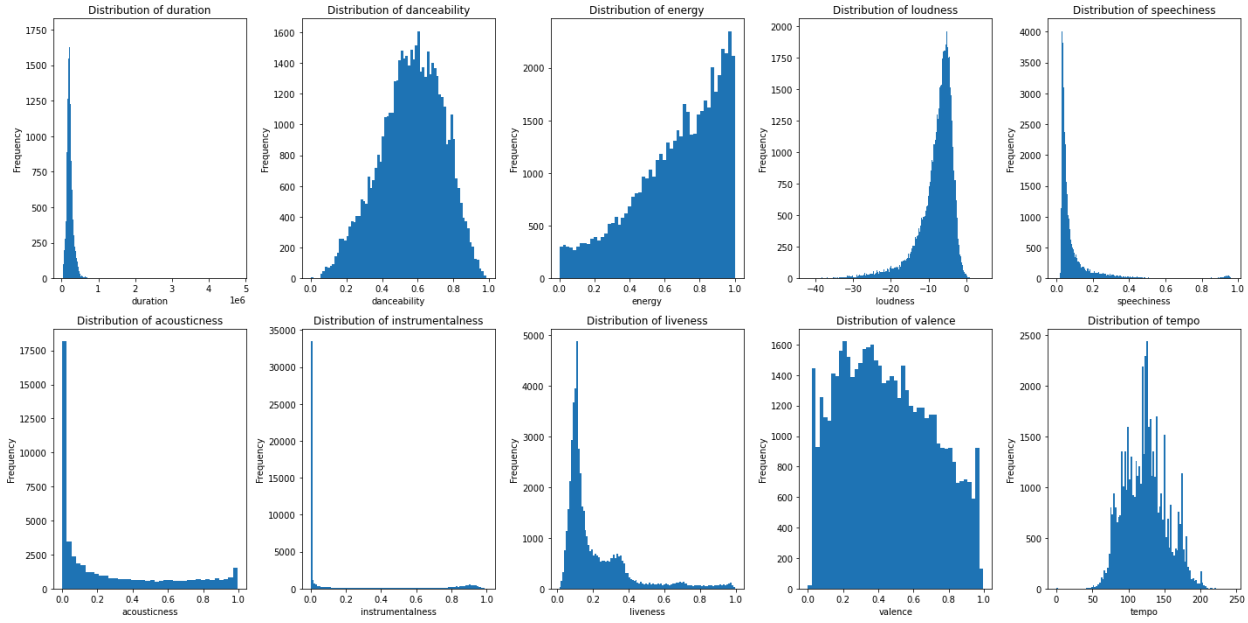
Capstone Project

For preprocessing, I decided to fill in all missing data values with either the column median or mode. I first separated the data to categorical features and numerical features, and filled in the missing values with the feature mode or median, respectively. I decided to use the median instead of the mean for the numerical columns because I thought that it would representent the data distribution better since all of them are not normally distributed (I looked ahead at Question 1). I also used my N-number to set the random seed of my code, N-17226773, and used it to set the random_state when splitting into training and testing groups as according to the instructions.

```
random.seed(17226773)

data = pd.read_csv('spotify52kData.csv')

## Seperating numerical and categorical data
numericals = data.select_dtypes(include=['int64','float64']).columns
categoricals = data.select_dtypes(include=['bool','object']).columns

## Fill missing values with mean
data[numericals] = data[numericals].fillna(data[numericals].median())

## Fill missing values with mode
for i in categoricals:
    mode = data[i].mode()[0]
    data[i] = data[i].fillna(mode)
```

