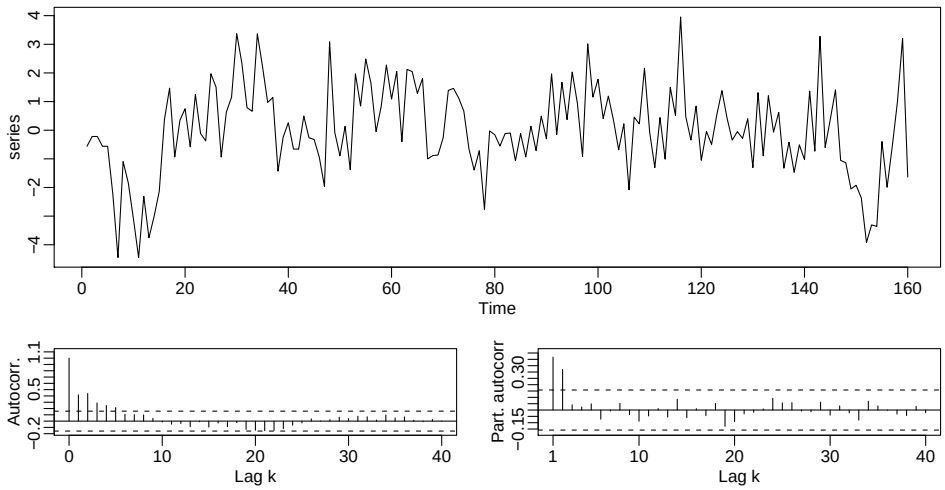
Solution to Series 6

1. a)
```
> r.bel.lm <- lm(NURSING ~ ., data=beluga)
> summary(r.bel.lm)
Call:
lm(formula = NURSING ~ ., data = d.beluga)

Residuals:
     Min       1Q   Median       3Q      Max
-4.44568 -0.90180 -0.08505  1.09525  3.95477

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.5602842  0.5502170   1.018  0.31012
PERIOD       0.0001998  0.0031937   0.063  0.95020
BOUTS        0.8784967  0.3336237   2.633  0.00932 **
LOCKONS      2.3903512  0.2035042  11.746  < 2e-16 ***
DAYNIGHT    -0.3416237  0.2510156  -1.361  0.17550
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 1.582 on 155 degrees of freedom
Multiple R-Squared: 0.842,     Adjusted R-squared: 0.8379
F-statistic: 206.5 on 4 and 155 DF,  p-value: < 2.2e-16


> d.resid <- ts(resid(r.bel.lm))
> plot(d.resid)
> acf(d.resid, lag=40)
> pacf(d.resid, lag=40)
```



The correlogram of the residuals shows that significant correlation is present. Consequently, all confidence intervals and tests in the output of lm can be wildly inaccurate. It is thus impossible for zoologists to conclude which explanatory variables are needed in the model.

b) Due to the partial autocorrelations present, an AR(2) model for the residuals makes sense. Note that the ordinary autocorrelations make up a dampened sine curve, a property typical of AR processes. We can use the Burg algorithm to estimate both AR parameters:

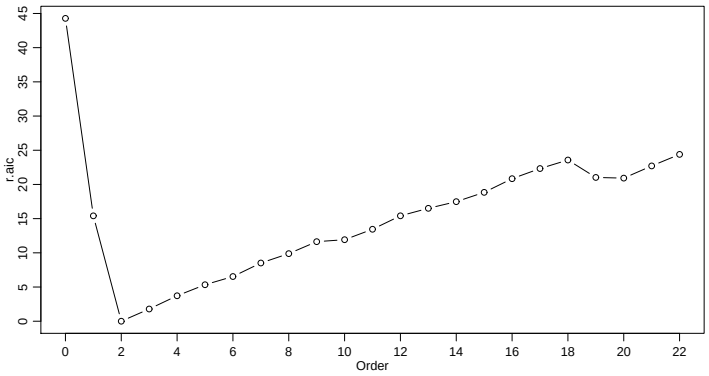
```
> r.burg <- ar(d.resid, method="burg", order.max=2, aic=F)
> str(r.burg)
```

in R, we obtain:

$\alpha_1 = 0.284, \alpha_2 = 0.321$.

