Relations between Music Taste and Self-reported Mental Health

Xinyi Wang

Data Science Institute, Brown University

https://github.com/XinyiW5/myrepositoryxinyi.git

1 Introduction

The project focuses on the relations between people's music taste and their self-reported mental health conditions. The topic is important because many people nowadays are experiencing mental health problems. If we can know the correlations between music and mental health, such as listening to what kind of music can cause or reduce what certain kinds of mental health problems, we can apply the rules on the future mental health therapy area. Also, music therapy will be extremely beneficial for patients since listening to music is much more accessible than getting medicines, seeing doctors, or even getting surgeries. The dataset is got from Kaggle, and the data were collected through both Google Form posted on social media and posters spread out in public locations such as libraries and parks. The participants are not restricted by age and locations.

2 EDA

There are four mental health problem features included in the dataset. They are "Anxiety", "OCD", "Insomnia" and "Depression". They are ordinal features and have levels ranging from zero to ten, while zero means "you barely experience the mental health problem" and 10 means "you frequently experience the mental health problem". I chose "Anxiety" as my target variable, since it has the most points for level of five and above.

The problem is a classification problem, and the plot of “Anxiety” distribution is attached here.

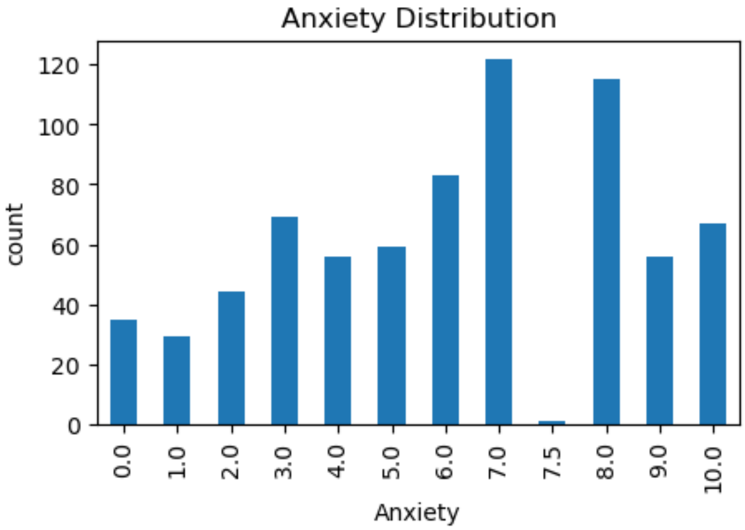


Figure 1 Anxiety Distribution

We can see a very interesting thing from the plot that we have an extremely low counts on level of ‘7.5’, and there’s no other half level having points in their category. To better know about the counts on level ‘7.5’, I checked the value counts on each level and there’s only one point in this level. Since I’m working on a classification problem, I want to make the target be considered as different categories but not numbers. So, I need to keep the target feature values as integers without decimals. Also, since there is only one point in the level of ‘7.5’ and there are no other “0.5” levels, it’s plausible to count “7.5” as “8” for the goal of keeping all the points as integers.

The updated “Anxiety” distribution plot is attached below.

