Block 3 Epic

A Machine-Learning Project around
Music & Mental Health

Group 3
Ellice Nelson - 23371323
Alisia Kazimierek - 23387076
Nandakishore Vinayakrishnan - 23070854

Table of Contents

Dataset Description

Our chosen dataset is titled **‘Music & Mental Health Survey Results’**, which we will refer to as **MXMH**. It originally contained 33 features and 736 samples. This level of samples and features were necessary to bring about conclusive insights. It also had many null values sporadically in each column.

Domain Research

We find studying this dataset incredibly important, as it offers us an incredibly unique opportunity to explore the relationship between music and mental health. We can analyse the dataset to find patterns and correlations that speak volumes about the deep link that exists between music and mental health.

Recently, a therapeutic practice known as “Music Therapy” has begun to grow in popularity. It involves the therapist performing “musical interventions” on a patient by advising the patient to sing, or listen to certain types of music with the intention of achieving a set goal. Music Therapy has begun to show results, an example of such being Russell E. Hilliard’s study on its effects in improving the Quality of Life of patients with terminal cancer, showing that music therapists address the patient’s Quality of Life concerns 30 times more than a standard counsellor, and 28 times more than a nurse. A music therapist, aside from a Chaplain, was the only profession that addressed the patient’s spiritual needs too. In this aspect, a Music Therapist provided spiritual support 16 times more than a counsellor and a nurse.

With the rise of Music Therapy, there has arisen significant demand in understanding how music affects someone. This project seeks to assist that process by analysing a user’s taste in music across several genres, and correlating it to their mental health condition. This information could potentially aid music therapists in tailoring treatment plans in accordance with the needs of their patients.

Research Questions

1. **Can we predict an individual's mental health range based on music preferences?**

2. **If yes, which genres have the greatest effect on mental health?**

We chose this question as our primary research question due to our research on Music Therapy showing us that there exists a gap in research with regards to the connection between a person’s preference in genre to their mental health. With further research, this model could potentially aid music therapists in personalising strategies for their patients. It could also help

stage interventions, as a music therapist can more effectively help a patient if they know that listening to a certain genre is causing a detrimental effect on mental health.

OLAP Questions

1. What is the correlation between anxiety and depression
2. What is the correlation between depression and insomnia
3. What is the correlation between OCD and insomnia
4. What are the top 3 genres in the dataset
5. What is the relationship between bpm and anxiety
6. How does music preferences (genres of music) relate to mental health outcomes?
7. How does the BPM of music relate to the mental health of individuals? Are there specific ranges of BPM associated with lower (or higher) levels of mental health issues?

OLTP Questions

1. What favourite genre group experiences the most anxiety?
2. What age group likes to listen to the most?
3. How many 18 year olds experience some form of mental illness?
4. How many people's favourite genre is rock?
5. How many people in the largest genre experience some form of mental illness?

Data Science Workflow

The DSW is the most important part of any successful data analytics project. Our DSW follows the general structure of any DSW but is better tailored to the task at hand. It covers the main stages that we undertook to bring out our conclusion and gather our insights. It succeeds in tracking the essential and non-negotiable parts of the process we went through.

