Sabina Prochowski Introduction to Data Science – DS UA 112 Capstone Project

In this capstone project, I created a different Jupyter Notebook for each question which meant that I handled the data cleaning each time and differently depending on what part of the dataset the question was asking about (in other words, I did not do a strict cleaning to the entire dataset at once). I will briefly explain what I did for data cleaning/transformation and handling dimension reduction in each question. For question 1, I loaded the entire dataset into a variable data and cleaned it via data.dropna(subset=names)[names] (where names holds a list of the columns necessary for this question) so that this new dataset would only contain the sensation seeking and movie experience columns while also removing rows which contain missing values along that subset. Then, I found the PCA for sensation seeking and movie seeking separately, which I will discuss further in the report below. For question 2, I use the same method to only store the personality columns in a dataframe (also removing rows with missing values within all personality factors) and then performed a PCA. For question 3, I only used the subset of data including movies and did not need to utilize a PCA. For question 4, I also create a subset of data only including the columns for the Shrek movie and gender identity (no PCA since there is only 1 column for each variable). For question 5, I read in the entire dataset, but only kept   
The Lion King” column and only child column (also not needing a PCA for the same reason). For question 6, I also read in the entire dataset, but only kept the “Wolf of Wall Street” and social viewing preference columns (also not needing a PCA for the same reason). For question 7, I read in the entire dataset, but only included the movies from the different franchises listed in the question. For question 8, I use the personality factors in a dataframe and drop rows with missing values, and then conduct a PCA. For question 9, I filter out rows with missing values across gender identity, only child, and social viewing preference. For question 10, I separate the dataset to movies and factors. For the factors dataset, I clear out rows with missing values across all factors that are not movies. Then, I conduct a PCA on them.

For question 1, to find the relationship between sensation seeking and movie experience, I loaded in the dataset and only kept the sensation seeking and movie experience columns, from which I dropped the rows of missing values. Then, I conducted a PCA on sensation seeking that originally had 19 factors (using the z-score of the sensation subset of the cleaned data) and then found that after the dimension reduction, Kaiser criterion suggests 6 factors while elbow criterion suggests 1 factor. I also conducted PCA on movie experience, which originally had 9 movie experience factors (using the z-score of the movie experience subset of the cleaned data), and after the dimension reduction, Kaiser criterion suggested that there are 2 factors while the elbow method suggested only 1. To answer this question, I wanted to use a correlation matrix, but was initially unsure of how many factors to decide on for the sensation dimension reduction. I was first thinking of choosing 2 factors from sensation and 2 factors of the movie experience dimension reduction so that I can have a simple 4x4 matrix. However, after some thought and testing, I realized I would be losing a lot of the data from the sensation by only using 2 factors and in fact, 2 factors only account for 27.061% of the model’s variance (where the first factor accounts for 17.693% and the second accounts for 9.368%). 6 factors account for 53.803% percent of the sensation data variance which better aligns with the 2 movie experience factors that account for 50.875% of the movie experience variance and thus, would be a better fit. Also, intuitively speaking, sensation behaviors are plentiful and varying to mold to just 2 categories. Furthermore, I then store all 8 PCAs from both sensation seeking and movie experience into one dataframe and name them PCA\_E1, PCA\_E2, PCA\_S1, PCA\_S2 and so on where the letter after the underscore represents which set it’s coming from (\_E comes from the movie experience and \_S comes from sensation seeking) and the number after / at the end represents which principal component it is. Then, I run a correlation on the dataframe to get the 8x8 correlation matrix, which is shown below. I also use a heap map to better visualize what this means so that I can answer the question. Based on the heap map, I can conclude that there is no relationship between sensation seeking and movie experience because as seen by the scale, correlations generally fall below the 0.2 mark and near the 0.0 mark (besides the obvious diagonal which is a correlation on each value itself).

Table

Description automatically generatedChart

Description automatically generated

For question 2, to find evidence of personality types based on the data of the research participants, I found the PCA of personality types and then found clusters. Performing a PCA in this case makes sense as there are a total of 43 personality columns and doing a PCA before clustering is efficient as k-mean clustering is more efficient for lower dimensional data. So, first, I read in the dataset and only keep columns 421 to 462 holding the personality data and drop rows with missing values throughout. Then, using z-scored data of the personality data, I find the PCA using the same method provided in the code session and the Kaiser criterion finds there to be 8 factors while the elbow method finds 1 factor. I use the Kaiser approach because only 1 factor is not a fair measure of the data and realistically speaking, there is more than one personality type. I store the 8 factors in a np.column\_stack and use that to compute k-means using the code provided in the clustering code session. I plot a number of clusters vs sum of silhouette scores to determine the ideal fit of clusters. As seen below, there is a peak at 6 clusters, but when I run 6 clusters on my data, 2 of the points coincide with each other with another one that is also within close proximity to them. I decide on 3 clusters because the points are far apart and the sum of silhouette scores is much higher and even with 4 or 5 clusters, two of the points always end up right next to / on top of each other. Though the silhouette score for 2 clusters is highest, realistically, personalities tend to not be binary. Thus, below is the data represented by clusters. To answer the question, there are 3 personality types based on the clustering.