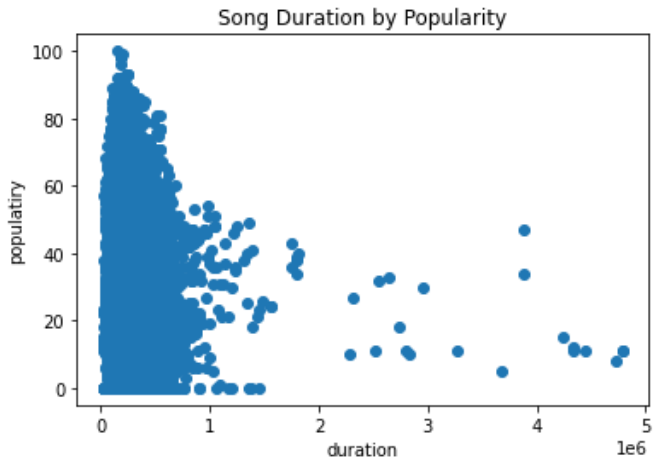
1. To visualize each of the distributions mentioned in the instructions, I made a 2x5 figure with histograms for each feature using matplotlib. The x-axis is each feature and the y-axis is its frequency. None of the graphs appeared to be normally distributed and when using a Kolmogorov–Smirnov test (which tests how similar the underlying distributions are) between each distribution and a standard normal distribution, I got extremely small p-values, less than

0.05, and concluded that none of the distributions were approximately normal.




```
Question 1:
KS test p-value for duration distribution:  0.0
KS test p-value for danceability distribution:  0.0
KS test p-value for energy distribution:  0.0
KS test p-value for loudness distribution:  0.0
KS test p-value for speechiness distribution:  0.0
KS test p-value for acousticness distribution:  0.0
KS test p-value for instrumentalness distribution:  0.0
KS test p-value for liveness distribution:  0.0
KS test p-value for valence distribution:  0.0
KS test p-value for tempo distribution:  0.0
```

2. As according to the instructions, I included a scatterplot between ‘duration’ and ‘popularity’. Just by looking at the plot, I was unsure if there was a correlation between them. I calculated both the Pearson and Spearman correlation coefficient and got r = -0.055 and ρ = -0.037, both extremely small values for correlation coefficients. It can be concluded that there is a little to no negative correlation between song duration and popularity.




```
Question 2:
Pearson's r:  -0.054651195936375914
Spearman's rho:  -0.03728567620648788
```

3. To test if explicitly rated songs are more popular than songs that are not explicit, I decided to use a Mann-Whitney test, because ‘popularity’ isn’t normally distributed and we have two groups, explicit and not explicit, that are independent. I first extracted the ‘explicit’ and ‘popularity’ columns from the data and separated the ‘popularity’ column into two groups, whether the song was explicit or not. Using the test from scipy, I tested the two and got a p-value of 3.06e-19, which is less than 0.05. Therefore, we reject the null hypothesis and conclude that there is a difference (with respect to the median) between the two groups. Looking at the median popularity rating for explicit and not explicit songs, I concluded that explicitly rated songs are more popular than songs that are not explicit.

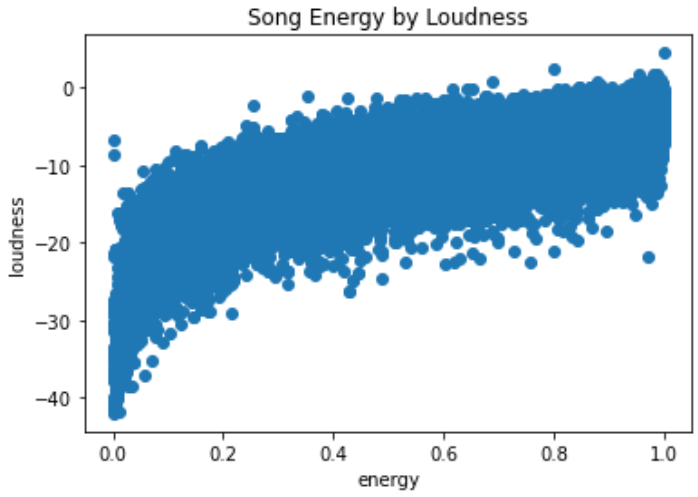
```
Question 3:
p-value:  3.0679199339114678e-19
Explicit Songs Median: 34.0
Not explicit Songs Median: 33.0
```