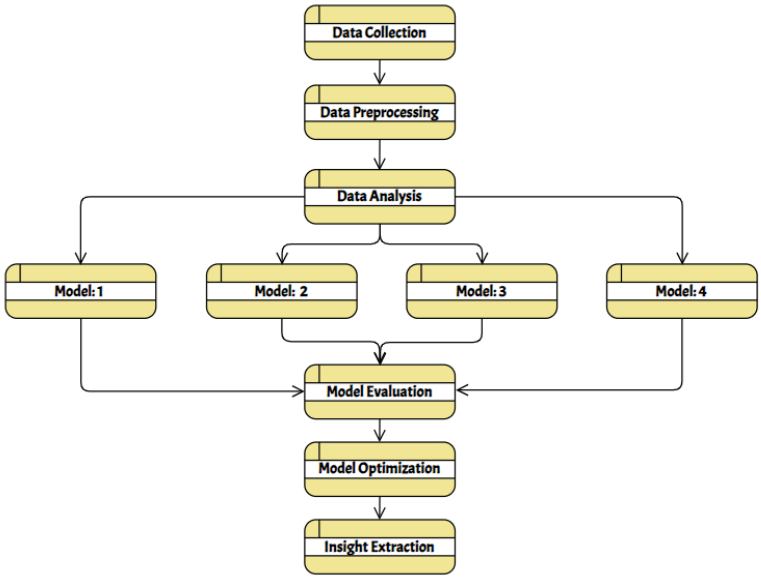


Figure 1 : Data Science Workflow diagram

Data Collection - Finding a suitable dataset in the domain we are looking for.
Data Preprocessing - Remove irrelevant columns, outliers, and null values.
Data Analysis - Visually explore our data and understand the importance/impact of outliers
Models - Implement possibly suitable predictive models for our dataset to answer research questions.
Model Evaluation - See how effective/accurate our models are in answering the research questions and how they could be improved.
Model Optimization - Employ optimization techniques to increase the accuracy of our models.
Insight Extraction - Format our findings in a comprehensible, concise manner.

Contribution

The working of this project followed an agile approach in the form of 3 one-week sprints with the general titles ‘Dataset Research’, ‘Data and Model Exploration’ and ‘Findings and Insights’ respectively. Each sprint constituted different levels of responsibilities for each member.

Member	**Contributions**
Alisia	Research questions, Data exploration/ descriptive analysis, Multiple Linear Regression Model
Ellice	OLAP and OLTP questions, Preprocessing, Random Forest Regression Model

Nanda	Domain Research, Naive Bayes Classification, Multi-Layer Perceptron Classification

Table 1: Contributions Table

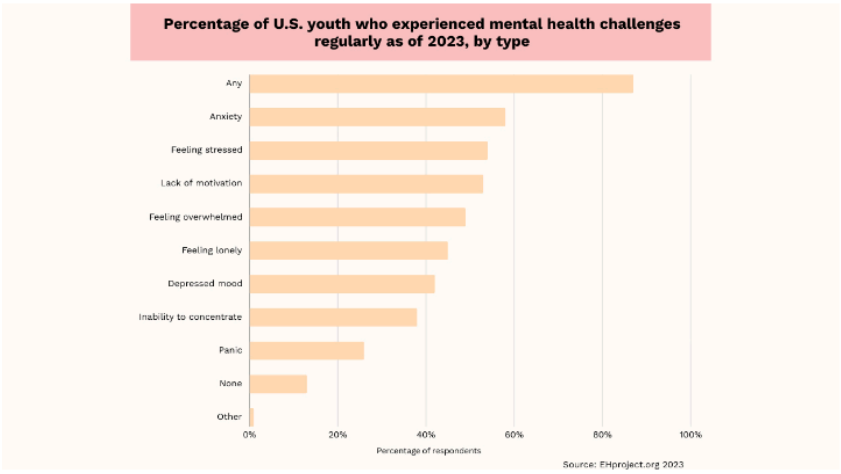
Data Collection

In search of a dataset we searched for a very relevant and important topic. Music is one of the things that bring us all together and we were hypothesising whether it is more than just something that is nice to listen to but also a possible preventative measure in terms of mental health issues. We found our MXMH dataset on Kaggle as an open source dataset with public rights. It was last updated one year ago.

https://www.kaggle.com/datasets/catherinerasgaitis/mxmh-survey-results

Why We Chose MXMH

The main reason we chose MXMH is because of the many possible valuable insights that could potentially be found, regarding possible mental health triggers relating to music. These insights could alter the way doctors care for mentally ill patients or simply help ordinary people protect their mental health in an easy way. Mental health is a very relevant topic and is becoming more and more prevalent as the days go on. The percentage of people with a mental health illness rapidly increases year by year. This could possibly be due to a correlation in the increase of music's prevalence in our lives. Without a doubt, **finding ways to manage and improve mental health** would be a very invaluable tool for the entire world



Over 80% of U.S. youth experience regular mental health challenges. Photo: Team Design

Figure 2: Graph of Percentage of youths who experience mental health challenges

Data Preprocessing

The first step of the preprocessing was removing null values. This was done using the dropna() function. This removed 120 samples from the dataset.

