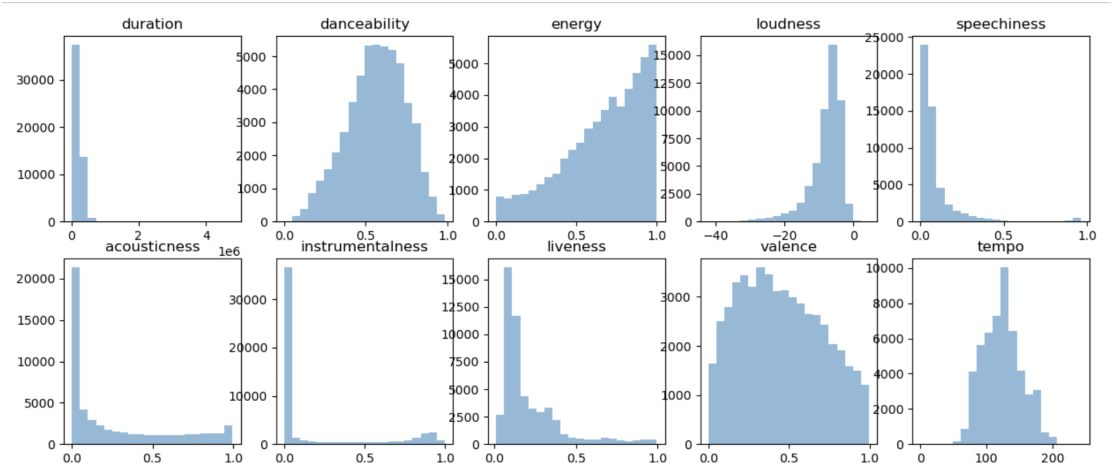
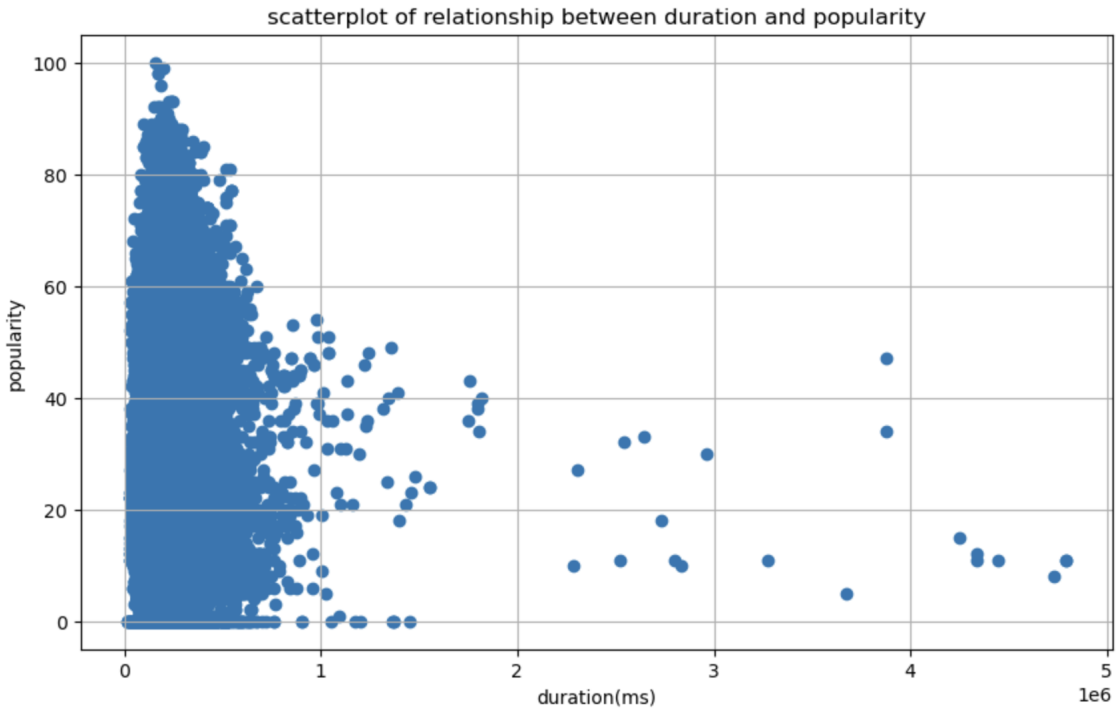
- Before I do the following questions, I seed the random number generator with my N-number and visualize the dataset first.
- Q1: for this question, I use features as my variable to store all the 10 features. Create a subplot with size (15, 6) and 2 x 5 frames. Use a loop to go through all the features. Axs[row, col] locates each feature to each subplot and .hist(data[feature], bins = 20, alpha = 0.5) to create histogram with 20 bins in each plot. Set the title to the feature name. It seems like none of the distributions is actually normal.