4. For the same reasons as Question 3, we can use the Mann-Whitney test to determine whether songs in the major key are more popular than songs in the minor key. I extracted the ‘mode’ and ‘popularity’ columns from the data and separated the popularied into two, whether the song was major or minor, ‘1’ meaning the song was in the major key and ‘0’ meaning the song was in the

minor key. Using the test from scipy, I tested the two and got a p-value of 2.018e-19. Therefore, we reject the null hypothesis and conclude that there is a difference (with respect to the median) between the two groups. Looking at the median popular rating for major and minor songs, I concluded that songs in the minor key are more popular than songs in the major key.

```
Question 4:
p-value:  2.0175287554899416e-06
Major Key Songs Median: 32.0
Minor Key Songs Median: 34.0
```

5. To see whether 'energy' largely reflects the 'loudness' of a song, I checked the correlation between the two. When visualizing the scatterplot of the two features, I wasn't completely sure if the relationship appeared linear or not. As a result, I decided to check both the Pearson and Spearman correlation coefficient and got r = 0.77 and ρ = 0.73. With this coefficient, I can conclude that there is a relatively strong positive correlation between song energy and loudness and can substantiate the claim made.




```
Question 5:
Pearson's rho:  0.7748808291850275
Spearman's rho:  0.7306382054765808
```

6. To see which feature mentioned in Question 1 is best at predicting 'popularity' the best, I decided to run many simple linear regression models between each feature and 'popularity' and

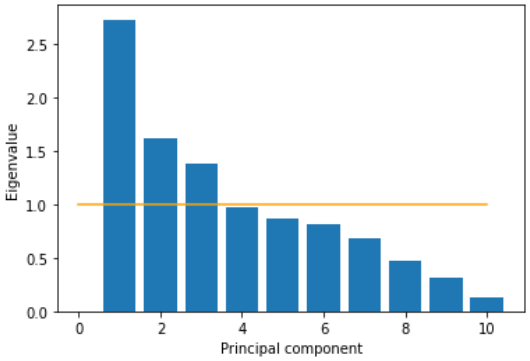
compare each of their R^2 values. R^2 is the coefficient of determination which indicates how well the feature explains the variability of song popularity, meaning the greatest R^2 value would predict song popularity the best. When doing so, I got that instrumentalness was the best feature to predict song popularity with R^2 value of 0.021. However, this value is extremely low for a coefficient of determination value, as R^2 ranges between 0 to 1, which means that instrumentalness (alone) is not a good predictor for 'popularity' in general. This is clear since, when interpreted, only about 2.1% of the variability in popularity can be explained by the variability in instrumentalness.

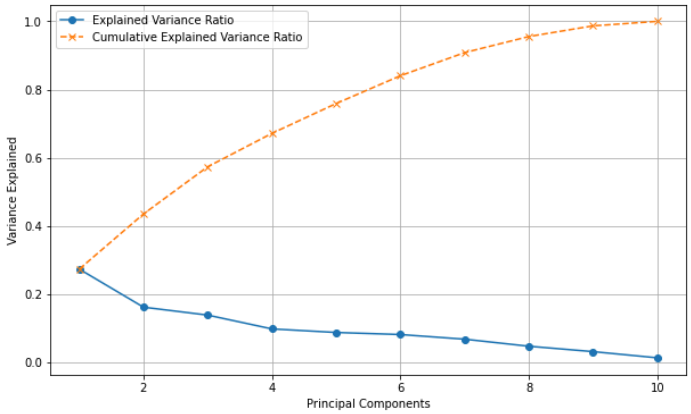
```
Question 6
r-Squared predicted on duration:  0.00298675321727615
r-Squared predicted on danceability:  0.0013807029448721364
r-Squared predicted on energy:  0.0031275710260059153
r-Squared predicted on loudness:  0.0036252482924016283
r-Squared predicted on speechiness:  0.0023554207449602016
r-Squared predicted on acousticness:  0.0006881908421283445
r-Squared predicted on instrumentalness:  0.021016959224749554
r-Squared predicted on liveness:  0.001922474338847624
r-Squared predicted on valence:  0.0012794062738639145
r-Squared predicted on tempo:  6.926535337070661e-06

Best Feature: instrumentalness with R-Squared:  0.021016959224749554
```