The second step was to identify any outliers and remove them to cater for the main cases.

The third step was removing columns that were irrelevant to the conclusions we were trying to draw. The columns we removed are ‘Permissions’, ‘Timestamp’, ‘While working’, ‘Instrumentalist’, ‘Composer’, ‘Exploratory’. This was done with a simple drop(axis=1) function.

Finally we standardized the data using sklearn’s standard scalar.

Data Exploration

Outlier Identification

Some outliers were found visually. For example, the outliers in the ‘Age’ category were found using a histogram. It can be clearly seen that there are very few entries over 74. As a result, we removed those rows.

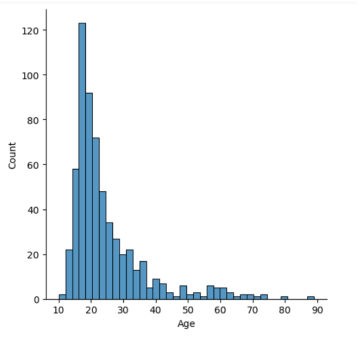


Figure 3: Frequency Distribution histogram of age

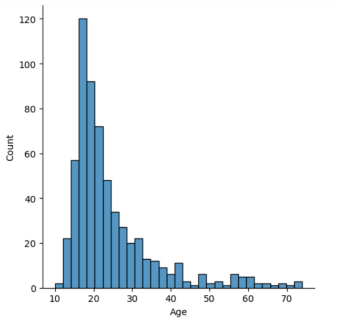


Figure 4:Frequency Distribution histogram after outliers removed

Outliers were identified in the ‘BPM’ feature in a different way using z-scores. A graph would not be able to contain the very large outliers in MXMH. Anything with a z-score greater than 3 or less than -3 was removed. This is because according to standard distribution, 99.5% of data lies between -3 and 3 and in our data analysis, we learn that ‘BPM’ has a very clear standard distribution.

The Interquartile Range (IQR) method was used to identify outliers in the ‘Hours per day’ feature. The IQR method concerns calculating the interquartile range of a given feature, which is the range between the first and the third quartile of the values of that feature. Then, outliers are identified by selecting the values that fall below the lower bound (Q1-1.5*IQR) and those above

the upper bound (Q3+1.5*IQR). The box-plot graph does all this and presents the outliers visually.

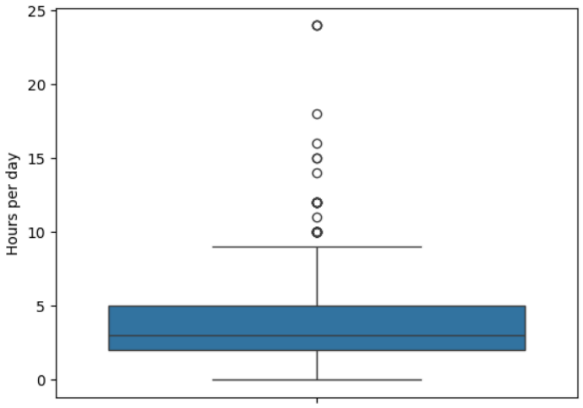


Figure 5: Boxplot of ‘Hours per day’

Mental health

Firstly, the distribution of mental health scores in the dataset exhibits an interesting pattern. We can see in the histogram below that most scores are within the range of 5 to 30. This suggests that most individuals report moderate levels of mental health scores and very few actually reported extreme scores. The mean score of 17.212 confirms this observation. Additionally, our left-skewed histogram for anxiety shows us that most of the individuals had elevated anxiety levels. The histogram for depression has multiple peaks but generally high levels of depressive symptoms. On the other hand , OCD is right-skewed with most respondents reporting 0 as their level of OCD. The distribution of insomnia scores shows us an interesting pattern, with approximately 50 respondents for each level between 1 and 9. However ,nearly 125 respondents report a score of 0, while 25 respondents report a score of 10 This suggests that most of the respondents experience either severe sleep disturbances or none at all.