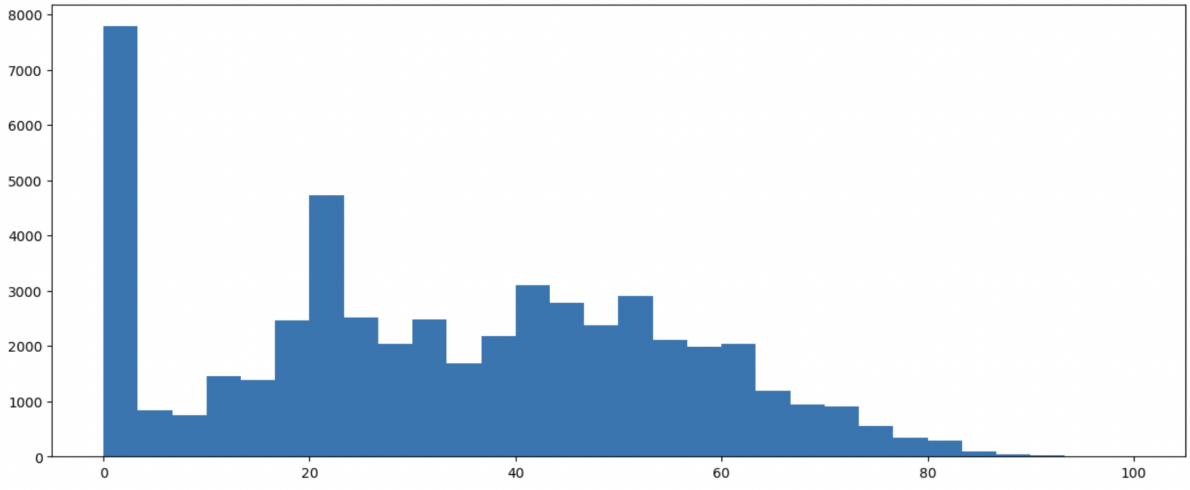


- Q2: create a scatter plot with figsize of (10, 6). On the x-axis is the duration in ms and the y-axis is the popularity. Given the popularity, it seems like there's no particular relationship between the song length and the popularity. Also, given the correlation coefficient that we calculated(-0.05), the relationship between song length and popularity

is not that strong.



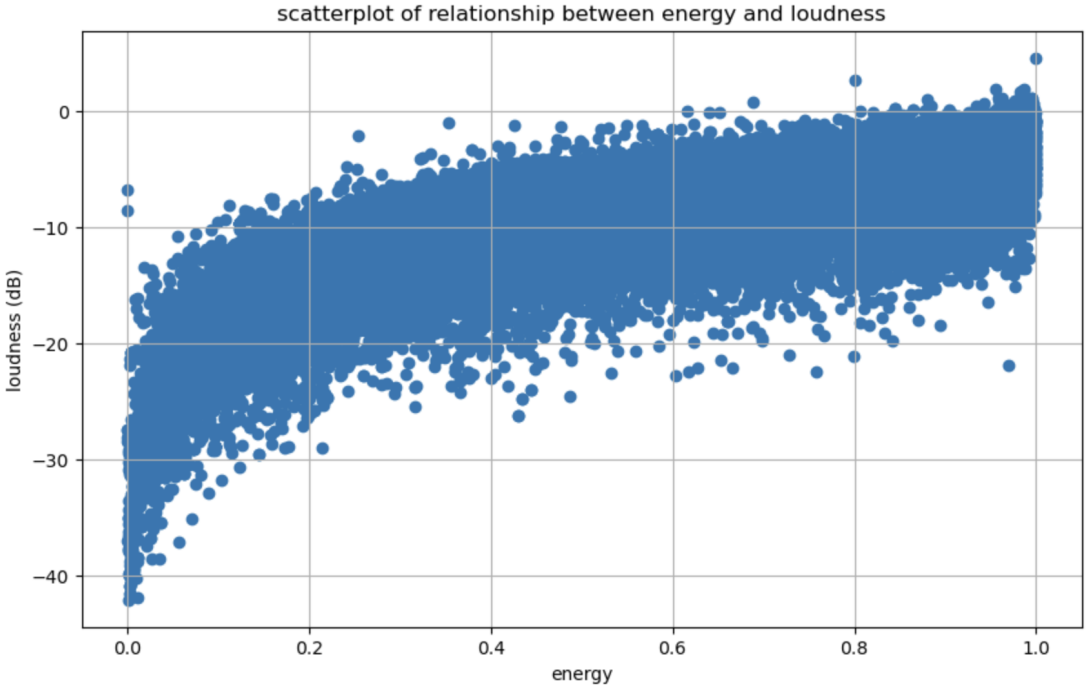
- Q3: Before I decided which significance test to use, I used a histogram to see the distribution of popularity.



