

MATH4341 - Spatio-temporal statistics

Term 2: Temporal modelling and time series analysis

Term 2, Lab 4 (and practice test)

Name: **Ben Rickard**

Overview

This lab document is an R Markdown file. It should be downloaded to your computer, and then loaded up into RStudio. Follow the instructions provided, writing code to solve the required problems into the provided code blocks, as directed. Regularly check on progress by “knitting” the document to a PDF and checking that the output is correct. You may find it helpful to have a separate R session window for writing, testing and debugging code solutions, before pasting back into the markdown document - that is up to you. But do make sure that all of your solutions are in this markdown document in the indicated places in the code blocks and that the document compiles into a PDF correctly.

This practical is constructed in a similar way to the practical test that will take place in week 10 (although this practical is shorter, to attempt to fit into a 50 minute session). At the end of the real test in week 10, you will compile your completed document to a PDF file and upload the PDF file (only) to Gradescope for marking. For this practical, you should make sure that you are able to compile your document to a PDF, but you don't need to upload anything anywhere at the end of the session. You should simply check later that your completed PDF looks similar to the solution file that will be provided after the session.

Note that whilst the week 10 test will be conducted under exam conditions (ie. silence, no cooperation), during this lab it's fine to discuss with your neighbours or ask for my help if you get stuck. Although the real test will be conducted in exam conditions, it will be *open-book*. You will be able to consult your lecture notes and other material on Blackboard Ultra that relates directly to this module. You **should not** use AI assistants.

The dataset

The following code block loads up the required library and plots the dataset that is to be the subject of our analysis.

```
library(astsa) # required library
```

```
## Warning: package 'astsa' was built under R version 4.4.2
```

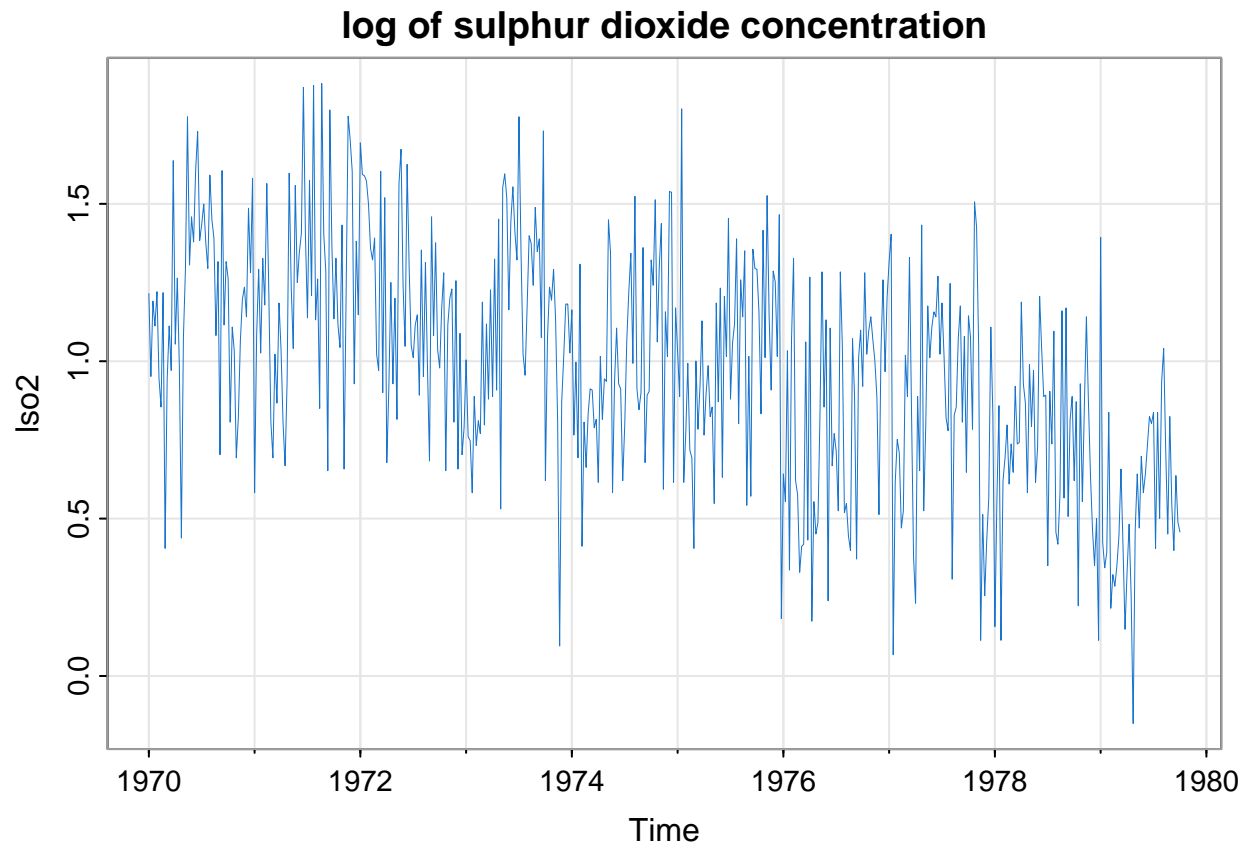
```
lso2 = log(so2) # log of the SO2 data (the log is more Gaussian)
tsp(lso2) # start, end, frequency
```

```
## [1] 1970.00 1979.75 52.00
```

```
n = length(lso2)
n
```

```
## [1] 508
```

```
tsplot(lso2, col=4, lwd=0.5,
      main="log of sulphur dioxide concentration")
```



The dataset is weekly (frequency 52) data.

