**Capstone Project 1 Data Wrangling**

My data set was provided by KKBox and made available on its Kaggle competition page. The dataset consisted of csv files of 5 categories.

1. Train - has two columns: userid (MSNO), and binary is\_churn values.
2. Sample Submission - which is the test set in same format as the train and as the final submission should be.
3. Transactions - a log of user transactions which are all subscription purchases and include dates and types of plans bought.
4. User Logs - daily user logs describing listening behaviors of a user.
5. Members - basic user information such as gender,

I started by just opening up a script in the Spyder environment from Anaconda and using pd.read\_csv to convert the data into dataframes. I quickly realized that I faced a problem with the size of these files. The two largest files were the Transactions file at about 1.7 GB and the User Logs file at almost 30 GB. So I realized I could not keep all these in memory as dataframes at the same time. Fortunately the machine I’m using now does have enough memory to read the Transactions file as a dataframe with almost 22 million rows and using the nrows argument, the first 20 million rows of the User Logs file. After speaking with my mentor I realized that even if I can only just use a fraction of the user logs data I can still move forward and successfully complete my project.

Next I explored the dataframes for missing values and outliers. It turns out that the only NaN values were in the gender column of the members dataframe. In that column there are 4,429,505 NaN’s, 1,195,355 male, and 1,144,613 female. This leads me to believe that gender may not be a feature to be included in the model as their are more missing values than complete ones. Also I looked for outliers in the numerical columns using the .describe() dataframe method. The member age column had a min of -7000 and a max of 2015. Looking at the value counts of the age column shows that the most common age is reported as 0 with 4.5 million entries and there are many other impossible ages including +200 and negative values. Again this make me consider just not using age as a feature, as it would require additional filtering and cleaning and the majority of this data is nonsensical.

Since with the machine I’m using I can only keep a fraction of the user\_logs file in memory I also wanted to check if the fraction I’m using contained enough unique users to be viable for the machine learning algorithm I’m going to be using. Reading the first 20 million rows of the user\_logs csv resulted in 1.7 million unique users which would be about about 25% of 6.7 million users in the members data. Using the first 40 million rows resulted in 1.9 million unique users, and for the first 80 million rows I get 2.3 million uniques. Much more than that and I run out of memory. So doubling the memory requirement seems to yield marginal increments of data completeness and I should be successful only using the first 20-80 million rows of the user\_logs file.