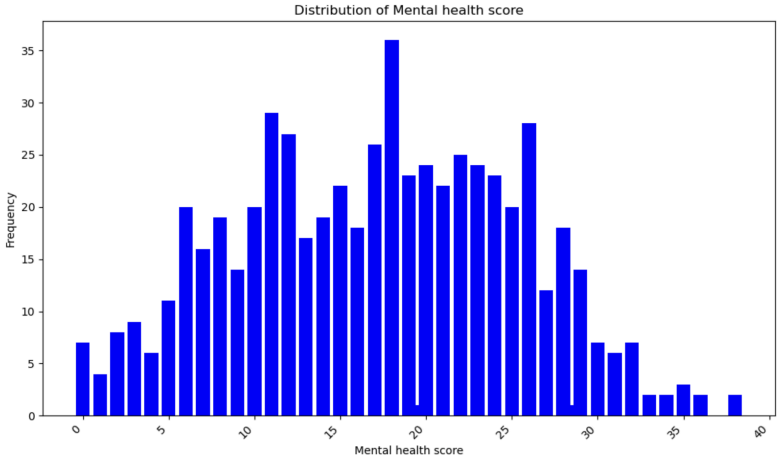


Figure 6: Frequency Distribution Histogram of Mental health scores

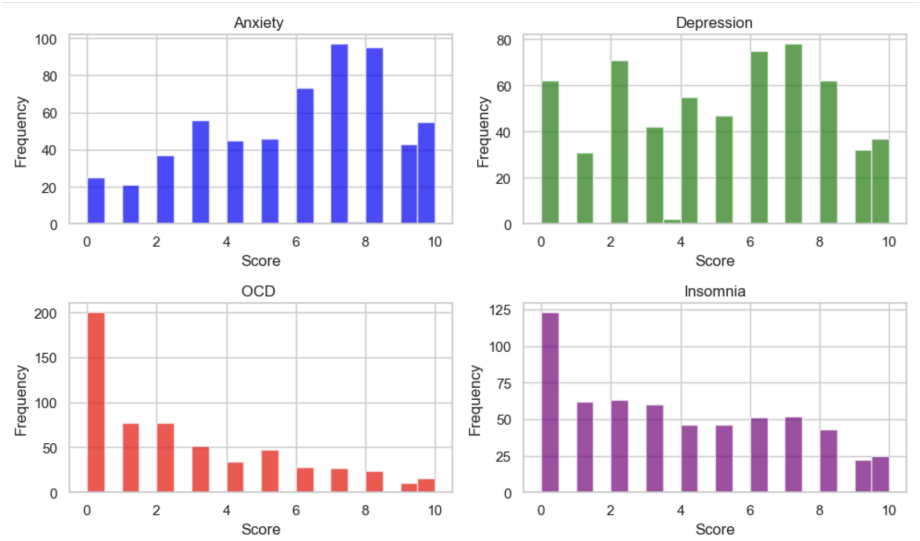


Figure 7 : Frequency Distribution of Anxiety, Depression , OCD and Insomnia

BPM

The BPM distribution in our dataset shows us intriguing patterns in the respondents' music preferences. In the bar chart below you can see that there is an overwhelming amount that favour 120 bpm in their music. However, if we look beyond that we can see that there are various peaks showing us that the respondents have diverse musical tastes.