Chart, line chart

Description automatically generated Chart, scatter chart

Description automatically generated

For question 3, to find if movies that are more popular are rated highter than movies that are less popular, popularity was intrepreted as the number of ratings each movie received. This question does not require PCA since all movies are independent. Thus, the counts were found for all movies. Then, a median split was performed and movies with counts below the median were considered to be less popular and those with counts equaling or greater than the median counts (198) were considered more popular. So, less popular and more popular movies were placed into separate dataframes. Then, I found the mean of means for each, less popular movies and more popular movies. I find the mean of means because a singular mean gives the average rating for each movie in each dataset. The mean of means on both datasets returns the average rating for more popular movies and less popular movies. The more popular movies did indeed have a higher average rating, which was approximately 2.8708 whereas the average rating of the less popular movies was approximately 2.4009. To test if this difference was significant, an independent t-test (since these movies’ ratings do not depend on each other in any way) was used this way, ttest\_ind(lessPop.mean(), morePop.mean()) where lessPop.mean() is an array of all of the movie ratings for the less popular movies and morePop.mean() is an array of of all of the movie ratings for the more popular movies. The p-value is 1.17499e^-52, which is less than the standard alpha level of 0.05. Thus, the null hypothesis is rejected where the null hypothesis is that the means of both samples are the same. Since the result is statistically significant and we know that from our mean of means, the popular movies had a higher average rating than the less popular movies, we can conclude that movies that are more popular rated are higher than movies that are less popular. This can also be proven in another way with a visualization by a simple linear regression on the number of ratings vs the average rating of all movies. As seen below, there is a clear positive relationship between popularity and rating, which shows that the more ratings there are for a movie (the more popular), the higher the average rating there is.

Chart, line chart

Description automatically generated

For question 4, to find if “Shrek (2001)” is gendered, I create a dataframe only consisting of the “Shrek (2001)” column and gender identity column. Then, to remove missing values across, I do df[~np.isnan(df).any(axis=1)]. From there, I find the value counts of females and males in the dataset and separate the data of females and the data of males into separate datasets. Then, the mean of both datasets is found and I use Welch’s independent samples t-test. Since there are unequal sample sizes of 743 females to 241 males (after deleting rows with missing values), we can assume that there are unequal variances, which is why Welch’s independent samples t-test is used. To do Welch’s t-test, stats.ttest\_ind(female, male, equal\_var = False) is used and the p-value is found to be 0.2483 which is greater than the standard alpha value of 0.05. That means that the null hypothesis failed to be rejected in which this case would be that the means of females to males are equal or in other words, female and male viewers do not rate it differently. The actual means are in fact very close to each other where female’s average rating is 3.1555 and male’s average rating is 3.083. Thus, the answer to this question is that the enjoyment of “Shrek (2001)” is not gendered.

For question 5, to find if only children enjoy “The Lion King (1994)” more than people with siblings, I followed the same method as the prior question. I create a dataframe only consisting of “The Lion King (1994)” and sibship status while also removing missing values across (row-wise removal). Then, I found the value counts to be 776 with siblings and 151 only children, which means they are very disproportionate sample sizes, implying unequal variances. Then, I separate the data of the only children from the children with siblings. I then conduct Welch’s independent t-test and find the p-value to be 0.06103 which is greater than the standard alpha level of significant of 0.05, which means the null hypothesis (of the means being equal) cannot be rejected. So, there is no difference in if only children or people with siblings enjoy “The Lion King (1994)” more than the other. In this case, the mean rating of the only children is found to be 3.3477 and the mean rating of the siblings is 3.4820.

