Chpater 2 Selected Computer Exercises

魏上傑

2023-04-25

目錄


```
library(astsa)
```

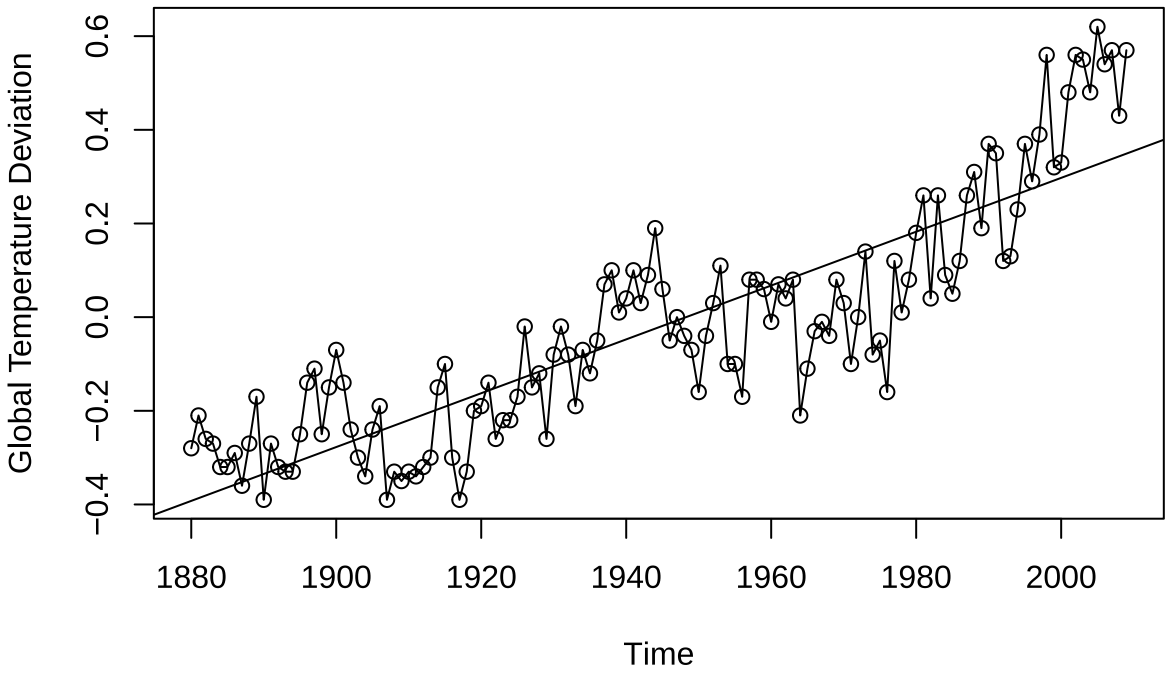
1 Section 2.2

1.1 Example 2.1

```r
summary(fit <- lm(gtemp~time(gtemp)))
```

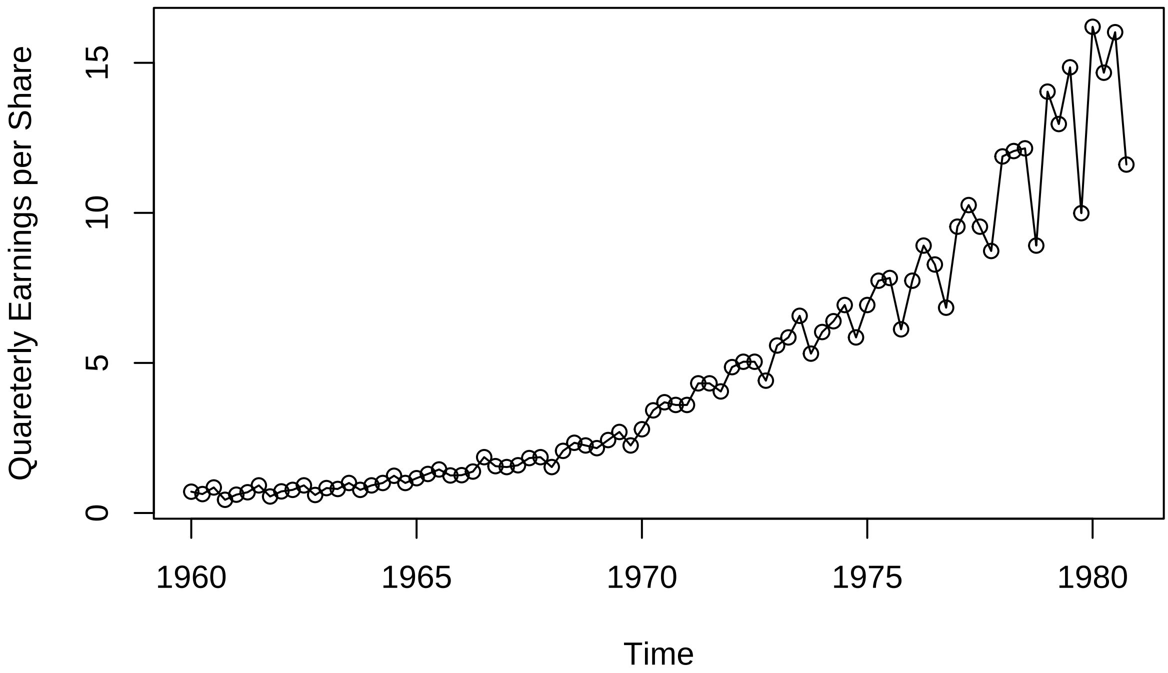
```
##
## Call:
## lm(formula = gtemp ~ time(gtemp))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31946 -0.09722  0.00084  0.08245  0.29383
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.120e+01  5.689e-01  -19.69   <2e-16 ***
## time(gtemp)  5.749e-03  2.925e-04   19.65   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1251 on 128 degrees of freedom
## Multiple R-squared:  0.7511, Adjusted R-squared:  0.7492
## F-statistic: 386.3 on 1 and 128 DF,  p-value: < 2.2e-16
```

```r
plot(gtemp, type="o", ylab="Global Temperature Deviation")
abline(fit) # add regression line to the plot
```



1.2 Problem 2.1

```
# Figure 1.1
library(astsa)
plot(jj, type="o", ylab="Quareterly Earnings per Share")
```



For the Johnson & Johnson data, say y_t, shown in Figure 1.1, let $x_t = log(y_t)$.

(a) Fit the regression model $x_t = \beta_t + \alpha_1 Q_1(t) + \alpha_2 Q_2(t) + \alpha_3 Q_3(t) + \alpha_4 Q_4(t) + w_t$, where $Q_i(t) = 1$ if time t corresponds to quarter i=1,2,3,4, and zero otherwise. The $Q_i(t)$'s are called indicator variables. We will assume for now that w_t is a Gaussian white noise sequence. What is the interpretation of the parameters $\beta, \alpha_1, \alpha_2, \alpha_3$, and α_4? (Detailed code in given in Appendix R on page 574)

`na.action` in `lm()` is to retain the time series attributes for the residuals and fitted values.

```
trend <- time(jj) - 1970 # help to 'center' time
# time() create the vector of times where a time series was sampled

Q <- factor(rep(1:4, 21)) # make (Q)uarter factors
reg <- lm(log(jj)~ 0 + trend + Q, na.action = NULL) # no intercept

# na.action = NULL means no actions to deal with NA

# model.matrix(reg) # view the model matrix
```

```
summary(reg) # view the result
```

```
##
```

```
## Call:
## lm(formula = log(jj) ~ 0 + trend + Q, na.action = NULL)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.29318 -0.09062 -0.01180  0.08460  0.27644
##
## Coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## trend 0.167172   0.002259   74.00   <2e-16 ***
## Q1    1.052793   0.027359   38.48   <2e-16 ***
## Q2    1.080916   0.027365   39.50   <2e-16 ***
## Q3    1.151024   0.027383   42.03   <2e-16 ***
## Q4    0.882266   0.027412   32.19   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1254 on 79 degrees of freedom
## Multiple R-squared:  0.9935, Adjusted R-squared:  0.9931
## F-statistic:  2407 on 5 and 79 DF,  p-value: < 2.2e-16
```

(b) What happens if you include an intercept term in the model in (a)?

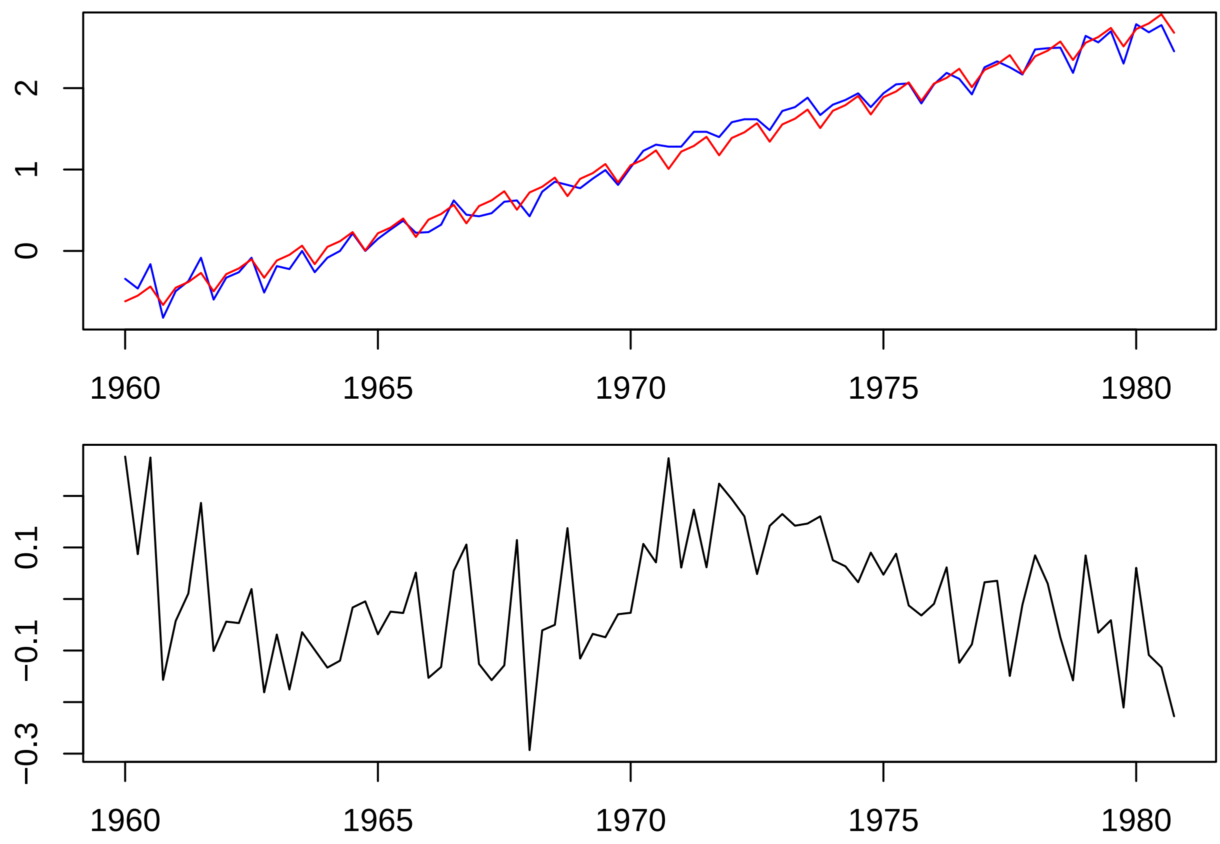
```
reg_with_intercept <- lm(log(jj)~ trend + Q, na.action = NULL)
summary(reg_with_intercept)
```

```
##
## Call:
## lm(formula = log(jj) ~ trend + Q, na.action = NULL)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.29318 -0.09062 -0.01180  0.08460  0.27644
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.052793   0.027359  38.480  < 2e-16 ***
## trend       0.167172   0.002259  73.999  < 2e-16 ***
## Q2          0.028123   0.038696   0.727   0.4695
## Q3          0.098231   0.038708   2.538   0.0131 *
```

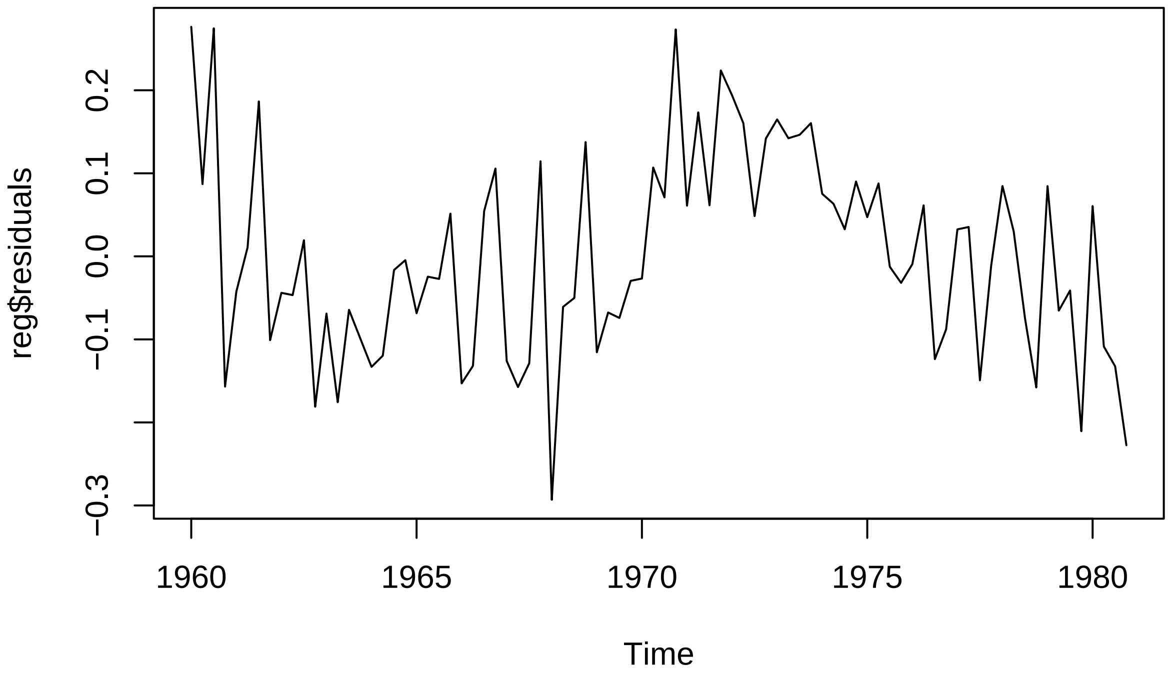
```
## Q4          -0.170527   0.038729  -4.403 3.31e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1254 on 79 degrees of freedom
## Multiple R-squared:  0.9859, Adjusted R-squared:  0.9852
## F-statistic:  1379 on 4 and 79 DF,  p-value: < 2.2e-16
```

