**Module Assignment**

**Module 7**

**QMB-6304 Foundations of Business Statistics**

**#Debjani Sarma**

**rm(list=ls())**

**library(car)**

Write a simple R script to execute the following data preprocessing and statistical analysis. Where required show analytical output and interpretations.

**Preprocessing**

1. Load the file "6304 Module 7 Data.xlsx" into R. This data shows the number of visitors to the United States from the Commonwealth of Australia on a quarterly basis from years 1998 to 2012. The data shown is scaled in thousands of people.

**commonwealth=rio::import("6304 Module 7 Assignment Data.xlsx")**

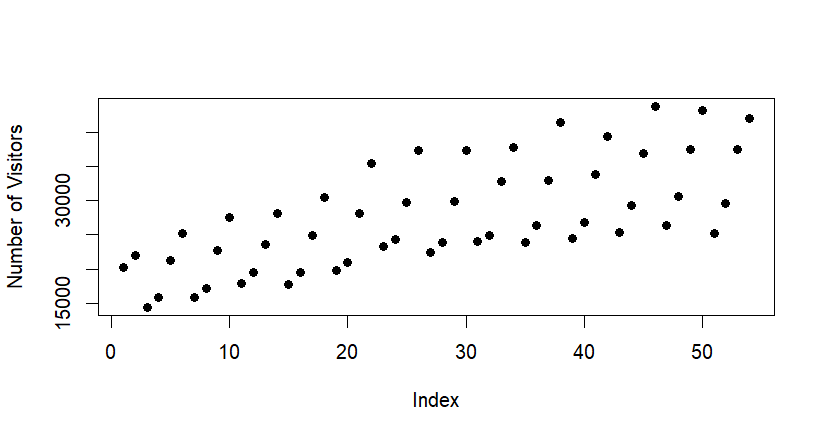
**colnames(commonwealth) = tolower(make.names(colnames(commonwealth)))**

1. Create a new "index" variable in the data frame which will be an identifying sequential numbering of rows from 1 to the number of rows in the data frame. This will be the only data set used for your analysis.

**commonwealth$index = 1:nrow(commonwealth)**

**attach(commonwealth)**

**Analysis**

1. Show a plot of the data using the number of visitors as the "y" variable in the plot.

**plot(index,australia.visitors, pch=19, xlab = "Index",**

**ylab = "Number of Visitors")**

1. Using all the data parameterize a base time series simple regression model using "index" as the independent variable. Show the summary of your regression output.

**> commonwealth.lm = lm(australia.visitors ~ index, data = commonwealth)**

**> summary(commonwealth.lm)**

**Call:**

**lm(formula = australia.visitors ~ index, data = commonwealth)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-10405.0 -4299.8 875.2 3320.7 10194.1**

**Coefficients:**

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

**(Intercept) 18314.62 1528.44 11.983 < 2e-16 \*\*\***

**index 339.28 48.35 7.017 4.65e-09 \*\*\***

**---**

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

**Residual standard error: 5538 on 52 degrees of freedom**

**Multiple R-squared: 0.4863, Adjusted R-squared: 0.4765**

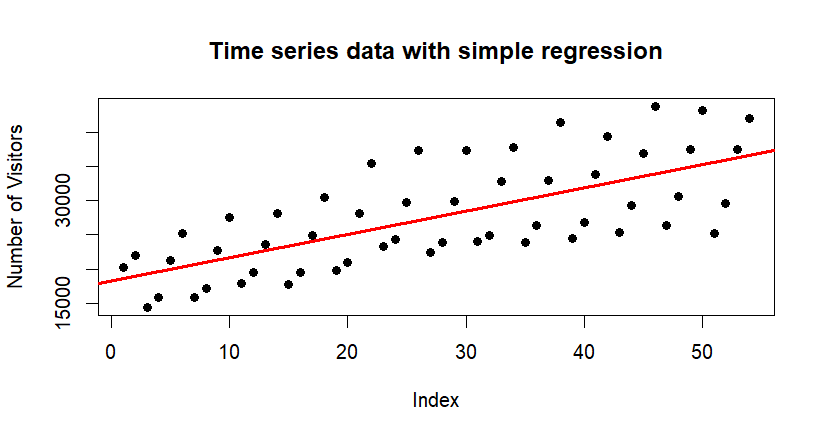
**F-statistic: 49.23 on 1 and 52 DF, p-value: 4.654e-09**

1. Drawing on Analysis Part 1 above, show a properly titled plot of the time series data with the simple regression line layered on the graph in a contrasting color.