Note: We can also use the AIC plot to determine the order of the process:

```
> r.aic <- ar(d.resid, method="burg")$aic
> plot(0:(length(r.aic)-1), r.aic, xlab="Order", type="b")
```



It seems that $p = 2$ is a good order to take.

c) We have

$$\begin{aligned} Y_t &= \beta_0 + \beta_1 \cdot t + \beta_2 X_{2,t} + \beta_3 X_{3,t} + \beta_4 X_{4,t} + E_t \qquad (t = 1, \ldots, 160) \\ \text{with} \quad & E_t = \alpha_1 E_{t-1} + \alpha_2 E_{t-2} + U_t \qquad U_t \text{ i.i.d. }, \; E[U_t] = 0, \; \mathrm{Var}[U_t] = \sigma^2 \; , \end{aligned}$$

where $Y_t =$ `NURSING`, $X_{1,t} = t =$ `PERIOD`, $X_{2,t} =$ `BOUTS`, $X_{3,t} =$ `LOCKONS` and $X_{4,t} =$ `DAYNIGHT`. Computing $Y_t^* = Y_t - \alpha_1 Y_{t-1} - \alpha_2 Y_{t-2}$:

$$\begin{aligned} Y_t^* &= Y_t - \alpha_1 Y_{t-1} - \alpha_2 Y_{t-2} \\ &= \beta_0 + \beta_1 \cdot t + \beta_2 X_{2,t} + \beta_3 X_{3,t} + \beta_4 X_{4,t} + E_t \\ &\quad -\alpha_1\big(\beta_0 + \beta_1 \cdot (t-1) + \beta_2 X_{2,t-1} + \beta_3 X_{3,t-1} + \beta_4 X_{4,t-1} + E_{t-1}\big) \\ &\quad -\alpha_2\big(\beta_0 + \beta_1 \cdot (t-2) + \beta_2 X_{2,t-2} + \beta_3 X_{3,t-2} + \beta_4 X_{4,t-2} + E_{t-2}\big) \\ &= \beta_0(1 - \alpha_1 - \alpha_2) + \beta_1(t - \alpha_1(t-1) - \alpha_2(t-2)) \\ &\quad +\beta_2(X_{2,t} - \alpha_1 X_{2,t-1} - \alpha_2 X_{2,t-2}) + \ldots + E_t - \alpha_1 E_{t-1} - \alpha_2 E_{t-2} \\ &= \beta_o^* + \beta_1 X_{1,t}^* + \beta_2 X_{2,t}^* + \beta_3 X_{3,t}^* + \beta_4 X_{4,t}^* + U_t \end{aligned}$$

The explanatory variables and the target must all be transformed as follows:

$$x_t^* = x_t - \widehat{\alpha}_1 x_{t-1} - \widehat{\alpha}_2 x_{t-2} = x_t - 0.284 \cdot x_{t-1} - 0.321 \cdot x_{t-2}$$

d) (*) The transformation, and the subsequent normal regression, can be performed in R using the following code. Note that the residuals now no longer exhibit correlation.

```
> t.ar <- r.burg$ar
> ## Transform the entire multivariate time series
> d.beluga.tr <- d.beluga - t.ar[1]*lag(d.beluga,-1) - t.ar[2]*lag(d.beluga,-2)
> ## Set new (meaningful) colnames
> colnames(d.beluga.tr) <- paste(colnames(d.beluga),".tr",sep="")
[1] "PERIOD.tr"   "BOUTS.tr"    "NURSING.tr"  "LOCKONS.tr"  "DAYNIGHT.tr"
> t.intercept <- rep((1-t.ar[1]-t.ar[2]),nrow(d.beluga.tr))
> r.lm.tr <- lm(NURSING.tr ~ -1 + t.intercept + PERIOD.tr + BOUTS.tr +
+               LOCKONS.tr + DAYNIGHT.tr, data=d.beluga.tr)
> plot(r.lm.tr$resid)
> acf(r.lm.tr$resid)
> pacf(r.lm.tr$resid)
```

