**DS 710**

**Homework 8**

**R assignment – Soujit Polireddy**

1. In this problem, you will create and apply a function that rates cities based on how appealing they are for you to live in.

a.  What factors are important to you in deciding where to live?  The data set Best Cities.csv contains data on 10 U.S. cities, obtained from <http://www.census.gov/quickfacts/>, [www.walkscore.com](http://www.walkscore.com), and <http://www.wunderground.com/>.  Develop your own formula to rate how pleasant a city would be to live in, in your opinion, based on the variables in this data set.

* Your formula should yield a single number, which is higher for cities which are more pleasant.
* Your formula should use at least 3 different variables from the data set.
* Your formula should not rely on a comparison between cities.
* If there are other variables that you consider to be important, you may add them to the data set.  If so, upload the modified version of the data set with your homework, and include a link to the data source.
* Be creative!

The three main conditions I look for when choosing a city to live in are: Lowest Temperatures, population density and traveling time I must invest from getting point to point.

|  |
| --- |
|  |
| 1. Min Temp, F, 2014 2. Population per square mile, 2010 3. (Mean travel time to work (minutes), workers age 16 years+, 2009-2013) |
|  |
|  |

I want to live in comfortable and serene area. Where the temperatures aren’t too extreme from either side. I don’t want the temperatures to be freezing cold during its peak winter seasons in its respective areas where it gets the coldest. In addition to that, I’d greatly appreciate an area where the traveling time isn’t greatly hindered by traffic or where density of the population is so much that it get hectic to even travel around.

# is added to final city score + number means better - means worse

#20 minutes - the cities time we add based on better or worse

# subtract population density / 1000 (i.e 14000 per sq mile / 1000 = 14) higher the worse

# 1 to 10 Minnesota is 1.0 and Miami is 7.6

b. Create an R function that computes the pleasantness score of a city, based on a vector of data about it.

mycityData **=** read.csv**(**"C:/Users/pedbv9699/Documents/GitHub/ds710assignment8/Best Cities.csv" header **=** T**)**

attach**(**mycityData**)**

City\_Livabilty\_Index **<-** **function(**x**){**

cityScore **=** 100

# this is travel time in minutes/ the different between the normal 20 minute commute(According to me)

# is added to final city score + number means better - means worse

#20 minutes - the cities time we add based on better or worse

commuteScore **=** **(**20 **-** as.numeric**(**x**[**2**]))**

# subtract population density / 1000 (i.e 14000 per sq mile / 1000 = 14) higher the worse

populationDensityScore **=** as.numeric**(**x**[**14**])** **/** 1000

# 1 to 10 Minnesota is 1.0 and Miami is 7.6

tempScore **=** **((**32 **+** as.numeric**(**x**[**18**])))** **/** 10

cityScore **=** cityScore **+** commuteScore **-** populationDensityScore **+** tempScore

return**(**cityScore**)**

**}**

c.Use apply() to apply your function to each city in the Best Cities.csv data set.  Based on your criteria, which city is the best for you?  Does this assessment seem accurate?  If not, what would you want to change about your formula?

**>** print**(**sort**(**index, decreasing **=**T**))**

Charlottesville.city..Virginia Madison.city..Wisconsin

105.256 101.071

Miami.city..Florida Minneapolis.city..Minnesota

98.245 96.142

Austin.city..Texas Portland.city..Oregon

96.125 95.690

Seattle.city..Washington San.Francisco.city..California

94.822 91.865

Philadelphia.city..Pennsylvania Los.Angeles.city..California

85.080 72.003

New.York.city..New.York

43.140

I think this is really good indication, I started at 100 as a base score, and I just didn’t want to see negative numbers. I think my top cities are Chartosville, Madision, Miami. Population density is less, traffic seems a little better than compared to others temperatures are acceptable.

2.  Can we use statistical analysis of word lengths to identify the author of an anonymous essay?  In Homework 7, you wrote a Python function that counted the lengths of words in the 1770 essay by “A Mourner”.  Analysis of other articles published in *The Boston Gazette and Country Journal* in early 1770 finds that John Hancock wrote a 121-word article with a mean word length of 4.69 and standard deviation of 2.60.

a. Use R to assess whether it is plausible that John Hancock was “A Mourner”.