Then I decided to use The Mann-Whitney U test to find the difference between explicitly rated songs and songs that are not explicit. Do the test using the existing library and set

the alpha to be 0.05. We found out that the p-value is 3.0679199339114678e-19, which is smaller than our threshold. Then we concluded that Explicit songs are more popular.

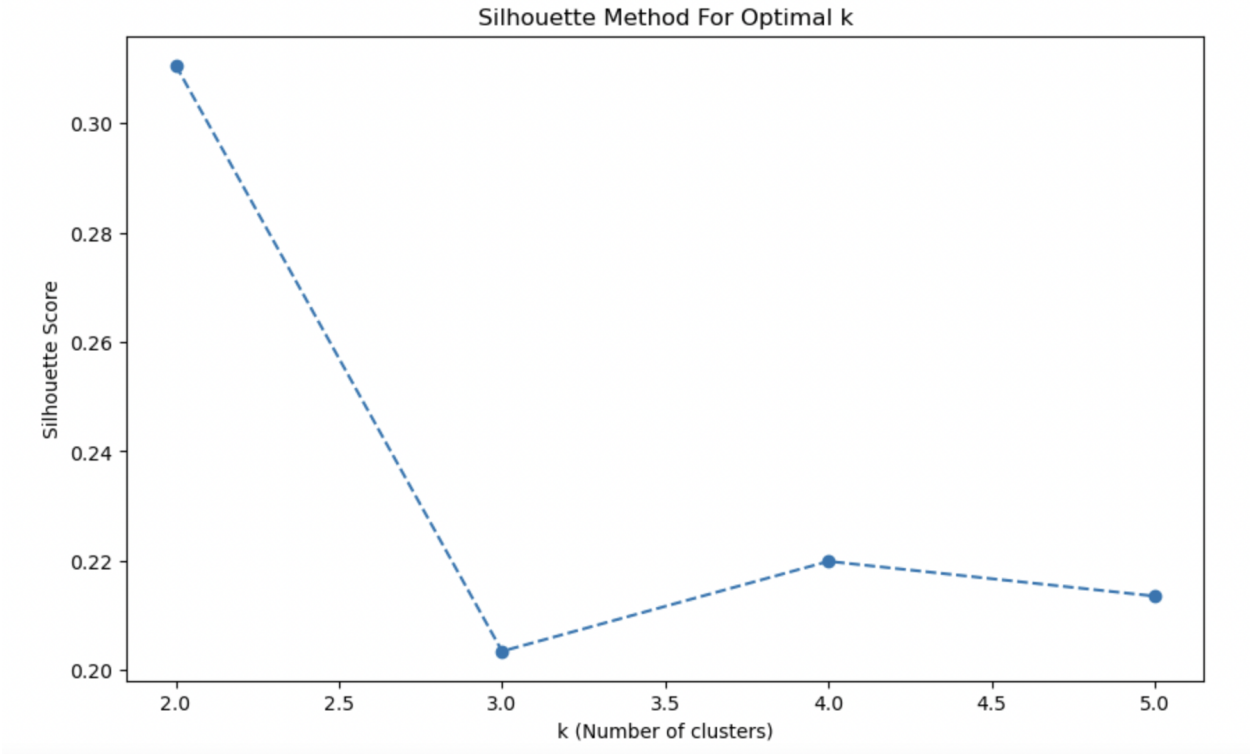
- Q4: We do the same thing again as question 3. Then we found out that p-value is equal to 2.0175287554899416e-06, which can conclude that Minor songs are more popular.
- Q5: Using the scatter plot, we see that energy and loudness have an increasing monotonic relationship. Also, the correlation coefficient between energy and loudness is 0.77, which implies energy can potentially reflect loudness of the song(strong positive correlation exists).

