“Rhythms of the Mind”: Music as Therapy for Mental Well-Being

Bhattacharya Brothers
comprising
Soham Bhattacharya & Darpan Bhattacharya
(B2430059 & B2430044, MSc BDA, Sem 1)
under the guidance of
Br. Tamal Mj

Department of Computer Science
Ramakrishna Mission Vivekananda Educational and Research Institute

30 November 2024

Objective

- To answer the following question:
- Based on the music tastes and lifestyle of an individual, does music improve or worsen their mental health conditions?

Objective

- To answer the following question:
- Based on the music tastes and lifestyle of an individual, does music improve or worsen their mental health conditions?

Motivation

- We like music.
- Mental health of our generation isn't that great. (Don't believe me? Look around you.)
- We had a project at hand, and needed data for it. This was one of the first okay-ish datasets that we came across that could be used for our project.

Dataset

- We used a dataset called “Music and Mental Health Survey results”[dat] which is publicly available on kaggle.
- The raw dataset, of shape 736 x 33, contained many features such as age, # hours per day, while working, frequency of music listened to in different genres, etc.

The Dataset

	Timestamp	Age	Primary streaming service	Hours per day	While working	Instrumentalist	Composer	Fav genre	Exploratory	Foreign languages	...	Frequency [R&B]	Frequ [Rap]
0	8/27/2022 19:29:02	18.0	Spotify	3.0	Yes	Yes	Yes	Latin	Yes	Yes	...	Sometimes	Very freque
1	8/27/2022 19:57:31	63.0	Pandora	1.5	Yes	No	No	Rock	Yes	No	...	Sometimes	Rarely
2	8/27/2022 21:28:18	18.0	Spotify	4.0	No	No	No	Video game music	No	Yes	...	Never	Rarely
3	8/27/2022 21:40:40	61.0	YouTube Music	2.5	Yes	No	Yes	Jazz	Yes	Yes	...	Sometimes	Never
4	8/27/2022 21:54:47	18.0	Spotify	4.0	Yes	No	No	R&B	Yes	No	...	Very frequently	Very freque

ratory	Foreign languages	...	Frequency [R&B]	Frequency [Rap]	Frequency [Rock]	Frequency [Video game music]	Anxiety	Depression	Insomnia	OCD	Music effects	Permissions
	Yes	...	Sometimes	Very frequently	Never	Sometimes	3.0	0.0	1.0	0.0	NaN	I understand.
	No	...	Sometimes	Rarely	Very frequently	Rarely	7.0	2.0	2.0	1.0	NaN	I understand.
	Yes	...	Never	Rarely	Rarely	Very frequently	7.0	7.0	10.0	2.0	No effect	I understand.
	Yes	...	Sometimes	Never	Never	Never	9.0	7.0	3.0	3.0	Improve	I understand.
	No	...	Very frequently	Very frequently	Never	Rarely	7.0	2.0	5.0	9.0	Improve	I understand.

Figure 1: The raw dataset

Data Preprocessing: Our reaction

Figure 2: Our reaction on actually looking at the dataset

Data Preprocessing: What we did

- **Data Cleaning**: dropped missing entries
- **Feature Selection**: removed unnecessary features
- **Feature Engineering**: performed feature scaling by normalization
- **Train-Test split**: split dataset into 70-30 ratio

Data Preprocessing: What we did

- **Data Cleaning**: dropped missing entries
- **Feature Selection**: removed unnecessary features
- **Feature Engineering**: performed feature scaling by normalization
- **Train-Test split**: split dataset into 70-30 ratio

Data Preprocessing: What we did

- **Data Cleaning**: dropped missing entries
- **Feature Selection**: removed unnecessary features
- **Feature Engineering**: performed feature scaling by normalization
- **Train-Test split**: split dataset into 70-30 ratio

Data Preprocessing: What we did

- **Data Cleaning**: dropped missing entries
- **Feature Selection**: removed unnecessary features
- **Feature Engineering**: performed feature scaling by normalization
- **Train-Test split**: split dataset into 70-30 ratio

Overview

- Our problem is primarily a multi-class classification problem.
- The models that we used to test our problems are:

Models Used

- **Logistic Regression**:
  - It is a binary classifier.
  - We used OvR strategy using `lbfgs` solver for multi-class classification
- **Softmax Regression**:
  - Logistic, but cooler.
- **Random Forest Classifier**:
  - It is an ensemble learning method.
  - It combines multiple decision trees to improve prediction accuracy and prevent overfitting.