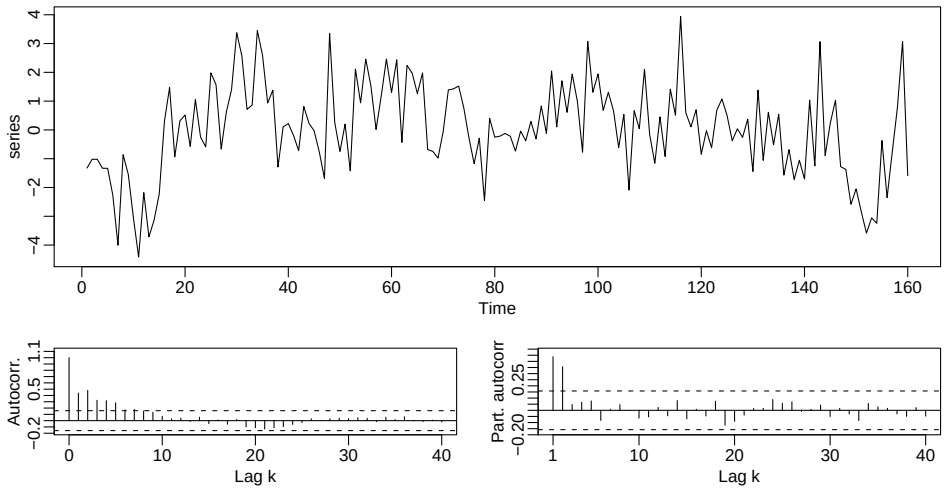
e) The procedure `gls()` can be used for much more general models than those you have already seen. The argument `correlation` can be used for specifying the correlation structure of the residuals. In principle an AR(p) model is merely a special case of the ARMA(p, q) model taking $q = 0$. This explains the overly complex expression `corARMA(value=c(...,...), p=2, q=0, fixed=F)`. The AR coefficients computed in Part b) can be used as starting values by specifying them in the argument `value`. Errors in different time periods can be specified as being correlated by means of the argument `form=` $\sim$ `PERIOD` of `corARMA`. This is necessary, as the entries in the data matrix can be arranged in any way.

R-output from `summary(r.bel.gls)`:

```
Generalized least squares fit by maximum likelihood
  Model: NURSING ~ BOUTS + LOCKONS + DAYNIGHT + PERIOD
  Data: d.beluga
      AIC      BIC   logLik
  560.396 584.9974 -272.198

Correlation Structure: ARMA(2,0)
 Formula: ~PERIOD
 Parameter estimate(s):
     Phi1      Phi2
0.2739964 0.3653668

Coefficients:
                Value Std.Error   t-value p-value
(Intercept)  1.3218871 0.7678364  1.721574  0.0871
BOUTS        0.2961684 0.3370588  0.878685  0.3809
LOCKONS      2.5681923 0.1964012 13.076257  <.0001
DAYNIGHT    -0.3080293 0.1549160 -1.988363  0.0485
PERIOD       0.0024982 0.0062754  0.398090  0.6911

 Correlation:
         (Intr) BOUTS  LOCKON DAYNIG
BOUTS    -0.303
LOCKONS  -0.101 -0.811
DAYNIGHT -0.014 -0.135  0.067
PERIOD   -0.607 -0.233  0.251  0.024

Standardized residuals:
        Min          Q1         Med          Q3         Max
-2.80055625 -0.58763749  0.01738824  0.65602061  2.49854120

Residual standard error: 1.577031
Degrees of freedom: 160 total; 155 residual
```

These coefficient estimates differ markedly from those in Part a). We obtain $\alpha_1 = 0.274$ and $\alpha_2 = 0.365$, which can be found in the above R output at `Parameter estimate(s)` (here labelled as `Phi1` and `Phi2`). In particular note that the standard errors of the explanatory variables sometimes differ greatly from those in the regression model.

Residual analysis:



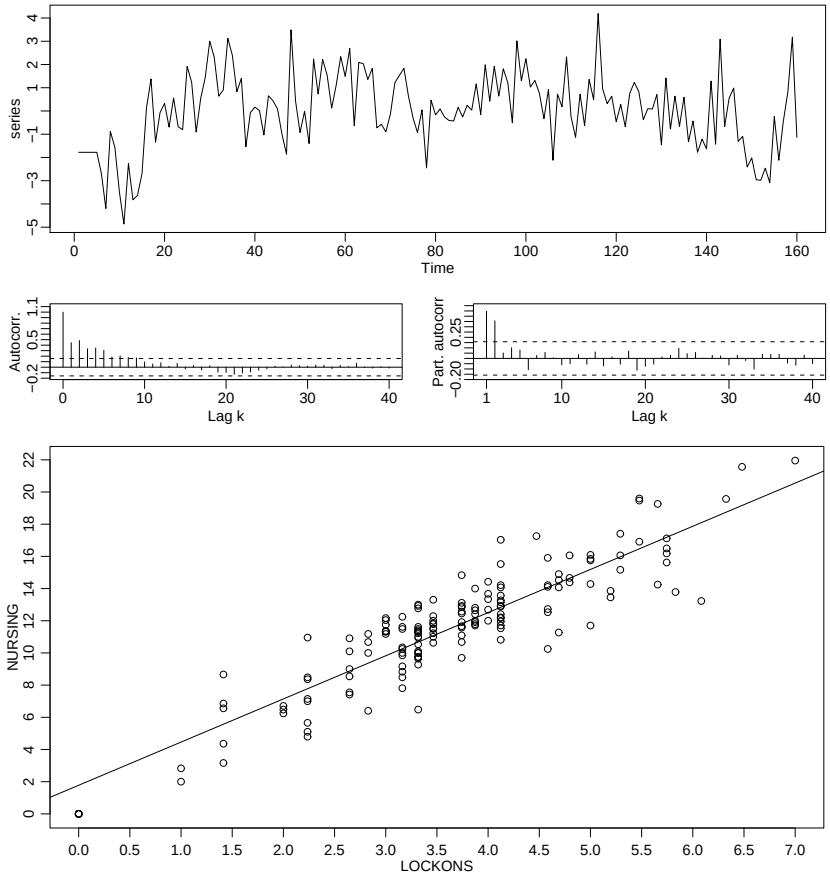
There are only small differences to the model using ordinary regression. This is because residuals denote the difference between observations and model-derived fitted values – and the least squares estimates of coefficients do make sense here. It is merely the standard errors of the least squares method that are wrong. The residuals form an AR(2) process; thus the chosen correlation structure is correct.