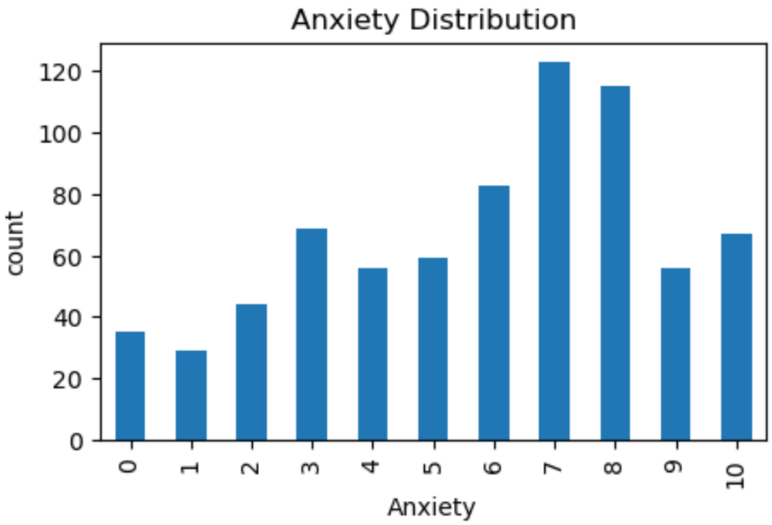


Figure 2 Anxiety Distribution after Processing

From the updated plot, we can see the dataset is not very balanced. But I still take the dataset as a balanced one, since we can't say someone's personal feeling is wrong. For instance, we can't say there are definitely many people barely experiencing anxiety, and we can't say some people should not feel that much anxious as they said. So, we can't say it's an imbalanced dataset. A possible reason to the "imbalanced look" of the distribution is people joining the survey might not know how to correctly measure their mental health conditions. For example, people who are actually at level of '5' may fill out the form as level '7' due to the insufficient knowledge of anxiety level standards, which can lead to the imbalanced look of the plot.

Then, I checked the distribution of "age" feature. Seeing from the plot attached below, most people in the dataset are at their twenties, which means after we train the models and get some conclusions, they might be useful only when applying to groups at the similar age. Also, high concentration at age of twenty may cause age bias to the dataset, such as determining what music types are more popular in this dataset, since people at different ages tend to listen to different music.

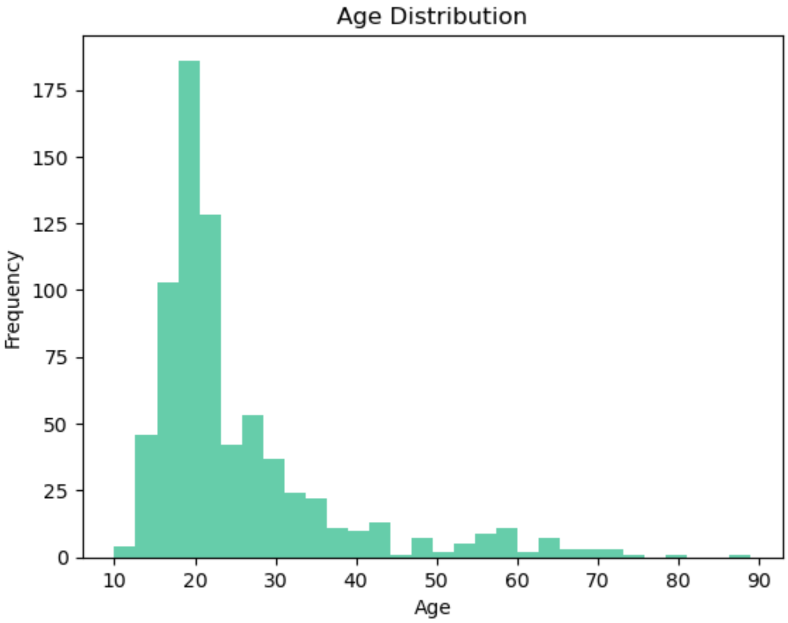


Figure 3 Age Distribution

3 Methods

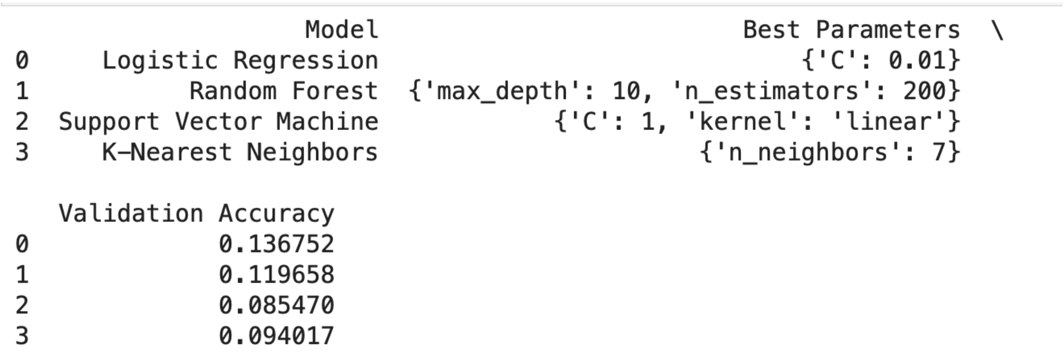
To train the data, I need to preprocess them first. There are missing values in eight features, including categorical, ordinal and numerical features. So, I used OneHotEncoder, OrdinalEncoder and IterativeImputer to deal with them. Then, I applied KFold splitting method on my dataset, since the dataset is IID and balanced.

After all the preprocessing, I chose ‘accuracy’ as my evaluation metrics and applied five ML algorithms on the dataset, they are logistic regression model, random forest model, SVC model, kNN model and XGB model. I’ll explain each model below.

