Data cleaning

There are two datasets in my capstone project 1: one is the water injection data in Oklahoma from 2006 and 2017 and the other one is seismic data in Oklahoma from 2000 to 2018. The objective is to find correlation between water injection and seismic activities in Oklahoma in the past decade.

Step 1: import data
These two data sets are from the website. I utilized request.get() command to obtain the data in Json type and converted into dataframe. One main issue what I encounter is that there is data limitation for each request from API and I was only able to obtain 2000 data points for each request. To tackle this issue, I utilized while loop and resultoffset command to obtain the full set of data.

Step 2: data cleaning
I utilized command info() and head(), columns to get a general idea of the data. I dropped unnecessary columns (for example, well names, well numbers) in the injection data as they are not needed in my analysis. I found that there are injection volumes for each year and also for each month from year 2006 and 2017, thus there are more than 100 columns. In my analysis, I would like to analyze the data on a yearly basis. As a result, I created a new injection data by selecting only the yearly volume. Similar issues occur to the seismic data as well. I also dropped some unnecessary columns as well. In addition, the time I initially imported are not in the right format, then I utilized to_datetime command to transform into the right time format. After that, I created a new column to collect the year information of seismic events based on the time.

Step 3: group data
This step is the most important steps in data cleaning. In these two data sets, there is location information for each injection activity and seismic activity by telling the latitude and longitude. As a result, each data points are scattered points in space. In order to find correlations, I have to group these scattered points into some regions. I first looked at the seismic map in Oklahoma and figured out that there are two main seismic activity regions and then measured the latitude and longitude of these two regions (In region one, latitude [36.25, 37], longitude [-99, -97.7]; and in region two, latitude [35.5, 36.6] and longitude [-97.7, -96.7]). Then I grouped the injection and seismic data points into groups by generation a new column named region based on the value of latitude and longitude: if the data is in region one, then the value is 1, if the data is in region two, then the value is 2, otherwise the data is NaN.

Step 4. Drop NaN values
I utilized info() command to check if there is any NaN values in the dataframe. I found that that the only NaN values occur in the column group. Then I dropped the NaN values by using the dropna function.