John Ortiz

Project Proposal

My question of interest stemmed from my love of videogames. I have played video games almost my entire life (earliest I remember was 7), my drive of competition with my brother pushed me further down that rabbit hole. In all my time gaming, not once have I ever asked which video games are most closely related to each other? Now that is a very broad question as there is a lot of video games out there and it seems that there are a lot of factors that could go into that. I have found a data set on video game sales that could be used to cluster videogames by multiple factors described below.

The dataset comes from Kaggle(Link Here) that originated from vgchartz.com. The dataset contains data of videogames ranks, genre, ESRB rating (Everyone, Teen, Mature, Etc.), platform, publisher, critics score, user scores, sales (NA-North America, PAL- Europe, JP-Japan, Global) and more. It contains a total of 23 columns and 37102 unique values. It seems to be a fairly large data that will definitely need some data cleaning: as I could already see some columns that will either be not useful or to sparse.

I intend to use the K-means algorithm to cluster the video games based on **genre**, **ESRB_Rating, Platform, publisher**, critic score, user scores(looked sparse at glance) , sales(NA, PAL, JP or Global), and year of release of the game. I bolded the four categorical variables that I would need to change into indicator variables, before I can even start K-means. I found a way to change them is by using pandas.get_dummies (). After handling these we will need to find a k-value, trying a couple different ones, than using the best for the rest of the time. On homework 3, of the K-means implementation I did not try to use hierarchical clustering to get k clusters from a subset of the data so I would like to try it with this implementation and compare it to K-means with random initialization. I am not sure what a hypothesis of a K-mean algorithm would look like, but I would guess it is about how many clusters we will use. I believe the true k number of clusters will be 4 and that if I compare the different clustering with random initialization and hierarchical clustering, they should have a positive adjusted_rand_score (from sklearn).

Aside: Please let me know if using the get_dummies() for the categorical variables is the right way to go.