Models Used

- **Support Vector Machine**:
  - It aims to maximize the margin between the hyperplane and the data points.
  - It works well with large number of features, like in our case.
- **XGBoost**:
  - It is popular, so we used it.
  - Apparently it performs well in competitive ML environments.
- **LightGBM**:
  - It stands for "Light Gradient Boosting Machine".
  - Used for the same reason as above.

- **Voting Classifer**:
  - It consists of both *hard voting* (majority voting) and *soft voting* (probability voting).
  - We used both.
  - It helped us to compare between the different models and hopefully provide an overall better prediction.

Drumroll

Figure 3: Results incoming...

Results

- We obtained a highest prediction accuracy of **0.767568**. We obtained it using both *Logistic Regression without hyperparamaeter tuning* and *Softmax Regression*.
- We obtained a mean prediction accuracy of 0.751801 across all our models.

Results Table

Classifier	Training Accuracy	Test Accuracy
Logistic Regression w hpt	0.755814	0.756757
Logistic Regression w/o hpt	0.765116	0.767568
Softmax Regression	0.765116	0.767568
RandomForest w hpt	1.000000	0.756757
RandomForest w/o hpt	1.000000	0.735135
SVM w hpt	0.755814	0.756757
SVM w/o hpt	0.762791	0.756757
XGBoost w/o hpt	1.000000	0.745946
XGBoost w hpt	0.800000	0.735135
LightGBM	1.000000	0.751351
Voting Classifier (Hard)	0.981395	0.751351
Voting Classifier (Soft)	1.000000	0.740541