(c) Graph the data, x_t, and superimpose the fitted values, say $\hat{x}_t$, on the graph. Examine the residuals, $x_t - \hat{x}_t$, and state your conclusions. Does it appear that the model fits the data well (do the residuals look white)?

```r
par(mar=c(2,2,1,1))
par(mfrow=c(2,1))
plot(log(jj), col="blue")
lines(reg$fitted.values, col="red")
resid <- log(jj)-reg$fitted.values
plot(resid)
```

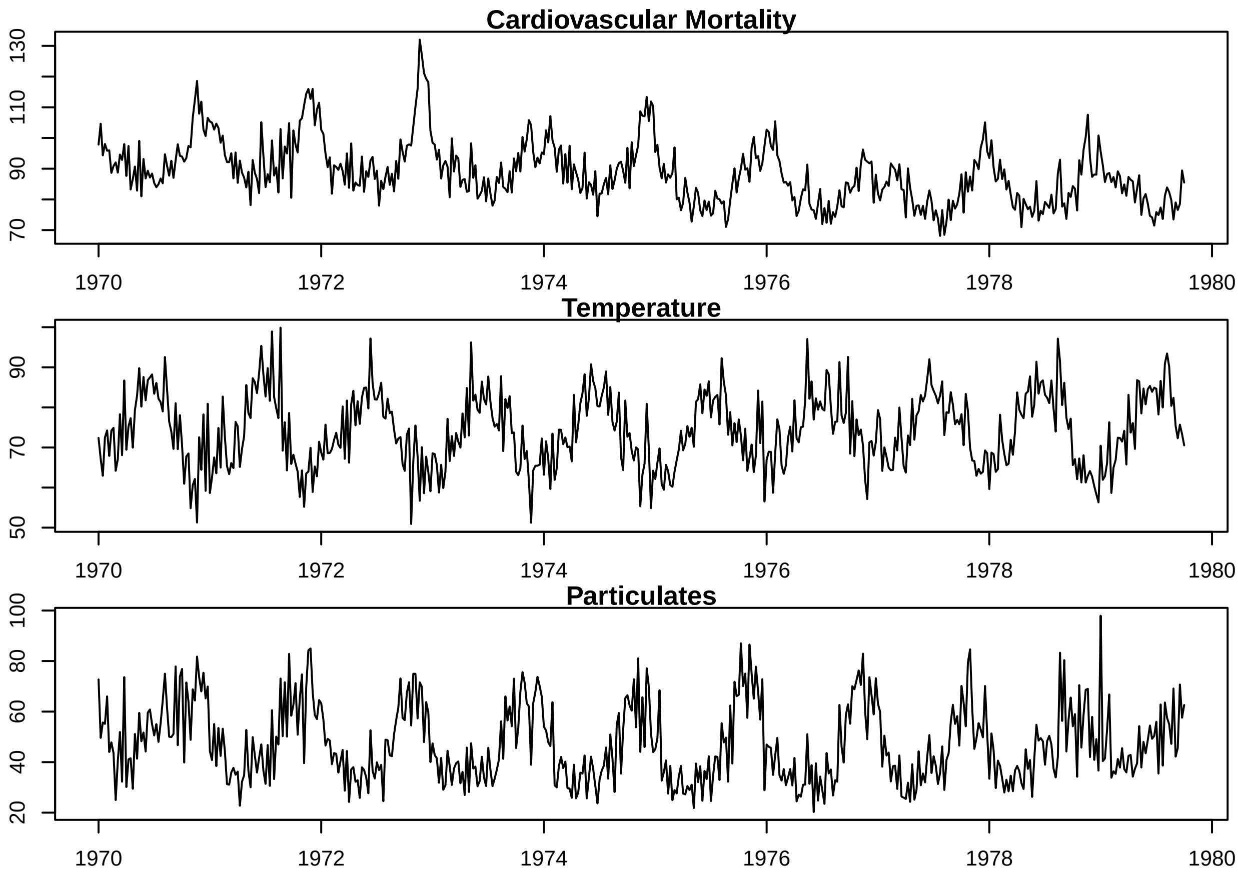



```r
# same as above
plot(reg$residuals)
```



1.3 Example 2.2

```
par(mfrow=c(3,1))
par(mar=c(2,2,1,1))
plot(cmort, main="Cardiovascular Mortality", xlab="", ylab="")
plot(tempr, main="Temperature", xlab="", ylab="")
plot(part, main="Particulates", xlab="", ylab="")
```



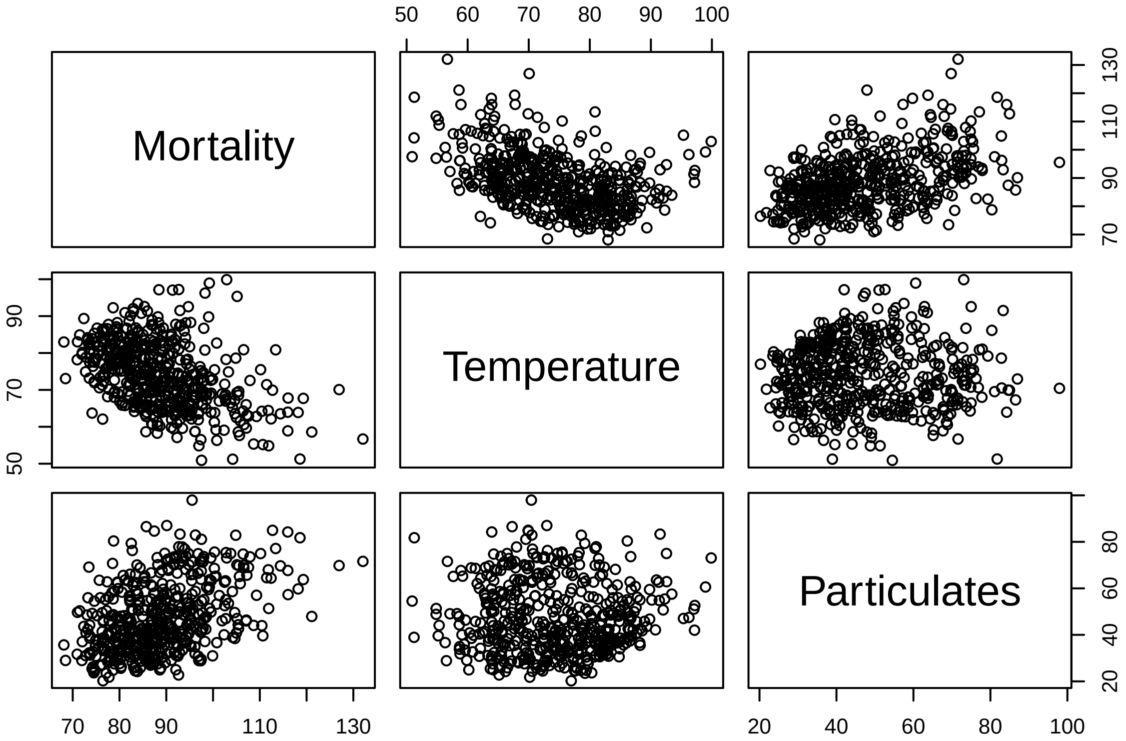
The model is $M_t = \beta_1 + \beta_2 t + \beta_3(T_t - T.) + \beta_4(T_t - T.)^2 + \beta_5 P_t + w_t$

This model contains curvilinear temperature and pollution, also, the temperature is mean-adjusted.

```
mean(tempr) # T.
```

```
## [1] 74.26041
```

```
pairs(cbind(Mortality=cmort, Temperature=tempr, Particulates=part))
```




```
temp = tempr - mean(tempr) # center temperature
temp2 = temp^2
trend = time(cmort)
# time, notice the scale of this term will affect the intercept
# but leave other coefficients unchanged
# Note that the scale here is not adjusted properly

fit = lm(cmort~ trend +temp + temp2+ part, na.action=NULL)
summary(fit) # regression result
```

```
##
## Call:
## lm(formula = cmort ~ trend + temp + temp2 + part, na.action = NULL)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.0760  -4.2153  -0.4878   3.7435  29.2448
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.831e+03  1.996e+02   14.19  < 2e-16 ***
## trend       -1.396e+00  1.010e-01  -13.82  < 2e-16 ***
```

```
## temp         -4.725e-01  3.162e-02  -14.94  < 2e-16 ***
## temp2         2.259e-02  2.827e-03    7.99 9.26e-15 ***
## part          2.554e-01  1.886e-02   13.54  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.385 on 503 degrees of freedom
## Multiple R-squared:  0.5954, Adjusted R-squared:  0.5922
## F-statistic:   185 on 4 and 503 DF,  p-value: < 2.2e-16
```

ANOVA: These tests have been used in the past in a stepwise manner, where variables are added or deleted when the values from the F- test either exceed or fail to exceed some predetermined levels. The procedure, called stepwise multiple regression, is useful in arriving at a set of useful variables.

```
summary(aov(fit)) # ANOVA table (compare to next line)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## trend         1  10667   10667  261.62 <2e-16 ***
## temp          1   8607    8607  211.09 <2e-16 ***
## temp2         1   3429    3429   84.09 <2e-16 ***
## part          1   7476    7476  183.36 <2e-16 ***
## Residuals   503  20508      41
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(lm(cmort~ cbind(trend, temp, temp2, part)))) # Table 2.1
```

```
##                                  Df Sum Sq Mean Sq F value Pr(>F)
## cbind(trend, temp, temp2, part)   4  30178    7545     185 <2e-16 ***
## Residuals                       503  20508      41
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
trend = time(cmort)
head(trend)
```

```
## [1] 1970.000 1970.019 1970.038 1970.058 1970.077 1970.096
```

```
tail(trend)
```

```
## [1] 1979.654 1979.673 1979.692 1979.712 1979.731 1979.750
```

```
# use different scale, only estimated intercept will change
# you should use this one

# create a time series sequence from 1 to 508 with a frequency of 52
# adjust the scale of trend to match the yearly frequency
trend <- ts(1:length(cmort), start = c(1970, 1), frequency = 52)
# frequency =52 means yearly data
# type time(cmort) and you'll see why we adjust this way
head(trend)
```

```
## [1] 1 2 3 4 5 6
```

```
tail(trend)
```

```
## [1] 503 504 505 506 507 508
```

```
# Notice that you can also use
# trend <- time(cmort)-mean(time(cmort))
```

```
fit = lm(cmort~ trend +temp + temp2+ part, na.action=NULL)
summary(fit)
```

```
##
## Call:
## lm(formula = cmort ~ trend + temp + temp2 + part, na.action = NULL)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.0760  -4.2153  -0.4878   3.7435  29.2448
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 81.592238   1.102148   74.03  < 2e-16 ***
## trend       -0.026844   0.001942  -13.82  < 2e-16 ***
## temp        -0.472469   0.031622  -14.94  < 2e-16 ***
## temp2        0.022588   0.002827    7.99 9.26e-15 ***
```

```
## part          0.255350   0.018857   13.54  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.385 on 503 degrees of freedom
## Multiple R-squared:  0.5954, Adjusted R-squared:  0.5922
## F-statistic:   185 on 4 and 503 DF,  p-value: < 2.2e-16
```

$$\hat{M}_t = 81.59 - .027_{(.002)}t - .473_{(.032)}(T_t - 74.6) + .023_{(.003)}(T_t - 74.6)^2 + .255_{(.019)}P_t$$

```
num = length(cmort) #sample size
AIC(fit)/num - log(2*pi) # AIC
```

```
## [1] 4.721732
```

```
AIC(fit, k=log(num))/num - log(2*pi) # BIC
```

```
## [1] 4.771699
```

```
(AICc = log(sum(resid(fit)^2)/num) + (num+5)/(num-5-2)) # AICc
```

```
## [1] 4.722062
```