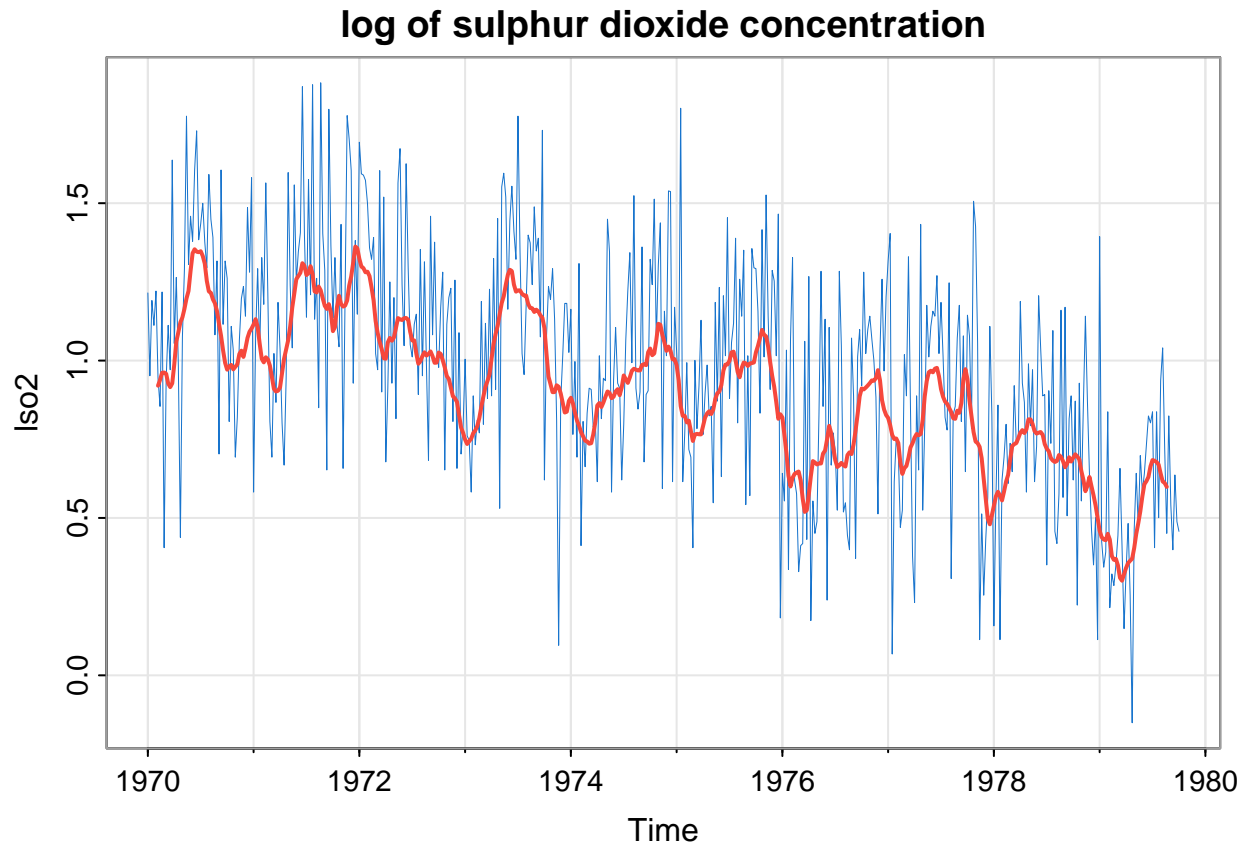
Q1.

Smoothing

1.1

We will begin with some simple smoothing of the time series data to reveal possible overall trends and seasonality. Compute a 12-week moving average of the data, and overlay it onto the plot below with a red line of thickness 2. You might want to consult Chapter 1 of the notes.

```
tsplot(lso2, col=4, lwd=0.5,
      main="log of sulphur dioxide concentration")
## YOUR CODE FOR 1.1 HERE:
lines(filter(lso2, c(1/24,rep(1/12, 10),1/24)), col=2, lwd=2)
```

