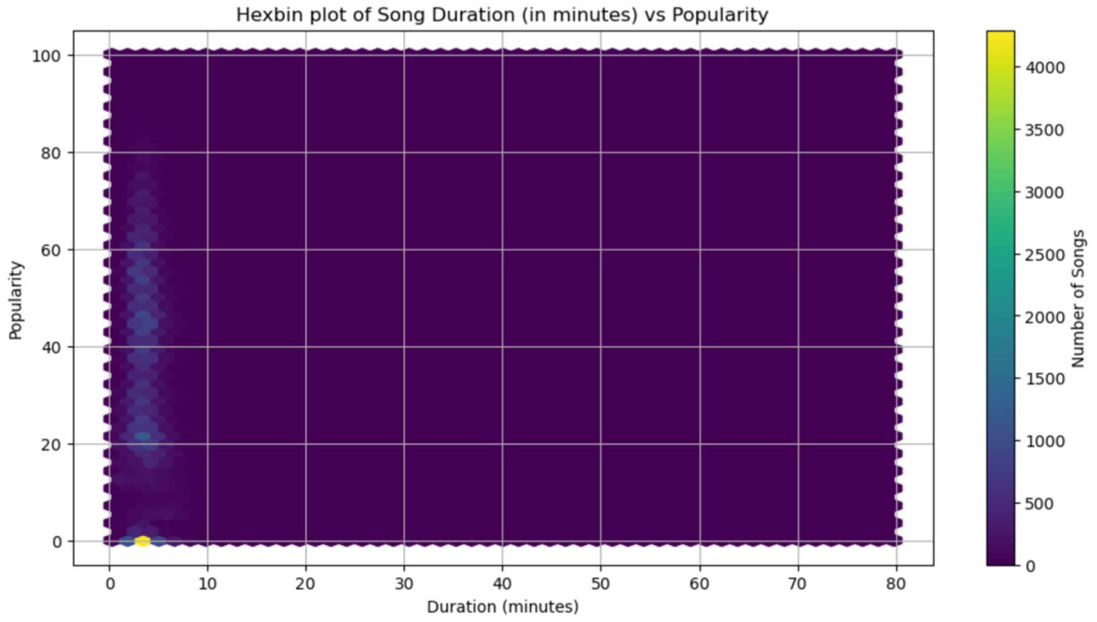
Contributor: Chen Hua, Haoda Yu, Haomin Yu (code, idea), ChatGPT (part of code)
In this report: question 1, 4, 5 & Extra credit has used Chen Hua's N number as random seed (15601674); question 2,6,8 has used Haoda Yu's N number as random seed (11375906); question 3, 7, 9,10 has used Haomin Yu's N number as random seed (10572260).

1)

In order to find the relationship between song length and popularity, we first select duration and popularity from the dataset. To understand the duration of the song, I turn it into the unit of minutes. And I calculated the correlation between these two variables and got -0.055. And I used the Hexbin plot to visualize the relationship between duration and popularity. There is no obvious strong relationship between song duration and popularity. The duration of most songs is concentrated in a short range, and the popularity distribution is relatively wide. Although the correlation coefficient we calculated before shows a slight negative correlation, it can be seen more clearly from this figure that this relationship is not obvious other factors may have a bigger impact on a song's popularity.



2) H_0: Explicitly rated songs are **not** popular than songs that are not explicit.

H_1: Explicitly rated songs are popular than songs that are not explicit.

For question 2 we first divide data into two sets "Explicit Rated (E)" and "Non-Explicitly Rated (non-E)", E group has 5597 songs and non-E has 46403 songs. Next, we do the following steps 1000 times (use Haoda Yu's N number as seed), do down sampling non-E songs number to equal the E songs number, then do a U test to these two sets of data. After this we calculate the mean value of U statistic and p-value.