f) Successively eliminating redundant variables (`PERIOD, BOUTS` and then `DAYNIGHT`) reduces the model.

R output from `summary(r.red.bel.gls)`:

```
 Generalized least squares fit by maximum likelihood
  Model: NURSING ~ LOCKONS
  Data: d.beluga
       AIC      BIC    logLik
  559.1555 574.5314 -274.5778

Correlation Structure: ARMA(2,0)
 Formula: ~PERIOD
 Parameter estimate(s):
     Phi1      Phi2
0.2803981 0.3696418

Coefficients:
              Value Std.Error   t-value p-value
(Intercept) 1.778230 0.5048868  3.522037   6e-04
LOCKONS     2.682246 0.1147227 23.380250  <.0001

 Correlation:
        (Intr)
LOCKONS -0.804

Standardized residuals:
        Min          Q1         Med          Q3         Max
-3.01255719 -0.57430640  0.05979804  0.69560485  2.59582932

Residual standard error: 1.614887
Degrees of freedom: 160 total; 158 residual
```

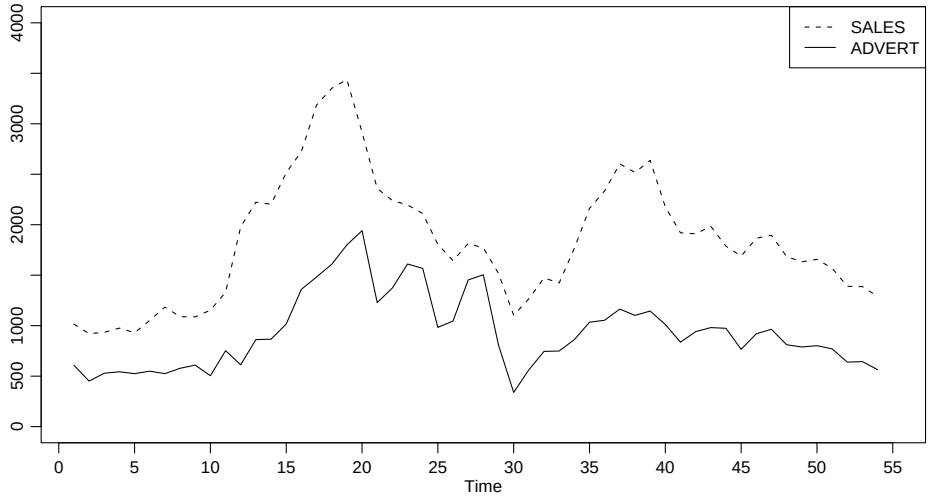
Note:
As you are not using an ordinary lm object, you cannot use the function `step()`. You will need to eliminate variables individually until all remaining variables are significant.
The analysis of residuals does not show any breach of the assumptions on errors, i.e. the residuals do still constitute an AR(2) process as assumed in the construction of the model. The fitted line is given in the last plot.



R commands for these plots:

```
> plot(ts(resid(r.red.bel.gls)))
> acf(ts(resid(r.red.bel.gls)))
> pacf(ts(resid(r.red.bel.gls)))
```
and
```
> plot(d.beluga[,4], d.beluga[,3], xlab="LOCKONS", ylab="NURSING")
> abline(r.red.bel.gls)
```

2. **a)** In the time series plot, the dependence of the two series is evident. When advertising expenditure increases (ADVERT), so do sales (SALES) (or vice versa?).