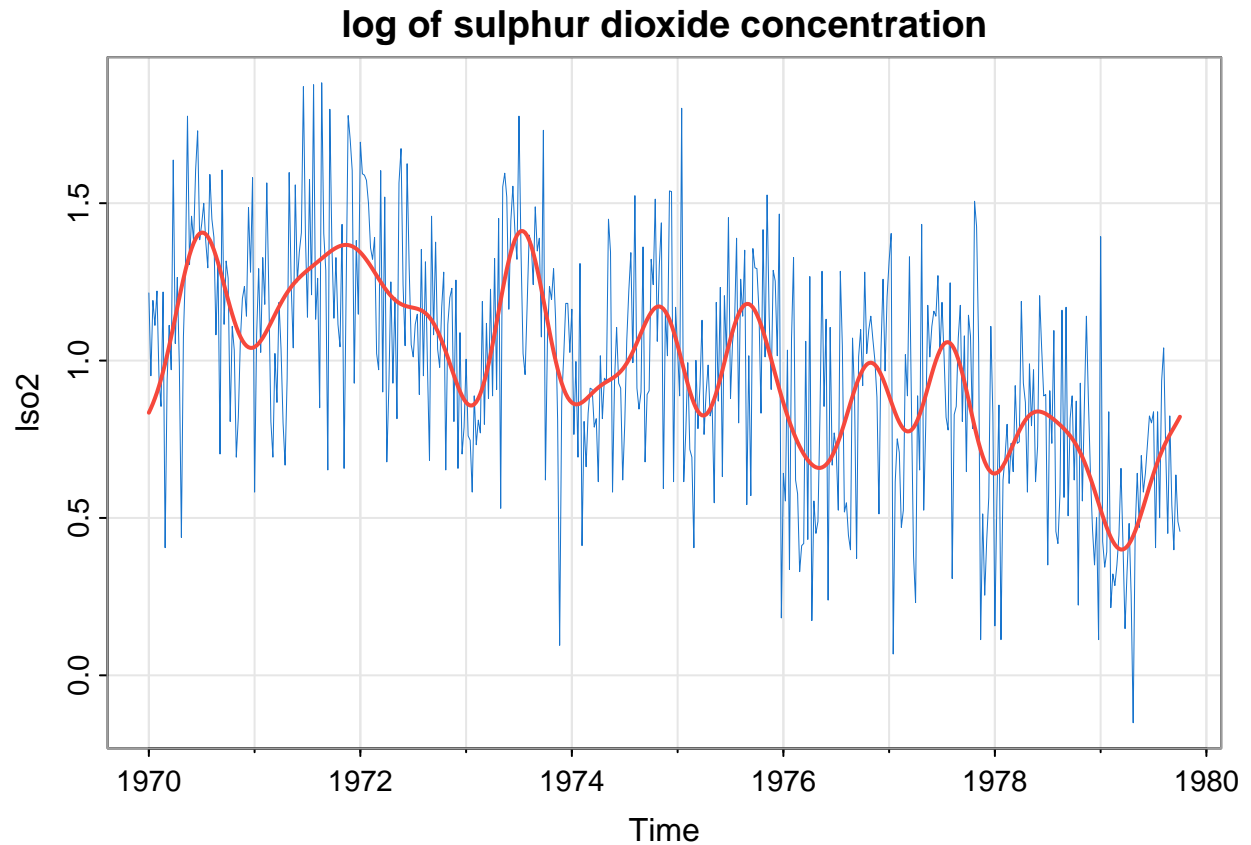


This hints at a gradual downward trend with some annual seasonality.

1.2

A different way to smooth the data is to just keep the low frequency coefficients from a DFT and zero out the rest. Produce a smoothing of the data in this way by just keeping the first 15 DFT frequency components. Again, overlay the smooth on the plot below using a red line of thickness 2. You might want to consult Chapter 5 of the notes.

```
tsplot(lso2, col=4, lwd=0.5,
      main="log of sulphur dioxide concentration")
## An inverse FFT function is likely to be useful
ifft = function(x) fft(x, inverse=TRUE) / length(x)
## YOUR CODE FOR 1.2 HERE:
keep <- 15
ft <- fft(lso2)
ft[(keep+2):(n-keep)]=0
sm <- Re(ifft(ft))
lines(sm, col=2, lwd=2)
```