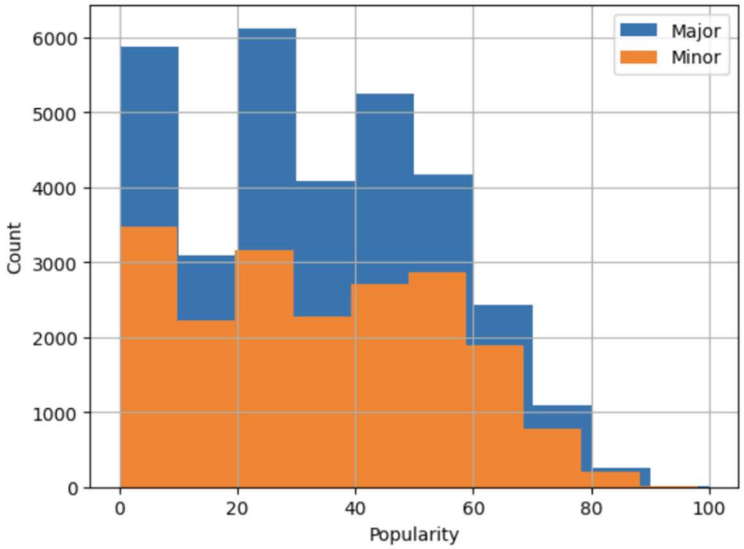
We choose this procedure because two sets of data have significant difference on variance and are both not normal distribution, so U test is the best choice here also down sampling can compensate for the imbalance in data volume and doing it 1000 times then take the mean value can give us a more robust result.

Finally, we have average U statistic = 16806045.6165, Average p-value = $8.45*10^{-9}$. With

the extremely low p-values from Mann-Whitney U test, we may confidently reject the H_0 and accept the H_1. This leads us to conclude that explicitly rated songs **are popular** than songs that are not explicit.

3)

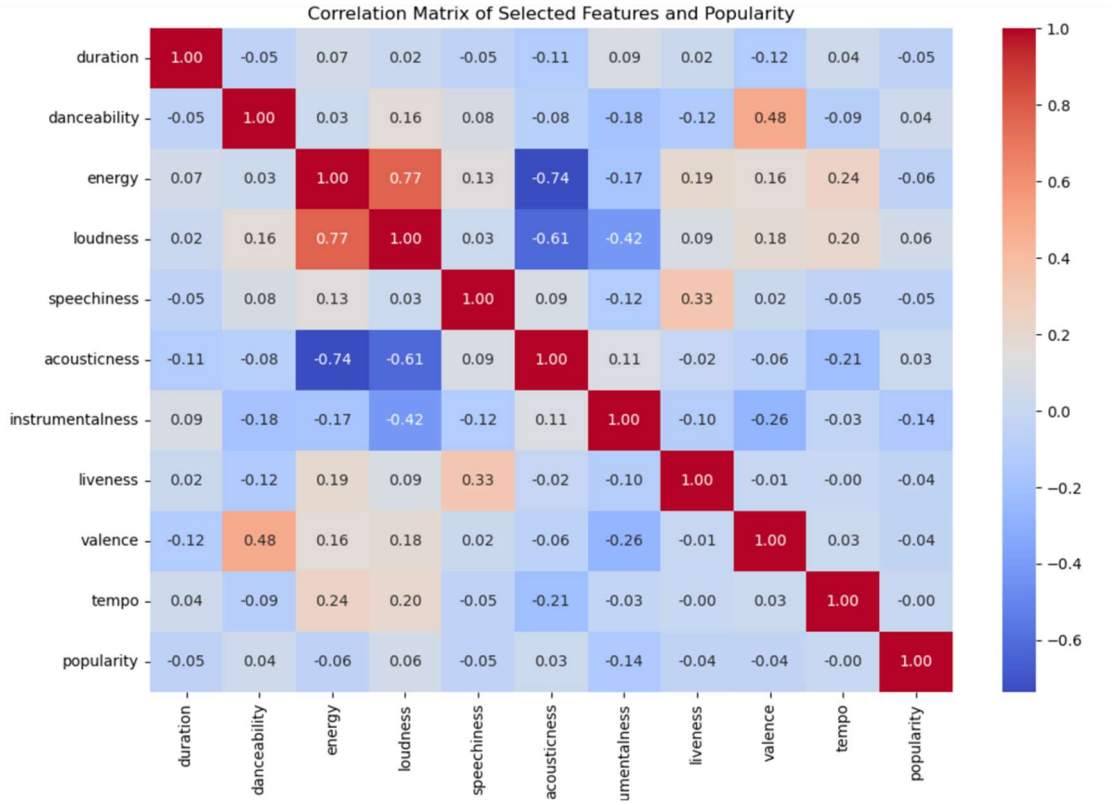
To determine whether major songs are more popular, we can use T-test or Mann-Whitney-U test. T-test is preferred because it has greater statistical power. However, T-test assumes that the data is normally distributed. Therefore, we first use KS test to check whether the data is normally distributed or not. We set the significance level to 0.05 and found that the data is not normally distributed. It is also evident from the histogram of the data that it is not normally distributed.