1.4 Example 2.3

Performing lagged regression in R is a little difficult because the series must be aligned prior to running the regression. The easiest way to do this is to create a data frame that we call fish using ts.intersect, which aligns the lagged series.

```
fish <- ts.intersect(rec, soiL6=lag(soi, -6), dframe = TRUE)
summary(lm(rec ~ soiL6, data = fish, na.action = NULL))
```

```
##
## Call:
## lm(formula = rec ~ soiL6, data = fish, na.action = NULL)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -65.187 -18.234   0.354  16.580  55.790
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   65.790      1.088   60.47   <2e-16 ***
## soiL6        -44.283      2.781  -15.92   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.5 on 445 degrees of freedom
## Multiple R-squared:  0.3629, Adjusted R-squared:  0.3615
## F-statistic: 253.5 on 1 and 445 DF,  p-value: < 2.2e-16
```

$\hat{R}_t = 65.79 - 44.28_{(2.78)} S_{t-6}$

Notice the use of `ts.intersect()`

```
head(ts.intersect(rec, soi),12)
```

```
##          rec    soi
##  [1,] 68.63  0.377
##  [2,] 68.63  0.246
##  [3,] 68.63  0.311
##  [4,] 68.63  0.104
##  [5,] 68.63 -0.016
##  [6,] 68.63  0.235
##  [7,] 59.16  0.137
##  [8,] 48.70  0.191
##  [9,] 47.54 -0.016
## [10,] 50.91  0.290
## [11,] 44.70  0.038
## [12,] 42.85 -0.016
```

```
head(ts.intersect(rec, soiL6=lag(soi, -6)))
```

```
##         rec  soiL6
## [1,] 59.16  0.377
## [2,] 48.70  0.246
## [3,] 47.54  0.311
## [4,] 50.91  0.104
## [5,] 44.70 -0.016
## [6,] 42.85  0.235
```

```
tail(ts.intersect(rec, soi), 12)
```

```
##              rec        soi
## [442,] 79.20000  0.1260000
## [443,] 87.83000  0.3330000
## [444,] 88.20000  0.5190000
## [445,] 94.83000  0.3990000
## [446,] 98.66001  0.5190000
## [447,] 94.83999  0.4320000
## [448,] 83.06000  0.3550000
## [449,] 61.42000 -0.1260000
## [450,] 47.47000 -0.5080001
## [451,] 31.81000 -0.3880000
## [452,] 22.95000  0.3880000
## [453,] 17.87000  0.0710000
```

```
tail(ts.intersect(rec, soiL6=lag(soi, -6)))
```

```
##           rec soiL6
## [442,] 83.06 0.126
## [443,] 61.42 0.333
## [444,] 47.47 0.519
## [445,] 31.81 0.399
## [446,] 22.95 0.519
## [447,] 17.87 0.432
```

1.5 Problem 2.2

For the mortality data examined in Example 2.2:

(a) Add another component to the regression in (2.25) that accounts for the particulate count four weeks prior; that is, add P_{t-4} to the regression in (2.25). State your conclusion.

```
temp = tempr - mean(tempr) # center temperature
temp2 = temp^2
# create a time series sequence from 1 to 508 with a frequency of 52
# adjust the scale of trend to match the yearly frequency
trend <- ts(1:length(cmort), start = c(1970, 1), frequency = 52)
# frequency =52 means yearly data
```

```r
# type time(cmort) and you'll see why we adjust this way

Mortal <- ts.intersect(cmort, trend, temp, temp2, part, partL4= lag(part, -4), dframe = TR
mod.fit=lm(cmort ~ trend + temp + temp2 + part+ partL4 , data=Mortal, na.action = NULL)

summary(mod.fit)
```

```
##
## Call:
## lm(formula = cmort ~ trend + temp + temp2 + part + partL4, data = Mortal,
##     na.action = NULL)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.228  -4.314  -0.614   3.713  27.800
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 79.239918   1.224693  64.702  < 2e-16 ***
## trend       -0.026641   0.001935 -13.765  < 2e-16 ***
## temp        -0.405808   0.035279 -11.503  < 2e-16 ***
## temp2        0.021547   0.002803   7.688 8.02e-14 ***
## part         0.202882   0.022658   8.954  < 2e-16 ***
## partL4       0.103037   0.024846   4.147 3.96e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.287 on 498 degrees of freedom
## Multiple R-squared:  0.608,  Adjusted R-squared:  0.6041
## F-statistic: 154.5 on 5 and 498 DF,  p-value: < 2.2e-16
```

```r
AIC(mod.fit)/num - log(2*pi) # AIC
```

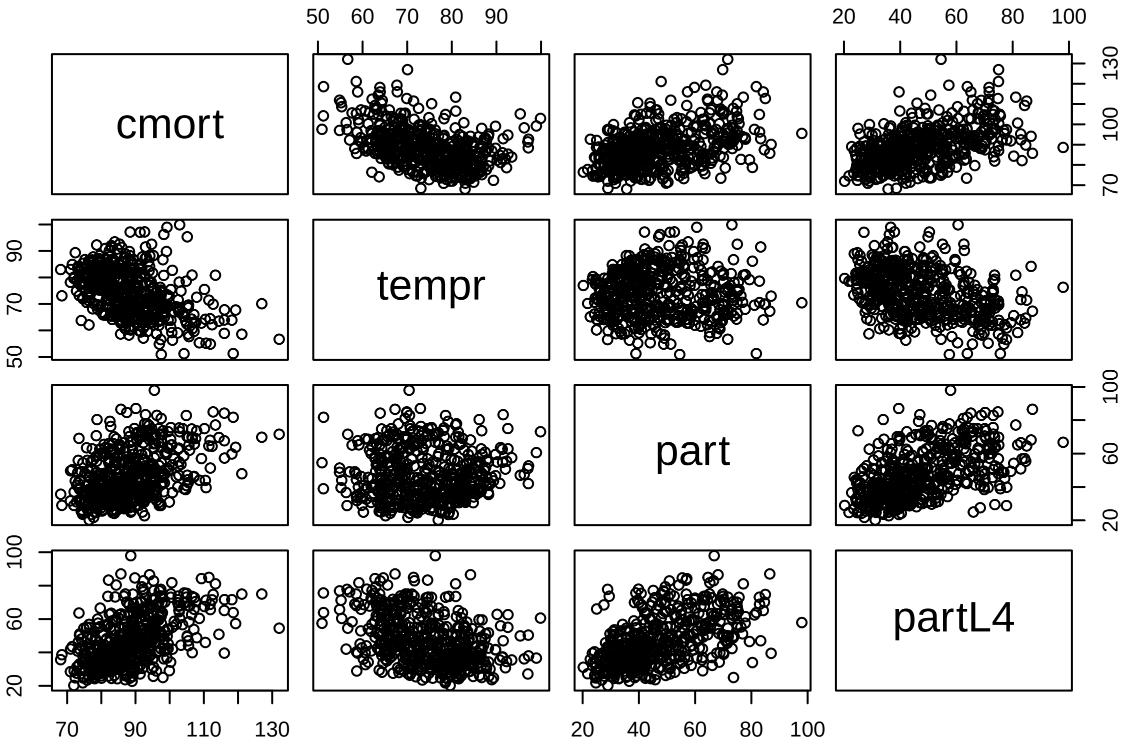
```
## [1] 4.641492
```

Adding P_{t-4} to the model increased the R^2 (which is to be suspected when adding parameters to a model), but also decreased the AIC. The F-statistic show that the model including P_{t-4} makes a better model.

(b) Draw a scatterplot matrix of M_t, T_t, P_t and P_{t-4} and then calculate the pairwise correlations between the series. Compare the relationship between M_t and P_t versus M_t and P_{t-4}.

```
my_data <- ts.intersect(cmort, tempr, part, partL4= lag(part, -4), dframe = TRUE)

pairs(my_data)
```




```
cor(my_data)
```

```
##              cmort       tempr        part      partL4
## cmort    1.0000000  -0.4369648   0.4422896   0.5209993
## tempr   -0.4369648   1.0000000  -0.0148241  -0.3990848
## part     0.4422896  -0.0148241   1.0000000   0.5340505
## partL4   0.5209993  -0.3990848   0.5340505   1.0000000
```

The correlation between mortality and the particulate count four weeks prior (0.5209993) is slightly stronger than that of the particulate count (0.4422896). This further strengthens the fact that adding the particulate count four weeks prior was a good idea.

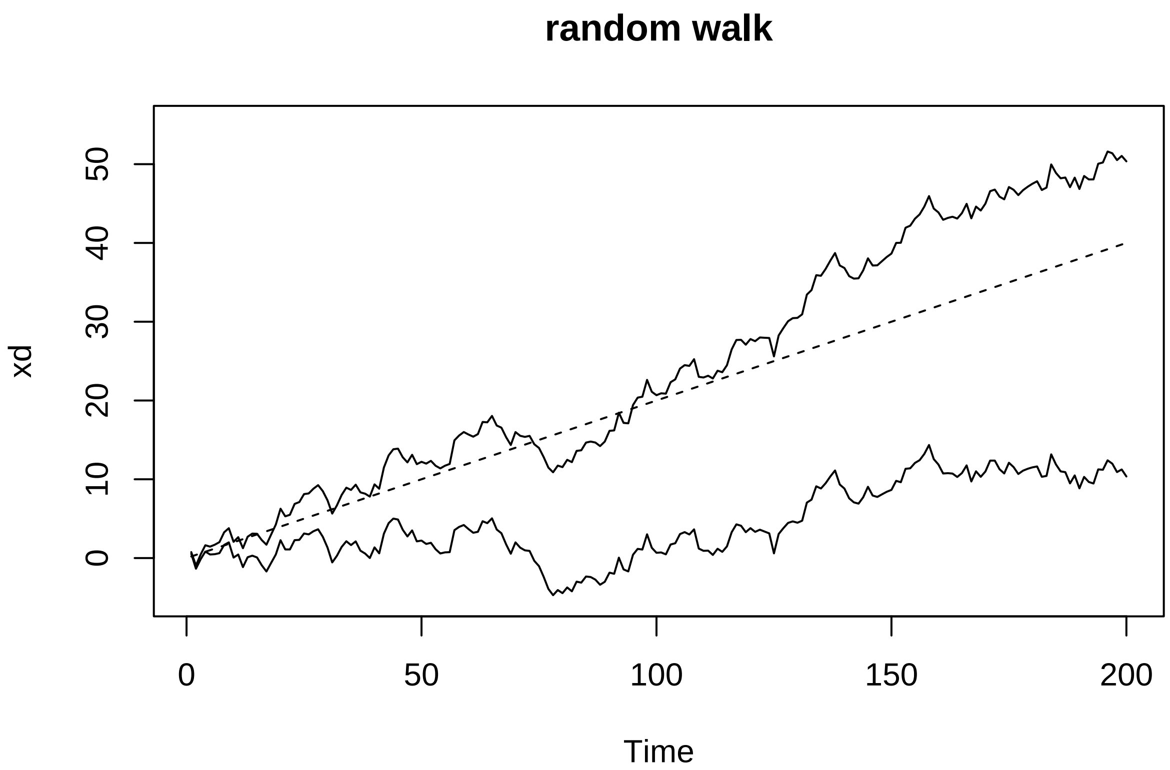
1.6 Example 1.11

Random Walk with Drift

Note that we've seen this before.

$$x_t = \delta t + \sum_{j=1}^{t} w_j \tag{1.4}$$

```
set.seed(154) # so you can reproduce the results
w = rnorm(200,0,1); x = cumsum(w) # two commands in one line
wd = w +.2; xd = cumsum(wd)
plot.ts(xd, ylim=c(-5,55), main="random walk")
lines(x); lines(.2*(1:200), lty="dashed")
```