For the logistic regression model, I tuned the parameter “logisticregression__C” on values of 0.001, 0.01, 0.1, 1, 10, and 100, and the best model parameter is 0.1. For random forest algorithm, I tuned parameter “randomforestclassifier__max_depth” on values 1, 3, 10, 30, and 100, and tuned parameter“randomforestclassifier__max_features” on values 0.5, 0.75 and 1.0. The best model parameter is max depth of 3 and max features of 0.75. For SVC model, I tuned “svc__C” on 0.1, 1, and 10, and tuned “svc__kernel” on 'linear' and 'rbf', with the best model parameter of 1 for “svc__C” and 'rbf' for “svc__kernel”. For kNN algorithm, I tuned “kneighborsclassifier__n_neighbors” on values 30, 50 and 70, and 50 is the best parameter. Lastly, I tuned parameters for XGB model. They are "xgbclassifier__learning_rate" of value 0.03, "xgbclassifier__n_estimators" of 50, "xgbclassifier__seed" on 0, “xgbclassifier__max_depth” on 1, 3, and 10, "xgbclassifier__missing" on ‘np.nan’, "xgbclassifier__colsample_bytree" on 0.9, and "xgbclassifier__subsample" of value 0.66. The best parameter for XGB is 1 for “xgbclassifier__max_depth”, with the rest keeping the same.

4 Result

It took many efforts to get to the right models. At first, I used “mean” as my metrics for imputer to apply on the numerical missing values and then applied 4 models, which were logistic regression, random forest, SVC and kNN. The summary table of their best parameters and validation scores is below.

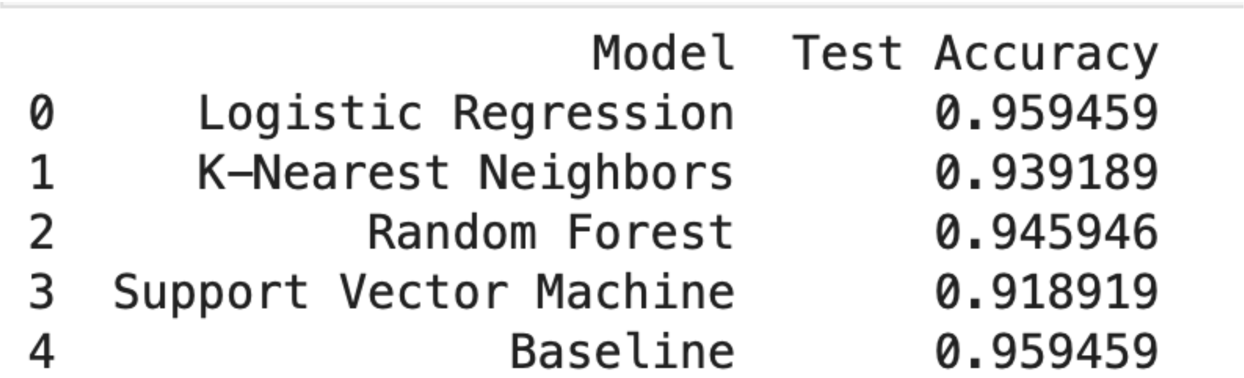


```
                    Model                             Best Parameters  \
0     Logistic Regression                                  {'C': 0.01}
1           Random Forest  {'max_depth': 10, 'n_estimators': 200}
2  Support Vector Machine           {'C': 1, 'kernel': 'linear'}
3     K-Nearest Neighbors                        {'n_neighbors': 7}

   Validation Accuracy
0             0.136752
1             0.119658
2             0.085470
3             0.094017
```

Figure 4 Summary Table of Parameters and Validation Accuracies before Correction

We can see from the table that the validation accuracy is extremely low for every model. I also built the baseline model and computed the test scores on them. The table is below.



```
                    Model  Test Accuracy
0     Logistic Regression       0.959459
1     K-Nearest Neighbors       0.939189
2           Random Forest       0.945946
3  Support Vector Machine       0.918919
4                Baseline       0.959459
```

Figure 5 Test Accuracies of 5 Models before Correction

We can see that in opposite, the test accuracies are extremely high for every model, which makes me realize that there’s something wrong with my training process. So, I adjusted my imputer method to IterativeImputer and also added a new model, XGB model, to train my dataset. The new validation score and test score for each model is recorded in the following table.