Then we use Mann-Whitney-U test to test the hypothesis. We set the significance level to 0.05 and found that we can reject the null hypothesis, which means minor key songs are more popular than major key songs.

4)

Firstly, we checked whether there is missing data in the database. And did correlation matrix to see the relationship between each feature. And train_test_split was used to select the train set and test set. After that logistic regression was used to train the models respectively. The COD and RMSE were used to evaluate the accuracy of the two models, and I also calculated the importance of each feature for popularity prediction under the two models to determine which features are more important. The results are shown in the figure. From the data, we can see that the accuracies of models are very low. Instrumentalness is more important than the other features. However, the low accuracy of predicting popularity from these features is reasonable because the popularity of a piece of music may have a higher correlation with the singer and the listening experience. It is more efficient to consider multiple features at the same time.




```
({'duration': 0.003756891592990752,
  'danceability': 0.001977090862231079,
  'energy': 0.004449112036058356,
  'loudness': 0.002849207560424505,
  'speechiness': 0.0020563842530981757,
  'acousticness': 0.0011416615545840614,
  'instrumentalness': 0.021104664943677243,
  'liveness': 0.0013590571693934406,
  'valence': 0.0019805389814274 71,
  'tempo': 1.4633492949056581e-05},
 {'duration': 21.645897388982238,
  'danceability': 21.66522409372876,
  'energy': 21.638375963614628,
  'loudness': 21.655756007841426,
  'speechiness': 21.66436342049667,
  'acousticness': 21.674290006357904,
  'instrumentalness': 21.45660764283447,
  'liveness': 21.671931237437885,
  'valence': 21.665186667564022,
  'tempo': 21.686514285568688})
```

5)

I first selected the desired features and used cross-validation to select the train set and test set and standardized the train dataset and test dataset. The train dataset is used to train the model using linear regression, and the Ridge Regression Model is used for regularization, and R^2 and RMSE are calculated to evaluate the model. Two models get similar R^2 and RMSE, which are approximately equal to 0.0498 and 21.039 shown in the figure. The R^2 of the model using all features has a significant improvement but is still too small. The similarity in performance between the linear and ridge regression models suggests that the data might inherently have limitations in predicting song popularity based on these features alone. In conclusion, while there is a marginal improvement in using a combined features model over individual feature models, the ability to predict song popularity from these features is still limited.

	Model Type	R^2 Value	RMSE Value
0	Linear Regression	0.049832	21.039199
1	Ridge Regression	0.049832	21.039199

6)

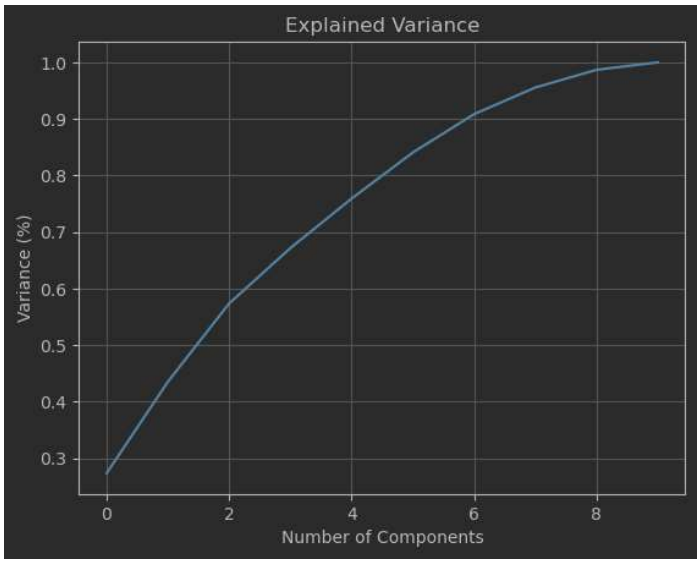
For question 6 we first import those 10 columns data and standardize them (StandardScaler) then we apply PCA to them with 95% confidence interval. Next, we calculate the total number of principal components and each components explained variance ratio. We got 8 principal components, in these components there are only 3 can be consider as meaningful principal components with cumulative variance ratio of 57.36% (Table 1). We have also drawn a cumulative variance graph (Shown below), we can find at least two elbows (approx. at [2,0.6] and [6,0.9]) from the graph by eyeball it, we have also found the eigenvalue of principal component, first three have over 1 eigenvalue (2.7339, 1.6174, 1.3846), all other have less than 1 eigenvalue. Then we use iterations try to find the best silhouette scores and number of clusters by try it from 2 to 20 (use Haoda Yu's N number as K means seed), finally we have best number of clusters at 2 and score of 0.4066. With these principal components, we finally find 2 clusters. This is far less than the genre (52) we have.