**plot(index,australia.visitors, pch=19 , xlab = "Index",**

**ylab = "Number of Visitors", main="Time series data with simple regression")**

**abline(commonwealth.lm,col="red",lwd=3)**

****

1. Execute and interpret a Durbin-Watson test on your model results.

**> durbin.out=durbinWatsonTest(commonwealth.lm)**

**> durbin.out**

**lag Autocorrelation D-W Statistic p-value**

**1 -0.08170513 2.143781 0.682**

**Alternative hypothesis: rho != 0**

Autocorrelation: The autocorrelation coefficient of -0.0817 suggests that there is a negative autocorrelation between residuals of the model. This indicates that if a residual is above(or below) the mean, the next residual is likely to be below(or above)the mean.

D-W Statistic which typically ranges from 0 to 4. Here,

D-W Statistic=2.143781 means there is no significant autocorrelation present in the residuals.

p-value : The p-value of 0.686 suggests that we fail to reject the null hypothesis (which states that there is no autocorrelation).

Alternative hypothesis: rho != 0 means there is significant autocorrelation in the residuals of the model. It indicates that the residuals are not independent and may be related to their previous values.

1. Note the original data appears to have a pronounced cyclical pattern. Assuming the complete cycles are four quarters long, construct a set of seasonal indices which describe the typical annual fluctuations in visitors. Use these indices to deseasonalize the visitors data. Store this deseasonalized data in a column in the original data frame.

**#Making Seasonal Indices**

**indices=data.frame(quarter=1:4,average=0,index=0)**

**for(i in 1:4) {**

**count=0**

**for(j in 1:nrow(commonwealth)) {**

**if(i==commonwealth$quarter[j]) {**

**indices$average[i]=indices$average[i]+australia.visitors[j]**

**count=count+1**

**}**

**}**

**indices$average[i]=indices$average[i]/count**

**indices$index[i]=indices$average[i]/mean(australia.visitors)**

**}**

**#Deseasonalizing**

**commonwealth$deseason.visitors=0**

**for(i in 1:4){**

**for(j in 1:nrow(commonwealth)){**

**if(i==commonwealth$quarter[j]){**

**commonwealth$deseason.visitors[j]=australia.visitors[j]/indices$index[i]**

**}**

**}**

**}**

1. Using the deseasonalized data parameterize four different regression models. A simple regression model will be the base case to be followed by second order, third order, and fourth order polynomial models which attempt to describe the longer-term secular fluctuations in the deseasonalized data.

**#Conducting the deseasonalized regression**

**desreg.out=lm(commonwealth$deseason.visitors ~ index,data=commonwealth)**

**index2 = index^2**

**desreg.out2=lm(commonwealth$deseason.visitors ~ index + index2,data=commonwealth)**

**index3 = index^3**

**desreg.out3=lm(commonwealth$deseason.visitors ~ index + index2 + index3,data=commonwealth)**

**index4 = index^4**

**desreg.out4=lm(commonwealth$deseason.visitors ~ index + index2 + index3 + index4,data=commonwealth)**

1. Reseasonalize the fitted values for each of the four models, storing the reseasonalized values in separate columns in the original data frame. Drawing on Analysis Part 3 above, construct a plot showing the original data and the fitted values for each of the four regression models. Show the four sets of fitted values plots in contrasting colors and title the graph appropriately.