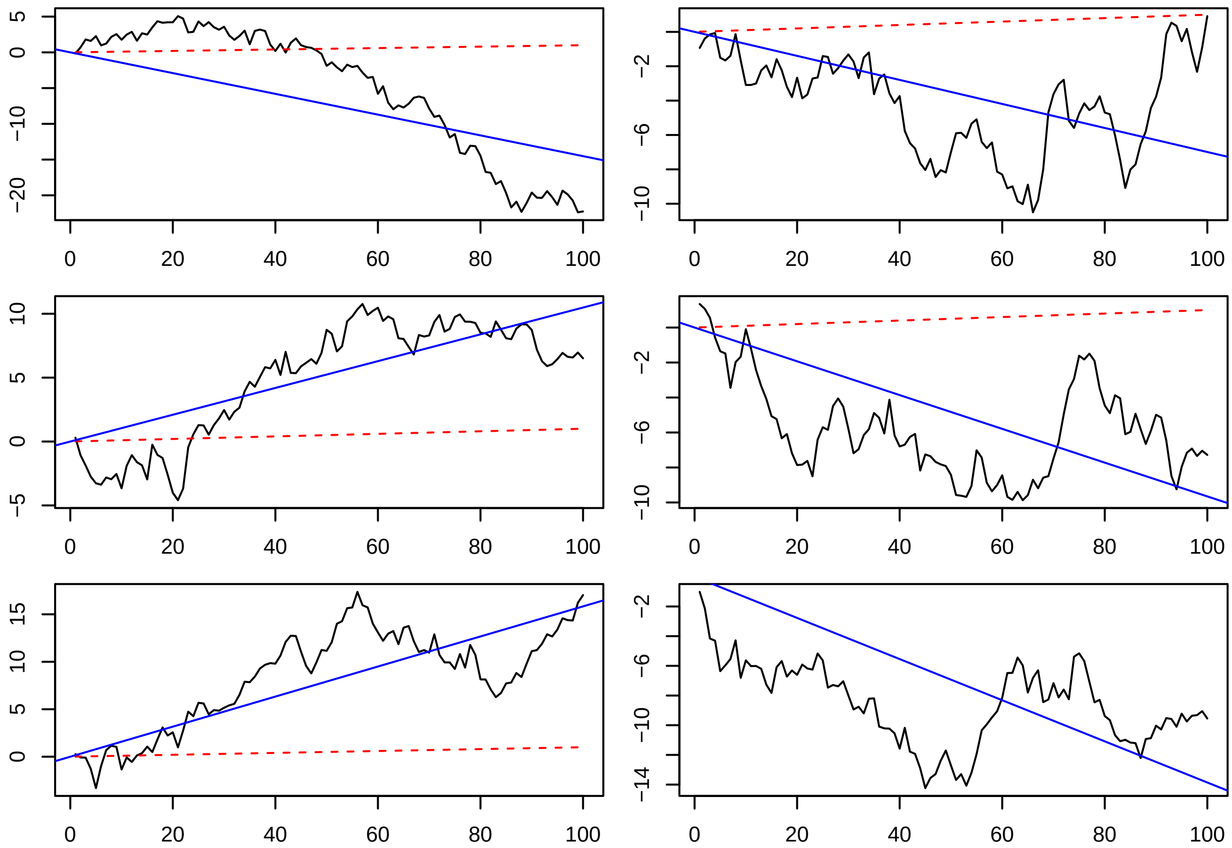


1.7 Problem 2.3

Repeat the following exercise six times and then discuss the results. Generate a random walk with drift, (1.4), of length n = 100 with $\delta = .01$ and $\sigma_w = 1$. Call the data x_t for t = 1,...,100. Fit the regression $x_t = \beta_t + w_t$ using least squares. Plot the data, the mean function (i.e., $\mu_t = .01t$) and the fitted line, $\hat{x}_t = \hat{\beta}t$, on the same graph. Discuss your results.

```
par(mar=c(2,2,1,1))
par(mfcol=c(3,2)) #set up graphics
for(i in 1:6){
  x= ts(cumsum(rnorm(100, .01, 1))) # the data, delta=0.01
  reg = lm(x ~ 0 + time(x), na.action = NULL) # the regression
```

```
  plot(x) # plot data
  lines(.01*time(x), col="red", lty ="dashed")
  # plot mean function mu_t =.01t
  abline(reg, col="blue")
  # plot regression line hat x_t = \hat beta t
}
```



It appears that the regression line fits the random walk data better than the mean function. The regression line will always try to fit the data best regardless of the drift. The mean function does not care what the data looks like and therefore, does not fit the data best. In some cases, it may be hard to tell which line best fits the data, as they could be very similar.

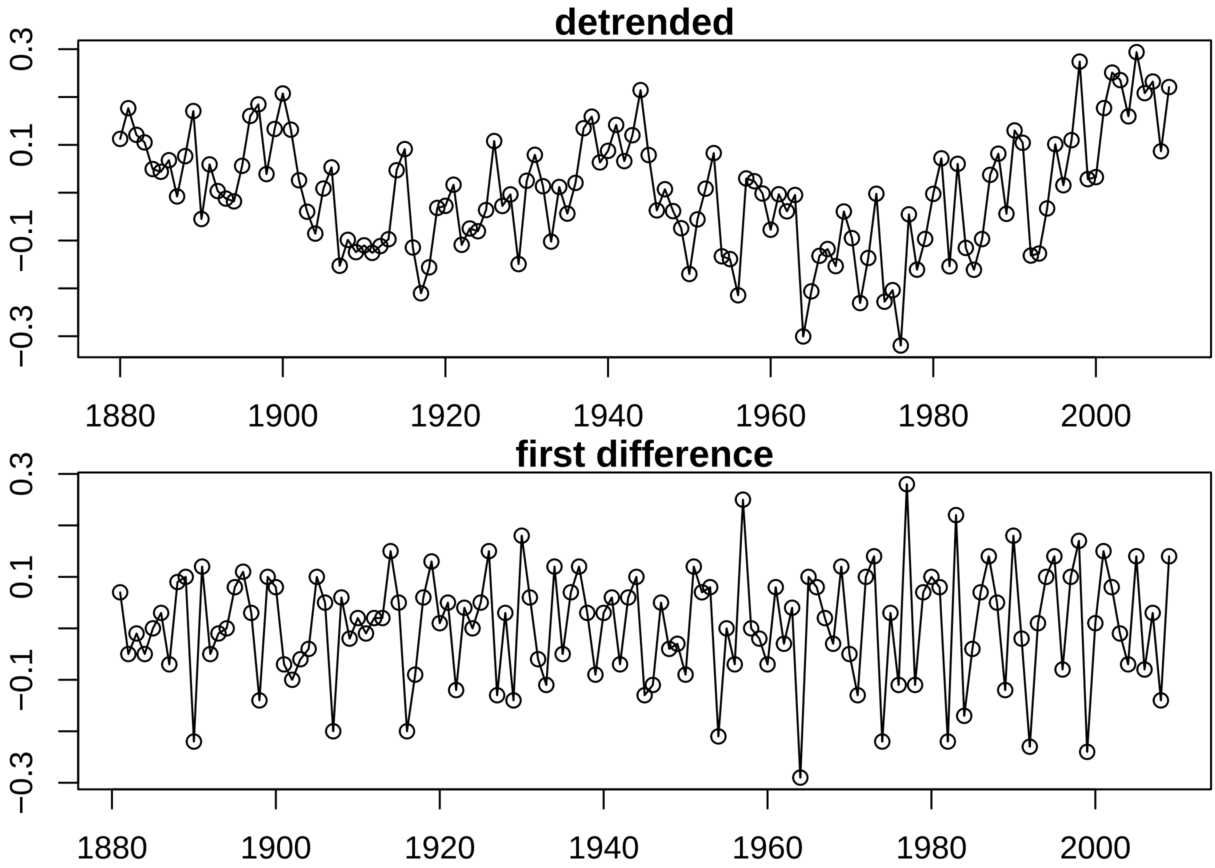
2 Section 2.3

2.1 Global Temperature Example

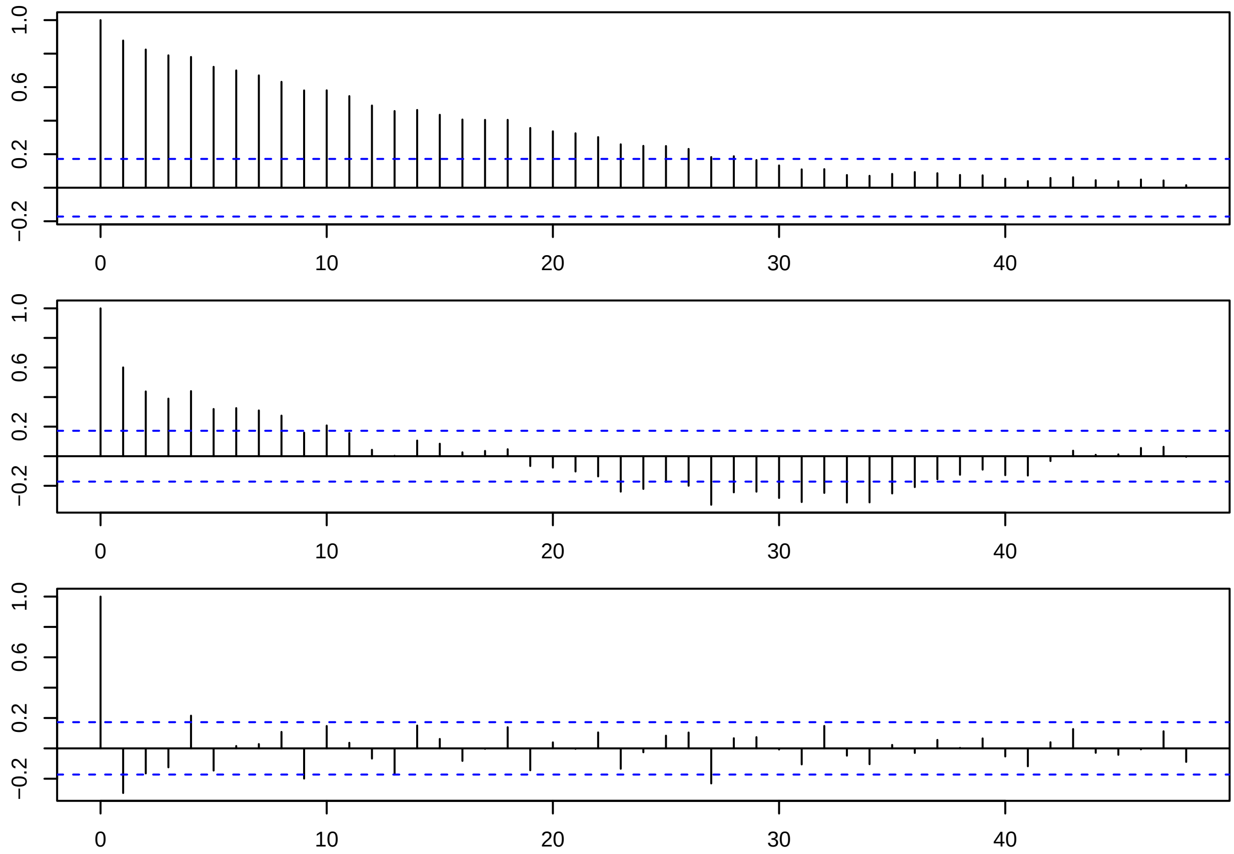
```
# Detrending global temperature

fit = lm(gtemp~time(gtemp), na.action=NULL) # regress gtemp on time
par(mfrow=c(2,1))
par(mar=c(2,2,1,1))
```

```
plot(resid(fit), type="o", main="detrended")
plot(diff(gtemp), type="o", main="first difference")
```




```
par(mfrow=c(3,1)) # plot ACFs
par(mar=c(2,2,1,1))
acf(gtemp, 48, main="gtemp")
acf(resid(fit), 48, main="detrended")
acf(diff(gtemp), 48, main="first difference")
```




```
mean(diff(gtemp)) # =0.00659 (drift)
```

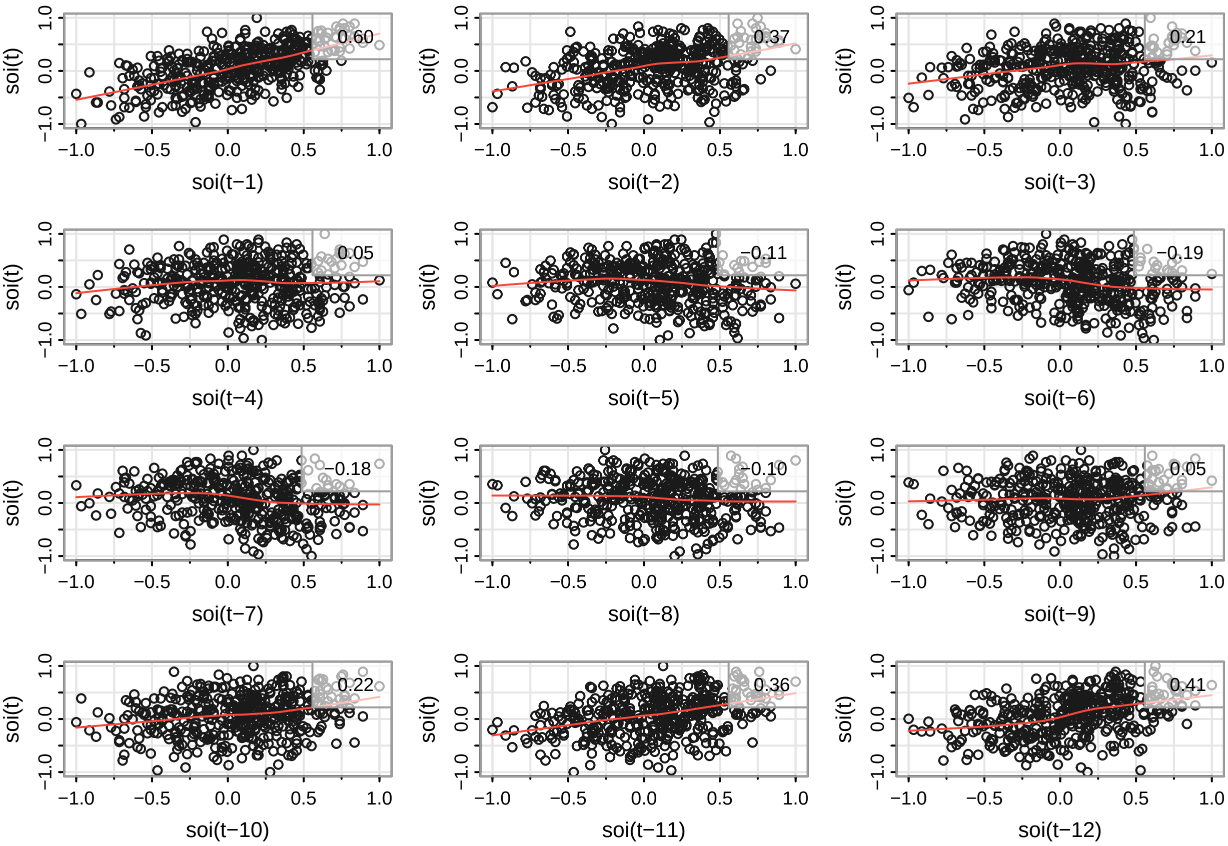
```
## [1] 0.006589147
```

```
sd(diff(gtemp))/sqrt(length(diff(gtemp))) # =0.00966 (SE)
```

```
## [1] 0.009658972
```