So, the result is that from 10 features we can extract 3 principal components, they have total 57.36% proportion, and with these principal components we can divide them into 2 clusters. But it is unable to genre labels in column 20.

Tabel 1

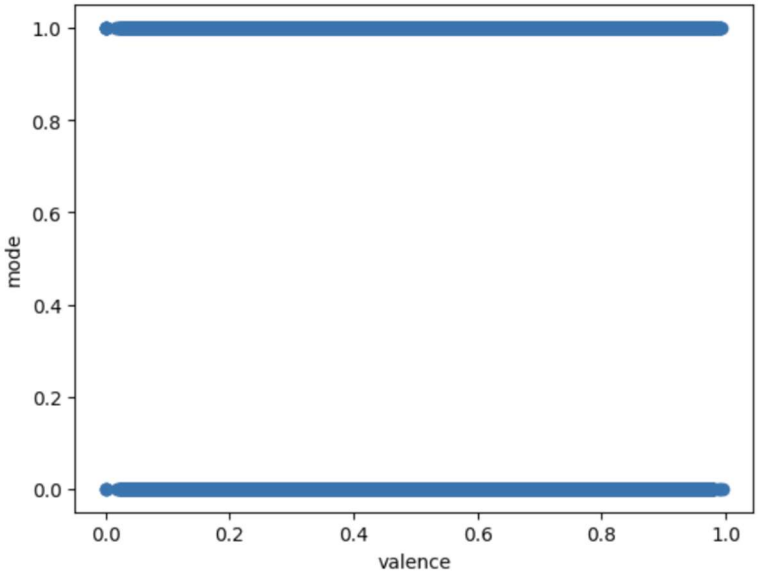
Principal Component	Explained Variance Ratio	Cumulative Variance Ratio
1	0.2734	0.2734
2	0.1617	0.4351

3	0.1385	0.5736
4	0.0980	0.6715
5	0.0875	0.7591
6	0.0815	0.8405
7	0.0678	0.9084
8	0.0472	0.9555



7)

This is a problem of predicting the mode using a single feature (valence) as input. We can first plot the data in a 2D plane with valence as x axis and mode as y axis.



From the figure above, we can see that there is no clear pattern that mode is related to valence, which indicates that it would be hard to fit any model to make a good prediction. But we can still try and see how the prediction model would perform. We split the data into 80% and 20% as training and validation data and fit a logistic regression model and a SVM model to the training data. The prediction accuracy on validation of both models is 62.11%, which is not very good considering it's a binary classification problem. We further see that the

predictions of the model are always 1, indicating that the model did not learn anything useful, which complies with our observation that there is no clear pattern that mode is related to valence.

8)

Considering we have only 2 clusters from question 6 result, this would be a big challenge to predict genre and may lead to a result with bias. So, we decided to use 10 features mentioned in question 4 to predict the genre. Here we choose to use a classification model called random forest to do the prediction work.