Note:  Because we only have summary statistics (mean and standard deviation) for John Hancock’s article, the t.test function won’t work to do a two-sample t-test.  You may either assume that the average word length of all of John Hancock’s writing (not just this sample) was 4.69 (and do a one-sample t-test), or install the BSDA package and use tsum.test() instead, as discussed here:  <http://stats.stackexchange.com/questions/30394/how-to-perform-two-sample-t-tests-in-r-by-inputting-sample-statistics-rather-tha>

Note 2: The null hypothesis for a 2-sample t-test of this question is

i.e., that A Mourner and Hancock have the *same* mean word length. In other words, the null hypothesis is that it *is* plausible that Hancock was “A Mourner.”

**>** vec1 **=** c**(**3, 7, 8, 3, 7, 3, 3, 6, 2, 3, 3, 2, 3, 4, 3, 8, 10, 2, 3, 3, 7, 4, 2, 10, 6, 3, 4, 9, 3, 6, 4, 2, 4, 2, 6, 4, 3, 8, 5, 2, 5, 4, 8, 11, 2, 6, 4, 4, 3, 3, 7, 2, 7, 3, 4, 2, 11, 2, 6, 5, 4, 8, 2, 3, 7, 2, 4, 6, 4, 3, 5, 6, 2, 3, 5, 10, 5, 6, 5, 4, 8, 8, 8, 2, 3, 8, 7, 2, 3, 6, 3, 6, 2, 3, 9, 3, 6, 4, 3, 3, 7, 3, 5, 2, 9, 3, 8, 8, 2, 6, 4, 3, 4, 5, 2, 3, 3, 4, 2, 7, 5, 6, 8, 4, 3, 7, 6, 6, 5, 2, 3, 6, 12, 7, 6, 2, 5, 5, 5, 6, 2, 5, 2, 3, 1, 7, 6, 3, 5, 4, 4, 1, 6, 3**)**

**>** head**(**vec1**)**

**[**1**]** 3 7 8 3 7 3

**>** length**(**vec1**)**

**[**1**]** 154

**>** alpha **=** .05

**>** mMean **=** 4.69

**>** mStandardDev **=** 2.60

**>** mtest **=** t.test**(**vec1,mu **=** 4.69, conf.level **=** alpha, alternative **=** "less"**)**

**>** print**(**mtest**)**

One Sample t**-**test

data**:** vec1

t **=** **-**0.0437, df **=** 153, p**-**value **=** 0.4826

alternative hypothesis**:** true mean is less than 4.69

5 percent confidence interval**:**

**-Inf** 4.371706

sample estimates**:**

mean of x

4.681818

p\_value **=** mtest**$**p.value

# pvalue = 0.4826 and alpha = .05 P-value is NOT less that alpha

print**(**"likely A Mourner was authored by John Hancock"**))**

So we cannot reject the null hypothesis, and say it is likely that A mourner could be written by John Hancock

3. In this problem, you will investigate whether sentences with more words tend to contain longer words.

1. In Homework 7, you used Python to count the number of words and mean length of words in each sentence of *Pride and Prejudice*.  Load this data set into R and make a scatterplot of mean length of words versus number of words per sentence.  Does linear regression appear appropriate here?

