The first task we did in the clean-up of the ogs-oilandgas-well-locations.xlsx file was to separate GAS, GIW (gas injection well), WIW (water injection well), and BDW (brine disposal well) into different excel sheets for easier manipulation. We uploaded the file to the GitHub repository, but we realized some column titles were left out on some of the well types, which we promptly fixed. Some issues with the data itself was the location coordinates. There were two sets, one labeled “BH…” and the other labeled with “Wh…” or “WH…”. The “BH…” label turns out to most likely stand for bottom hole, but we still have no clue what Wh stands for. We ended up just getting rid of the “BH…” coordinates altogether because both coordinate labels were very close in value with each other, and most were identical.

One of our crew members found another data set containing possible all the wells in fifty states of the United States of America. This data set had to be split into seven csv files in order to fit onto GitHub. The files were named with this format: “wellsX.csv” where X was a number from one to seven. Three of the 7 file names were saved incorrectly at first, “wells5”, “wells6”, and “wells7” were spelled with capital “W”. We fixed that before working on cleaning up the data. The data was pretty much a mess. All the csv files had the same column titles, but each state had different phrasing for the type of wells and how the columns were filled out. Before we tackled getting most of the types to be phrased the same way, we got rid of some of the columns. We decided to remove two columns, “spud date”, and “API” column. For many of the data entrees, “spud date” was blank, so we decided that we shouldn’t use it. The “API” column we got rid of because we didn’t really need to know the unique identifying number of each of the wells, we were just interested in the locations of the wells.

For wells1 and wells2, we started getting rid of the entrees that didn’t have a type filled out. We stopped doing that once we realized some of the states used the “status” columns to label what type of well it was. Also, we could still use wells without a type, we would just have to label in either unknown or other type of well. We also decided to keep wells that were labeled abandoned in the “status” columns because abandoned wells were still dangerous to environment.

The main issue for cleaning up the data was to figure out what phrasing the states were using to describe the wells. At first we decided to use python to grab all the types and count them with a default dictionary, like we did with frequency calculations of articles. We were going to make a list of ones we combined. But what we ended up just doing, was to go through each of the csv files in excel and find what they used. We changed entrees that had types such as “OIL”, “oil”, “OIL WELL” or “Oil Well” to just “Oil Well”. Gas type wells were changed to “Gas Well”. To change these, we made use of the find and replace function in excel. Wells for disposing of salt water were labeled as “Brine Disposal Well”. Many were either labeled as “SWS”, “Salt Water”, et cet. Injection wells were either labeled as “Injection Well”, or “Water Injection Well”. We kept track of as many types as we could in a separate text file for referencing. For the states that used the status column as describing the well type, we manually moved the sections into the “type” column. We needed to clean up some of the types to easily pull them from the csv files into python for our map. We did not want to have 6 or 7 different things to search for, when we specifically wanted oil wells or gas wells.

One thing we realized we may need is the spud date. Not all states have them, so we will end up focusing on one particular state for this data.

Also, OK and TX were not showing up on the map we initially created. We were able to get OK on it once we changed our for-loop to include our final wells, wells7.csv, since it was not being read into it. TX was not in original data set and it apparently costs to get the data so we are letting TX bite the dust.

One of the things Ryan has been considering is using k means clustering or a similar argument to find some average points to represent many wells in a particular state. I was looking on some forums, and some of them pointed out k means clustering wasn’t good for distances with latitude and longitude. K means sort of worked, in the sense it gave some of those points, but I have been attempting to use DBSCAN, Ward’s Hierarchal Clustering, and mean shift ML techniques. These techniques are available on sklearn. DBSCAN and Ward’s Hierarchal Clustering both ended up crashing my laptop when using the full data set I have for Kansas. I eventually got DBSCAN to work and visually displayed it with the help of example code on sklearn. However, DBSCAN did not separate the wells into reasonable groups. Basically, it either had one cluster, which was the entirety of the wells, or had several clusters, where one was most of the wells. I believe this isn’t going to work because DBSCAN usually goes by finding groups that are grouped by different whitespace, which the wells are very dense.

We decided to focus on one state; this was to cut down on the amount of data we use in our programs. The first thing I (Ryan) did was to get all the Kansas data from the set that had spud date data into a separate csv file. In python, I separated the wells that had type of Oil Well, Gas Well, Brine Disposal Well, and Injection Well into separate lists of coordinate pair lists. I also made a list with all the wells.