Training Accuracy Scores

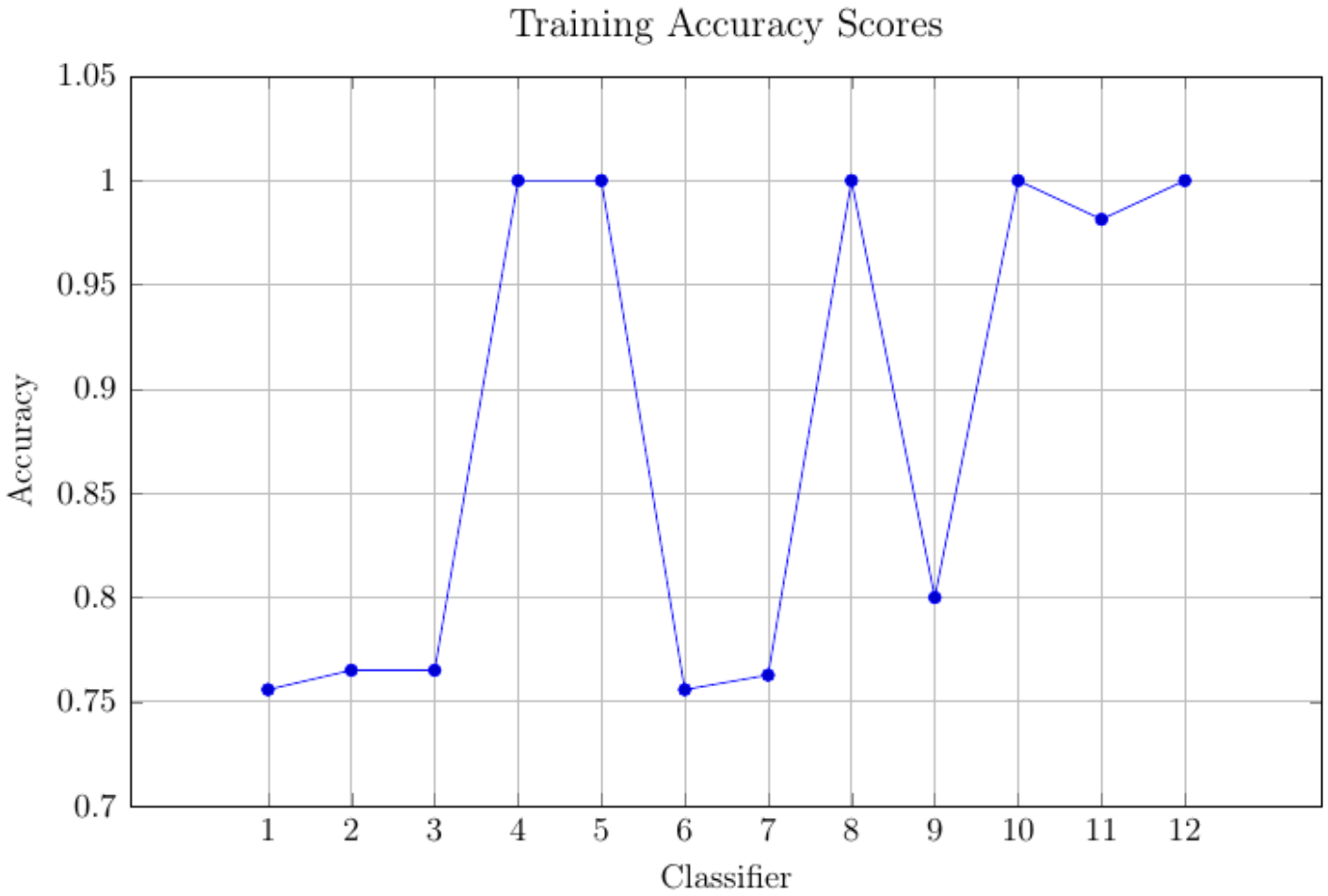


Figure 4: Training Accuracy Scores

Testing Accuracy Scores

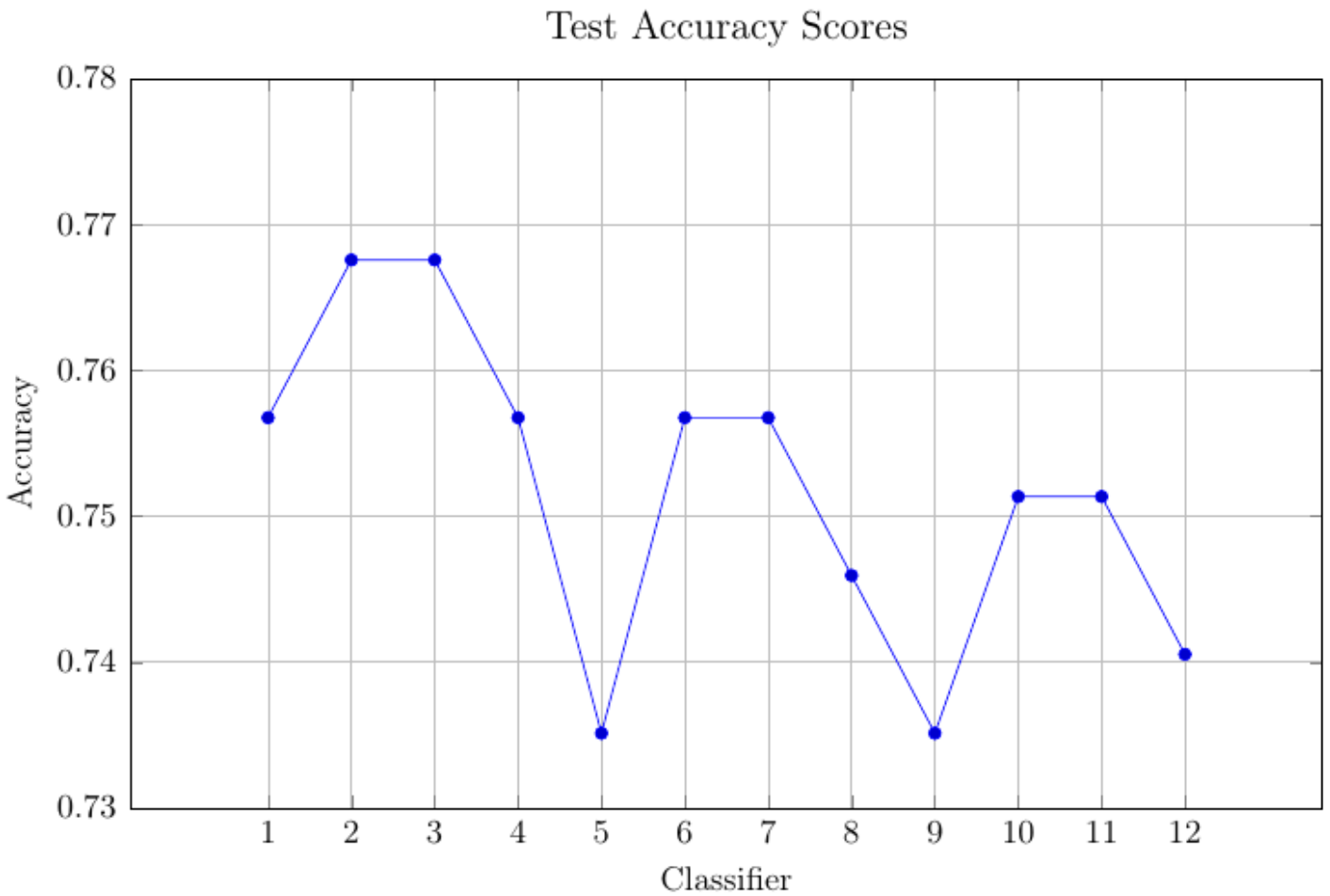


