**6304 Module 7 Assignment**

**Time Series Regression**

***Chad F. Lutz***



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 monthly production of beer in Australia from January of 1956 to December 1978. The data shown is scaled in megaliters(1 Megaliter = 1million liters).
2. Rename the following three columns in this fashion:

|  |  |
| --- | --- |
| **Old name** | **New name** |
| X | index |
| Month | date |
| Monthly.beer.production | production |

1. Create two new columns in the data frame accounting for the “year” and the “month” of the production, respectively. Use the method demonstrated in the lecture on this topic to extract information in the new columns from the existing “date” column.

**beer=import("6304 Module 7 Assignment Data.xlsx")**

**colnames(beer)=c("index","date","production")**

**beer$year=as.numeric(format(beer$date,'%Y'))**

**beer$month=as.numeric(format(beer$date,'%m'))**

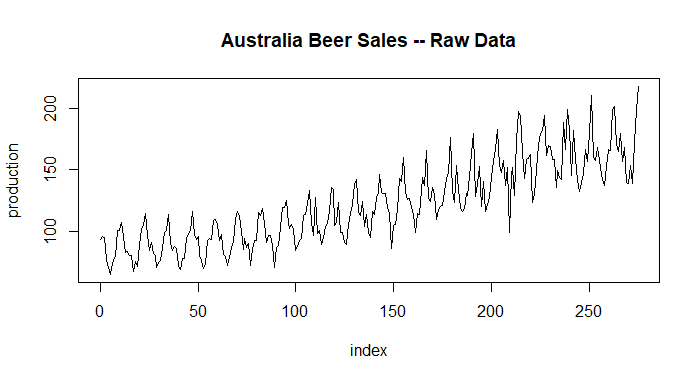
**Analysis**

1. Show a line plot of the data using the index as ‘x’ and production as "y" variable. Show appropriate main titles and axis titles on the graph.

**attach(beer)**

**plot(index,production,pch=19,type="l",**

**main= "Australia Beer Sales -- Raw Data")**



1. Using all the rows parameterize a base time series simple regression model using "index" as the independent variable and production as dependent variable. Show the summary of your regression output. From this, state the slope of your regression line and the correlation coefficient between actual and predicted values for production.

***The slope is 0.3541 with correlation of 0.853***

**beer.out=lm(production~index,data=beer)**

**cor(beer$production,beer.out$fitted.values)**

**[1] 0.8528959**

**summary(beer.out)**

**Call:**

**lm(formula = production ~ index, data = beer)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-48.175 -12.695 -1.649 11.210 48.452**

**Coefficients:**

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

**(Intercept) 72.8636 2.0810 35.01 <2e-16 \*\*\***

**index 0.3541 0.0131 27.04 <2e-16 \*\*\***

**---**

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

**Residual standard error: 17.33 on 274 degrees of freedom**

**Multiple R-squared: 0.7274, Adjusted R-squared: 0.7264**

**F-statistic: 731.3 on 1 and 274 DF, p-value: < 2.2e-16**

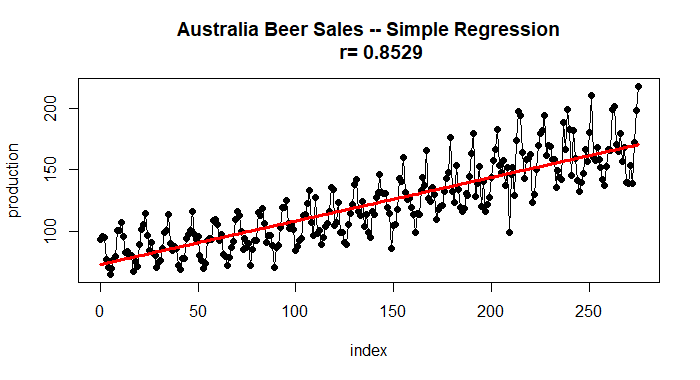
1. Drawing on Analysis Part 1 above, show a properly titled plot of the time series data with the simple regression line layered on the graph in a contrasting color.