For question 6, to find if people who like to watch movies socially enjoy “The Wolf of Wall Street (2013)” more than those who prefer to watch them alone, I also use the same basic approach as 4 and 5, but notice that the sample sizes are much closer than the previous questions, which implies this might be a better fit interpretation. I create a dataframe only consisting of “The Wolf of Wall Street (2013)” column and viewing preference column while also performing a row-wise removal of missing values. Then, I find the value counts of those that enjoy movies best alone as 393 and those who prefer to watch them socially to be 270. However, though the sample sizes are still much closer than the questions previously, we cannot assume equal variance. Those, we use Welch’s t-test of independence and find the p-value to be approximately 0.1214, which is above the alpha level of significance of 0.05. Thus, we fail to reject the null hypothesis, which in this case is that the means between the two are the same. Thus, we can conclude that those who have watched “The Wolf Wall Street (2013)” socially enjoy it just the same as those who have watched it alone. The means, again, were close together where those who enjoyed the movie socially gave it an average rating of approximately 3.0333 and those who enjoyed it alone gave an average rating of about 3.1438.

For question 7, to measure inconsistent quality of the franchices as experienced by viewers, I compared the measures of dispersion. Thus, for example, for the Star War movies, I found the means of all 6 movies (parts I-VII excluding part III since it was not included in the dataset) and then found the standard deviation on top of those means to find the measure of dispersion, which was approximately 0.3072. For the rest of the franchises, I follow the same method. For the (four) Harry Potter movies, the standard deviation was approximately 0.0257. For the (three) Matrix movies, the standard deviation was approximately 0.2299. For the (four) Indiana Jones movies, the standard deviation was approximately 0.1792. For the (three) Jurassic Park movies, the standard deviation was approximately 0.1821. For the (three) Pirates of the Caribbean movies, the standard deviation was approximately 0.1105. For the (three) Toy Story movies, the standard deviation was approximately 0.0784. For the (three) Batman movies, the standard deviation was approximately 0.4516. As seen in the bar graph below, Harry Potter was the most consistent along all of its movies, meaning the average rating had little variation across Harry Potter movies. The bar graph then shows that the most consistent to least consistent franchises are as follows, Harry Potter, Toy Story, Pirates of Caribbean, Indiana Jones, Jurassic Park, Matrix, Star Wars, and Batman. Even though Batman would be considered the least consistent, about half a rating is not an extreme difference for the rating distribution of Batman movies. In fact, it would account for 10% variation of a total rating, since the ratings of each movie are rated 1-5. These results matched my expectation because realistically, those who watch all movies in a franchise decide to keep watching because they are enjoying the series and thus, the ratings remain more consistent. Unfortunately, there is no precise measure of what would be considered inconsistent. Thus, the bar graph shows the most inconsitent to most consistent quality of franchises neatly in descending order.

Chart, bar chart

Description automatically generated

For question 8, I used a random forest as a prediction model to predict movie ratings for all 400 movies from personality factors only. First, I do a row-wise removal of missing values for all personality factors. Then, I conduct a PCA on the personality factors that originally had 43 columns (using its z-score data conversion) and then found that after the dimension reduction, Kaiser criterion suggests 8 factors while elbow criterion suggests 1 factor. I decide to use to 8 factors (since personality would be better categorized into 8 groups rather than 1 since realistically, there is more than 1 personality type) and put all 8 PCAs into a dataframe which is used for the random forest. Please refer to the code file named ‘Capstone Q8’ I attached to follow along. I begin with a for loop that iterates 400 times (because there are 400 movies) in which one at a time, stores a dataset including the 8 personality PCAs and all ratings for a single movie while conducting a row-wise removal each time specific to that movie and the PCAs. (rather than doing a row-wise removal across all factors and all movies because that would only leave 2 people who rated all movies, certainly creating a poor model when the dataset includes 1097 people). Then, I set x as the 8 PCAs and y as the movie’s actual ratings and its predicted ratings. To avoid overfitting the model, I use cross-validation. I set it to the standard so that 30% of the data will be used for the test set and 70% for the training set. For the cross-validation of the random forest, I set the estimators too start at 200 and go up to 2000 with an interval of 10, the max features to consider at every split as the default, the max depth / levels of the tree to start at 10 and go up to 110 with an interval of 11. Then, I also set the minimum number of samples required to split a node, the minimum number of samples required at each leaf node, and the method of selecting samples for each tree, all to default values. Then, I use all of those values for the random grid. The model makes models based on the multiple models set by the cross validation to find the best model with the lowest root mean squared error. I collect all of the best root mean squared errors across all movies in a list and find the average and thus, the accuracy of the model to be approximately 1.106. I also conduct a linear regression using the same parameters mentioned above, but find the root mean square error to be slightly worse with an error of 1.1162. A random forest is preffered because it is a better algorithm with more hyperparameters (where each tree has a specific number of splits and leafs in this case to determine the ratings rather than a best fit line with so many multivariate variables that tends to consequentially, be more biased). In each loop, as mentioned before, I store the dataframe, which I include examples of below.

Table

Description automatically generated

Above is an example dataset stored for the movie, Twister along with its predicted values given by the random forest model.