7. In order to use all the features from Question 1, I built a multiple linear regression model and set the independent variables as all the features and the dependent variable as 'popularity'. In doing so, I got a R^2 value of about 0.048. This value can also account for model improvement, as from the simple regression models from Question 6, the best coefficient of determination value was 0.021. This means that this model had a 2.7% increase in variance accounted for compared to the model that only uses instrumentalness to predict popularity. Again however, this value is still an extremely low R^2 value which still doesn't make it a good predictor of song popularity.

```
Question 7
r_squared:  0.04767961428671152 5
```

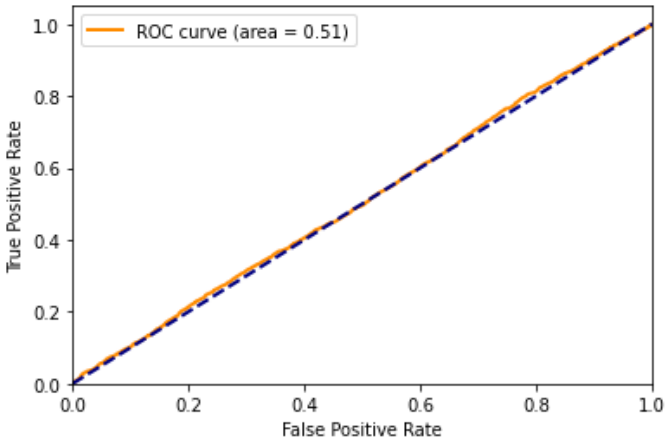
8. To find how many meaningful principle competents I can extract from the features from Question 1, I need to do principal component analysis. I first standardized the features using stats.zscore and then fit it on the PCA model from sklearn. When looking at the screeplot I created, I first decided to use the Kaiser criterion to determine how many factors were meaningful, which gave me the first three factors. However, the explained variance of the three factors was only 0.57 when calculating the cumulative explained variance and since we are not worried about computational efficiency of interpretability for this project, I then decided to use the factors that account for >90% of the eigensum to get more relevant features. Therefore, to answer the question, I extracted 7 meaningful principle components that accounted for 90.84% of the variance.





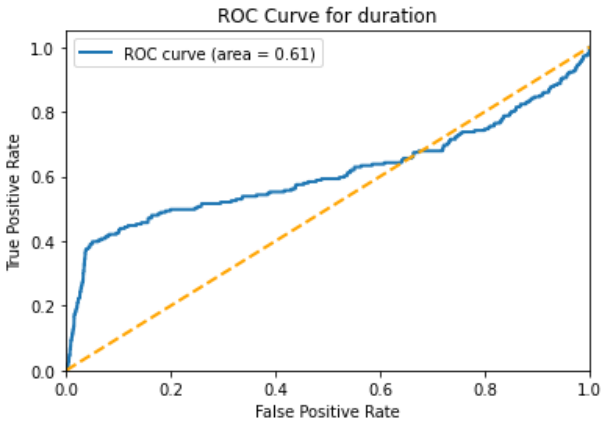
9. We can use valence to predict whether a song will be in the major or minor key using logistic regression since mode is a binary feature. First, I did a train and test split of the data in order to test for accuracy of the model, seeding the randomness by using my N-number. I then used the logistic regression model from sklearn and fit my data. I determine the AUC to quantify the model accuracy and graph the ROC curve to help visualize it. With an AUC value basically 0.5,

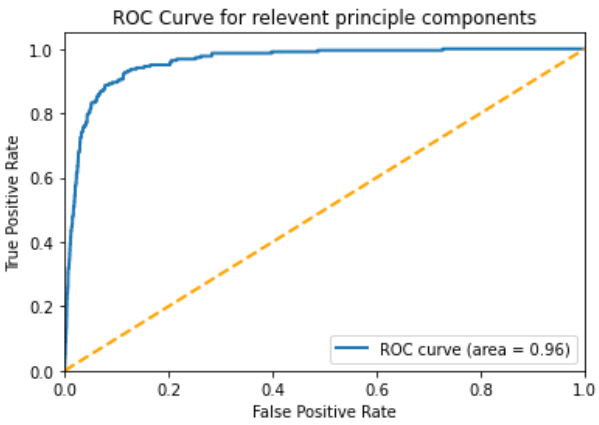
this means that the model is horrible at predicting the mode of a song; it is no better than a 50/50 guess since AUC measure begins at 0.5. When looking at the other features from the data, I picked ‘acousticness’ to hopefully be a better predictor because I assumed that minor songs would have a sadder tone which may have more acoustic instruments. I got an AUC value of 0.56 using this, which means it's a better predictor than ‘valence’, but still not that good for a prediction model in general.




```
Question 9
AUC for valence: 0.5070225655237692
AUC for acousticness: 0.5594186627948495
```