**plot(index,production,pch=19,type="o",**

**main=paste("Australia Beer Sales -- Simple Regression**

**r=",round(cor(beer$production,beer.out$fitted.values),4)))**

**points(index,beer.out$fitted.values,type="l",col="red",lwd=3)**



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

***Not much indication of autocorrelation as 1.5-2.5 is normal.***

**durbinWatsonTest(beer.out)**

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

**1 0.5350599 0.8973878 0**

**Alternative hypothesis: rho != 0**

1. Note the original data appears to have a pronounced cyclical pattern. Assuming the complete cycles are 12 months long, construct a set of seasonal indices which describe the typical annual fluctuations in production. Use these indices to deseasonalize the production data. Store this deseasonalized data in a column in the original data frame.

**indices=data.frame(month=1:12,average=0,index=0)**

**for(i in 1:12) {**

**count=0**

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

**if(i==beer$month[j]) {**

**indices$average[i]=indices$average[i]+beer$production[j]**

**count=count+1**

**}**

**}**

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

**indices$index[i]=indices$average[i]/mean(beer$production)}**

**for(i in 1:12) {**

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

**if(i==beer$month[j]) {**

**beer$deseason[j]=beer$production[j]/indices$index[i]**

**}**

**}**

**}**

1. Using the deseasonalized data parameterize two different regression models. A simple regression model will be the base case and a second order polynomial (x2) model which attempts to describe the non-linear secular fluctuations in the deseasonalized data.

**desbeer.out=lm(deseason~index,data=beer)**

**beer$index2=beer$index^2**

**sqbeer.out=lm(deseason~index+index2,data=beer)**

**summary(desbeer.out)**

**Call:**

**lm(formula = deseason ~ index, data = beer)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-27.434 -6.525 -0.431 4.735 26.192**

**Coefficients:**

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

**(Intercept) 73.214609 1.047160 69.92 <2e-16 \*\*\***

**index 0.351567 0.006589 53.35 <2e-16 \*\*\***

**Residual standard error: 8.722 on 274 degrees of freedom**

**Multiple R-squared: 0.9122, Adjusted R-squared: 0.9119**

**F-statistic: 2847 on 1 and 274 DF, p-value: < 2.2e-16**

**summary(sqbeer.out)**

**Call:**

**lm(formula = deseason ~ index + index2, data = beer)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-34.104 -4.882 -0.064 4.102 25.932**

**Coefficients:**

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

**(Intercept) 8.035e+01 1.455e+00 55.235 < 2e-16 \*\*\***

**index 1.954e-01 2.444e-02 7.994 3.71e-14 \*\*\***

**index2 5.680e-04 8.602e-05 6.604 2.09e-10 \*\*\***

**Residual standard error: 8.114 on 273 degrees of freedom**

**Multiple R-squared: 0.9243, Adjusted R-squared: 0.9237**

**F-statistic: 1666 on 2 and 273 DF, p-value: < 2.2e-16**

1. Reseasonalize the fitted values for each of the two models. Drawing on Analysis Parts 1 and 3 above, construct a plot showing the original data and the fitted values for each of the two regression models. From a visual review, which model appears to have the better fit to the original beer production data?

***With simple glance they look nearly identical, however, with closer inspection you can see an R value in the title indicating the squared model is a better fit. The squared model seems to match the peeks and valleys better than the standard seasonal model.***

