**Stat 510 Week 1 Homework**

**Submit answers in a Word document or a pdf to the designated ANGEL Drop Box in the Drop Boxes folder.**

1. The data for this problem are the October levels of Lake Erie in the United States for 40 consecutive years. The dataset is named eriedata.dat and is linked in the Datasets folder of the course website. Download the file to your computer. We suggest that you create a Stat510 folder on your computer, if you haven’t already done so. Then within that folder, create a Datasets folder. Download datasets to your Datasets folder.

1. Start the R program. (You can use Minitab or SAS if you want, but I’m giving the instructions for R here. At the end of this document, there’s a brief list of Minitab menu sequences for this assignment.)   
     
   Use **File>Change dir ..** (**Mac Users – Misc > Change Working Dir …** ) as a menu sequence and select the folder in which you have stored the dataset for this problem. That will make that directory the “working” directory. Then, enter these three commands:

x = scan ("eriedata.dat")

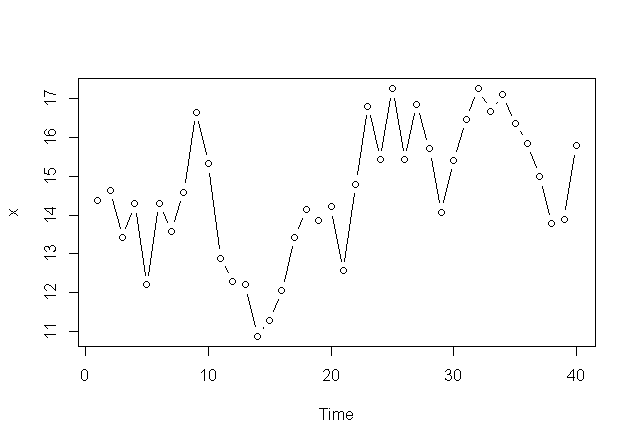
x = ts (x)

plot (x, type=”b”)

The first command reads the data into a variable (object really) named x. The second command designates x as a time series object, a necessary step to make some of the time series commands work right. The third command creates a time series plot of x, the type = "b" part is optional – it puts symbols on the plot where data points fall.

As the answer to this part, copy and paste the time series plot of the data. (Right click the graph, copy it as a bitmap, and then switch to Word and paste. It can be resized there.

The plot is:



**B**. Refer to the plot made for part A. Briefly discuss features of the plot. See Lesson 1.1 for this week to see what we’re worried about (trend, seasonality, outliers, etc.)

Some features of the plot:

* There is no consistent trend and the series wanders up and down.
* We have annual data and therefore deduce that there is no seasonality
* There are no obvious outliers.
* From this plot it’s difficult to judge whether the variance is constant or not.

**C.** Use these two commands to get a plot of the autocorrelation function and a printed version of the values.

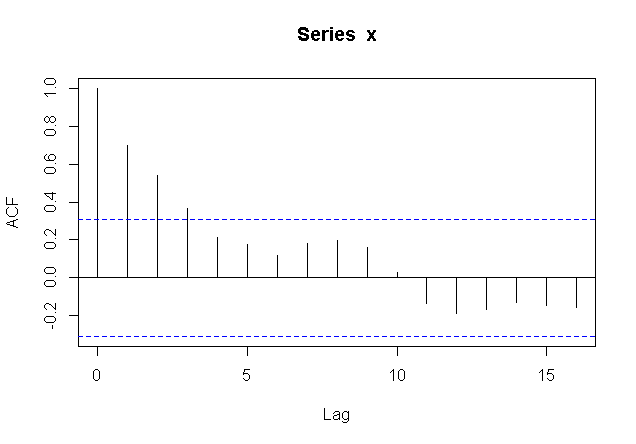
acf1 = acf(x)

acf1

*Notes:* The acf will include a lag 0 correlation which is simply the correlation between the variable and itself, so it must = 1. Ignore the lag 0 autocorrelation. The single command acf(x) all by itself will produce a plot of the ACF but won’t give the numerical values.

As an answer to this part, first copy and paste the value of the acf given as a result of the second command. Then, discuss whether you think the pattern of the first few autocorrelations resembles the theoretical pattern of an AR(1) model.