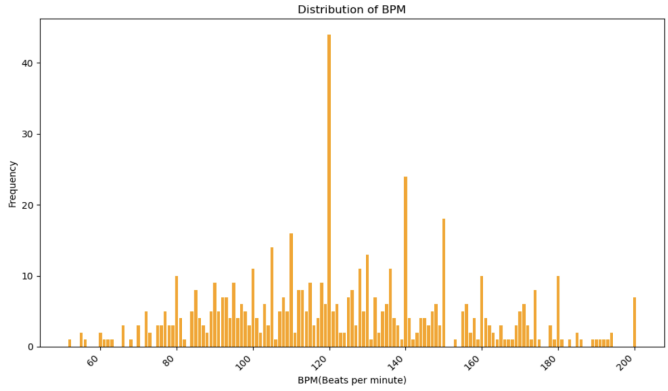
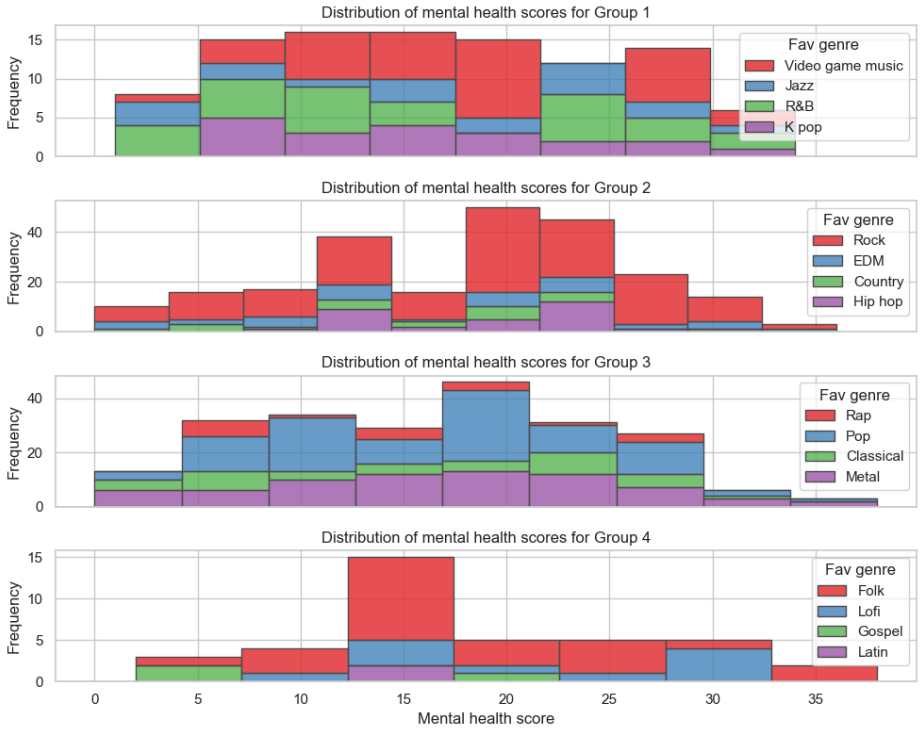


Figure 8: Frequency Distribution Histogram of BPM

Impact of Music Preferences on Mental Health

Since we are attempting to predict an individual's health based on their music preference, it was interesting to see the mental health score people were having based on their favourite music genre. We created a mental health score column , highest being 40. This column is the addition of an individual's level of anxiety , depression , OCD and insomnia . The higher an individual's mental health score , the worse their mental health is. After looking at the dataset in this histogram, we saw that the genre with the lowest mean mental health score is ‘Gospel’ with a



score of 10.67. While the genre with the highest mean mental health score is ‘Lofi’ with a score of 21.70.

Figure 9: Subplots containing stacked histograms for distribution of mental health score across different groups of favourite genres.

By examining the cross-tabulation results between music effects and favourite genre, we were able to see how individuals perceive the impact of music on their mood. People who believe music improved their mood often favoured genres like Rock, Pop and Metal , which suggests energetic and uplifting music could be a factor in bettering a person's mental health. However, there is a small portion that listen to the same type of music but believe it had no effect at all. Further exploration and research into this could give us deeper insights into the effect of music on well-being.

