MATH3026/4022: Time Series Analysis/Time Series and Forecasting
Computer Practical 3

During this practical, we will use R by loading RStudio and running commands. When writing R commands, it is good practice to write comments so that you understand your code.

Before beginning the practical, it is recommended that you create a folder named `MATH3026` or `MATH4022` within your `Documents` folder on your University workspace. Then, within your `MATH3026` or `MATH4022` folder, create a new folder called `Practical3` in which you can save all of the material covered during this practical session.

You should set your working directory to be the `Practical3` folder that you've created. In RStudio, this can be done by selecting the `Session` menu and then `Set Working Directory` and `Choose Directory`, navigating to the correct folder. You can then easily save your R code and plots in this directory.

1. In this question, we will be using the dataset `Nile`. This dataset records the annual flow (in $10^8 m^3$) of the River Nile at Aswan, Egypt, from 1871–1970 (available in the R `datasets` package). Note: you might need to install the R `datasets` package and then run the command `library(datasets)` to access the `Nile` dataset, depending on the version of R/RStudio that you are using

   (a) Ensure that the working directory in RStudio is set to your `Practical3` folder (see instructions at the beginning of the practical sheet for guidance on how to do this). Once the dataset is available in RStudio you can view the data by typing

   ```
   Nile
   ```

   and then pressing Enter.

   (b) Run the following commands to produce a time plot, a plot of the sample ACF against the lag and a plot of the sample PACF against the lag for the Nile flow data. Does this time series appear to be stationary?

   ```
   #Code to produce a time plot
   ts.plot(Nile,xlab="Year",ylab="Annual flow")

   #Code to produce a plot of the sample ACF against the lag
   acf(Nile)

   #Code to produce a plot of the sample PACF against the lag
   pacf(Nile)
   ```

(c) Now run the following command to define the time series `Nile_diff` as the first difference of the time series `Nile`:

```
Nile_diff<-diff(Nile)
```

Produce a time plot and plots of the sample ACF and PACF against the lag for the `Nile_diff` time series. Does the time series appear to be stationary?

(d) Now we will consider fitting an ARIMA time series model to the `Nile` time series. The following command can be run to fit an ARIMA model to these data using a likelihood-based fit (we'll call this model `model1`, note that you can name your models as you wish).

```
#Code to fit an ARIMA model to the Nile dataset
#Here p = order of the AR part of the model
#q = order of the MA part of the model
#d = order of differencing
#p, d and q should be replaced by integers of your choice
 model1<-arima(Nile,order=c(p,d,q),method="ML")

#For example, to fit an ARMA(1,1) model we'd run the command
 model1<-arima(Nile,order=c(1,0,1),method="ML")

#To view the parameter estimates, standard error estimates and
#other model features, run the command
 model1
```

As an example, we begin by fitting an $AR(1)$ model to the `Nile` dataset (note - not necessarily the most desirable model). To do this, run the command:

```
 model.AR1<-arima(Nile,order=c(1,0,0),method="ML")
```

What are the estimates of ϕ_1, μ and σ_z^2?

(e) Run the following command to extract the model residuals and store these as `resid.AR1`.

```
 resid.AR1<-residuals(model.AR1)
```

Produce a time plot and a plot of the sample ACF against the lag for these residuals. Do the residuals look like white noise?

(f) Finally, we'll examine the Ljung-Box test P-values with respect to the model residuals extracted in (e). Calculation of the Ljung-Box test P-values requires use of the bespoke function `LB_test`. The R code for this function can be found in the file `Ljung-Box-test.R`, available from the course moodle page.

Download the file `Ljung-Box-test.R` and open this in RStudio. Run the commands in this file, then run the commands below to produce a plot of the first ten Ljung-Box test P-values against the degrees of freedom.

```
#Since p+q=1, we run the following command to perform the first ten
#Ljung-Box tests for the model residuals (max.k=11)
AR1.LB<-LB_test(resid.AR1,max.k=11,p=1,q=0)

#To see the table of P-values, type
AR1.LB

#To produce a plot of the P-values against the degrees of freedom and
#add a blue dashed line at 0.05, we run the commands
plot(AR1.LB$deg_freedom,AR1.LB$LB_p_value,xlab="Degrees of freedom",ylab="P-
value",main="Ljung-Box test P-values",ylim=c(0,1))
abline(h=0.05,col="blue",lty=2)
```

What do the P-values of the Ljung-Box test suggest about the fit of the AR(1) model?

(g) By adapting the code from (d)–(f), find and fit an appropriate model to the time series `Nile` and report your model's parameter estimates and associated standard error estimates. Check the fit of your models by producing:

  (i) A time plot of the model residuals.
  (ii) A plot of the sample ACF of the model residuals against the lag.
  (iii) A plot of the first ten P-values for the Ljung-Box test.

You should consider how to obtain your final model by using appropriate hypothesis tests and/or information criteria to be sure that your final model is parsimonious.

HINT: Consider whether or not differencing might be appropriate for this time series (see part (c)).

2. The dataset `airquality` (from the R `datasets` package) contains various daily air quality and weather measurements made in New York City between May and September 1973. In this question we will work with some of these data.

(a) Begin by reading the `airquality` dataset into RStudio. This can be done by typing the command

```
data(airquality)
```

Note: you might need to install the R library `datasets` and then run the command `library(datasets)` prior to running the command above, depending on the version of R/RStudio that you are using.

We will work with the column `Temp` which gives the maximum daily temperature in degrees Fahrenheit at La Guardia Airport. To do this, we must tell R that these data are a time series. We do this by running the command below.

```
ny_temp<-ts(airquality$Temp,start=c(1973,121),frequency=365)

#Note: we use start=c(1973,121) because 1st May was the 121st
#day of 1973

#frequency = 365 implies we're using daily data.
```

(b) Produce a time plot, plot of the sample ACF against the lag and a plot of the sample PACF against the lag for the `ny_temp` data. Does this time series appear to be stationary?

(c) Now take the first difference of the `ny_temp` dataset, by running the command:

```
temp_diff<-diff(ny_temp)
```

Using the methods from question 1. choose an appropriate ARMA model to fit to the differenced New York temperature data. You should justify your choice of model and assess its fit.

Note: this process should involve fitting more than one model and then choosing the most appropriate (and parsimonious) model to fit to the data.