**>** plot**(**WordCount,Avg\_Word\_Len**)**

**>** word\_Test **=** read.csv**(**"C:/Users/pedbv9699/Documents/GitHub/ds710assignment7/1c.csv",header **=** T**)**

**>** attach**(**word\_Test**)**

**>** head**(**word\_Test**)**

WordCount Avg\_Word\_Len

1 27 6.148148

2 26 5.192308

3 1 9.000000

4 59 7.677966

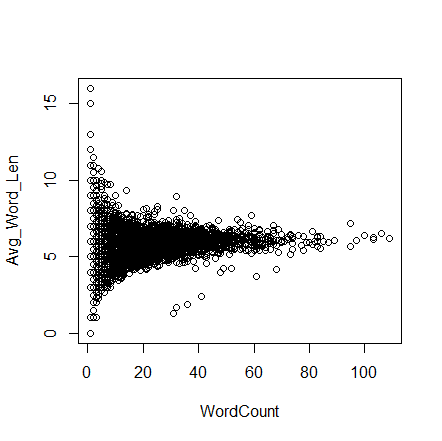
5 44 5.863636

6 3 4.333333

**>** plot**(**WordCount,Avg\_Word\_Len**)**

Null Hypothesis, there is no relation with average word length

Alternative, there is an association with average word length



**>** Word\_Count\_Model **=** lm**(**Avg\_Word\_Len **~** WordCount**)**

**>** Word\_Count\_Model

Call**:**

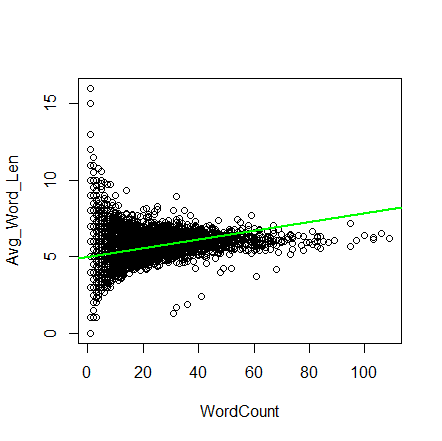
lm**(**formula **=** Avg\_Word\_Len **~** WordCount**)**

Coefficients**:**

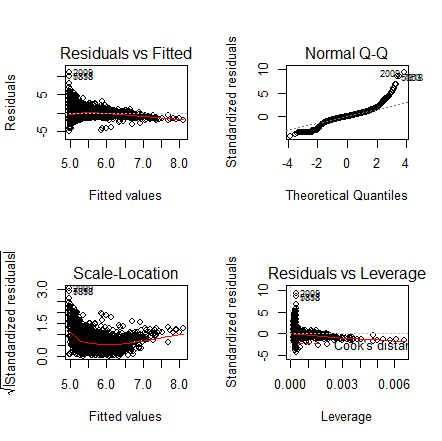
**(**Intercept**)** WordCount

4.95363 0.02877

**>** abline**(**Word\_Count\_Model , col **=** "green" ,lwd **=**2**)**



The line is pretty inclined, as word count increases so does avg word len. I am inclined to say there might be some association.



**>** summary**(**Word\_Count\_Model**)**

Call**:**

lm**(**formula **=** Avg\_Word\_Len **~** WordCount**)**

Residuals**:**

Min 1Q Median 3Q Max

**-**4.9824 **-**0.5428 0.0176 0.5955 11.0176

Coefficients**:**

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

**(**Intercept**)** 4.95363 0.02173 228.00 **<**2e**-**16 **\*\*\***

WordCount 0.02877 0.00102 28.19 **<**2e**-**16 **\*\*\***

**---**

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

Residual standard error**:** 1.18 on 7357 degrees of freedom

Multiple R**-**squared**:** 0.09751, Adjusted R**-**squared**:** 0.09739

F**-**statistic**:** 794.9 on 1 and 7357 DF, p**-**value**:** **<** 2.2e**-**16

The P-value being as small as it is, it is hard to not reject. A significance level of .05 and p val being 2.2e-16, we can reject the null hypothesis and say there is strong evidence to claim the association exists between wordCount and average word length

The residuals vs.fitted seems to show some good crossing of the cook’s distance line to the actual data points. It is not a bad fit.

Q-Q plot seems to be little less comforting at the corner end, the line sways off and begins to fade interms of parallel climb.

1. If your answer to part a was “no”, apply a transformation to make linear regression more appropriate.
2. Test whether sentences with more words tend to contain longer words.  State your conclusion in context.
3. Add a line to your scatterplot representing the regression model.  Write a sentence interpreting its slope.
4. Examine the residual diagnostic plots, and explain what they tell us in this case.

Submit a .doc, .docx, .rmd, or .pdf document containing your R code, plots, and interpretations.