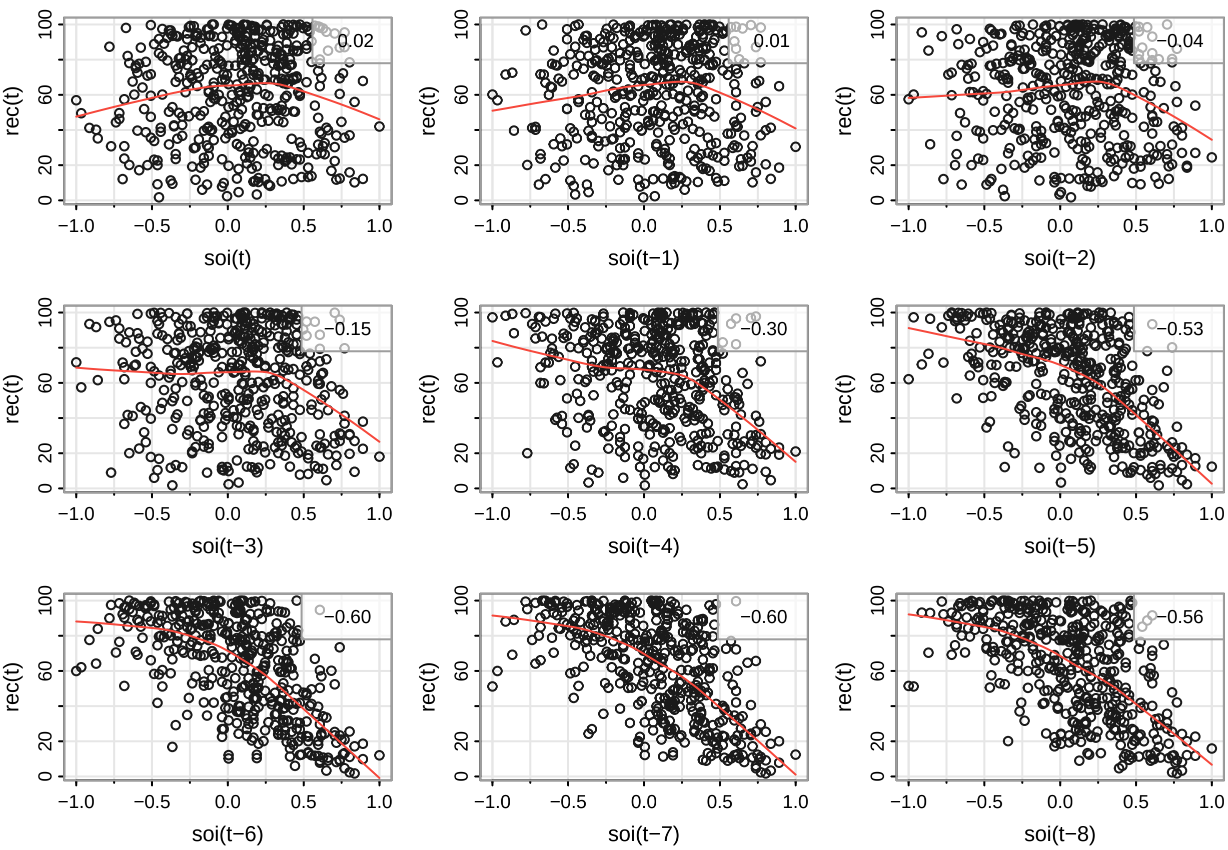
2.2 Example 2.7

```
# scatter plot
lag1.plot(soi, 12)
```



The values in the upper right corner are the sample autocorrelations and the lines are a lowess fit.

```
lag2.plot(soi, rec, 8)
```

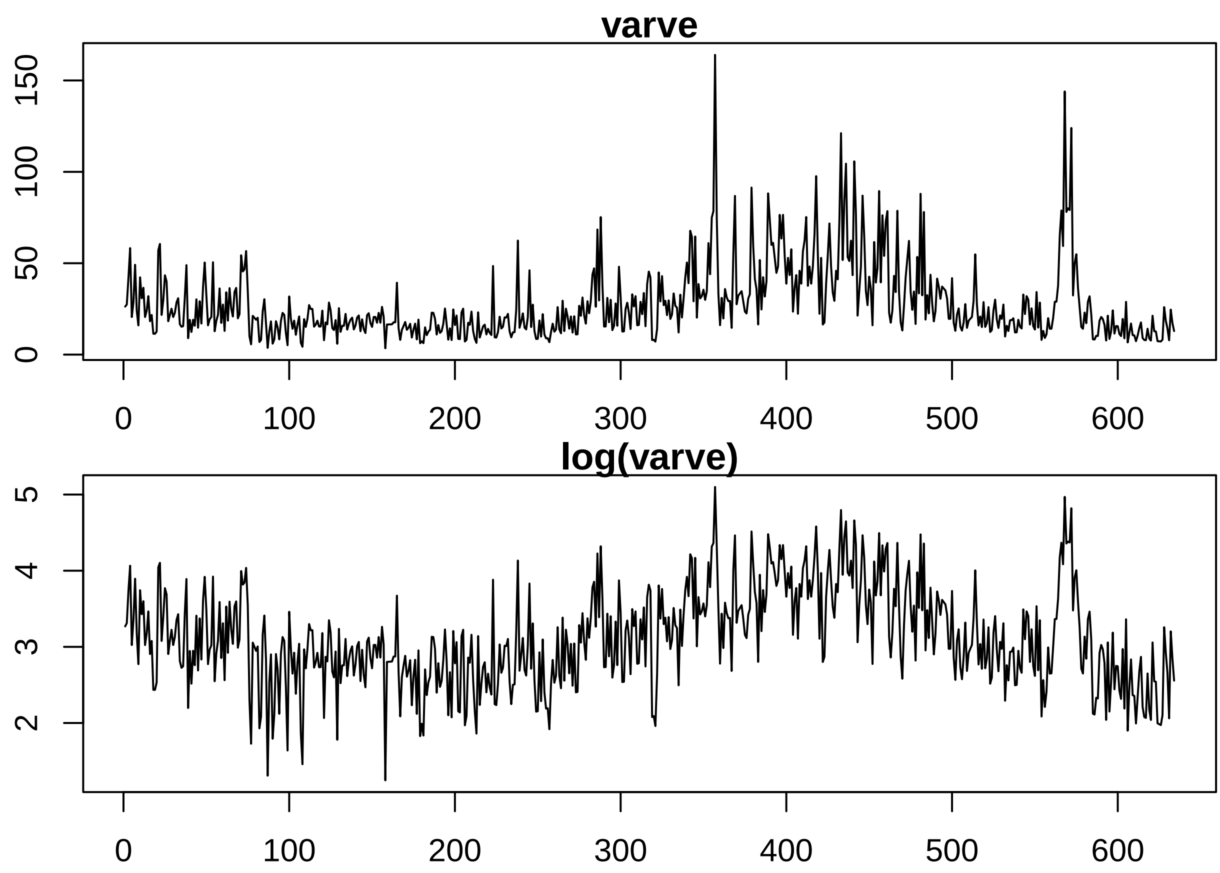


The values in the upper right corner are the sample cross-correlations and the lines are a lowess fit.

2.3 Problem 2.8

The glacial varve record plotted in Figure 2.6 exhibits some nonstationarity that can be improved by transforming to logarithms and some additional nonstationarity that can be corrected by differencing the logarithms.

```
# Figure 2.6
par(mar=c(2,2,1,1))
par(mfrow=c(2,1))
plot(varve, main="varve", ylab="")
plot(log(varve), main ="log(varve)", ylab="")
```



(a) Argue that the glacial varves series, say x_t, exhibits heteroscedasticity by computing the sample variance over the first half and the second half of the data. Argue that the transformation $y_t = log(x_t)$ stabilizes the variance over the series. Plot the histograms of x_t and y_t to see whether the approximation to normality is improved by transforming the data.

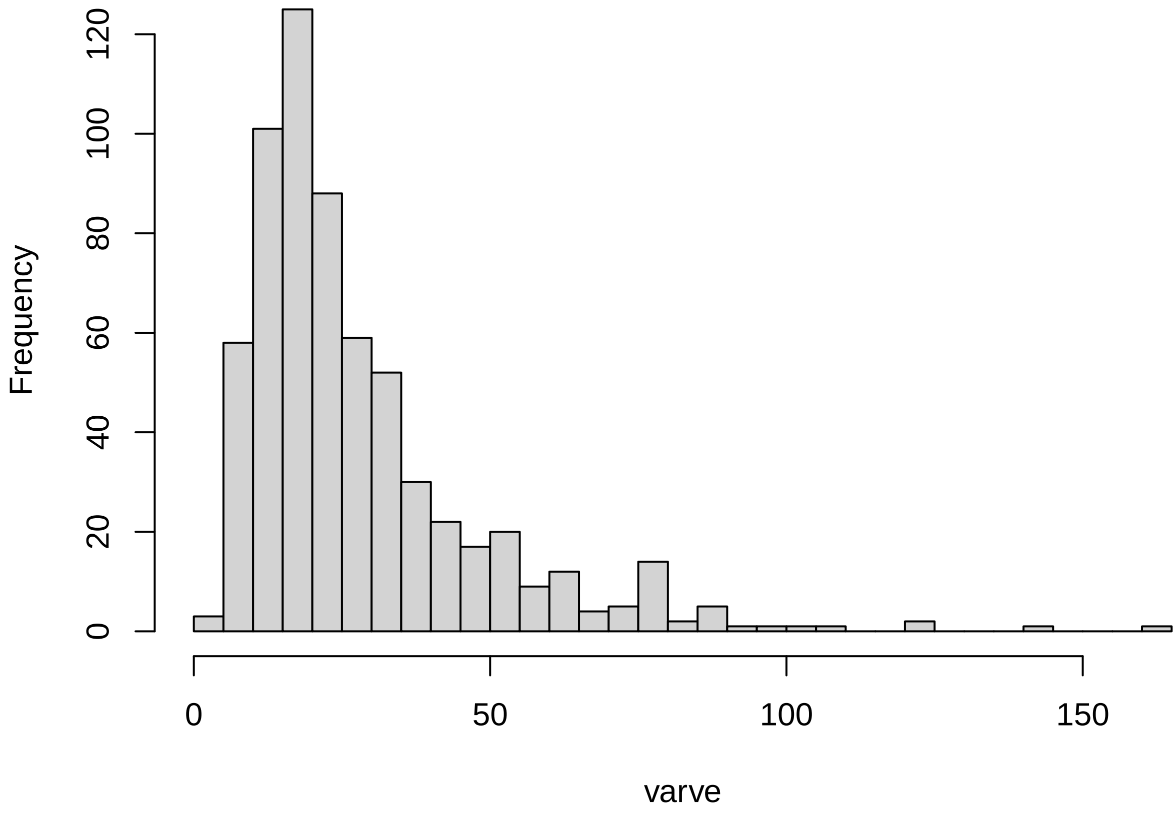
```
var(varve[1:length(varve)/2])
```

```
## [1] 132.501
```

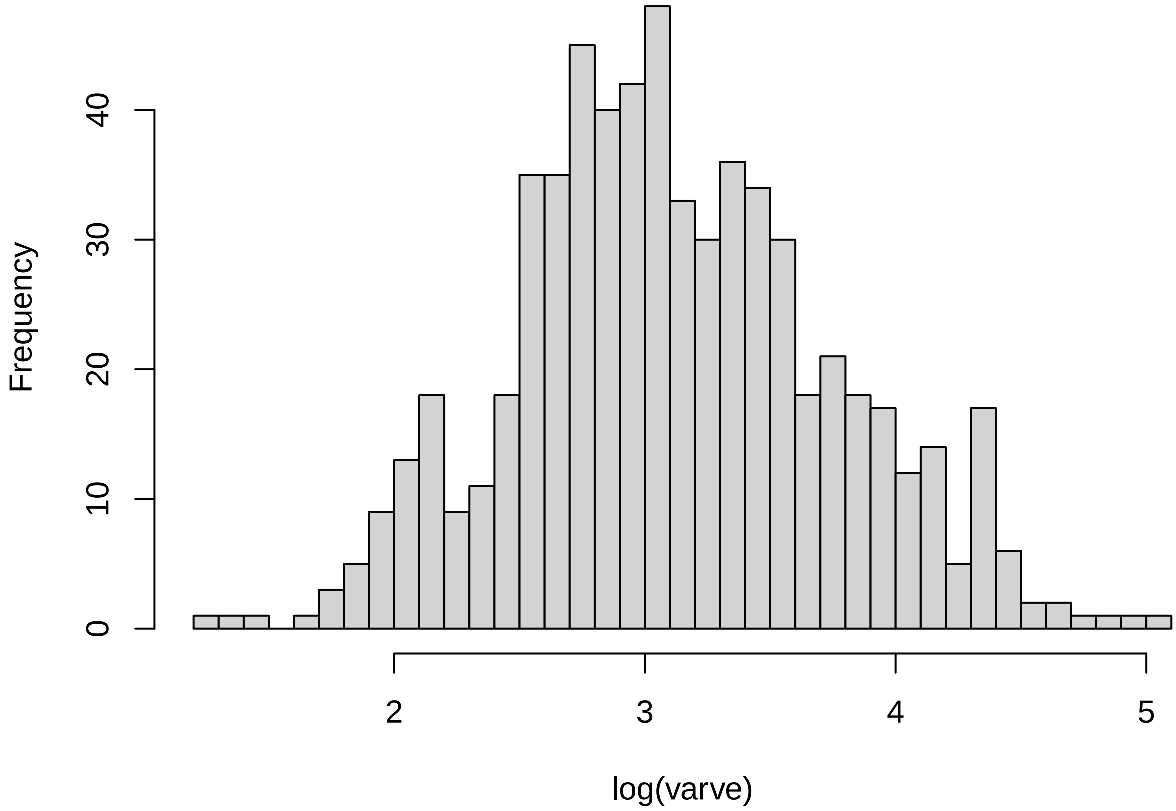
```
var(varve[length(varve)/2:length(varve)])
```

```
## [1] 89.62063
```

```
par(mai=c(0.9,0.9,0.1,0.1))
hist(varve, breaks=50, main=NULL)
```