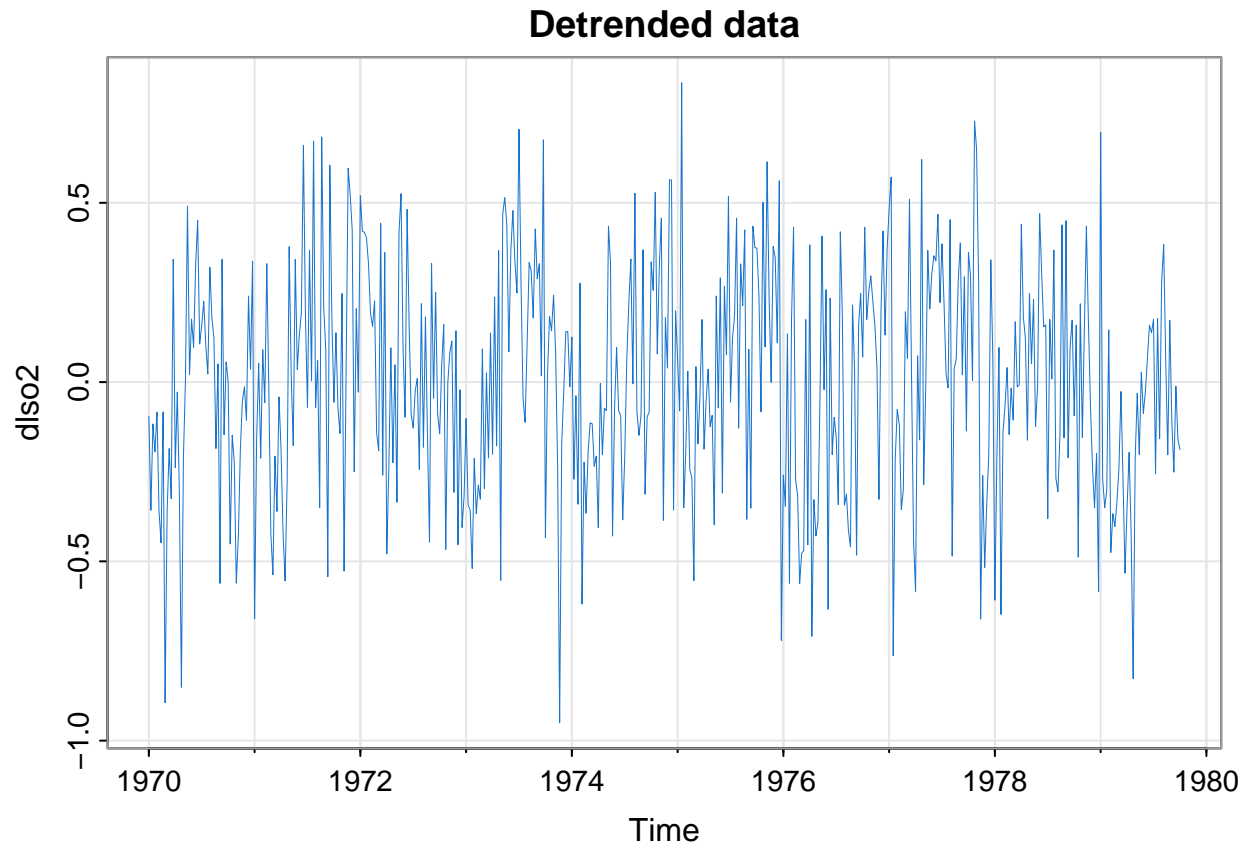


Q2.

ARMA model fitting

We will now turn our attention to ARMA modelling, and for this we will want to focus mainly on the detrended data.

```
dlso2 = detrend(lso2) # dlso2 is the detrended data, for ARMA model fitting
tsplot(dlso2, col=4, lwd=0.5, main="Detrended data")
```

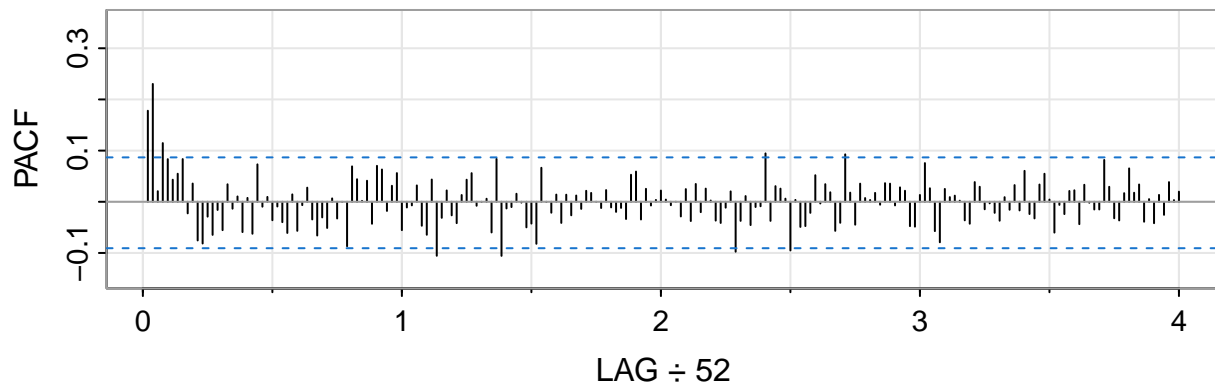
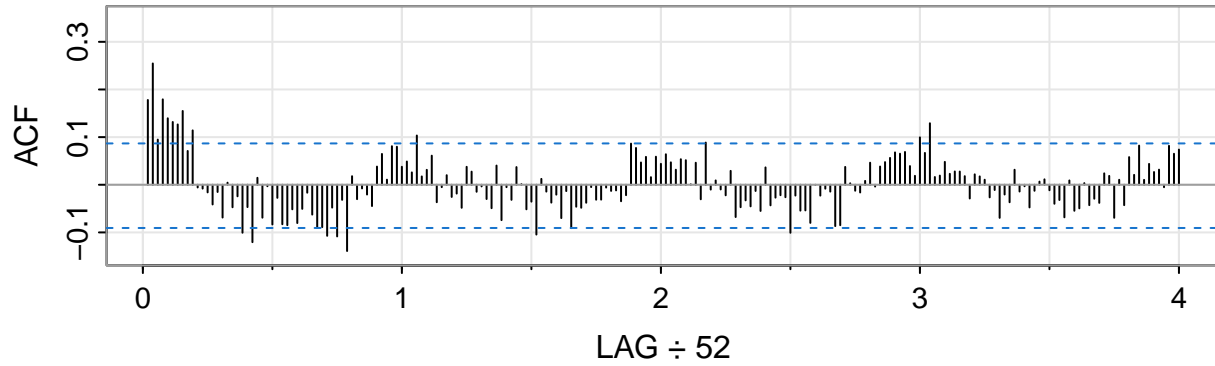


