***Determining and examining correlations between obesity rates in America and proximity to resources like grocery stores, parks and green spaces, centers for arts and culture, museums, shopping malls, etc.***

***Submitted by***

***Aleya Chowdhury***

***Abdolaziz Bozorgzadeh***

***Jane Yoon***

***Group Name: Make Data Mining Great Again***

***Software:***

Eclipse

Miniconda

Excel

RStudio

MatLab

***Languages:***

R

Python

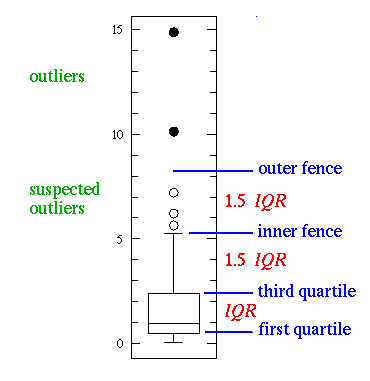
**Purpose:**

Obesity is correlated with health risk, according to data from National Health and Nutrition Examination Survey, In 2011–2014, the prevalence of obesity was just over 36% in adults and 17% in youth. Our purpose is to determine and examine the correlation between obesity rates in America and proximity resources like food environment factors- such as store or restaurant proximity, food prices, food and nutrition assistance programs and community characteristics which interact to influence food choices and diet quality.

**Outlier Detection:**

An outlier is a data object that deviates significantly from the rest of the objects, as if it were generated by a different mechanism. we may refer to data objects that are not outliers as “normal” or expected data. Similarly, we may refer to outliers as “abnormal” data. [1] To Calculate Outlier we use these formula

The lower fence is equal to the 1st quartile – IQR\*1.5. The upper fence is equal to the 3rd quartile + IQR\*1.5. We use excel to calculate outlier of different attributes from our data. Outlier detection is very significant to get correct correlation between obesity rates and proximity resources.



***Outlier detection***

For example, In our “Health” data set, for Per Capita Diabetics Adults 2009, we got upper fence 16.25 and lower fence 6.25. We deleted anything lower than 6.25 and anything higher than 16.25.

**Min-Max Normalization:**

Normalization is very important for data processing; we use Min-Max normalization technique to get correct correlation coefficient. To detect Min-Max normalization we code our data in Python. We use this following formula to find min-max normalization.

Screenshot of Min-Max normalization code:



In our data set we found some data are exact same for all county, we just deleted those data to do min-max normalization.

Screenshot of sample output of Min-Max normalization:



**KNN missing values:**

In our data according to the documentation of the source, blank and empty rows are considered missing values. So in order to perform normalization, there is need to identifying the missing values using the possible methods available to do so we need either with “ignoring or discarding data”, or “parameter estimation”, or “imputation”. Therefore, we have implemented KNN code in MatLab in to calculate the missing values with k=3. Since our data was big, it was required to split the columns by three and perform a KNN so it would reduce the percentage of error and the redundancy of the written code. The importance of the KNN implementation is that it preforms discriminant analysis where there are no reliable values for a given columns and rows. The logic that the code knn.m implementation follows is that first it sort distances from low to high and finds the three nearest neighbors “distances = sortrows(distances, n);”,when the distances are completely were found then it calculates the weights and sort the three original values with function “weight = (n/distn)/x;”, the program performs the same functions for all those columns that have those missing values. In our case it does it for three columns at a time since the data contains many columns that have different range values that might affect and distort the accuracy of the result.



**Pearson’s Correlation Coefficient**

The goal of this analysis is to determine the strength of the linear relationship between the following factors (by county):

x:

Percentage of obese adults in 2010,

Percentage of diabetic adults in 2010,

and percentage of obese children in 2011.

y:

Selected access attributes pertaining to the population’s ease of access to resources like grocery stores,

Selected assistance attributes pertaining to the amount of government supported programs that populations take advantage of,

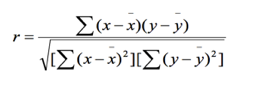
Selected prices/taxes attributes pertaining to the pricing of taxable goods, such as soda,

Selected restaurant attributes, for example the number of fast food restaurants per thousand people

Selected store attributes, for example the number of grocery stores per thousand people,

and socioeconomic attributes.