Figure 5: Testing Accuracy Scores

Combined Accuracy Scores

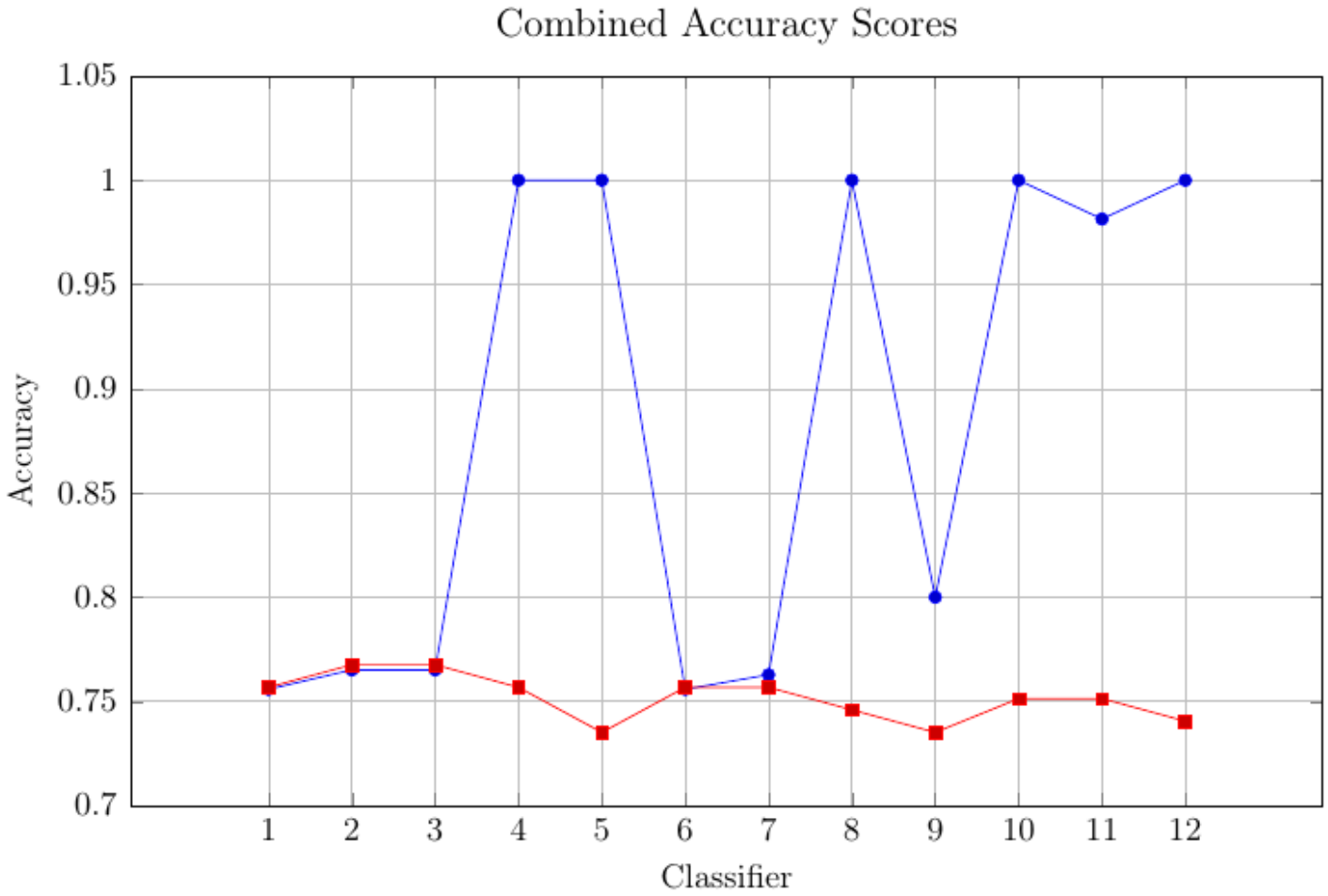
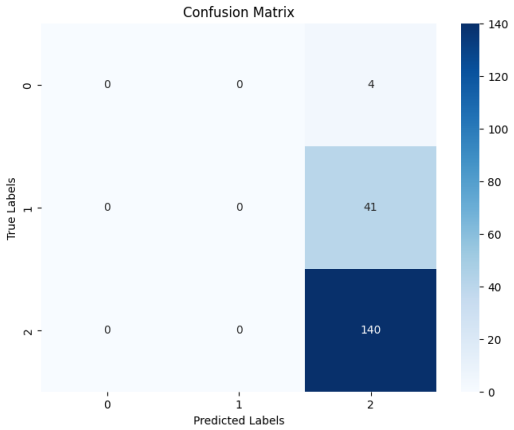