**#Reasonalizing fitted values**

**deseason.fitted1=desreg.out$fitted.values**

**commonwealth$reseason.fitted = 0**

**for(i in 1:4) {**

**for(j in 1:nrow(commonwealth)) {**

**if(i == commonwealth$quarter[j]) {**

**commonwealth$reseason.fitted1[j]=deseason.fitted1[j] \* indices$index[i]**

**}**

**}**

**}**

**deseason.fitted2=desreg.out2$fitted.values**

**commonwealth$reseason.fitted2 = 0**

**for(i in 1:4) {**

**for(j in 1:nrow(commonwealth)) {**

**if(i == commonwealth$quarter[j]) {**

**commonwealth$reseason.fitted2[j]=deseason.fitted2[j] \* indices$index[i]**

**}**

**}**

**}**

**deseason.fitted3=desreg.out3$fitted.values**

**commonwealth$reseason.fitted3 = 0**

**for(i in 1:4) {**

**for(j in 1:nrow(commonwealth)) {**

**if(i == commonwealth$quarter[j]) {**

**commonwealth$reseason.fitted3[j]=deseason.fitted3[j] \* indices$index[i]**

**}**

**}**

**}**

**deseason.fitted4=desreg.out4$fitted.values**

**commonwealth$reseason.fitted4 = 0**

**for(i in 1:4) {**

**for(j in 1:nrow(commonwealth)) {**

**if(i == commonwealth$quarter[j]) {**

**commonwealth$reseason.fitted4[j]=deseason.fitted4[j] \* indices$index[i]**

**}**

**}**

}

**##Plotting reseasonalized graph with original one**

**par(mfrow=c(1,1))**

**plot(index, australia.visitors, type="o",pch=19,**

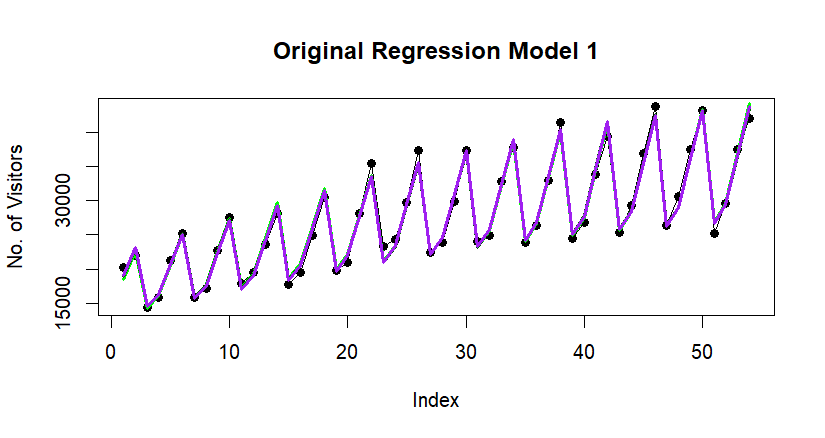
**main="Original Regression Model vs reseasonalized ",**

**xlab="Index", ylab="No. of Visitors")**

**lines(index, commonwealth$reseason.fitted, col="yellow", lwd=3)**

**lines(index, commonwealth$reseason.fitted2, col="green", lwd=3)**

**lines(index, commonwealth$reseason.fitted3, col="darkgreen", lwd=3)**

**lines(index, commonwealth$reseason.fitted4, col="purple", lwd=3)**

1. Select the model which in your view is the best fit to the deseasonalized data. Give a brief justification as to why you believe your selection is the best fit.

**correlation1=cor(commonwealth$deseason.visitors, deseason.fitted1)**

**correlation1**

**correlation2=cor(commonwealth$deseason.visitors, deseason.fitted2)**

**correlation2**

**correlation3=cor(commonwealth$deseason.visitors, deseason.fitted3)**

**correlation3**

**correlation4=cor(commonwealth$deseason.visitors, deseason.fitted4)**

**correlation4**

The correlation coefficient for the fourth-order polynomial regression is 0.9822292, which is the highest among all models. This indicates a very strong linear relationship between the actual deseasonalized visitors and the fitted values from this model.

Your deliverable will be a single MS-Word file showing 1) the R script which executes the above preprocessing and analysis instructions and 2) the results of those instructions and needed written interpretations. The first line of your script file should be a “#” comment line showing your name as it appears in Canvas. Results should be presented in the order in which they are listed here. Deliverable due time will be announced in class and on Canvas. **This is an individual assignment to be completed before you leave the classroom. No collaboration of any sort is allowed on this assignment.**