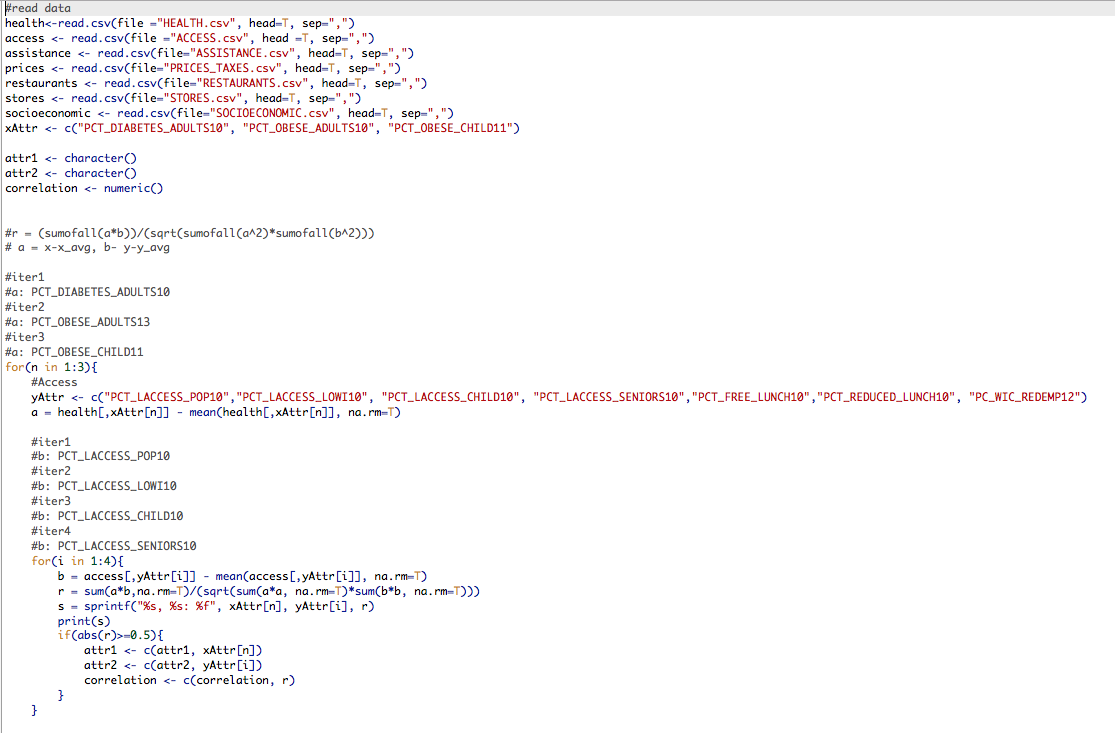
To determine these correlations, we iteratively calculate Pearson’s correlation coefficient between each pair of attributes (one x and one y attribute). The correlation coefficient (r) is calculated thusly:



where x is an x attribute for a specific county, x\_bar is the mean value of the same x attribute, y is a y attribute for a specific county, and y\_bar is the mean of the same y attribute.

The calculations were initially implemented using a Java program, but were translated to an R script later, since we could not reliably connect to the remote SQL server we set up. Also, R lends itself to database manipulation more naturally than Java; for example, it is possible to manipulate the values of an entire data structure in a single operation, rather than looping through the data structure’s values and updating them one-by-one.

Screenshot of the Pearson’s correlation coefficient code:

******

One limitation of this approach is that the correlation coefficient only discovers the *linear* correlations. It is possible that the relationships between attributes could be better described by a different function. Another limitation is that the correlations we discovered do not imply causality; a good example of this is the strong positive relationship we discovered between the percentage of people who take advantage of a free lunch program and the percentage of people who are obese. It is likely that there is some hidden attribute which is indirectly responsible for both (such as the poverty rate in the area), rather than the presence of one attribute being directly responsible for the other.

Output Analysis:

The Pearson correlation coefficient, r, can take a range of values from +1 to -1. A value of 0 indicates that there is no association between the two variables. A value greater than 0 indicates a positive association; that is, as the value of one variable increases, so does the value of the other variable. A value less than 0 indicates a negative association; that is, as the value of one variable increases, the value of the other variable decreases.



[***https://www.cdc.gov/nchs/data/databriefs/db219.pdf***](https://www.cdc.gov/nchs/data/databriefs/db219.pdf)

***Data Mining, Concepts and Techniques, Third Edition, Jiawei Han, Micheline Kamber, Jian Pei***