	Logistic Regression	Random Forest	SVC	kNN	XGB
Validation Score	0.214	0.233	0.226	0.221	0.192
Test Score	0.176	0.189	0.162	0.216	0.176

Figure 6 Validation and Test Scores of 5 Models

We can see that there is not a big difference between validation score and test score anymore, even though all the models still have relatively low validation and test scores. The best model is kNN model based on the test score.

After I trained the five models, I computed their feature importance with permutation.

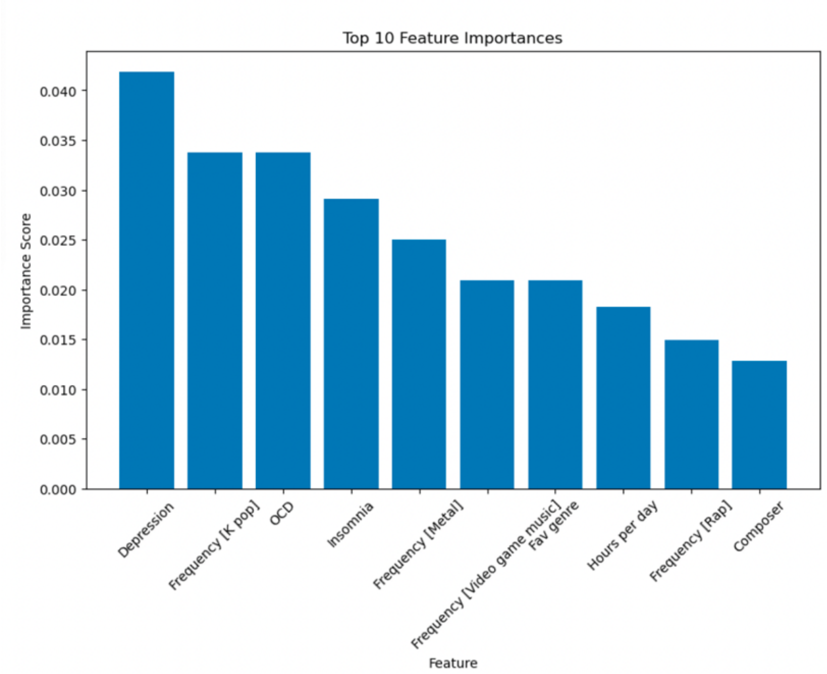


Figure 7 Logistic Regression Top 10 Important Features with Permutation

For logistic regression model, the top three important features are “Depression” level, the frequency listening to K-pop and the “OCD” level. The least important three features are the hours of listening to music per day, the frequency listening to rap, and the music composer.

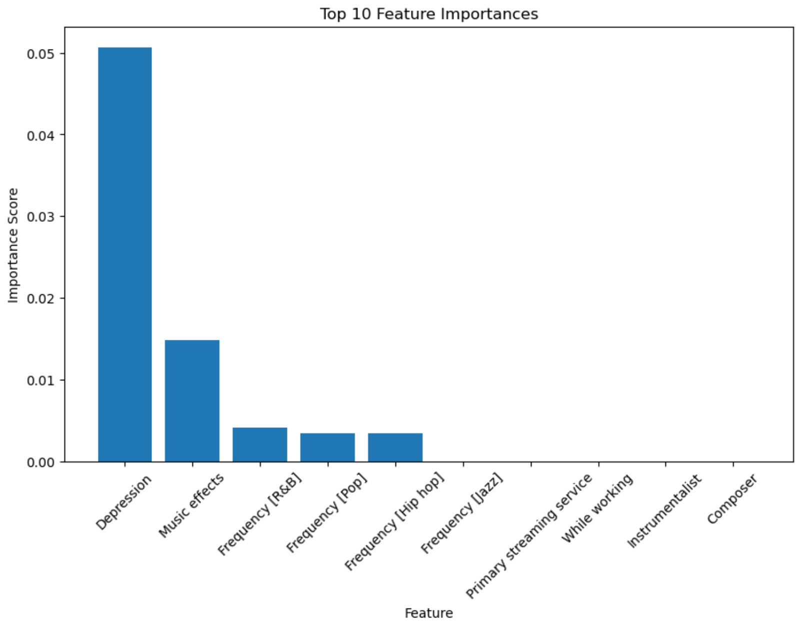


Figure 8 Random Forest Top 10 Important Features with Permutation

For random forest model, the top three important features are "Depression" level, the music effects and the frequency listening to R&B music. A very interesting point about the random forest model plot is that after the fifth important features, the importance scores are all zeros for the rest features.