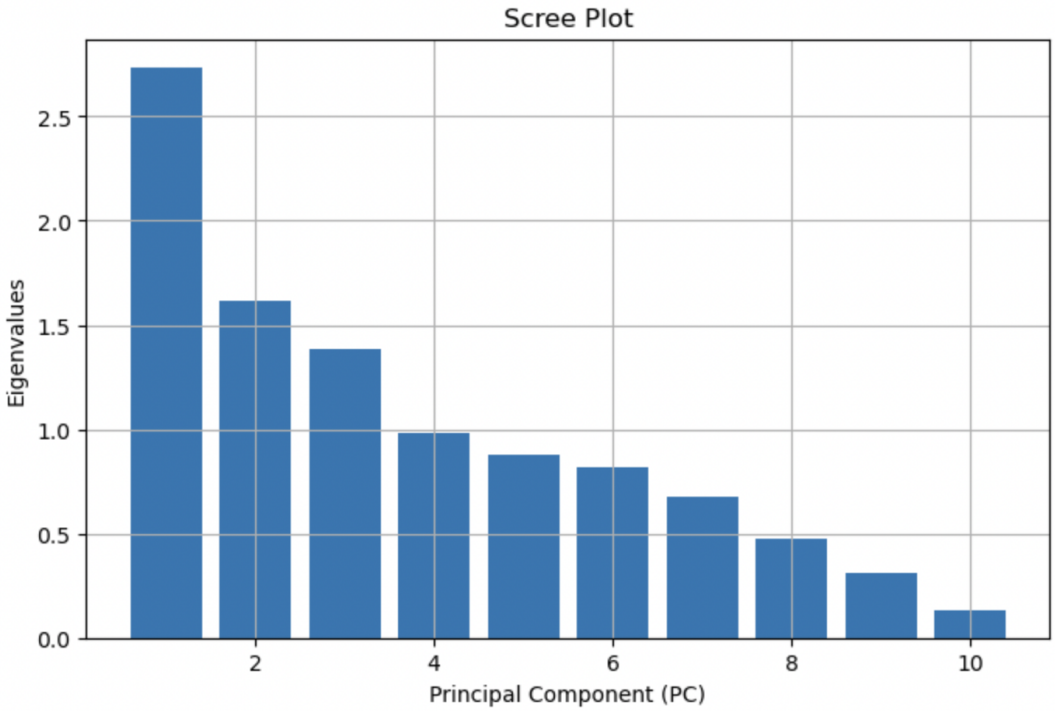


- Q6: Use the for loop to test each feature on popularity. For each loop, we do cross-validation by using the train_test_split to split the data into training and test sets. Use linear regression to predict the outcome and compute mean squared error for each feature. Get back the best result by finding features with lowest mean squared error. The feature with lowest mean squared error and highest correlation coefficient is

instrumentalness and the value is -0.1449 and 468.4163913465182 respectively, which is not a good model since the typical value is about 0-100.

- Q7: Now, instead of linear regression, I used random forest to use all the features to predict popularity. Turns out that the mean squared error is 279.2538968221127, which is slightly better than linear regression(468.4163913465182).
- Q8: We first scaled the data using StandardScaler. Perform PCA and display the scree plot. I extracted 6 principal components that accounted for 90% of variance. Then I transformed the data with 6 principal components. I then used the transformed data to perform k-means with k equals from 2 to 6. Record the silhouette score for each k-means and make a plot for it. Based on the plot, I concluded that k is equal to 2(2 clusters).





- Q9: I used logistic regression and calculated the R squared is roughly equal to-0.62, which I think it's a pretty bad model.
- Q10: I first used the library, LabelEncoder, to map the genre labels to numeric labels. Then I train the data with decision trees and report the class report.