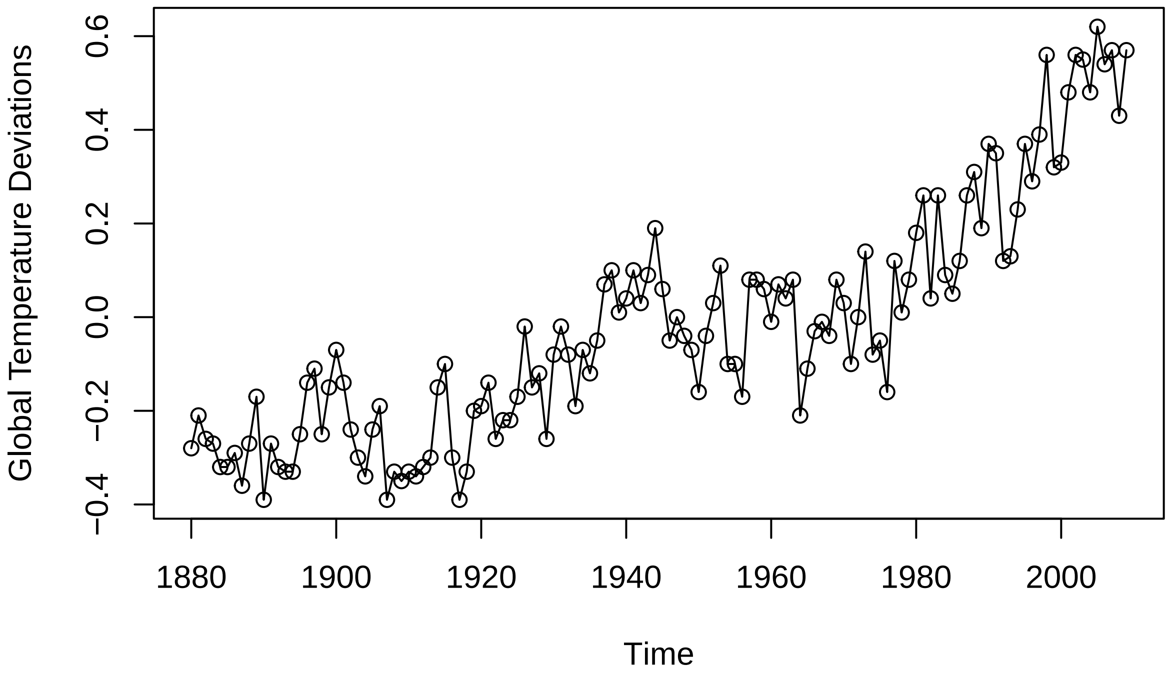



```
par(mai=c(0.9,0.9,0.1,0.1))
hist(log(varve), breaks = 50, main = NULL)
```



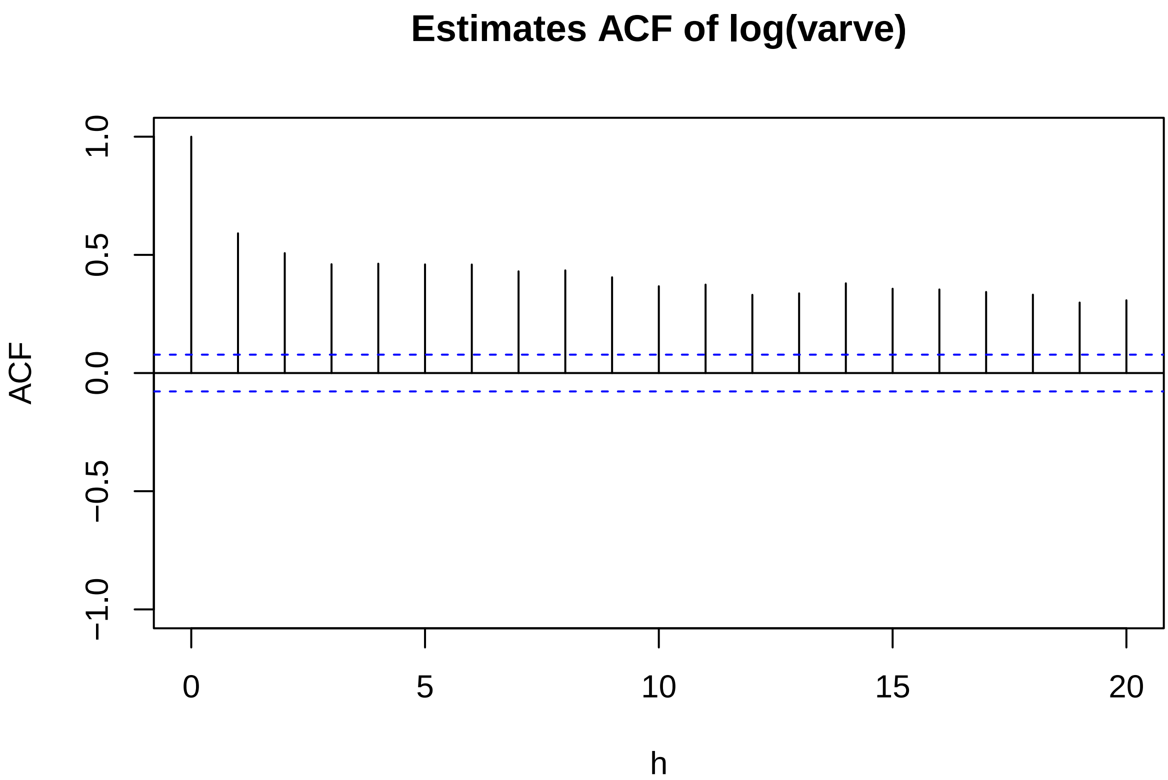
(b) Plot the series y_t. Do any time intervals, of the order 100 years, exist where one can observe behavior comparable to that observed in the global temperature records in Figure 1.2?

```
# Figure 1.2
plot(gtemp, type="o", ylab="Global Temperature Deviations")
```



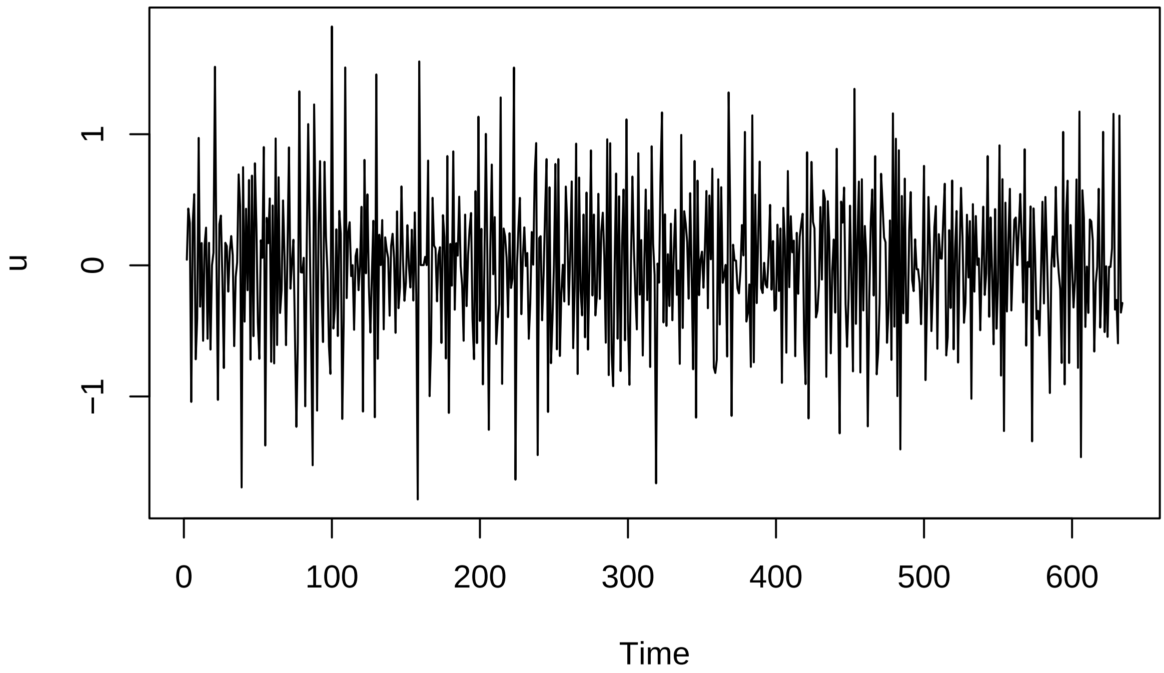
(c) Examine the sample ACF of y_t and comment

```
acf(log(varve), type = "correlation", lag.max=20, xlab="h", ylim=c(-1,1), main = "Estimate
```



(d) Compute the difference $u_t = y_t - y_{t-1}$, examine its time plot and sample ACF, and argue that differencing the logged varve data produces a reasonably stationary series. Can you think of a practical interpretation for u_t? Hint: For $|p|$ close to zero, $log(1+p) \approx p$; let $p = (y_t - y_{t-1})/y_{t-1}$.

```
u <- diff(log(varve), lag=1, difference=1)
plot(u)
```

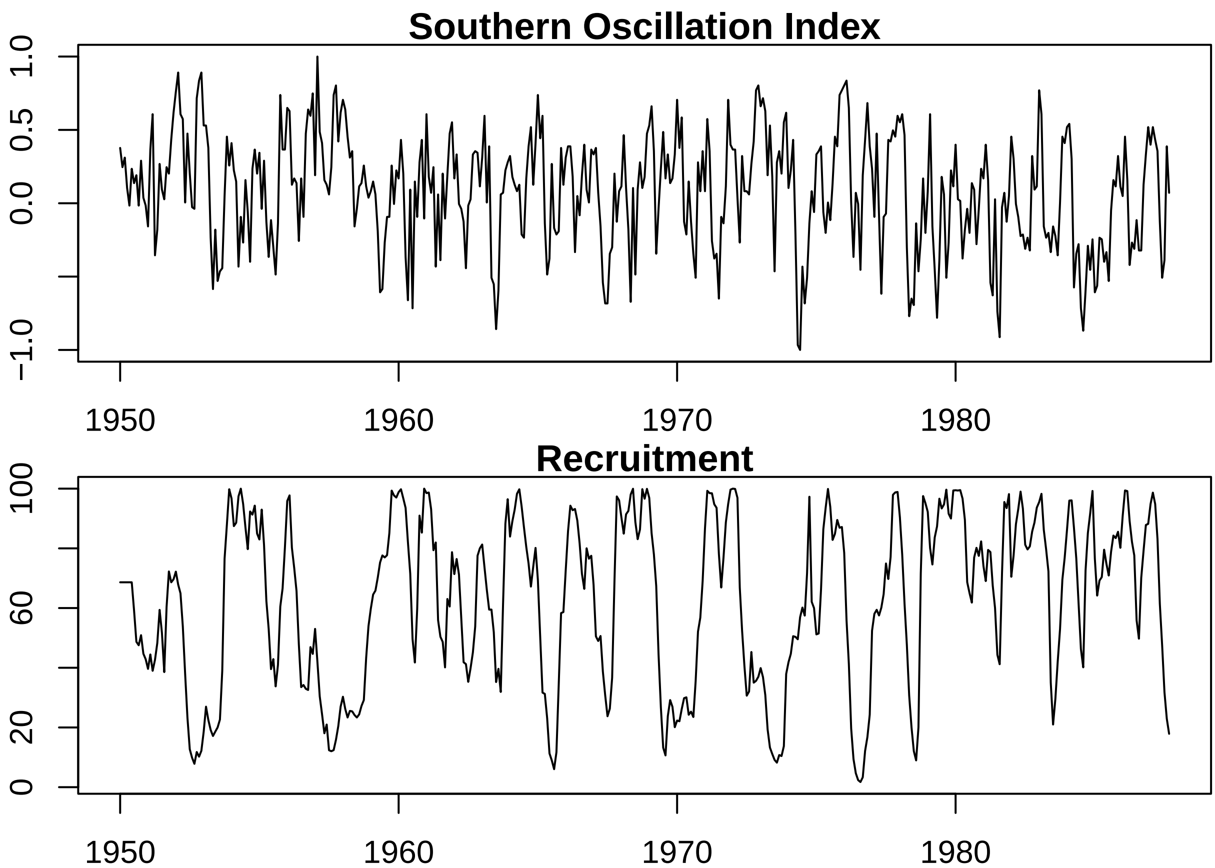


2.4 Problem 2.9

In this problem, we will explore the periodic nature of S_t, the SOI series displayed in Figure 1.5.

```
# Figure 1.5

par(mar=c(2,2,1,1))
par(mfrow=c(2,1)) #set up the graphics
plot(soi, ylab="", xlab="", main="Southern Oscillation Index")
plot(rec, ylab="", xlab="", main="Recruitment")
```