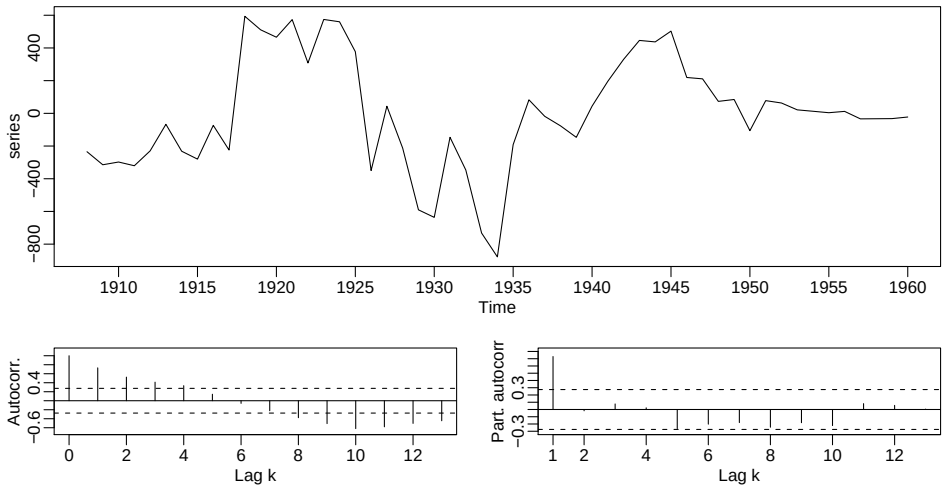
R commands to create such a plot:
```
> plot(d.advert$ADVERT, ylim=c(0,4000), ylab="")
> lines(d.advert$SALES, lty=2)
> legend(0, 4000, c("SALES","ADVERT"), lty=c(2,1))
```

b) We regard the model

$$\texttt{SALES}_t = \beta_0 + \beta_1\,\texttt{ADVERT}_t + \beta_2\,\texttt{ADVERT}_{t-1} + E_t\,.$$

R commands and output:
```
> r.lm1 <- lm(SALES ~ ADVERT + ADVERT1, data=d.advert.ts)
> summary(r.lm1)

Call:
lm(formula = SALES ~ ADVERT + ADVERT1, data = d.advert.ts)

Residuals:
    Min      1Q  Median      3Q     Max
-877.94 -224.37  -18.10  211.06  593.62

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 496.68768  135.76609   3.658  0.00061 ***
ADVERT        1.35243    0.22704   5.957 2.55e-07 ***
ADVERT1       0.08066    0.22753   0.355  0.72445
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 346.3 on 50 degrees of freedom
Multiple R-Squared: 0.7081,     Adjusted R-squared: 0.6965
F-statistic: 60.66 on 2 and 50 DF,  p-value: 4.263e-14

> r.res1 <- ts(resid(r.lm1), start=1908)
> plot(r.res1, type="l")
> acf(r.res1, lag=13)
```

```
> pacf(r.res1, lag=13)
```



The time series plot of residuals, and the corresponding correlograms, show that the errors are correlated and behave as an AR(1) process.

Consequences:

Correlation of residuals means that subsequently, the confidence intervals for coefficients β_0, β_1 and β_2 are inaccurate, which has an adverse effect on predictions and their precision. Since the setup of this exercise means that prediction is our main interest, this model really should be improved first.

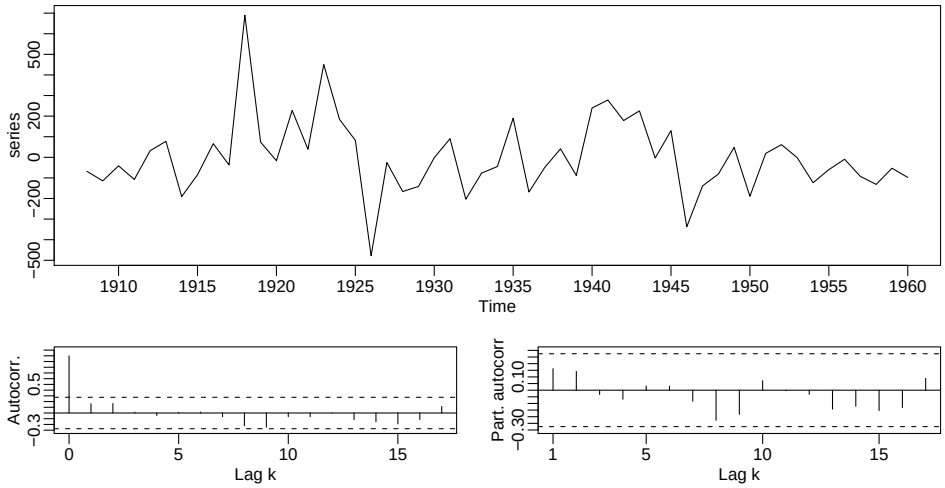
c) We extend the model from part b) by introducing the following variables $\texttt{SALES}_{t-1} = \texttt{SALES1}$:

$$\texttt{SALES}_t = \beta_0 + \beta_1\,\texttt{ADVERT}_t + \beta_2\,\texttt{ADVERT}_{t-1} + \beta_3\,\texttt{SALES}_{t-1} + E_t\,.$$

Note that the variable SALES serves both as a target and as an explanatory variable.

R commands and output:

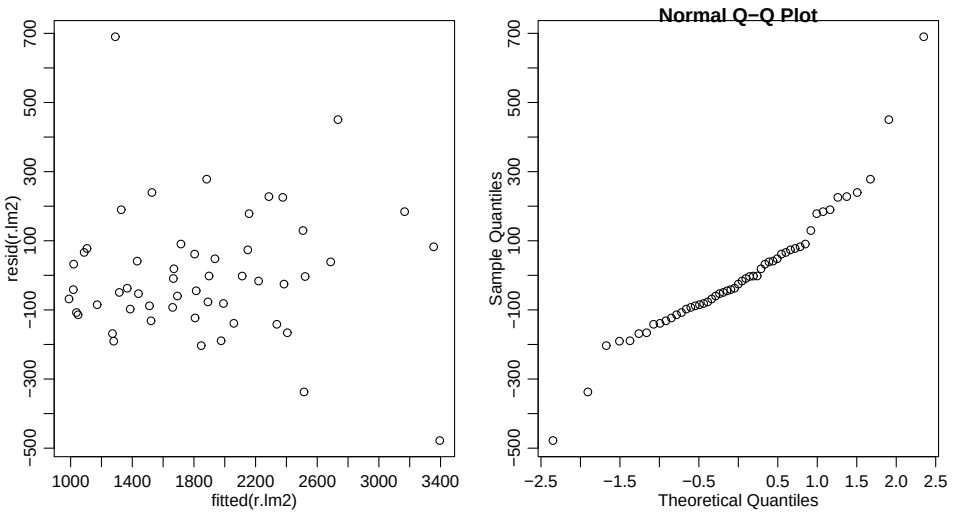
```
> r.lm2 <- lm(SALES ~., data=d.advert.ts)
> summary(r.lm2)
Call:
lm(formula = SALES ~ ., data = d.advert.ts)

Residuals:
    Min      1Q  Median      3Q     Max
-477.94  -97.66  -25.39   73.64  690.21

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 154.06533   80.49015   1.914 0.061458 .
ADVERT        0.58944    0.14232   4.142 0.000136 ***
SALES1        0.95546    0.08764  10.902 1.07e-14 ***
ADVERT1      -0.66006    0.14156  -4.663 2.43e-05 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 189 on 49 degrees of freedom
Multiple R-Squared: 0.9148,    Adjusted R-squared: 0.9096
F-statistic: 175.4 on 3 and 49 DF,  p-value:     0

> r.res2 <- ts(resid(r.lm2), start=1908)

> plot(r.res2)
> acf(r.res2)
> pacf(r.res2)
```



The plot of residuals - and moreover, the correlograms - no longer exhibit unwanted correlation. By including the additional variable $\texttt{SALES}_{t-1}$, we have succeeded in eliminating the autocorrelation of residuals from the model in b).

Checking the assumption on the distribution of residuals:

```
> plot(fitted(r.lm2), resid(r.lm2))
> qqnorm(resid(r.lm2))
```