The ACF plot:



The values are:

Autocorrelations of series ‘x’, by lag

0 1 2 3 4 5 6 7 8 9 10 11 12

1.000 0.698 0.541 0.363 0.212 0.174 0.117 0.182 0.197 0.161 0.028 -0.136 -0.191

13 14 15 16

-0.166 -0.133 -0.149 -0.160

We can deduce the following:

* This looks like the pattern of an AR(1) with a positive lag 1 autocorrelation. Note that the lag 2 correlation (0.541) is roughly equal to the squared value of the lag 1 correlation (0.487). The lag 3 correlation (0.363) is nearly equal to the cubed value of the lag 1 correlation (0.34), and the lag 4 correlation (0.212) nearly equals the fourth power of the lag 1 correlation (0.231).
* Thus an AR(1) model may be a suitable model

**D**. Use these four commands to do a regression for an AR(1) model:

lag1x = lag (x, -1)

y = cbind (x, lag1x)

ar1model = lm (y[,1] ~ y[,2])

summary (ar1model)

The first command creates the first lag of x. The second rather mysterious command creates a matrix named y in which the first column is x (original data) and the second is lag 1 of x. It turns out this is the only way to make the right link between x and lag 1 of x for the regression that’s coming. The third command does a linear model (lm) relating x to lag 1of x (the two column of y). Results from the regression are stored in an object names ar1model. The final command gives a summary of the regression results.

As an answer, copy and paste the summarized regression results. Then, write the regression equation.

The regression results are:

Call:

lm(formula = y[, 1] ~ y[, 2])

Residuals:

Min 1Q Median 3Q Max

-2.25526 -0.80864 -0.04491 1.08912 2.06151

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 4.2878 1.7231 2.488 0.0175 \*

y[, 2] 0.7078 0.1176 6.017 5.95e-07 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.25 on 37 degrees of freedom

(2 observations deleted due to missingness)

Multiple R-squared: 0.4946, Adjusted R-squared: 0.4809

F-statistic: 36.2 on 1 and 37 DF, p-value: 5.954e-07

From the results we can deduce:

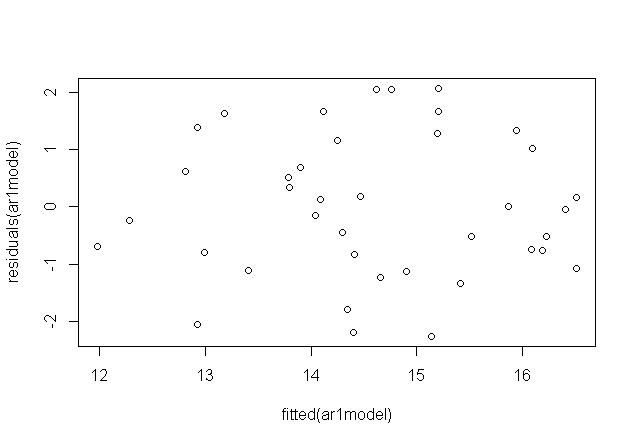
* **The equation is:** **level = 4.2878 + 0.7078 \* lag1**

**E**. Now we’ll look at a plot of residuals versus predicted values for the regression that we just did. The command is

plot(fitted(ar1model),residuals(ar1model))

Briefly discuss whether the plot looks the way it should for a good model. You don’t have to give the plot.

The plot is:



We can deduce: It doesn’t show any serious problems.

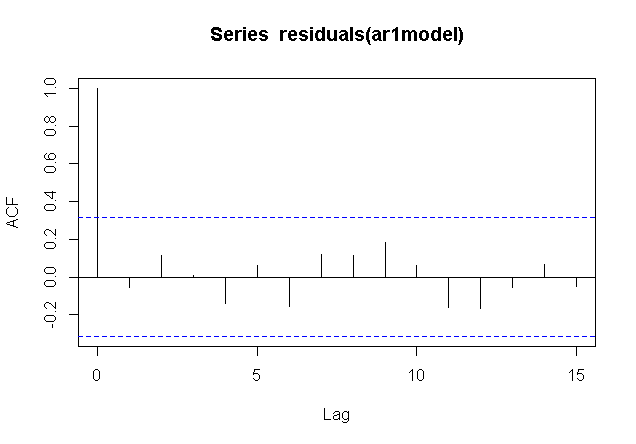
* There are no obvious outliers
* The mean appears to stay around 0 and the variance appears to be constant

**F.** Now we’ll determine the ACF of the residuals. The command is

acf(residuals(ar1model))

Briefly discuss whether the ACF of residuals looks like it should for a good model.

The ACF is:



The blue lines indicate the bounds for statistical significance. This is a good ACF for residuals - Nothing is significant and that’s what we want for the residuals.

**G**. Use the regression equation found in part D to predict the level in the next year past the series. You’ll need the value of x at the end of the series to do this. One way to find that is to enter the command

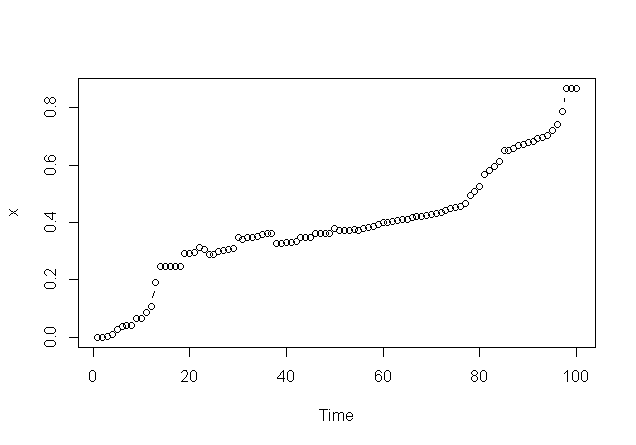
x

That will list the values of the series in order.