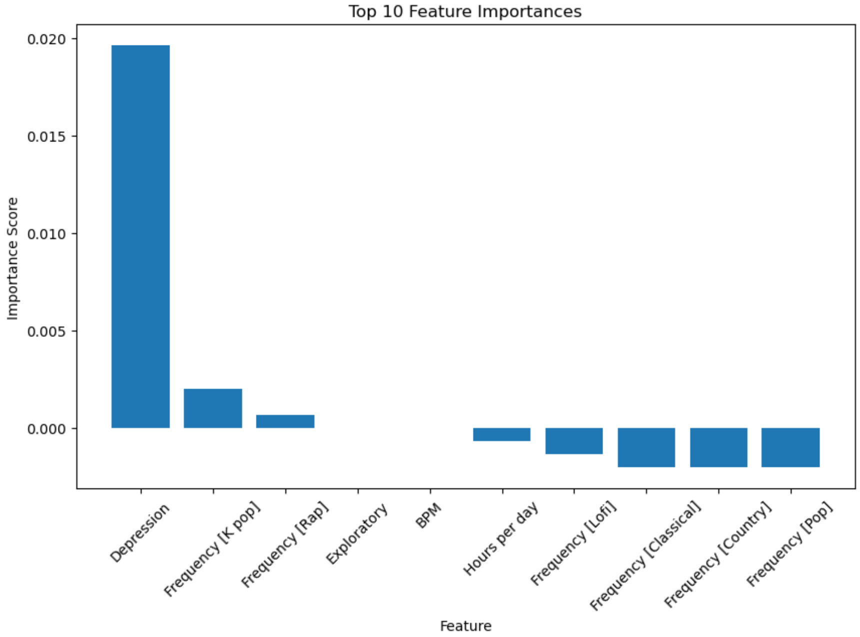


Figure 9 SVC Top 10 Important Features with Permutation

For SVC model, the top three important features are “Depression” level, the frequency listening to K-pop and the frequency listening to rap, while the importance of “Depression” level is extremely high compared to others. After these top three features, the rest feature importance are either equal to or below zero.

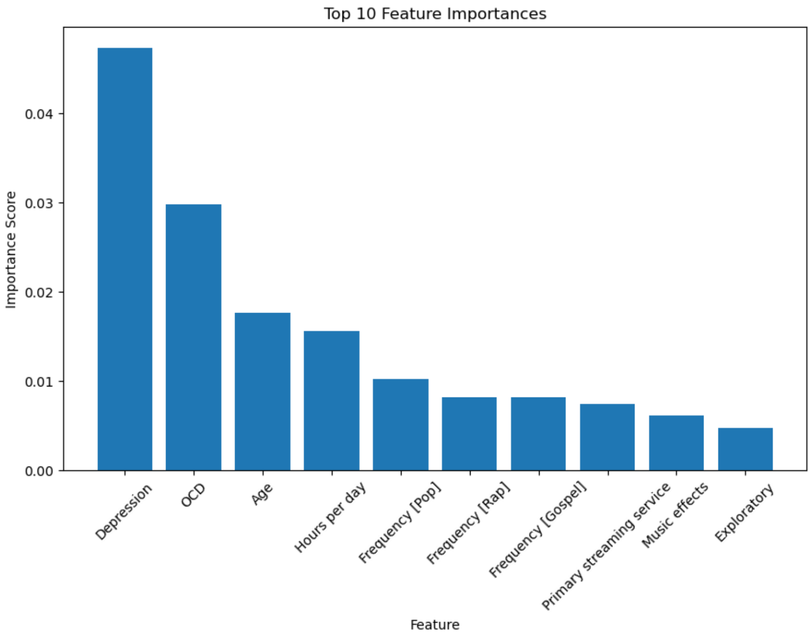


Figure 10 kNN Top 10 Important Feature with Permutation

For kNN model, the top three important features are “Depression” level, the “OCD” level, and age. The least important three features are the primary streaming effects, music effects and exploratory.

