Ameen Abdelghani

PUBG Data Cleaning Process

Dataset Description

My initial step to import pandas, matplotlib, numpy, and seaborn into the python notebook. Each of the 5 pubg csv files are around 2 gb. After loading each one individually, I begin exploring each of them trying to compare any similarities or differences, if there are missing values, any extreme outliers.

As I expected due to how the data is gathered from a videogame, and the dataset being from Kaggle, I learn that the files have zero missing values. There is no need to utilize fillna methods to replace missing values.

PUBG has two methods of playing the game, either in first-person, or third-person view. I did expect the files to contain data on both game modes, but after utilizing the *value_counts(dropna=False)* method, I learned that all 5 files only contain data on third-person view. For me personally this was disappointing since I primarily play first-person view.

Utilizing the *.info* and *.describe()* methods, I got a feel for the numbers of the data, and learned that each dataset contained about 11 million rows, with 15 columns. Every file was the same, but simply split up to reduce size.

Cleaning Process

Since I was dealing with somewhat large files especially, and when cleaning I wanted to only deal with information of interest, I needed to cut the data. I found that doing computations through Jupyter notebook could take time, even when working on a very powerful computer.

Once I loaded all of the files into a pandas dataframes, I cut them down to the 7 columns of interest I had. Once I did this to each file individually, I concat them into one dataframe, ignoring the index since it was irrelevant to the data. The reason I couldn't simply concat them and then cut the columns is because when I attempted this, I got a memory issue from Jupyter.

From here I narrowed down the dataframe to only include rows where player survival time was greater than 10 seconds, and distance walked is greater than 5 meters (determining what unit of measurement the file is using for distance and time required some analyzing, after experimenting with the numbers I believe it a fair assumption to say the units are seconds, for time, and meters, for distance). I did not wish to include players that barely moved or barely survived in my reports. Many of these players were likely away from the keyboard, or ‘afk’. And if they barely survived but a few seconds those players data wouldn’t help to answer the questions I seek to answer in my project.

Dividing The Data

For my analysis, I wanted to divide the data frames by three categories. PUBG has three main sub game modes: solo, duo, and squad. Squad modes are typically 4 players, but technically you can go in with less.

To divide the dataframe, I used a simple boolean structure, determine if the value under “party_size” is 1, 2, or 4, save them into the variable df_solo, df_duo, and df_squad respectively.