Predicted level: levelHat = 4.2878 + 0.7078 \* lag1 = 4.2878 + 0.7078 \* 15.787 = 15.4618

2. For this problem, use the dataset oildata.dat from the Dataset folder of the course website. Download the dataset to your computer. The data are a time series of a price index for oil in the United States for 100 consecutive months. For month t, this measure is  where x = actual price.

**A**. Refer back to part A of problem 1 to see how to read the data and create a time series plot of the data. Briefly describe the noteworthy features. What’s one obvious reason why the series is not a stationary series?



Notable features are:

* There is an upward trend, possibly a curved one.
* There are no obvious outliers.
* The variance across time is uncertain.

There is a clear upward trend, so the series may not be stationary. To create a (possibly) stationary series, we’ll examine the first differences yt = xt - xt-1

**B**. A first difference is defined as. For series with a strong trend, first differences may (or may not) be stationary. An example of how to create a first difference in R, is the following:

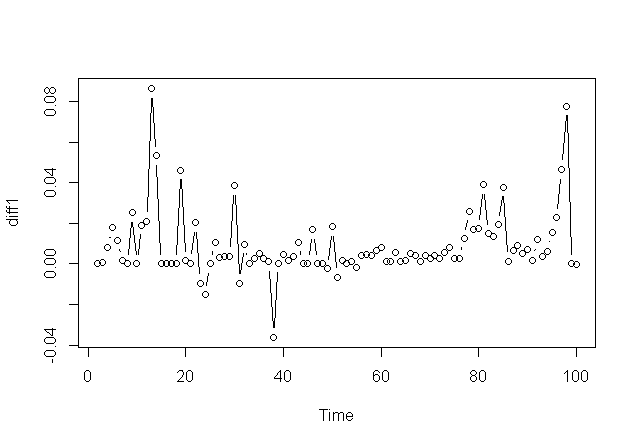
diff1 = diff (x, 1)

The diff1 name is arbitrary. You can call it anything you like. Also, the x within the parentheses is the name that you have already given the data. If you called the original data something else, then that’s what goes there.

Create first differences for this series and the plot the first differences. For example, if you called the first differences diff1, the command could be

plot (diff1, type = ‘b’)

As the answer to this part, give the plot and briefly describe any noteworthy features of the plot.



The notable features:

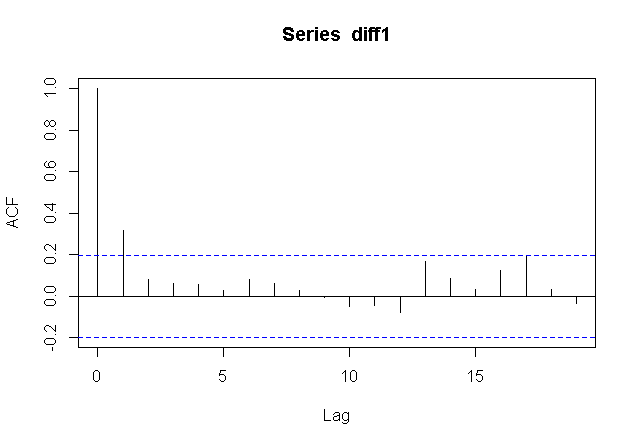
- There is no trend and the series wanders up and down. This indicates that the original series has been detrended

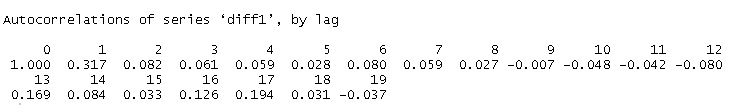
- There are some outliers in the plot.

- Constant variance across time doesn’t seem to hold in this plot.

**C.** Determine the ACF of the first differences. See part C of problem 1 for guidance.

As an answer to this part, copy and paste the value of the acf given as a result of the second command. Then, discuss whether you think the pattern of the first few autocorrelations resembles the theoretical pattern of an AR(1) model.





We deduce here that the ACF tapers more rapidly than they should for an AR(1). However it does adhere to the pattern. Note that the lag 2 correlation (0.082) is slightly less than the squared value of the lag 1 correlation (0.100489). The lag 3 correlation (0.061) is less than the cubed value of the lag 1 correlation (0.0318)

We must add that we also see that only 1st lag is significant and all the remaining lags are not significant. These points indicate that the autocorrelations resembles the theoretical pattern of an MA(1) model more than the AR(1) model.