10. I first had to convert the quantitative feature of ‘track_genre’ to a binary one, whether or not the song genre was classical. Then, like in the previous question, I first split the ‘duration’ and ‘classical’ feature into training and testing in order to test for model accuracy and later compare, seeding the randomness by using my N-number. I fit the data to the sklearn logistic regression model, determine the AUC value and plot the ROC curve and I do the same steps of splitting and fitting the data, this time using the first 7 relevant principle components determined in Question 8 as my predictor. The AUC value using ‘duration’ as the predictor was 0.61 while the one using the first 7 principle components was 0.95, meaning that the relevant principle components were a much better predictor of whether a song is classical or not than using just ‘duration’.

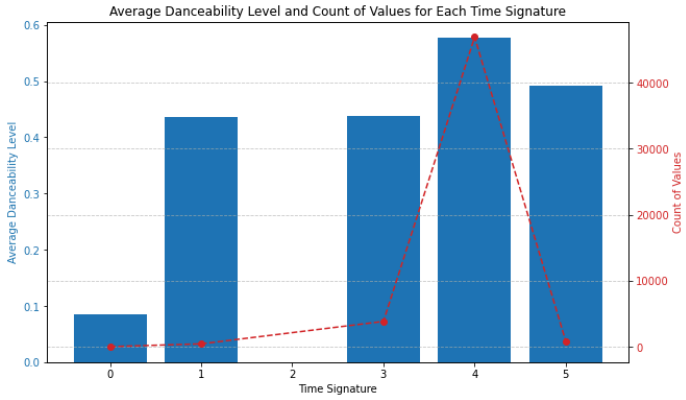


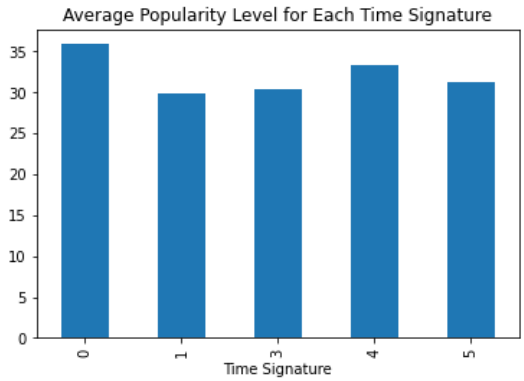



```
Question 10
AUC for duration: 0.6091300561576736
AUC for relevent principle components: 0.959780116902913
```

Extra Credit:

I wanted to use the ‘time_singature’ feature of the data and wanted to see if I could do anything interesting with it. When looking at the counts of the ‘time_singature’ feature, I noticed 9 songs labeled ‘0’ and I disregarded it as I assumed it represented missing data. I decided to first see the average ‘danceability’ level for each time signature. I initially assumed that time signature 4/4 would be the most common and also the most ‘danceable’, because I thought most dances were in 4/4 besides waltzes. When looking at the data, my prediction was correct, but I was also surprised at the ‘danceability’ levels for 1/4 and 5/4, which were at par or greater than that of 3/4. Seeing this interesting result, I also decided to look at the average ‘popularity’ level for each time signature and assumed songs in 4/4 would be the most popular. When visualizing the data, 4/4 was the most popular on average, but not by that much. From this, I concluded that there may not be enough data for songs in time signatures other than 4/4 to come to concrete conclusions, as this could be unrepresentative of the true population of ‘danceability’ and ‘popularity’ in these time signatures.


Average Danceability Level and Count of Values for Each Time Signature
Average Danceability Level
Count of Values
Time Signature
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0
10000
20000
30000
40000
0
1
2
3
4
5


Average Popularity Level for Each Time Signature
Time Signature
0
5
10
15
20
25
30
35
0
1
3
4
5