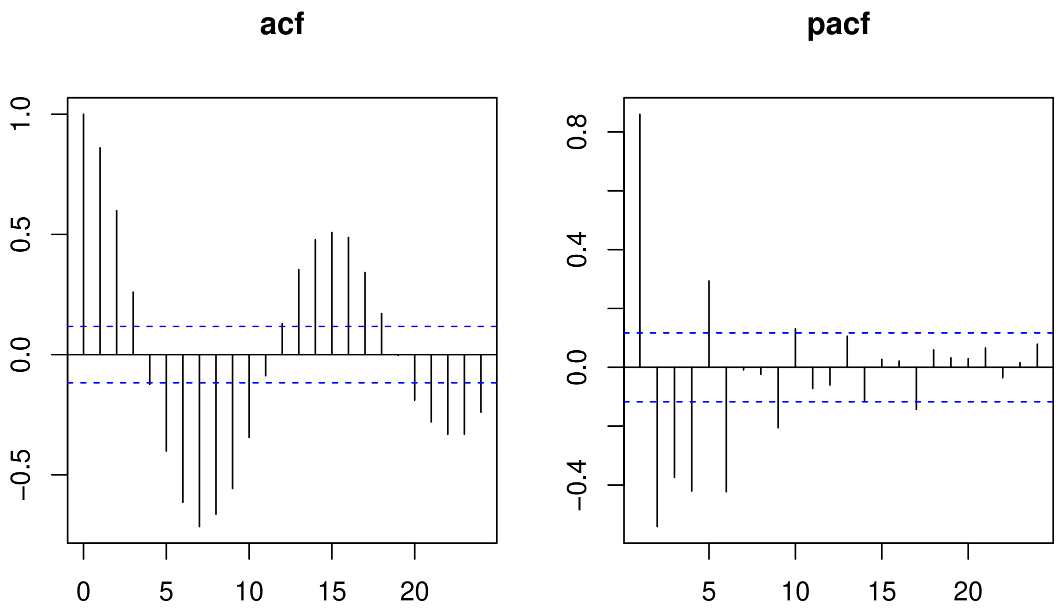
a)
```
load("Daten/force.Rdata")
plot(force)
```



b)
```r
force.train = window(force, end = 280)
```

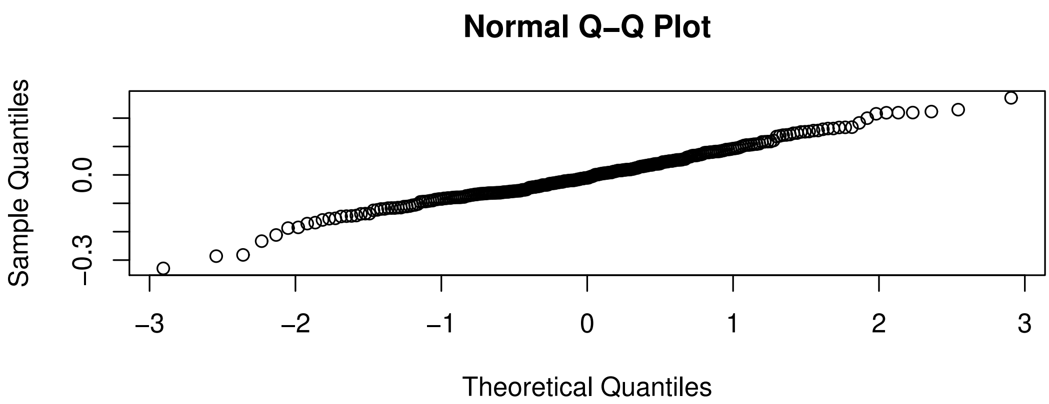
c) We compute the correlograms and we see that the acf has the typical oscillating/exponentially decreasing behaviour. The partial correlogram indicates that the order $p = 7$ could be appropriate.

```r
par(mfrow = c(1, 2), mar = c(3, 2, 3, 2))

acf(force.train, main = "acf")
pacf(force.train, main = "pacf")
```

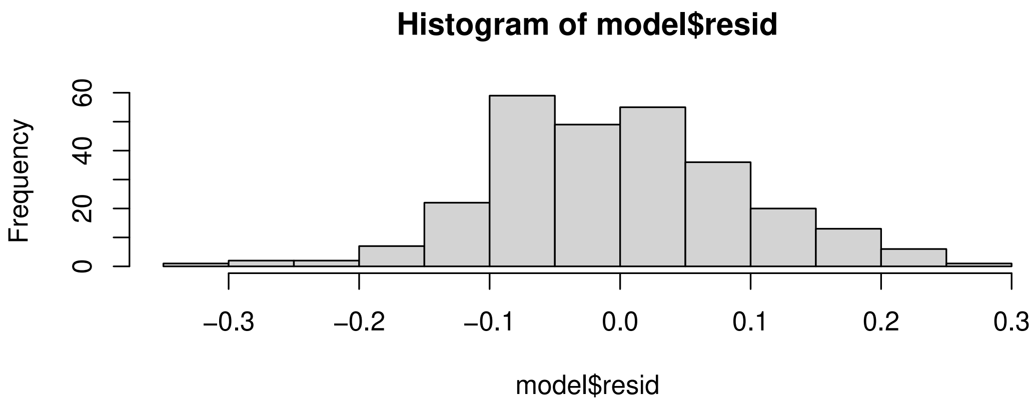


We fit an autoregressive model of order $p = 7$ and inspect the residuals.

```
model = ar(force.train, aic = F, order.max = 7)

qqnorm(model$resid)
```




```
hist(model$resid, nclass = 20)
```



The residuals are sufficiently close to normality from visual inspection.

d)
```
force.pred = predict(model, n.ahead = 40)$pred
force.se = predict(model, n.ahead = 40)$se

plot(force, lty = 3)
lines(force.train)
lines(force.pred, col = "red")

# 95% confidence intervals (optional)
lines(force.pred + 1.96 * force.se, col = "red", lty = 3)
lines(force.pred - 1.96 * force.se, col = "red", lty = 3)
```