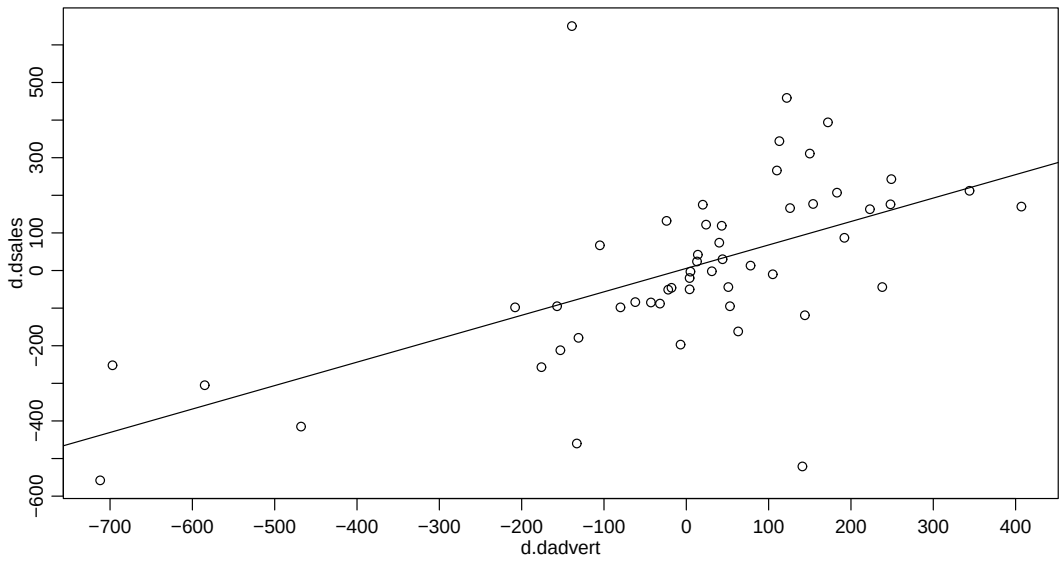


In the time series plot of residuals, and in the normal and Tukey-Anscombe plots, however, 2 outliers are visible. These observations should be looked at more closely. Simply omitting them is not an option, since this obviously causes problems for a time series. (Simply omitting outliers is a bad habit anyway.)

d) We regard the model

$$\texttt{D\_SALES}_t = \beta_0 + \beta_1\, \texttt{D\_ADVERT}_t + E_t\,,$$

where $\texttt{D\_SALES}_t = \texttt{SALES}_t - \texttt{SALES}_{t-1}$ and $\texttt{D\_ADVERT}_t = \texttt{ADVERT}_t - \texttt{ADVERT}_{t-1}$ are the first-order differences. The fitted line is shown in the following plot:



R commands and output:

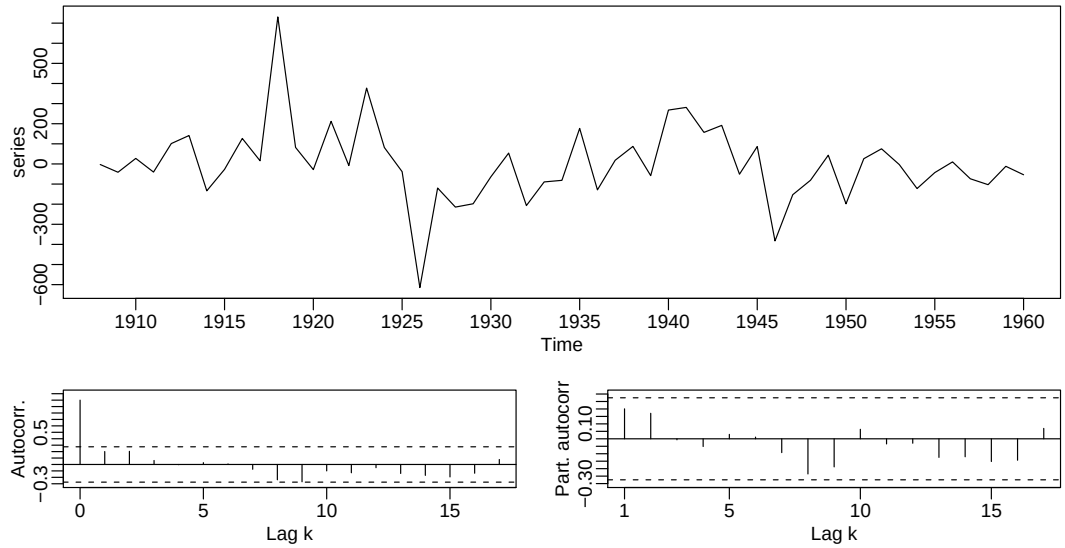
```
> r.lm3 <- lm(d.dsales ~ d.dadvert)
> summary(r.lm3)
Call:
lm(formula = d.dsales ~ d.dadvert)

Residuals:
    Min      1Q  Median      3Q     Max
-614.56  -81.46  -11.79   82.11  730.98

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.6685    26.4415   0.214    0.831
d.dadvert     0.6234     0.1206   5.168 3.98e-06 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 192.5 on 51 degrees of freedom
Multiple R-Squared: 0.3437,    Adjusted R-squared: 0.3308
F-statistic:  26.7 on 1 and 51 DF,  p-value: 3.982e-06
```