2.1

Plot the ACF and PACF of the detrended data, and on the basis of this, fit an appropriate auto-regressive model (using the `arima` function). Make sure that the fitted model and model coefficients are visible in your code output. You might want to consult Chapter 4 of the notes.

```
## YOUR CODE FOR 2.1 HERE:  
acf2(dlso2)
```

Series: dls02



```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.18 0.25 0.09 0.18 0.14 0.13 0.13 0.15 0.07 0.11 -0.01 -0.01 -0.02
## PACF 0.18 0.23 0.02 0.11 0.08 0.04 0.05 0.08 -0.02 0.04 -0.08 -0.08 -0.03
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF -0.04 -0.02 -0.07 0.00 -0.05 -0.02 -0.10 -0.05 -0.12 0.01 -0.07 0.00
## PACF -0.06 -0.02 -0.06 0.03 -0.01 0.01 -0.06 0.01 -0.06 0.07 -0.01 0.01
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF -0.08 -0.03 -0.08 -0.08 -0.05 -0.08 -0.05 -0.02 -0.06 -0.09 -0.09 -0.11
## PACF -0.04 -0.01 -0.04 -0.06 0.01 -0.06 -0.01 0.03 -0.03 -0.07 -0.03 -0.05
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49]
## ACF -0.05 -0.11 -0.03 -0.14 0.02 -0.03 -0.01 -0.02 -0.04 0.04 0.06 0.01
## PACF 0.01 -0.03 0.00 -0.09 0.07 0.04 0.00 0.04 -0.04 0.07 0.06 -0.02
##      [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60] [,61]
## ACF 0.08 0.08 0.04 0.05 0.03 0.10 0.02 0.03 0.06 -0.04 -0.01 0.02
## PACF 0.03 0.06 -0.06 -0.01 -0.01 0.03 -0.05 -0.06 0.04 -0.11 -0.03 0.02
##      [,62] [,63] [,64] [,65] [,66] [,67] [,68] [,69] [,70] [,71] [,72] [,73]
## ACF -0.03 -0.02 -0.05 0.04 0.03 -0.01 0 -0.03 -0.05 0.04 -0.07 -0.01
## PACF -0.03 -0.04 0.01 0.04 0.06 -0.01 0 0.01 -0.06 0.08 -0.11 -0.01
##      [,74] [,75] [,76] [,77] [,78] [,79] [,80] [,81] [,82] [,83] [,84] [,85]
## ACF -0.03 0.04 0 -0.05 -0.04 -0.10 0.01 -0.01 -0.04 -0.02 -0.07 -0.01
## PACF -0.01 0.02 0 -0.05 -0.04 -0.08 0.07 0.00 -0.02 0.01 -0.04 0.01
##      [,86] [,87] [,88] [,89] [,90] [,91] [,92] [,93] [,94] [,95] [,96] [,97]
## ACF -0.09 -0.05 -0.05 -0.04 0.00 -0.03 -0.03 -0.01 -0.01 -0.01 -0.03 -0.02
## PACF -0.03 0.01 -0.01 0.02 0.02 0.00 -0.01 0.02 -0.01 -0.02 -0.01 -0.03
##      [,98] [,99] [,100] [,101] [,102] [,103] [,104] [,105] [,106] [,107] [,108]
## ACF 0.09 0.08 0.05 0.06 0.02 0.06 0.04 0.06 0.05 0.03 0.05
```

```
## PACF 0.05 0.06 -0.03 0.03 -0.01 0.00 0.02 0.00 -0.01 0.00 -0.03
##      [,109] [,110] [,111] [,112] [,113] [,114] [,115] [,116] [,117] [,118]
## ACF 0.05 0.00 0.05 -0.03 0.09 -0.01 0.01 -0.01 -0.02 0.03
## PACF 0.02 -0.04 0.03 -0.02 0.03 0.00 -0.04 -0.04 -0.01 0.02
##      [,119] [,120] [,121] [,122] [,123] [,124] [,125] [,126] [,127] [,128]
## ACF -0.07 -0.05 -0.03 -0.05 -0.01 -0.05 0.04 -0.04 -0.03 -0.02
## PACF -0.10 -0.04 0.01 -0.05 -0.01 -0.01 0.09 -0.04 0.03 0.03
##      [,129] [,130] [,131] [,132] [,133] [,134] [,135] [,136] [,137] [,138]
## ACF -0.03 -0.1 -0.02 -0.05 -0.05 -0.08 0.00 -0.02 -0.01 -0.01
## PACF 0.01 -0.1 0.00 -0.05 -0.05 -0.02 0.05 0.00 0.03 0.02
##      [,139] [,140] [,141] [,142] [,143] [,144] [,145] [,146] [,147] [,148]
## ACF -0.09 -0.08 0.04 0.00 -0.01 -0.02 0.01 0.05 0.00 0.04
## PACF -0.06 -0.04 0.09 0.02 -0.04 0.04 0.01 0.00 0.02 -0.01
##      [,149] [,150] [,151] [,152] [,153] [,154] [,155] [,156] [,157] [,158]
## ACF 0.05 0.06 0.07 0.07 0.07 0.04 0.02 0.10 0.07 0.13
## PACF 0.04 0.04 -0.01 0.03 0.02 -0.05 -0.05 0.01 0.08 0.03
##      [,159] [,160] [,161] [,162] [,163] [,164] [,165] [,166] [,167] [,168]
## ACF 0.02 0.02 0.05 0.02 0.03 0.03 0.02 -0.03 0.02 0.02
## PACF -0.06 -0.08 0.03 0.01 0.01 0.00 -0.04 -0.04 0.04 0.03
##      [,169] [,170] [,171] [,172] [,173] [,174] [,175] [,176] [,177] [,178]
## ACF 0.01 -0.03 -0.01 -0.07 -0.02 -0.04 0.03 -0.01 0.00 -0.05
## PACF -0.01 0.00 -0.02 -0.04 0.01 -0.02 0.03 -0.02 0.06 -0.02
##      [,179] [,180] [,181] [,182] [,183] [,184] [,185] [,186] [,187] [,188]
## ACF -0.01 0.01 0.01 -0.01 -0.04 -0.03 -0.07 0.01 -0.05 -0.05
## PACF -0.03 0.03 0.05 0.00 -0.06 -0.01 -0.02 0.02 0.02 -0.04
##      [,189] [,190] [,191] [,192] [,193] [,194] [,195] [,196] [,197] [,198]
## ACF 0.00 -0.04 -0.03 -0.04 0.02 0.02 -0.07 0.01 -0.04 0.06
## PACF 0.03 0.00 -0.02 -0.01 0.08 0.03 -0.03 -0.04 0.02 0.07
##      [,199] [,200] [,201] [,202] [,203] [,204] [,205] [,206] [,207] [,208]
## ACF 0.02 0.08 0.01 0.04 0.03 0.03 -0.01 0.08 0.07 0.07
## PACF 0.02 0.03 -0.04 0.01 -0.04 0.01 -0.03 0.04 0.00 0.02
```

```
# PACF suggests that AR(4) may be most appropriate
fit <- arima(dlso2, c(4,0,0), include.mean = FALSE)
fit
```

```
##
## Call:
## arima(x = dlso2, order = c(4, 0, 0), include.mean = FALSE)
##
## Coefficients:
##      ar1      ar2      ar3      ar4
## 0.1297 0.2014 0.0059 0.1149
## s.e. 0.0440 0.0444 0.0444 0.0440
##
## sigma^2 estimated as 0.09065: log likelihood = -111.13, aic = 232.27
```

