THE UNIVERSITY OF TEXAS AT AUSTIN
McCombs School of Business

STA 372.5   **Spring 2019**

HOMEWORK #2 – Due Wednesday, February 13

1. Problem #2 on the 2015 midterm exam.

Suggested problems from previous exams

The following problems from previous years' exams do not need to be turned in, although I strongly recommend that you do them. Their answers will be distributed with the answers to this homework.

2015 exam: 1
2017 exam: 1
2018 exam: 1

You must turn in the answer to problem #2 below.

2. The file STA372_Homework2.dat on the *Data sets* page of the Canvas class website contains 44 quarters of quarterly sales data for The Gap. The three columns in the data set are *Time*, *Quarter* (this variable is *Qrt1*, *Qrt2*, *Qrt3*, *Qrt4*, *Qrt1*, *Qrt2*, …), and *Sales*.

   This problem should be done using R. When submitting your answers, please submit both the R output and R plots requested below.

   Create an R script to do the following:

   (a) Use the *read.table* command to read the data into a data frame named *Sales_table* and the *colnames* command to name the columns *Time*, *Quarter* and *Sales*, respectively. Compute log(*Sales*) and store it in *Sales_table$LogSales*. Print the first six rows of the data frame by typing *head(Sales_table)*.

   (b) Use *ggplot* to construct a plot of *Sales* vs. *Time* with a point representing each observation and the points connected with lines. Label the *x*- and *y*-axes "*Time*" and "*Sales*", respectively, and make the title of the plot "*Sales vs. Time*".

(c) Use *ggplot* to construct a plot of log(*Sales*) vs. *Time* with a point representing each observation and the points connected with lines. Label the *x*- and *y*-axes "*Time*" and "*log(Sales)*", respectively, and make the title of the plot "*log(Sales) vs. Time*".

Did the log transformation stabilize the variance?

(d) Use the *ts* command to create a time series object called *LogSales_time_series* with *Sales_table$LogSales* as the time series data and *frequency* = 4.

(e) Use the *stl* package to decompose *LogSales_time_series* into its trend, seasonal and irregular components, and store the components in *LogSales_time_series_components*.

(f) Print the time series components using by typing *LogSales_time_series_components*. Plot the components using the *plot(LogSales_time_series_components)* command.

(g) Compute the seasonally adjusted *log(Sales)* values and put them in the column *Sales_table$LogA*.

(h) Use *ggplot* to construct a plot of *LogA* vs. *Time* with a point representing each observation and the points connected with lines. Label the *x*- and *y*-axes "*Time*" and "*LogA*", respectively, and make the title of the plot "*Seasonally adjusted log(Sales)*".

(i) Compute the seasonally adjusted *Sales* values and put them in the column *Sales_table$A*. Print the first six rows of *Sales_table*.

(j) Use *ggplot* to construct a plot of *A* vs. *Time* with a point representing each observation and the points connected with lines. Label the *x*- and *y*-axes "*Time*" and "*A*", respectively, and make the title of the plot "*Seasonally adjusted Sales*".

Did the *stl* seasonal decomposition procedure do a good job seasonally adjusting *Sales*?

Hint: Before running the R script, don't forget to set the working directory (by clicking on *Session > Set Working Directory > To Source File Location* on the menu at the top of the screen). Also, before starting the script, be sure to type *install.packages("ggplot2")* at the "> prompt" in the console window to install *ggplot2*, and then type *library(ggplot2)* at the "> prompt" in the console window to load it from the library. Similarly, type *install.packages("fpp2")* at the "> prompt" to install *fpp2*, and then type *library(fpp2)* at the "> prompt" to load it.

The output you obtain from your R script should look similar to the output on the next page:

R output

```
   Time Quarter  Sales LogSales
1    1     Qtr1 105715 11.56850
2    2     Qtr2 120136 11.69638
3    3     Qtr3 181669 12.10994
4    4     Qtr4 239813 12.38761
5    5     Qtr1 159980 11.98280
6    6     Qtr2 164760 12.01225

 Call:

Components
          seasonal     trend      remainder
 1 Q1 -0.17186373 11.79106 -0.0506963987
 1 Q2 -0.16799917 11.88385 -0.0194668478
 1 Q3  0.05868272 11.97320  0.0780571010
 1 Q4  0.27799018 12.06138  0.0482435187
 2 Q1 -0.16859626 12.14624  0.0051607303
 2 Q2 -0.16527035 12.20839 -0.0308755135
 2 Q3  0.05429794 12.26914 -0.0004684644
 2 Q4  0.27383633 12.33088  0.0017088905

          --- Output removed ---

10 Q1 -0.15906972 13.68008  0.0090398017
10 Q2 -0.15200115 13.70925  0.0009558042
10 Q3  0.08230293 13.73989 -0.0184053805
10 Q4  0.23170646 13.77073  0.0036939944
11 Q1 -0.16050494 13.81370 -0.0017482264
11 Q2 -0.15365775 13.86096 -0.0327614239
11 Q3  0.08049325 13.90861 -0.0251727453
11 Q4  0.23510700 13.95734  0.0417691511

   Time Quarter  Sales LogSales    Seasonal     LogA        A
1    1     Qtr1 105715 11.56850 -0.17186373 11.74037 125538.3
2    2     Qtr2 120136 11.69638 -0.16799917 11.86438 142113.2
3    3     Qtr3 181669 12.10994  0.05868272 12.05126 171314.9
4    4     Qtr4 239813 12.38761  0.27799018 12.10962 181611.4
5    5     Qtr1 159980 11.98280 -0.16859626 12.15140 189359.1
6    6     Qtr2 164760 12.01225 -0.16527035 12.17752 194369.4
```