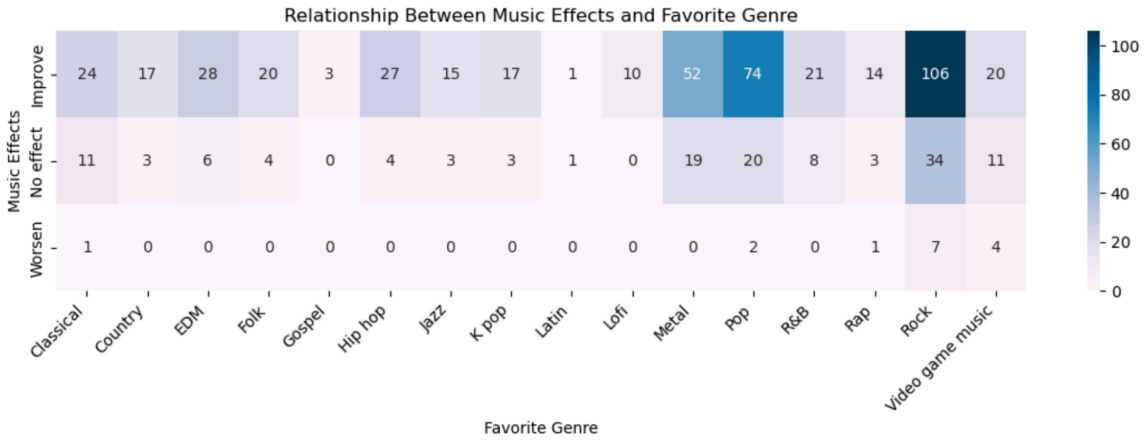


Figure 10: Heatmap to illustrate the relationship between ‘Music Effects’ and ‘Favourite Genre’

Age and Mental Health

After exploring mental health scores across different age groups, we can see an interesting trend. As you can see in the scatter plot below ,Younger individuals tend to have a higher mental health score . Under the age of 30 , we have 6 individuals with a mental health score of 35 or above while the highest with an age above 50 is 24.

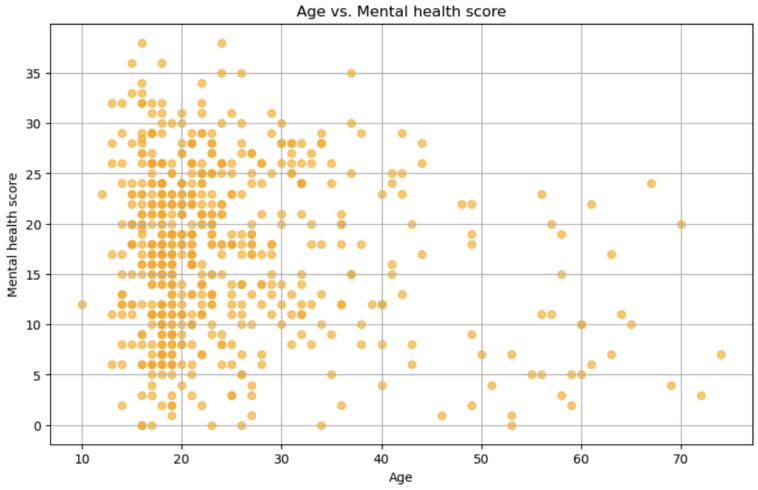


Figure 11: Scatterplot depicting the relationship between Age and Mental health score

Music genres

When visualising the distribution of favourite music genres in the dataset using a pie chart , we can learn alot about the dataset. Rock music emerges as the favourite among the respondents , with 24.7% calling Rock their favourite genre. The group with the best mental health as shown above , Gospel, only has 0.5% calling it their favourite genre. While the group with the worst

mental health , Lofi has 1.7% calling it their favourite genre.The variety in favourite music preferences provides us an insight into the diverse musical tastes among the respondents and highlights the importance of studying the multiple music preferences individuals have and how it relates to their mental health.

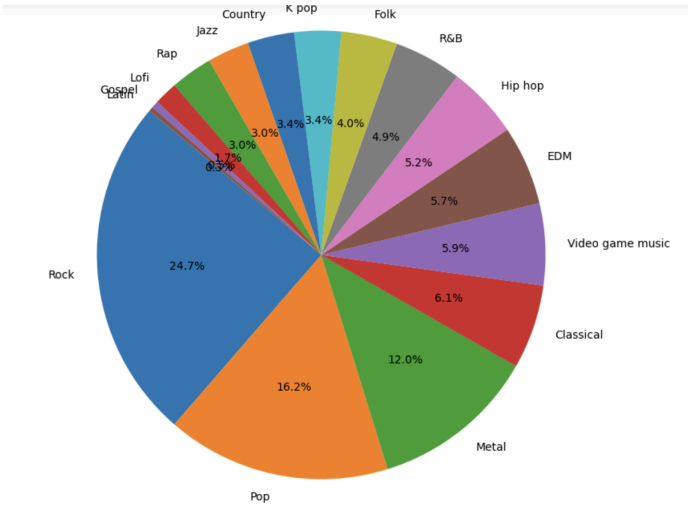


Figure 12: Pie chart of respondents favourite genres

Analysing the frequency of how often an individual listens to each genre of music gives us more knowledge in relation to music effects on mental health. The data below shows how often a respondent listens to different genres. Genres like Metal and Rap are very frequently listened to, while genres like Folk and K pop have the highest statistics for never being listened to.