Text

Description automatically generated with medium confidence

Above is the first four outputs out of 400 movies for the random forest. Run the code in the files I attached. Warning, it does take several minutes to load.

Inititally, for question 9, I used a random forest as a prediction model to predict movie ratings for all 400 movies from gender identity, sibship status, and social viewing preference. First, I do a row-wise removal of missing values for all three factors. No PCA is needed here as there is only one column for each of the 3 factors. Please refer to the code file named ‘Capstone Q9’ I attached to follow along to. I follow the same procedure for this question as the one previously where I begin with loop that iterates 400 times in which one at a time, stores a dataset including the 3 factors and all ratings for a single movie while conducting a row-wise removal each time specific to that movie and the factors. I set x as the three column factors and y as the movie’s actual ratings and its predicted ratings. Then, I apply the same cross validation method from the previous question and find the root mean square error (RMSE), also defined as the accuracy of the model, to be approximately 1.0669. However, after running a linear regression using the same method from the previous question, the accuracy of the model given by RMSE is 1.0619. Thus, the linear regression is a better choice, but only, slightly. However, observe that regardless of whether we utilize linear regression or random forest, these prediction models are better than the prediction models in question 8. This is expected because there were only 3 factors here rather than a PCA done on 43 columns to narrow down to 8 factors previously. In other words, the impact of gender identity, sibship status, and social viewing preference has less variation in terms of how people rate movies in comparison to personality features that creates more variation. Thus, to answer this question, though both prediction models have a very close accuracy, the linear regression has better accuracy. Below is a sample output of the first 4 out of the 400 movies for the linear regression.

Text

Description automatically generated

Since the linear regression saves each dataframe and names it with dataset\_ followed by the movie title, the specific movie’s predicted values can be seen. For example, to see dataset\_TheFastandtheFurious, we find the following.

Table

Description automatically generated

For question 10, to build a prediction model to predict movie ratings for all 400 movies from all available factors that are not movie ratings (columns 401-477), I use a random forest, again. This time, I store all factors from 401-477 in a dataset while performing a row-wise removal. A PCA is needed here because there are 77 factors to take into consideration. After conducting a PCA, there are 18 factors as given by the Kaiser criterion. Then, running the random forest with the same cross validation techniques as questions 8 and 9, while just changing x to the 18 PCAs and y as the movie’s actual ratings and its predicted ratings. Then, I find the average RMSE for all movies which shows the accuracy of the model to be approximately 1.0852. I did not expect this result because this accuracy claims to be better than that of the random forest model in question 8 which has a RMSE of 1.106 even though there were 18 PCAs used here and 8 used in number 8. My speculation was that the row-wise removal of missing values across 77 factors for this question would make it more inaccurate since a row-wise removal of a wider set of data would normally shrink the dataset more, making it more inaccurate. I also ran a linear regresstion to compare accuracies to determine whether a random forest or linear regression is a better fit for this prediction and find the accuracy of the model to be much worse with a RMSE of 1.2042. This matched the previous expectation I had because this model had the worst RMSE out of all models used in questions 8 to 10. Thus, to answer this question directly, since ideally, you want to your RMSE to be minimized, the random forest is the best prediction model for this prediction. Please refer to the code attached named “Capstone Q10” to better understand my approach.

Text

Description automatically generated

Above, we can see the first 4 yields of the 400 movie predictions given by a random forest.

Table

Description automatically generated

Above, we can see the dataframe for the movie “The Mummy Returns” and its predicted values.

For the extra credit, I was interested in finding if being an only child makes you more likely to want to watch movies alone or in the other case, if having siblings would make you more likely to enjoy watching movies with others. To find this, I created a dataframe containing the sibship status and social viewing preference columns. A row-wise removal of missing values using drop.na was not necessary here (there were no rows with missing values) as the fields with those who did not respond were marked as “-1” so I dropped all rows across that had “-1.” For this problem, I used matthew’s correlation coefficient since both of the factors only had binary outcomes. The matthew’s correlation coefficient is specifically used to measure correlation between two binary variables, which is why I utilized it for this problem rather than pearson’s correlation coefficient. To clarify, sibship status only identifies if the rater is an only child or not (has siblings) and the social viewing preference viewing only identifies if the rater prefer to watch movies alone or not (socially). Thus, I used matthews\_corrcoef from sklearn.metrics and found the coefficient to be 0.0995. Since this correlation is fairly close to 0 (where -1 represents an inverse prediction, 0 indicates an average random prediction, and 1 indicated a perfect prediction), there is no obvious relationship if any, or a (very) slightly positive one indicating that only children are more likely to enjoy movies alone.