Figure 6: Combined Accuracy Scores

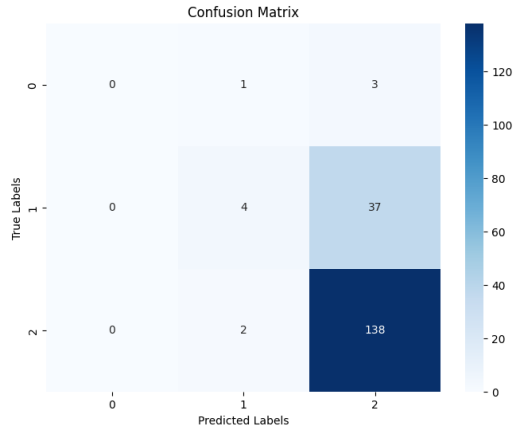
Legend for the charts

1 = Logistic Regression with hyperparameter tuning
2 = Logistic Regression without hyperparameter tuning
3 = Softmax Regression
4 = RandomForest with hyperparameter tuning
5 = RandomForest without hyperparameter tuning
6 = SVM with hyperparameter tuning
7 = SVM without hyperparameter tuning
8 = XGBoost without hyperparameter tuning
9 = XGBoost with hyperparameter tuning
10 = LightGBM
11 = Voting classifier with hard voting
12 = Voting classifer with soft voting