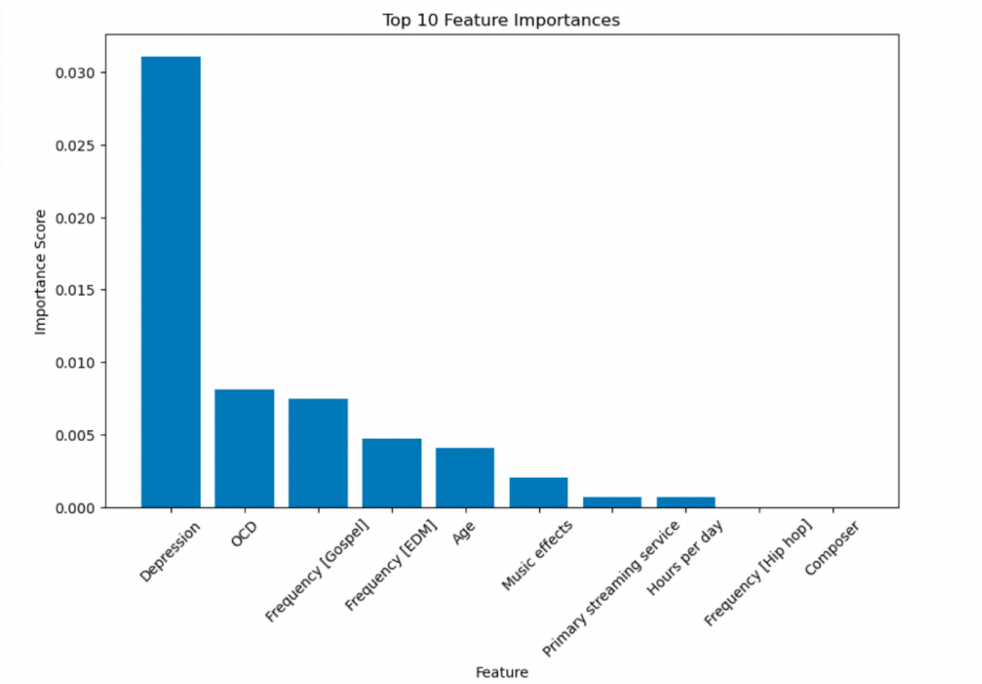


Figure 11 XGB Top 10 Important Features with Permutation

Lastly, for XGB model, the top three important features are “Depression” level, the “OCD” level, and frequency listening to Gospel music. The third to last important feature is the hours listening to music per day, with the rest two least important features with feature importance equal to zero.

To conclude, we can see that from the “Depression” level is always the most important feature for all models, while “OCD” level appears three times among the five models in the top three importance. It means that people having depression or OCD tend to be more anxious. To generalize that, people with one mental health problem may tend to have other mental health problems as well. Also, “Composer” appears three times to be the least important feature, which means who compose the music doesn’t affect mental health that much.

Also, for random forest and SVC models, there are features with zero or negative feature importance, with means that in fact, only very few factors are related with mental health, other details of music such as BPM, whether listening to music during work etc. don't affect mental health a lot.

5 Outlook

To improve the models, there are several aspects to consider. Firstly, we can see from the feature importance plots that other mental health problems are always in the top three important features, especially "Depression" and "OCD". So, for our next step, we can train our dataset on the relations between music types and the rest three mental health problems, so that we can figure out their correlations to further apply on the correlations between music types and "Anxiety".

Also, we noticed that the validation and test scores are not high for all the five models we have. A possible reason is that for the classification problem with accuracy score as the evaluation metrics, once the predicted class is not the same with the true class, it will be counted as "wrong", so that the accuracy score is low. To improve this, we can train the dataset again using regression models.

For the participant aspect, we can add more age groups to the dataset so that when we train the data and get some conclusions, the results can be more generalized to be applied to the whole society. Besides, we can teach people how to accurately measure their mental health conditions before they fill out the survey, to make sure the data we get are good to use and avoid the imbalanced look of the dataset.

6 References

Rasgaitis, C. (2022, November 21). *Music & Mental Health Survey Results*. Kaggle. https://www.kaggle.com/datasets/catherinerasgaitis/mxmh-survey-results

The transformative power of music in mental well-being. Psychiatry.org - The Transformative Power of Music in Mental Well-Being. (2023, August 1). https://www.psychiatry.org/news-room/apa-blogs/power-of-music-in-mental-well-being

Lorrie Kubicek, M.-B. (2022, July 25). *Can music improve our health and quality of life?*. Harvard Health. https://www.health.harvard.edu/blog/can-music-improve-our-health-and-quality-of-life-202207252786