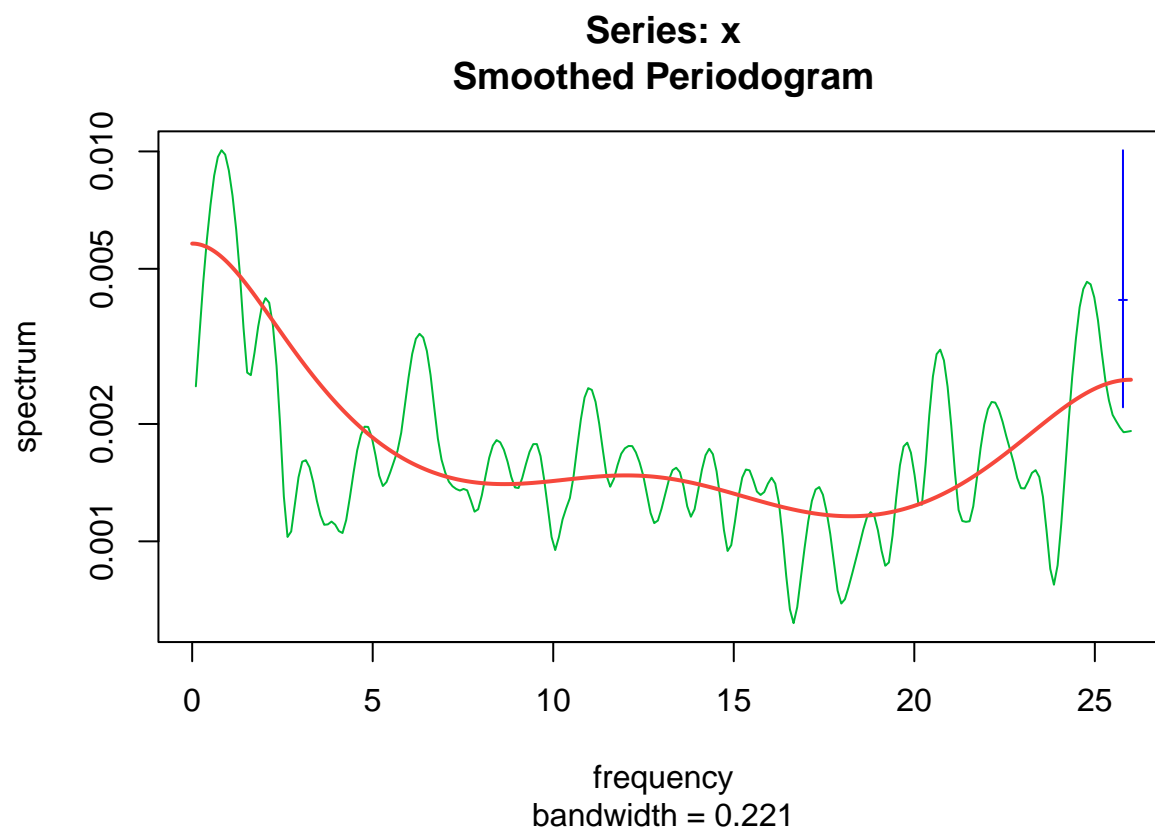
2.2

In order to understand the nature of the fitted auto-regressive model, produce an appropriately smoothed periodogram of the detrended data, and overlay on top the spectrum of the fitted model (with a thicker line). *Note that you may need to divide the spectrum of the fitted model by the frequency of the time series*