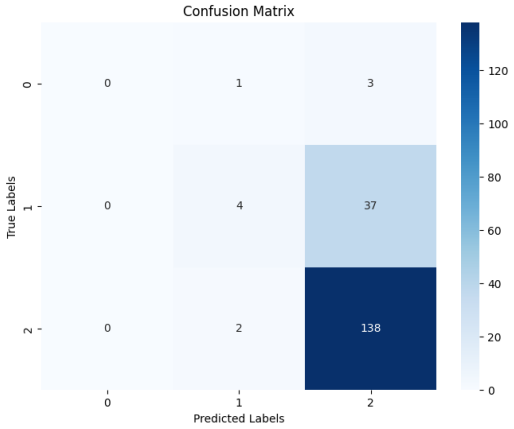
Confusion matrices



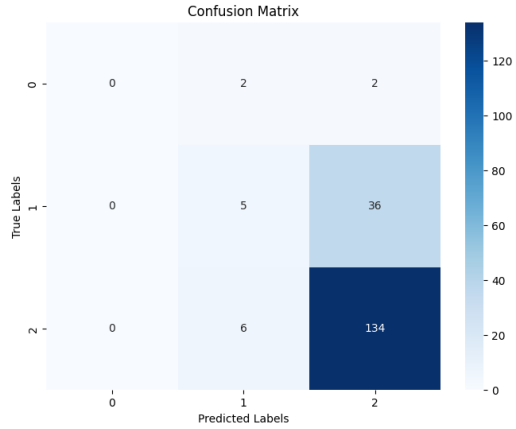
Log Reg. w/ hp



Log Reg. w/o hp

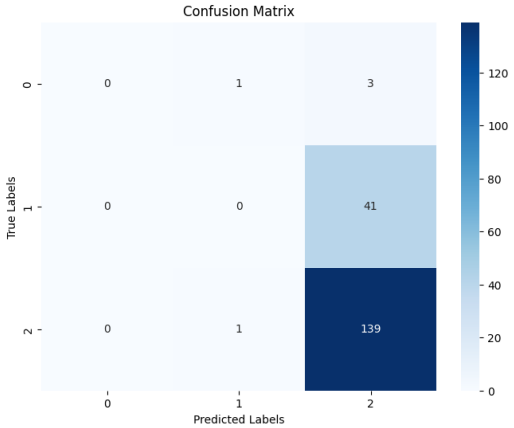


Softmax Reg.