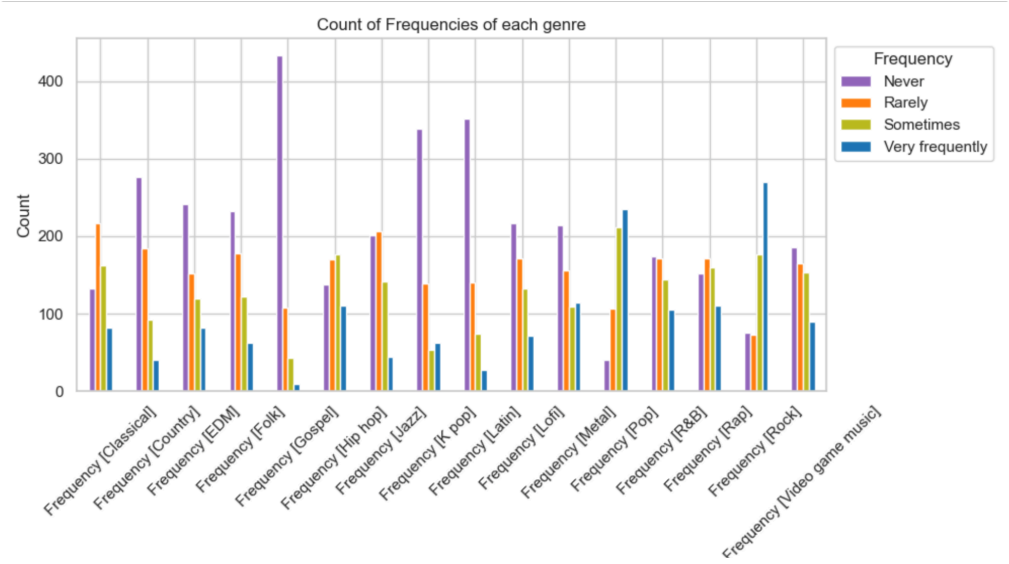


Figure 13: Grouped Barplot displaying frequency of times respondents listen to each genre

Model Implementation

Multiple Linear Regression

From the exploration of MXMH we identified that the data could require a regression model. This led us to attempt to use the Multiple Linear Regression model. We chose Multiple Linear Regression as it allows us to see how strong the relationship is between two or more independent variables and one dependent variable. Independent variables would be favourite music genres, BPM, age and hours spent listening to music etc. While our dependent would be the mental health score. The objective of using multiple linear regression analysis is to use the independent variables to predict the value of the single dependent value.

Random Forest Regression

Another model we attempted to use was the Random Forest model, specifically the regression version as predicting mental health scores would be a discrete value and not a specific class. This was used to attempt to answer our predictive question. Random Forest Regression is a supervised learning algorithm used for regression tasks, where the goal is to predict a continuous output variable based on input features. It constructs multiple decision trees during training and outputs the average prediction of the individual trees. The input features for the model were all the 23 columns left after cleaning and feature engineering, excluding the mental health columns. The target was the ‘Mental health score’ column. The model was then trained on a training set of 60% of this data, test set of 20% and 20% set aside for validation during the optimization process.

