Sarah Floris February 5th, 2017

Data Wrangling in My First Capstone Project

For my first capstone project, I am analyzing twitter's Search API and investigating the data obtained from this API in comparison to a Kaggle Dataset that describes a selection of video games, genres, ratings, and sales. Hopefully, I will find some unique trends and behaviors, providing insight into a video game's audience. The first step was to create a search path that allowed me to search specific hashtags that contained the names of the video games. I initially just started out doing one game, Battlefield 1. I used the Search API from Twitter which allowed me to search the tweets made in the last 30 days. However, when the data was obtained, it was in html and not quite readable. Thus, my second goal was to convert this not so readable HTML to JSON, so I can could look more specifically at the data that was available to me. After the conversion from HTML to Json using the Python's JSON library, I was able to flatten the multi-dimensional data frame into a 1D dataframe. Locations can show interesting trends such as where the videogames are most bought, who to market to, and so much more. Thus the main piece of data I was looking for was the location of the tweet and searching through the upper-level layer of data was unsuccessful; many coordinates and geolocations were set to False or unavailable. Still, I found out that the location of the user's twitter was indeed available. When I saw this data, I was thrilled and converted these locations to degrees, separating by latitude and longitude. These coordinates were then placed in my clean dataframe that I had created, and they allow me to easily graph a data frame unto a map, providing invaluable insight into the video game customer backgrounds. This data had quite a bit of none values so I used the pandas.dropna() function to clean it up. However, I soon realized when I use geocoders.google that it does not matter whether I have None or a user's location that is not an actual location (such as "world" or a hashtag as a location) because the geocoders.google library returns a None value if it is not able to get the city, state, or country. I was conveniently able to convert the addresses I had to coordinates by using google module from the geocoders package. I combined all this data within a clean dataframe with all of the values for easy graphing and analyzing.