**D**. A “moving average model of order 1” (abbreviated MA(1)) is defined as

 where wt are independently distributed with mean 0 and variance . That is, the value of x at time t = mean + random error at this time + random error from last time. The theoretical ACF of an MA1 is simple – the first lag autocorrelation is non-zero while all other autocorrelations are 0.

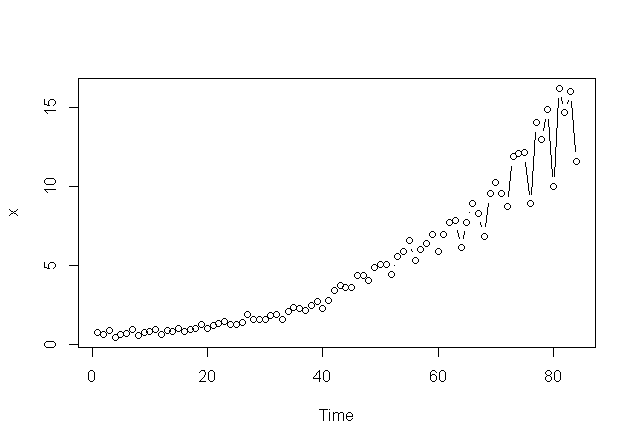
Refer back to the ACF found in part C. Explain why a MA(1) might work as a model for the first differences

We see that only 1st lag is significant and all the remaining lags are not significant. These points indicate that the autocorrelations resembles the theoretical pattern of an MA(1) model

Note: This brings up one of the “fun” parts of time series - more than one model might work for a series.

3. Use the dataset jj.dat from the Datasets folder. The series gives measure of quarterly profits of the Johnson & Johnson Corporation for 84 consecutive quarters.

**A.** Create a time series plot of the data. Give the plot as the answer to this part.



**B.** Refer to the plot in part A. Discuss noteworthy features of the plot. See Lesson 1.1 for this week to review what we’re looking for.

The noteworthy features are:

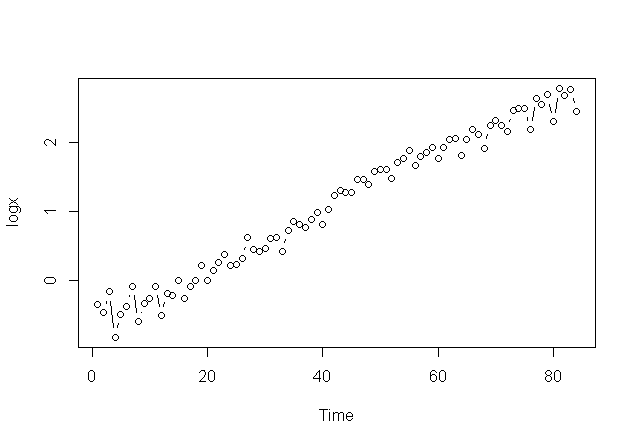
* There is an upward trend, possibly a curved one.
* There is seasonality – a regularly repeating pattern of highs and lows (maybe related to quarters of the year).
* There are no obvious outliers.
* There is increasing variation as we move across time.

**C**. Calculate a series that is the logarithm of the original series, and then plot the logarithm series. For example, if you had called the original series x, then this R sequence would do the job:

logx = log(x)

plot (logx, type = "b")

Give the plot of the logarithm data as the answer to this part and then discuss any differences between this plot and the plot for parts A and B. (For instance, are there changes in trend, variance issues, etc.)



The noteworthy features and differences where applicable are:

* There is an upward linear trend. This is different in that previously there was possibly a non linear upward trend
* There is seasonality – a regularly repeating pattern of highs and lows. Same as before.
* There are no obvious outliers. Same as before.
* The variation appears to be same as we move across time. This is different in that previously there was possibly increasing variance.

D. What regression model would you use to model the series in part C? Briefly describe the predictor variables in the model. See the second example of Lesson 1.1 for this week for guidance.

We note the following for the choice of model / predictor variables

* Since we see a linear trend, we can attempt to use t (the time index) as a predictor variable in a regression.
* Since we have quarterly data, with possible seasonal (quarterly) effects, we can define indicator variables such as Sj = 1 if observation is in quarter j of a year and 0 otherwise. There are 4 such indicators.

The model can be:

xt = β1t+α1S1+α2S2+α3S3+α4S4+ϵt

Where ϵt ∼iid N(0,σ2)

If we want to test a quadratic trend, the model is

xt=β1t+β2t2+α1S1+α2S2+α3S3+α4S4+ϵt

**MINITAB USERS**

For a time series plot, Stat>Time Series > Time Series Plot

For an ACF, Stat>Time Series >Autocorrelation

To lag a variable, Stat>Time Series >Lag

To determine a first difference, Stat>Time Series >Differences