R commands for plotting the fitted line:

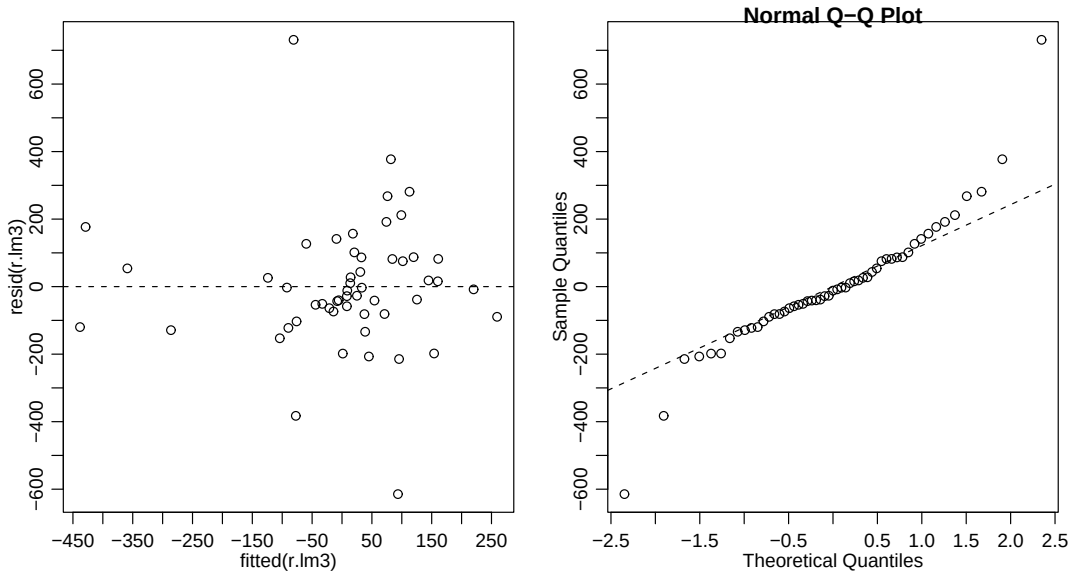
```
> plot(d.dadvert, d.dsales); abline(r.lm3)
```

Analysis of residuals:

```
> r.res3 <- ts(resid(r.lm3), start=1908)
> f.acf(r.res3)
```




```
> plot(fitted(r.lm3), resid(r.lm3))
> qqnorm(resid(r.lm3))
```



The correlograms do not exhibit any undesired correlation. All the ordinary and partial autocorrelations lie inside the confidence band.

However, the time series plot of residuals and the normal and Tukey-Anscombe plots again contain 2 outliers. The fitted model is

$$\texttt{D\_SALES}_t = 5.668 + 0.623 \cdot \texttt{D\_ADVERT}_t + E_t\,.$$

The intercept $\widehat{\beta}_0 = 5.668$ is not significant and could possibly be removed from the model.

e) Comparison of both models:

c) $\texttt{SALES}_t = \beta_0 + \beta_1\,\texttt{ADVERT}_t + \beta_2\,\texttt{ADVERT}_{t-1} + \beta_3\,\texttt{SALES}_{t-1} + E_t$

d) $\texttt{D\_SALES}_t = \beta_0 + \beta_1\,\texttt{D\_ADVERT}_t + E_t$ corresponds to the model $\texttt{SALES}_t = \beta_0 + \beta_1\,\texttt{ADVERT}_t - \beta_1\,\texttt{ADVERT}_{t-1} + \texttt{SALES}_{t-1} + E_t$

- In both models the errors satisfy the assumption of independence. However, both models breach the assumption on their distribution, and there are outliers.
- Both models contain the same explanatory variables, but the model in part d) contains restrictions on the regression coefficients (only 2 coefficients are estimated here!).
- The second model is somewhat simpler to interpret than the first one. However, model d) does not fit as well as model c): its R^2 is only 0.344 compared to $R^2 = 0.915$ in model c).

Notes — Outlook:
In this example it is difficult to determine which series influences the other one. The theory distinguishes two settings:

- Both series influence each other. Such models are called **bivariate autoregressive models**.
- Only one of the series (y_t) depends on the other one (x_t). Such models are termed **transfer function models**.

The connection between the two time series can be investigated using so-called **cross-correlations**, which you will encounter later. In both cases, however, both y_t and x_t must be assumed to be **stationary** time series.