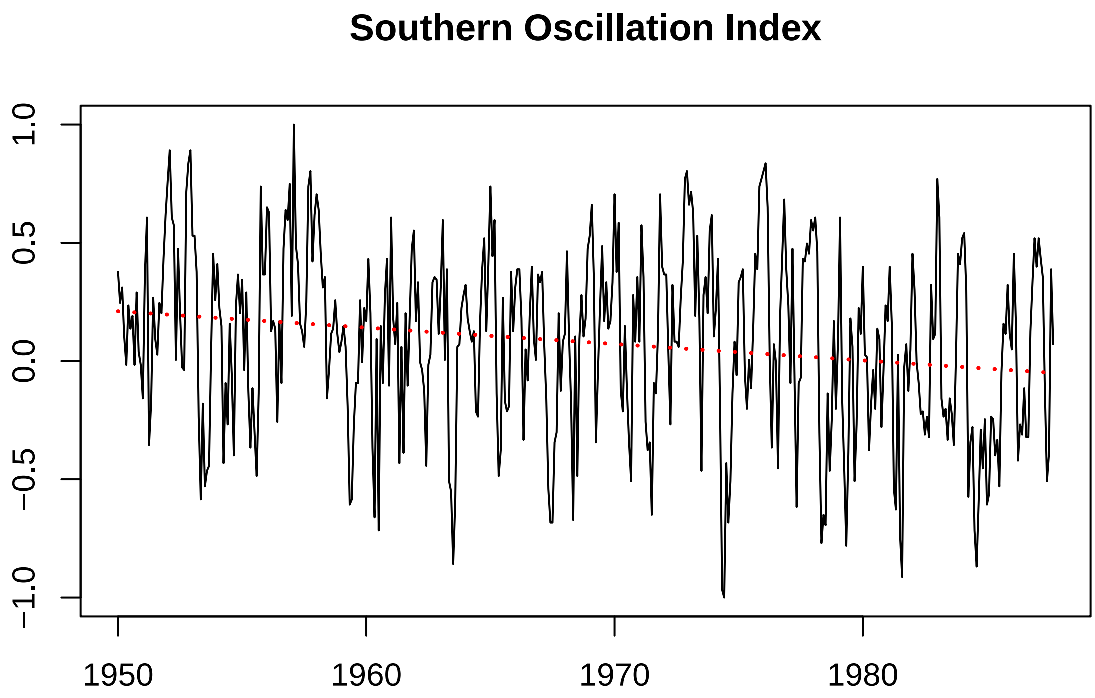
(a) Detrend the series by fitting a regression of St on time t. Is there a significant trend in the sea surface temperature? Comment.

```
trend <- time(seq(1 ,length(soi), length.out=length(soi)))
mod.fit <- lm(soi~ trend, na.action = NULL)
mod.fit
```

```
##
## Call:
## lm(formula = soi ~ trend, na.action = NULL)
##
## Coefficients:
## (Intercept)        trend
##   0.2109341   -0.0005766
```

The regression line is $Temperature = 0.2109 - 0.0006 * Time$

```
plot(soi, ylab="", xlab="", main="Southern Oscillation Index")
lines(mod.fit$fitted.values, col="red", lwd=2, lty="dotted")
```



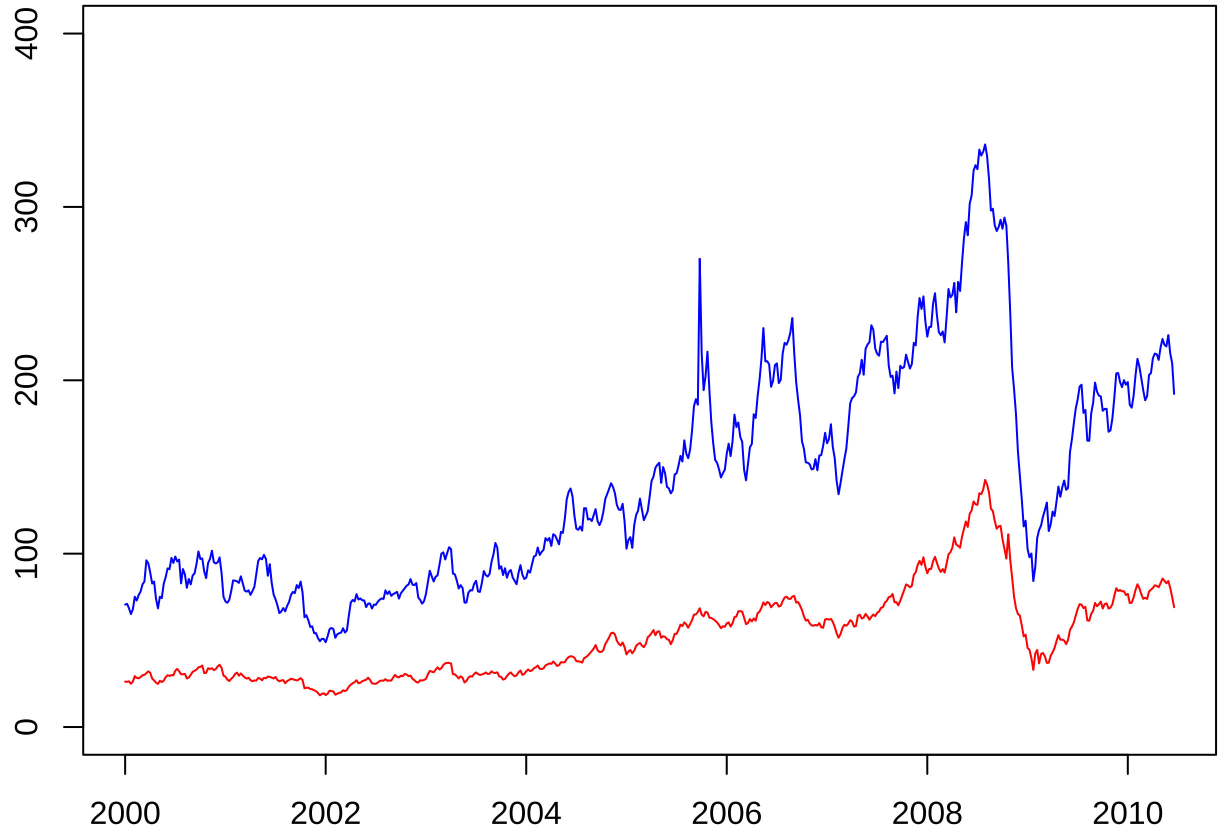
By plotting a regression line over the data and looking at the value of the slope, it is clear that there is a trend in the sea surface temperature.

2.5 Problem 2.11

Consider the two weekly time series `oil` and `gas`. The oil series is in dollars per barrel, while the gas series is in cents per gallon; see Appendix R for details.

(a) Plot the data on the same graph. Which of the simulated series displayed in §1.3 do these series most resemble? Do you believe the series are stationary (explain your answer)?

```
par(mar=(c(2,2,1,1)))
plot(oil, col="red", ylim=c(0,400))
lines(gas, col="blue")
```

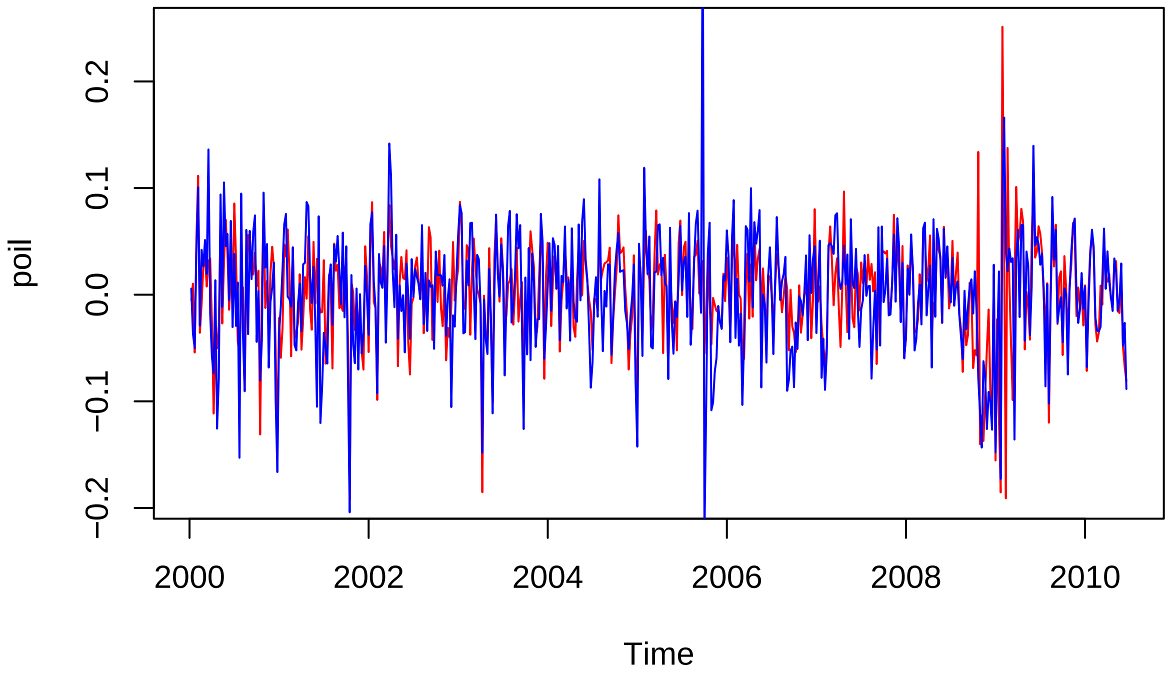


(b) In economics, it is often the percentage change in price (termed growth rate or return), rather than the absolute price change, that is important. Argue that a transformation of the form $y_t = \nabla log(x_t)$ might be applied to the data, where x_t is the oil or gas price series [see the hint in Problem 2.8(d)].

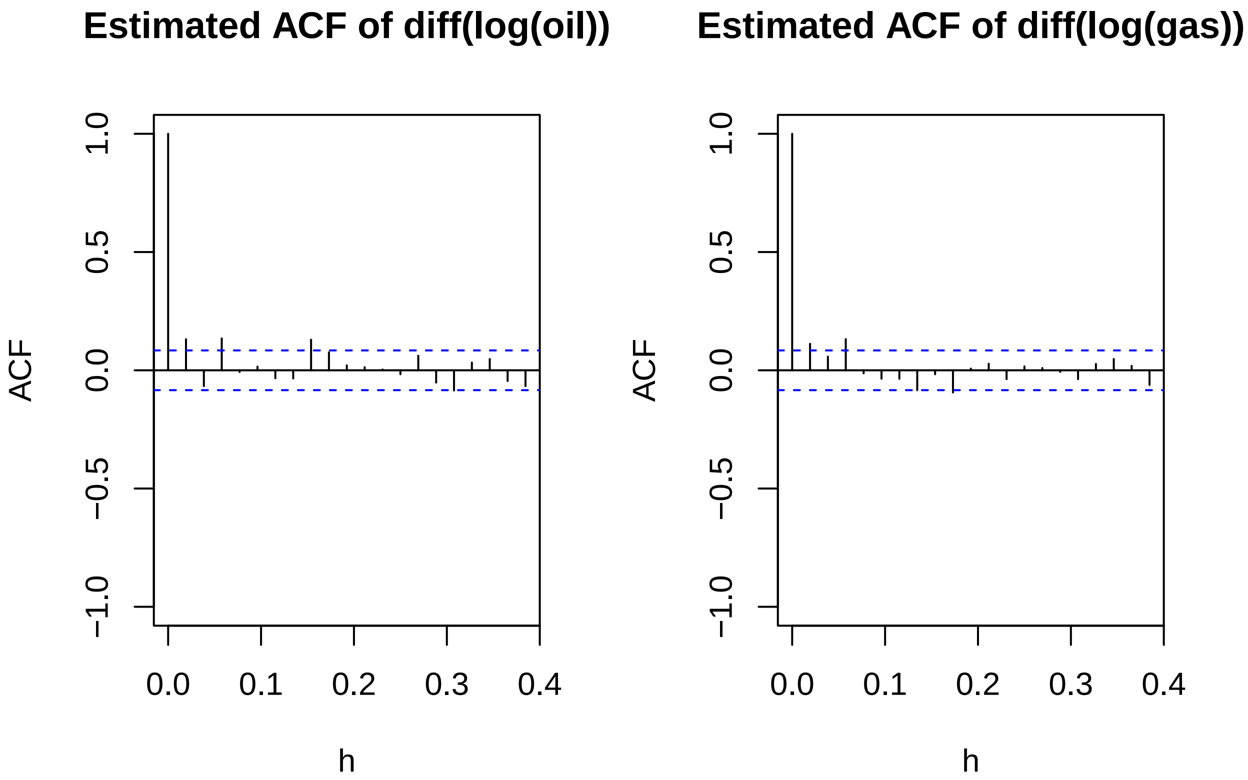
$$log(y_{t+1}) - log(y_t) = log(\frac{y_{t+1}}{y_t}) = log(1 + \frac{y_{t+1} - y_t}{y_t}) \approx \frac{y_{t+1} - y_t}{y_t} \tag{1}$$

(c) Transform the data as described in part (b), plot the data on the same graph, look at the sample ACFs of the transformed data, and comment. [Hint: poil = diff(log(oil)) and pgas = diff(log(gas)).]

```
poil <- diff(log(oil))
pgas <- diff(log(gas))
plot(poil, col="red")
lines(pgas, col="blue")
```