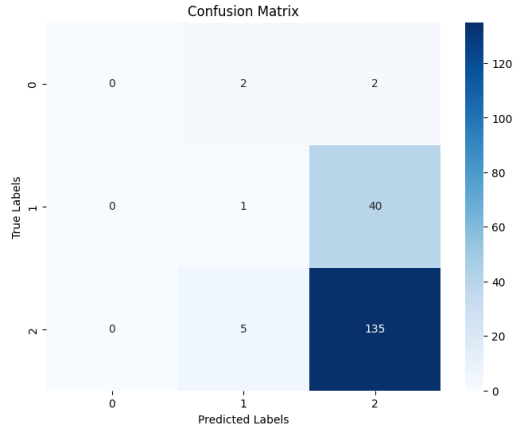


LightGBM

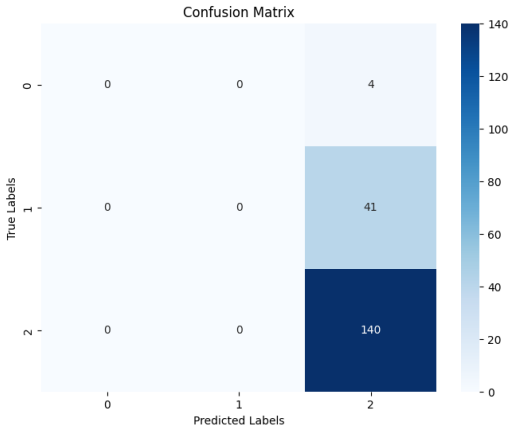
Confusion matrices



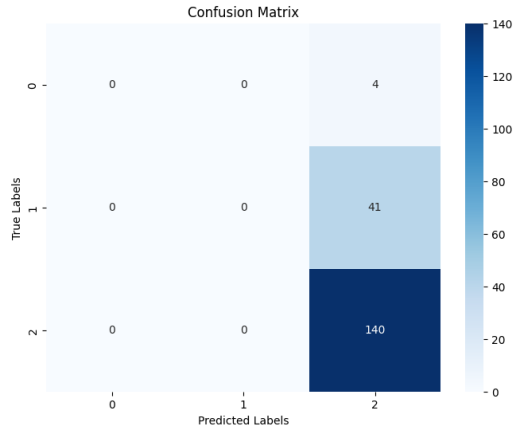
RF w/ hp



RF w/o hp

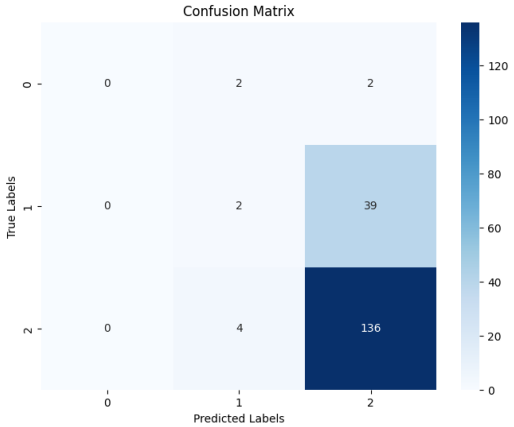


SVM w/ hp

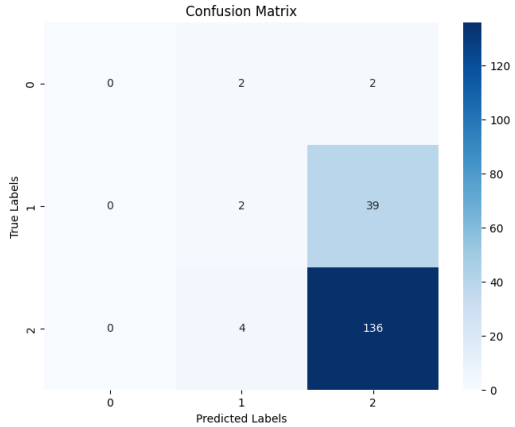


SVM w/o hp

Confusion matrices



XGB w/ hp



XGB w/o hp

Challenges & Learnings

- **No cross validation**: due to resource constraints
- **Dataset size**: it wasn't large enough to provide very impactful results

Challenges & Learnings

- **No cross validation**: due to resource constraints
- **Dataset size**: it wasn't large enough to provide very impactful results

Conclusion

- We obtained a prediction accuracy of ~ 0.76 from our project. Based on the analysis covered in our project our assumption that "given the music tastes and lifestyle of an individual, does music improve/worsen their mental health conditions" is indeed true.
- Our accuracy score depicts that we can detect an individual's mental health correctly with moderately high accuracy based on their living conditions and music tastes.

Conclusion

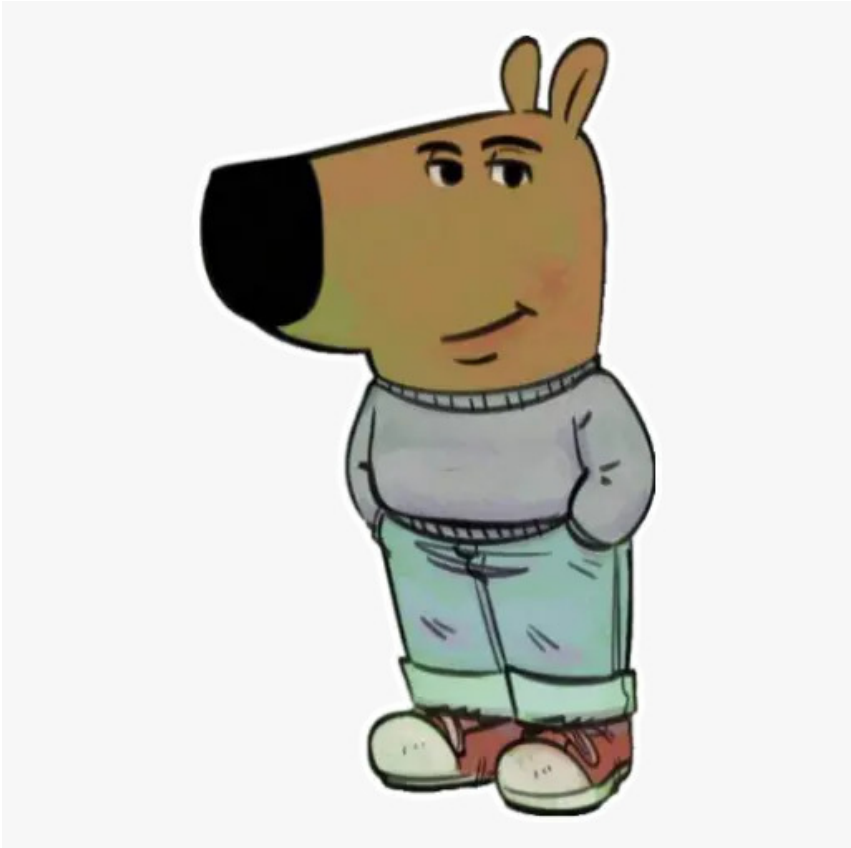
- We obtained a prediction accuracy of ~ 0.76 from our project. Based on the analysis covered in our project our assumption that "given the music tastes and lifestyle of an individual, does music improve/worsen their mental health conditions" is indeed true.
- Our accuracy score depicts that we can detect an individual's mental health correctly with moderately high accuracy based on their living conditions and music tastes.

References

[dat] dataset.
https://www.kaggle.com/datasets/catherinerasgaitis/mxmh-survey-results/data.
[Accessed Nov 2024].

All images not directly related to our work are taken from the internet.

Questions?



Thank You! :)