Capstone 1 Data Wrangling

First, I used pd.read\_excel() function to read in data from approximately 250 project proposals. Once they had been read, I selected the pertinent information from each proposal and assembled that information into one spreadsheet. Additionally, I set up code to add the relevant information from any future project proposals to the workbook.

Now that I had all my data in one sheet, I analyzed the data to see what needed to be cleaned. I initially reset the index, while dropping the existing one to more easily analyze the data in the future.

Looking at the ‘Vehicles Per Day in Front of Site’ column, it was easy to tell something must be done. While a human could read the data and understand its meaning relatively easily, a machine, especially with the intent of using a machine learning model, could not. I started by making the column data astype(str). Using regular expressions, I found numerical values in the data that were the amounts of vehicles, and not part of street names. Using that information, I summed up these numbers using lambdas and apply(sum) to have a single numerical value for the number of vehicles that passed by a given site per day.

Next, I dropped all the non-numerical data for ease of reading for the machine learning model. This involved dropping the first eight columns and an additional ‘Best Comp’ column.

Following this, any row where the ‘Accepted’ column equaled -1 was dropped. The -1 in this case indicated that the project had neither been accepted or rejected, these projects are projects still under review.

Knowing the self-storage industry, I created two new columns from the existing data that were likely to be of use to the model. These are the ‘Diff now’ and ‘Diff later’ columns. They represent the difference between the MSA SF per Capita and the corresponding existing SF per capita (now) and SF per capita including planned (later).

After this, I determined the amount of missing data in each column using the .info() function and saw that there was too much missing information on the asking price for each project. Thus the pricing data had to be dropped. The next column with a significant amount of missing data was the ‘Subject Rentable SF’ column. This column’s data was fairly consistent with few outliers, so it was determined that filling in the missing values could be acceptable. In the creation of the workbooks if the subject rentable SF is unknown it is filled in with 70000. Thus filling it in with the mean value of the ‘Subject Rentable SF’ column, which was very close to 70000 is viewed as acceptable.

After these fill ins, most of the data is available still and much cleaner than it was. Finally, I looked for outliers in the data. I found them using .describe() on the new dataframe. I found that the ‘Land Size (Acres)’, ‘Existing Sq. Ft. Per Capita’, ’Sq. Ft./Capita including Planned’, ‘Diff now’, and ‘Diff later’ have some clear outliers that are likely related to each other. Upon looking at the outliers they all seem to be consistent and not false or mistranslated data, they are extreme data points.

Now the dataframe is ready for EDA and potentially machine learning.