in order to put it on the same scale as the periodogram spectrum (due to the way the spectral density is often defined for time series with frequencies other than one).

YOUR CODE FOR 2.2 HERE:

```
spectrum(dlso2, spans=c(5, 7), col=3)
as <- arma.spec(ar=fit$coef, var.noise=fit$sigma2, freq=52, plot=FALSE)
lines(as$freq, as$spec[,1]/52, col=2, lwd=2)
```



Q3.

Forecasting

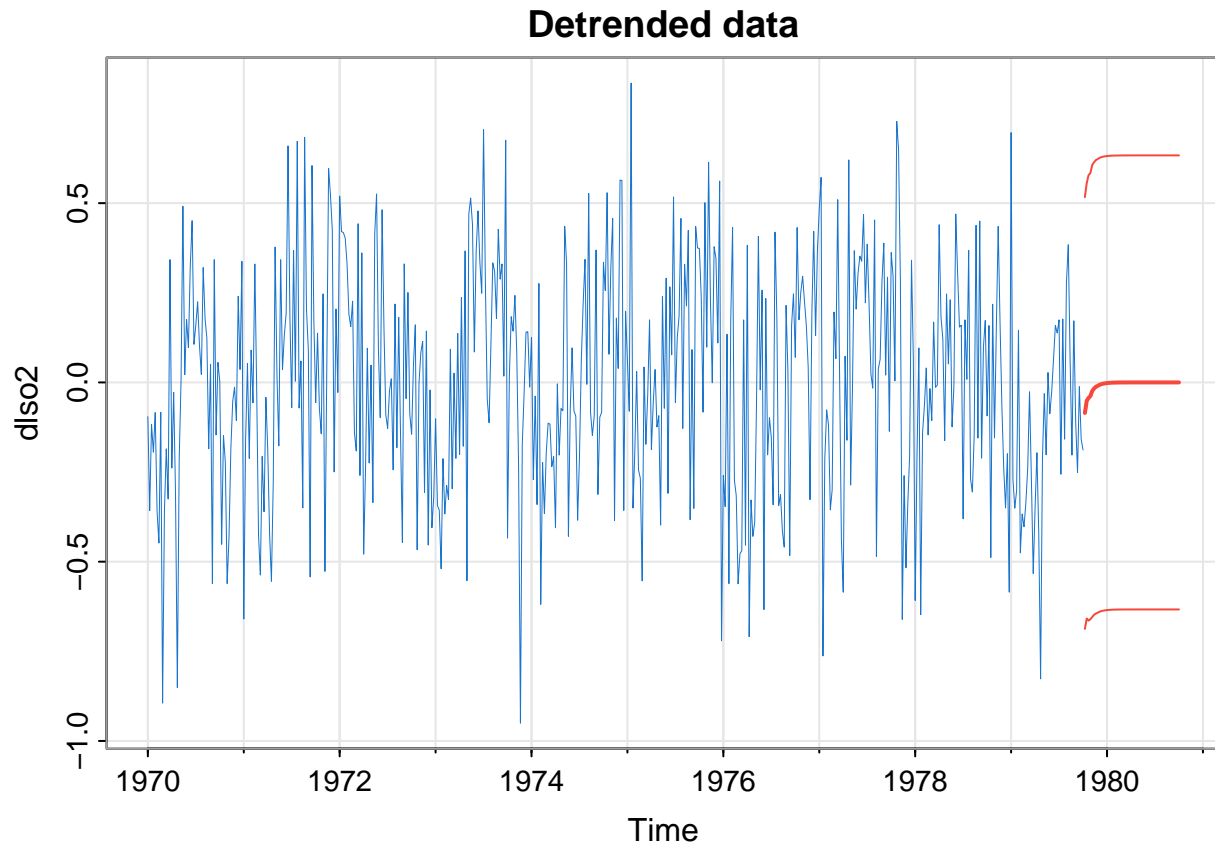
3.1

We now want to forecast the detrended data. Add one year (52 weeks) of forecasts to the plot below (in red), together with upper and lower 2 standard deviation intervals.

```
tsplot(dlso2, col=4, lwd=0.5, main="Detrended data",
       xlim=c(tsp(dlso2)[1], tsp(dlso2)[2]+1))
## YOUR CODE FOR 3.1 HERE:
fore <- predict(fit, n.ahead=52)
pred <- ts(fore$pred, start=tsp(dlso2)[2]+(1/52), freq=tsp(dlso2)[3])
lines(pred, col=2, lwd=2)
upp <- ts(fore$pred+2*fore$se, start=tsp(dlso2)[2]+(1/52), freq=tsp(dlso2)[3])
```