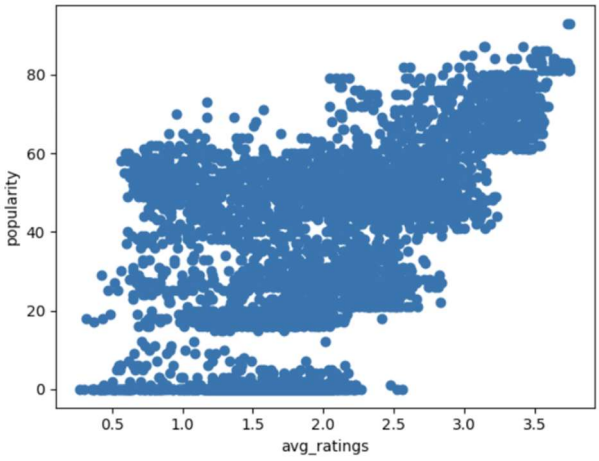
In this step we first import the data and standardize them (StandardScaler) and use test number = 0.3, which means 70% of data would be random (Haoda Yu's N number is random seed) selected and used to training model and 30% left will be used to test model's performance. Next, we create a random forest classifier (Use Haoda Yu's N number as random seed) to classified data with estimator of 100 (we have tried 10,100,1000, from 10 to 100 there is like 5% of accuracy benefit but increase it to higher number would not have much prediction accuracy increase, so the best accuracy/time is 100).

We have also tried to use XGBoost to do the classify, but the performance was similar to random forest. (accuracy = 0.34)

After doing this, we have a model of 35.78% accuracy, this is too low and not accurate enough do any genre classification work. The performance of this model is not very well, our expected accuracy should be higher than 50%.

9) a)

We can plot average ratings and popularity into a 2D plane, as shown in the following figure.



We can see that there is a tendency that higher average ratings indicate higher popularity. We also calculated the correlation coefficient between the two variables, which is 0.57. It also indicates that there is a positive correlation between these two variables.

9) b)

We can calculate how many ratings each song received, and the top 10 songs with most

ratings would be the “greatest hits”, which are shown below.

songNumber	artists	album_name	track_name
100	Motohiro Hata	言ノ葉	Rain
275	Aidan Hawken	Walking Blind (feat. Carina Round)	Walking Blind
57	Chord Overstreet	Sleepwalking in the Rain	Sleepwalking in the Rain
2841	Siouxsie and the Banshees	Helloween Party	Spellbound
2254	Welshly Arms	No Place Is Home	Legendary
4134	Bethel Music;Jonathan David Helser;Melissa Helser	Victory (Live)	Raise a Hallelujah - Live
91	The Civil Wars	Alternative Christmas 2022	I Heard The Bells On Christmas Day
3327	Yeah Yeah Yeahs;A-Trak	Halloween Party 2022	Heads Will Roll - A-Trak Remix
1749	Hugh Masekela	60's Gold	Grazing In The Grass
2563	The Servant	Collection	Orchestra

10)

To create an individualized recommendation, we start by measuring the similarity of music tastes between users. Then for a particular user, we can refer to those users that have similar tastes as this user to make our recommendations. To measure how similar two users are, we consider the songs that both users have given ratings and calculate the mean absolute difference between these songs and use this as the distance between the two users. After calculating the distance between all pairs of users, we obtain a distance matrix of shape 10000 by 10000, where each element is the distance between two users (i.e., the matrix is symmetric).

To provide recommendations for a specific user, we first identify the top 10 users who have similar preferences to this user. Then, we recommend the top 10 songs that have the highest average ratings from these 10 users. This assumes that users with similar tastes will enjoy the same music, making it an ideal recommendation for our user.

To compare our individualized recommendation with ‘greatest hits’, we calculate the mean of the number of intersected songs between our recommendation and the ‘greatest hits’, which is 0.0215. This indicates that our recommendation is very different from ‘greatest hits’.

We calculate the average ratings of our recommended songs and those of the 'greatest hits'. Note that not all of these songs have ratings, so we only consider those that do. The mean average rating of our recommended songs is 3.41, while the mean average rating of 'greatest hits' is only 2.09. This indicates that our personalized recommendation is better.

11) Extra credit

Perform PCA to identify the three features most strongly correlated with danceability and then build a model to predict danceability based on these features.

First select the desired features and standardize the features, then perform a PCA to find the three features to predict danceability, and use these three features and linear regression to predict the danceability. R^2 for this mode is 0.21, RMSE is 0.157.