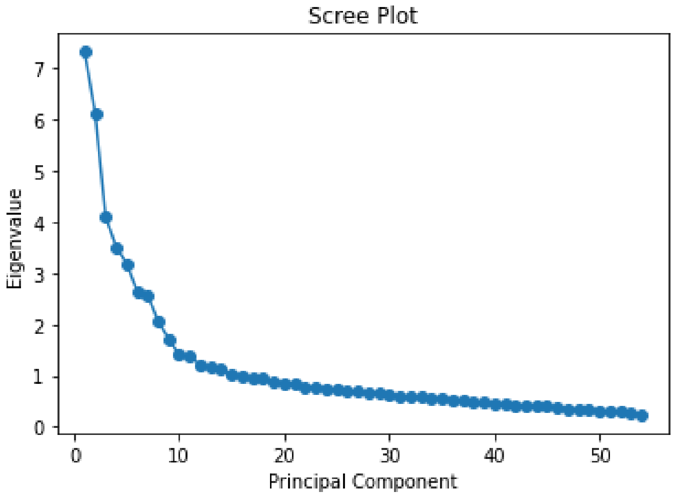
Jason Wang jw7383

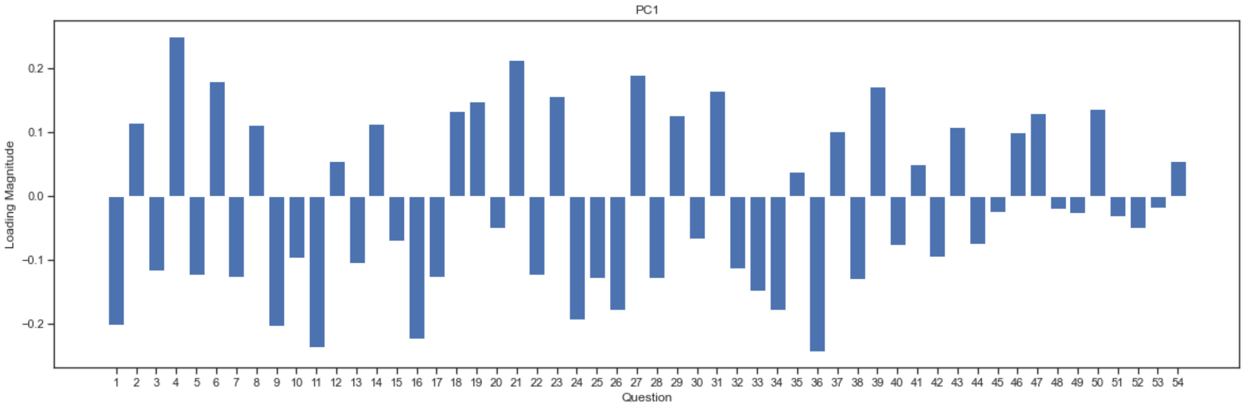
For project 3, we were asked to analyze the movie ratings dataset using machine learning methods. In this report, I will detail my approach for the first three parts of the project.

(Q1) First I applied a PCA on the 54 personality features in columns 421-474 to reduce the dimension of the dataset. To prepare the data for PCA, I first filled all NA's with the column median (since median is more appropriate than mean as a measure of central tendency for ordinal data) and then normalized the data since PCA expects normalized data. I then applied a PCA and plotted the corresponding scree plot as shown below:

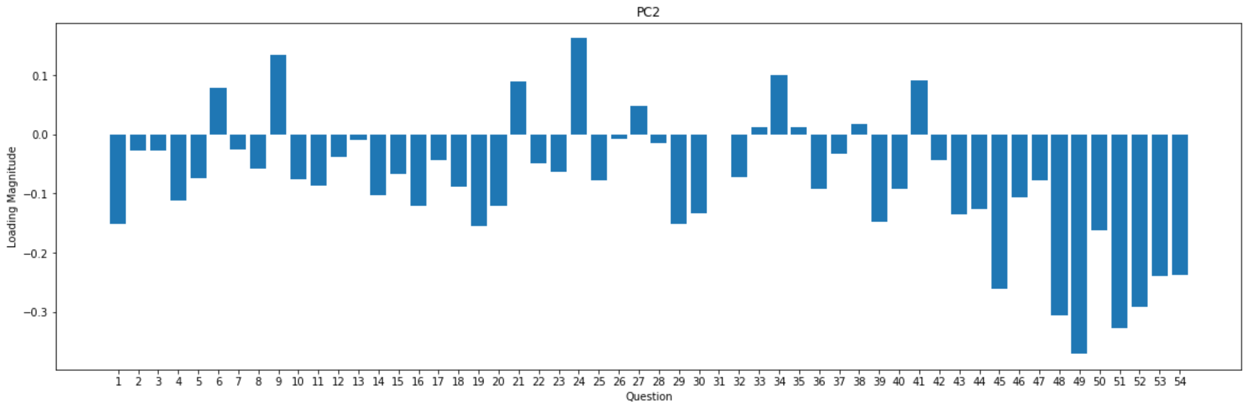


This scree plot plots the eigenvalues, which is equal to the explained variance, in descending order. I used the elbow criterion to decide to keep 2 principal components, noticing that there was a big drop off in the eigenvalue's magnitude after the second principal component. The elbow criterion is named as such because I chose to keep all the principal components before the "elbow" at the 3rd principal component.

I then plotted the loadings matrix for the first two principal columns, as shown below.

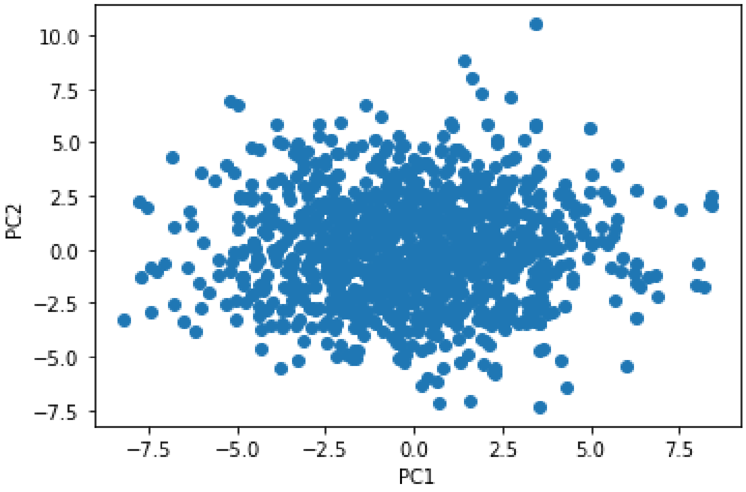


The first principal component has the largest positive loadings at questions 4 and 21, which are the questions "is depressed/blue" and "tends to be quiet" and the largest negative loadings at questions 11 and 36, which are the questions "is full of energy" and "is outgoing/sociable." Therefore, it is reasonable to infer that the first principal component is measuring how introverted an individual is.

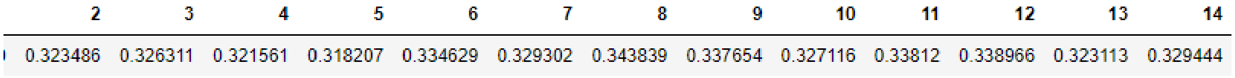


The second principal component all have very negative loadings for questions 45, 48, 49, 51, 52, 53, and 54, which are all questions related to how effected an individual is by movies. Therefore, it is reasonable to infer that the second principal component is measuring individuals who are not very interested or impacted by movies.

(Q2) Since PCA is simply an orthogonal transformation of the data, I can now plot the data in the new coordinate system. Since I am only choosing 2 principal components, the x-axis represents the datapoint in terms of the first principal component and the y-axis represents the datapoint in terms of the second principal component. The plot is below.



(Q3) Although there doesn't initially appear to be clusters, we can perform k-means clustering to partition the users by their survey answers. To do this, I first needed to determine the optimal number of clusters. I did this by computing the silhouette score from 2 to 14 clusters. The table of scores is below:



2	3	4	5	6	7	8	9	10	11	12	13	14
0.323486	0.326311	0.321561	0.318207	0.334629	0.329302	0.343839	0.337654	0.327116	0.33812	0.338966	0.323113	0.329444

Since the maximum silhouette score occurs at 8 clusters, that is our optimal number of clusters. `kMeans.labels_` then identifies which cluster each user belongs to.