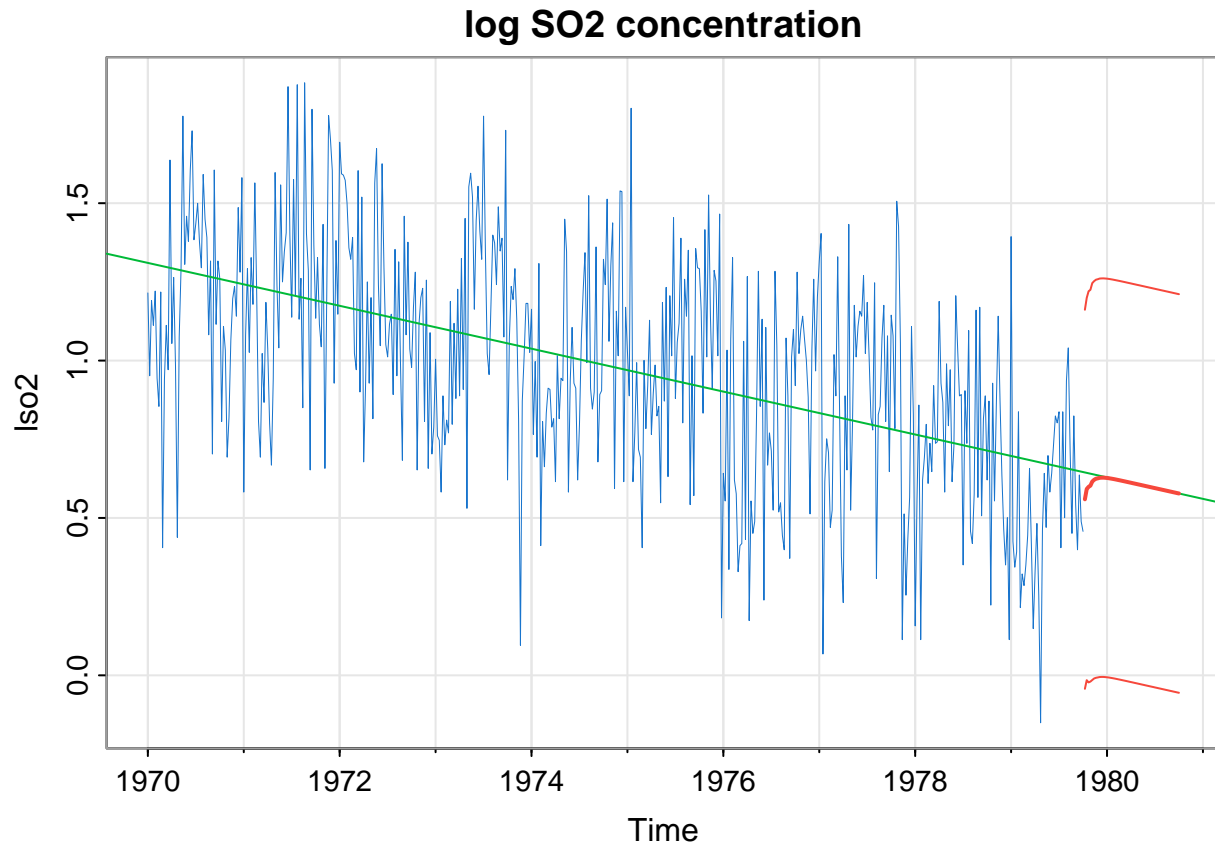
```
lines(upp, col=2)
low <- ts(fore$pred-2*fore$se, start=tsp(dlso2)[2]+(1/52), freq=tsp(dlso2)[3])
lines(low, col=2)
```



3.2

Of course we are really more interested in the non-detrended data (shown below). First, add a linear trend line to the plot (in green), and then, by adding the linear trend to the forecasts you computed above, add appropriate forecasts and upper and lower intervals (in red) to the plot below. *You may ignore the uncertainty in your regression line.*

```
tsplot(lso2, col=4, lwd=0.5, main="log SO2 concentration",
       xlim=c(tsp(dlso2)[1], tsp(dlso2)[2]+1))
## YOUR CODE FOR 3.2 HERE:
dat <- data.frame(y=lso2, x=time(lso2))
lfit <- lm(y ~ x, data=dat)
abline(lfit, col=3)
lines(pred + predict(lfit, data.frame(x=time(pred))), col=2, lwd=2)
lines(upp + predict(lfit, data.frame(x=time(pred))), col=2)
lines(low + predict(lfit, data.frame(x=time(pred))), col=2)
```



The end

The practical is now complete. If you have completed all of the code blocks as required, you should check that you can compile this completed file into a PDF file, and that the PDF file looks correct. That is all you need to do. You don't need to upload it anywhere. However, you should make sure that you know where the PDF file is located on your computer, so that you *could* upload it if you needed to. After the summative test in Week 10 (which will be constructed in a similar way to this lab), you will then upload your compiled PDF file (only) to Gradescope for marking, before the deadline.