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

**xx=beer$month[j]**

**beer$reseason.y.hat[j]=desbeer.out$fitted.values[j]\*indices$index[xx]**

**beer$reseason.error[j]=beer$production[j]-beer$reseason.y.hat[j]**

**}**

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

**xx=beer$month[j]**

**beer$sq.reseason.y.hat[j]=sqbeer.out$fitted.values[j]\*indices$index[xx]**

**beer$sq.reseason.error[j]=beer$production[j]-beer$sq.reseason.y.hat[j]**

**}**

**plot(beer$index,beer$production,type="o",lwd=3,**

**main=paste("Australia Beer Sales -- Season Adjusted**

**r=",round(cor(beer$production,beer$reseason.y.hat),4)))**

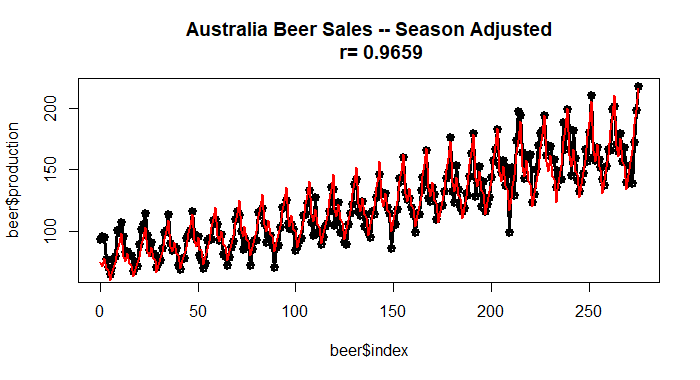
**points(beer$index,beer$reseason.y.hat,type="l",col="red",lwd=2)**

**plot(beer$index,beer$production,type="o",lwd=3,**

**main=paste("Australia Beer Sales -- Season Adjusted^2**

**r=",round(cor(beer$production,beer$sq.reseason.y.hat),4)))**

**points(beer$index,beer$sq.reseason.y.hat,type="l",col="red",lwd=2)**



A graph of a bar chart

Description automatically generated

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.

**#Chad Lutz**

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

**library(rio)**

**library(car)**

**beer=import("6304 Module 7 Assignment Data.xlsx")**

**names(beer)**

**colnames(beer)=c("index","date","production")**

**str(beer)**

**beer$year=as.numeric(format(beer$date,'%Y'))**

**beer$month=as.numeric(format(beer$date,'%m'))**

**str(beer)**

**attach(beer)**

**plot(index,production,pch=19,type="l",**

**main= "Australia Beer Sales -- Raw Data")**

**beer.out=lm(production~index,data=beer)**

**summary(beer.out)**

**cor(beer$production,beer.out$fitted.values)**

**plot(index,production,pch=19,type="o",**

**main=paste("Australia Beer Sales -- Simple Regression**

**r=",round(cor(beer$production,beer.out$fitted.values),4)))**

**points(index,beer.out$fitted.values,type="l",col="red",lwd=3)**

**durbinWatsonTest(beer.out)**

**indices=data.frame(month=1:12,average=0,index=0)**

**for(i in 1:12) {**

**count=0**

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

**if(i==beer$month[j]) {**

**indices$average[i]=indices$average[i]+beer$production[j]**

**count=count+1**

**}**

**}**

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

**indices$index[i]=indices$average[i]/mean(beer$production)}**

**for(i in 1:12) {**

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

**if(i==beer$month[j]) {**

**beer$deseason[j]=beer$production[j]/indices$index[i]**

**}**

**}**

**}**

**desbeer.out=lm(deseason~index,data=beer)**

**summary(desbeer.out)**

**beer$index2=beer$index^2**

**sqbeer.out=lm(deseason~index+index2,data=beer)**

**summary(sqbeer.out)**

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

**xx=beer$month[j]**

**beer$reseason.y.hat[j]=desbeer.out$fitted.values[j]\*indices$index[xx]**

**beer$reseason.error[j]=beer$production[j]-beer$reseason.y.hat[j]**

**}**

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

**xx=beer$month[j]**

**beer$sq.reseason.y.hat[j]=sqbeer.out$fitted.values[j]\*indices$index[xx]**

**beer$sq.reseason.error[j]=beer$production[j]-beer$sq.reseason.y.hat[j]**

**}**

**plot(beer$index,beer$production,type="o",lwd=3,**

**main=paste("Australia Beer Sales -- Season Adjusted**

**r=",round(cor(beer$production,beer$reseason.y.hat),4)))**

**points(beer$index,beer$reseason.y.hat,type="l",col="red",lwd=2)**

**plot(beer$index,beer$production,type="o",lwd=3,**

**main=paste("Australia Beer Sales -- Season Adjusted^2**

**r=",round(cor(beer$production,beer$sq.reseason.y.hat),4)))**

**points(beer$index,beer$sq.reseason.y.hat,type="l",col="red",lwd=2)**