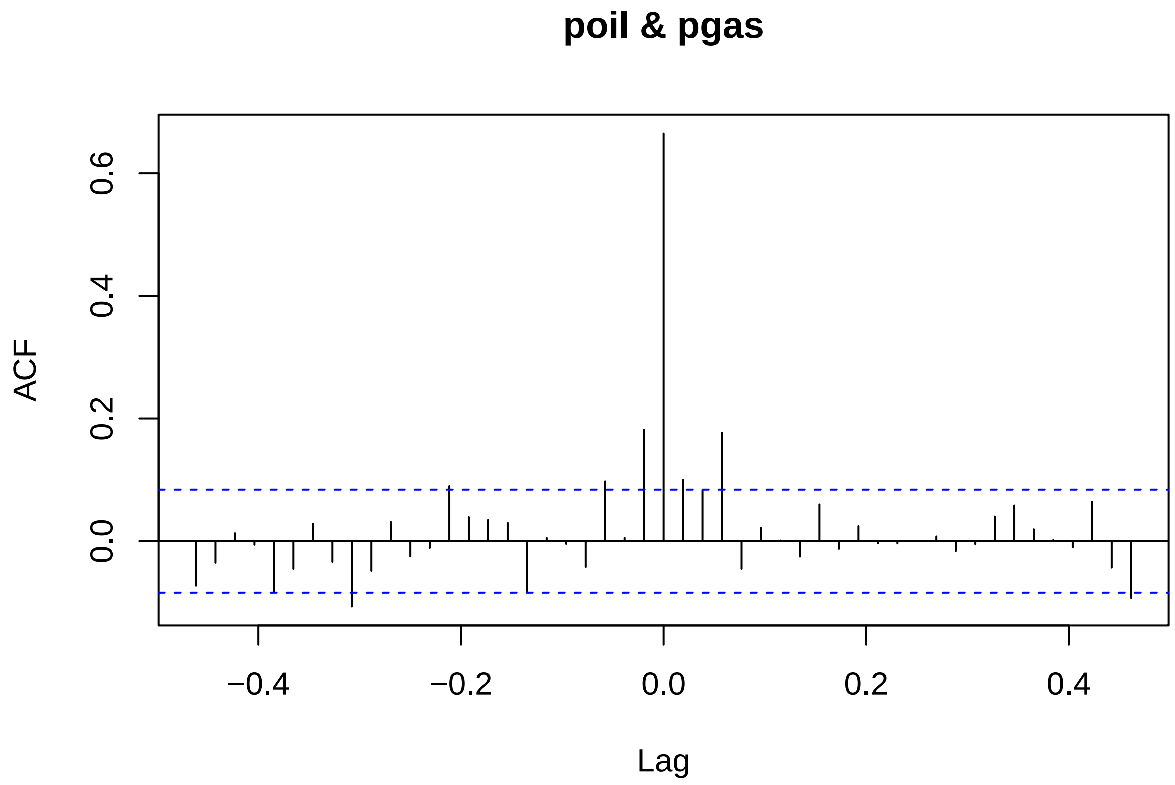



```
par(mfrow=c(1,2))
acf(poil, type="correlation", lag.max = 20, xlab="h",
    ylim=c(-1,1), main= "Estimated ACF of diff(log(oil))")
acf(pgas, type="correlation", lag.max = 20, xlab="h",
    ylim=c(-1,1), main= "Estimated ACF of diff(log(gas))")
```



(d) Plot the CCF of the transformed data and comment The small, but significant values when gas leads oil might be considered as feedback. [Hint: ccf(poil, pgas) will have poil leading for negative lag values.]

```
poil.pgas.ccf <- ccf(poil, pgas, type = "correlation")
```

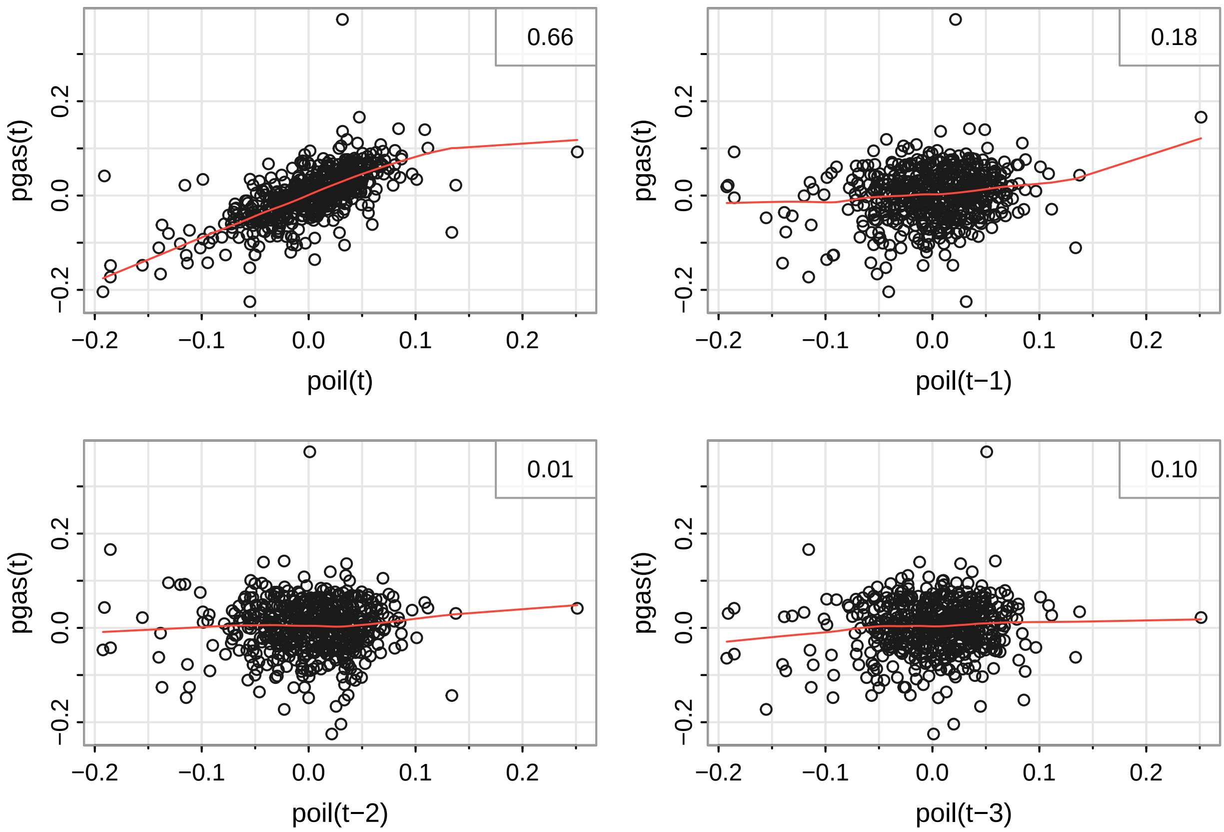



```
poil.pgas.ccf
```

```
##
## Autocorrelations of series 'X', by lag
##
## -0.4615 -0.4423 -0.4231 -0.4038 -0.3846 -0.3654 -0.3462 -0.3269 -0.3077 -0.2885
##  -0.073  -0.035   0.013  -0.006  -0.082  -0.045   0.028  -0.034  -0.107  -0.048
## -0.2692 -0.2500 -0.2308 -0.2115 -0.1923 -0.1731 -0.1538 -0.1346 -0.1154 -0.0962
##   0.031  -0.025  -0.011   0.090   0.039   0.035   0.030  -0.082   0.005  -0.004
## -0.0769 -0.0577 -0.0385 -0.0192  0.0000  0.0192  0.0385  0.0577  0.0769  0.0962
##  -0.042   0.097   0.005   0.182   0.665   0.100   0.082   0.177  -0.045   0.021
##  0.1154  0.1346  0.1538  0.1731  0.1923  0.2115  0.2308  0.2500  0.2692  0.2885
##   0.001  -0.025   0.060  -0.012   0.024  -0.003  -0.004   0.000   0.008  -0.016
##  0.3077  0.3269  0.3462  0.3654  0.3846  0.4038  0.4231  0.4423  0.4615
##  -0.005   0.040   0.058   0.019   0.002  -0.010   0.064  -0.043  -0.093
```

(e) Exhibit scatterplots of the oil and gas growth rate series for up to three weeks of lead time of oil prices; include a nonparametric smoother in each plot and comment on the results (e.g., Are there outliers? Are the relationships linear?). [Hint: lag.plot2(poil, pgas, 3).]

```
lag2.plot(poil, pgas, 3)
```



The values in the upper right corner are the sample cross-correlations and the lines are a lowess fit (can help to discover any nonlinearities, think of lowess as a robust method for fitting nonlinear regression).

(f) There have been a number of studies questioning whether gasoline prices respond more quickly when oil prices are rising than when oil prices are falling ("asymmetry"). We will attempt to explore this question here with simple lagged regression; we will ignore some obvious problems such as outliers and autocorrelated errors, so this will not be a definitive analysis. Let G_t and O_t denote the gas and oil growth rates.

(g) Fit the regression (and comment on the results)

$$G_t = \alpha_1 + \alpha_2 I_t + \beta_1 O_t + \beta_2 O_{t-1} + w_t$$

where $I_t = 1$ if $O_t \geq 0$ and 0 otherwise (I_t is the indicator of no growth or positive growth in oil price)

```
indi <- ifelse(poil<0, 0, 1)
mess <- ts.intersect(pgas, poil, poilL = lag(poil, -1), indi)
summary(fit <- lm(pgas~poil+poilL+indi, data=mess))
```

```
##
## Call:
## lm(formula = pgas ~ poil + poilL + indi, data = mess)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18451 -0.02161 -0.00038  0.02176  0.34342
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.006445   0.003464  -1.860  0.06338 .
## poil         0.683127   0.058369  11.704  < 2e-16 ***
## poilL        0.111927   0.038554   2.903  0.00385 **
## indi         0.012368   0.005516   2.242  0.02534 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04169 on 539 degrees of freedom
## Multiple R-squared:  0.4563, Adjusted R-squared:  0.4532
## F-statistic: 150.8 on 3 and 539 DF,  p-value: < 2.2e-16
```

(ii) What is the fitted model when there is negative growth in oil price at time t? What is the fitted model when there is no or positive growth in oil price? Do these results support the asymmetry hypothesis?

- With negative growth in oil price:

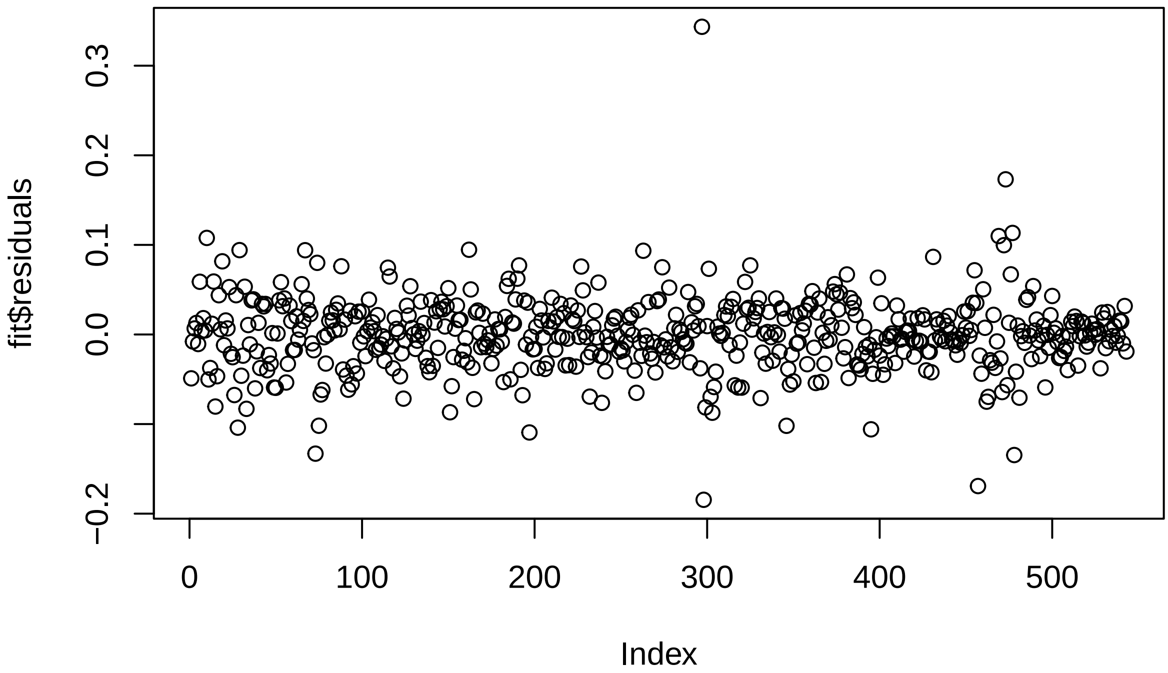
$$\hat{G}_t = -0.006 + 0.683O_t + 0.112O_{t-1}$$

- With no or positive growth in oil price:

$$\hat{G}_t = -0.006 + 0.012 + 0.683O_t + 0.112O_{t-1}$$

(iii) Analyze the residuals from the fit and comment

```
plot(fit$residuals)
```


fit$residuals
Index
0
100
200
300
400
500
−0.2
0.0
0.1
0.2
0.3