Naïve Bayes Classification

At this point, we decided to pivot to classification models to further explore the possibilities of the dataset. The next model we implemented was the Naïve Bayes Classification model. The aim of the model was to answer the question. Naïve Bayes is a type of machine learning model that utilises Bayes’ Theorem. It was chosen due to its computational efficiency, low variance, and its capability to continue working despite the existence of noise or missing values in the dataset. The Gaussian Naïve Bayes model was used, due to its computational efficiency, and due to the fact that it does not require hyperparameter tuning. It also works better with datasets derived from real-world information, as they usually follow a Gaussian(Normal) Distribution, which follows Gaussian Naïve Bayes’ natural assumption that the data passed to it follows such a distribution. Due to the high frequency of numerical features in the dataset, we found that Gaussian Naïve Bayes has a lot of potential for application in the machine learning model, and as discussed later in the report, provides greater accuracy than other Naïve Bayes models like Multinomial Naïve Bayes. The model was trained on 70% of the dataset, tested on another 15% and was validated on the remaining 15%.

Multi-Layer Perceptron Classification

At the recommendation of a professor, we also implemented a MLPClassifier model, in a similar fashion to the Naïve Bayes classifier. 70% of the dataset went to training the model, 15% to test, and the last 15% went to validating. Similarly to the Naïve Bayes implementation, a pipeline was first created. This pipeline contained the earlier defined pre-processing steps in the system, followed by the Multi-Layer Perceptron Classifier with a statement setting the random_state to 42, for reproducibility.

The MLP Classifier, or Multi-Layer Perceptron Classifier, is a neural-network based learning algorithm, typically used for real-world data which could have complex relationships that aren't readily obvious at first glance, like in our dataset. In a Multi-Layer Perceptron Classifier, input data is passed to a "neuron" or a "perceptron", the base foundational unit of the MLP Classifier, which then outputs a weighted sum(with weights passed to it), then performs an "activation function" upon the code to output a value, which it then passes to the next perceptron. It performs these functions forwards and backwards, before giving a prediction as an output.

Model Evaluation

Multiple Linear Regression

We first split our dataset into training and testing sets using an 80/20 ratio with 80% of the data allocated for training and 20% for testing. Then to assess how well the multiple linear regression model works, we analysed several metrics. The mean square error(MSE) tells us the average difference between our predicted mental health scores and the actual scores. We also calculated the R-squared value(R2), the mean absolute error(MAE) and the root mean square error(RMSE). The R2 gives an indication of how good a model fits a given dataset , the RMSE is the standard deviation of the errors which occur when a prediction is made on a dataset and the MAE is the mean size of the mistakes in collected predictions. From these metrics we learnt that multiple linear regression did poorly on all these so optimisation is needed.

```
Mean Square Error: 64.03927155313792
R-squared: 0.02660021288865655
Mean Absolute Error: 6.323278465656104
Root Mean Square Error: 8.002454095659527
```

Figure 14: Evaluation Metrics for Multiple Linear Regression

Despite these evaluation metrics, the scatterplot shows haphazard points that follow no direction.As the independent variable increases, there is no set expectation of what the dependent variable will do, showing that this is a no correlation scatterplot. Hence, it's clear that our current model requires optimisation to improve its predictive accuracy.

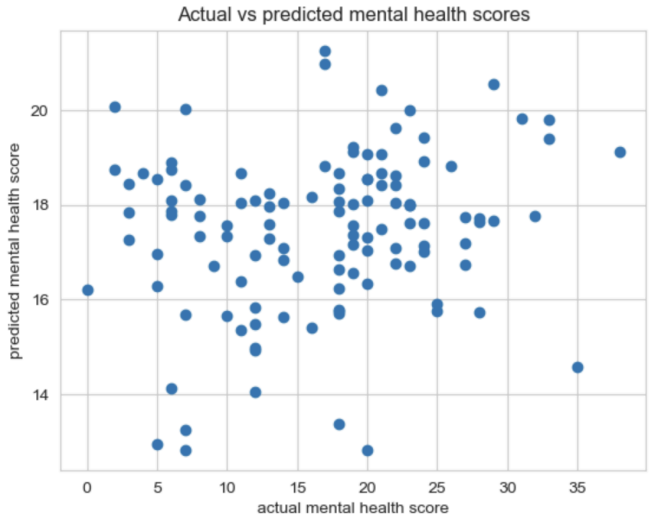


Figure 15: Scatter plot demonstrating the actual mental health score vs the predicted mental health score

Random Forest Regression

To determine the accuracy of the Random Forest Regression model we used a couple techniques. Firstly we graphed the prediction of the test set against the actual target value.

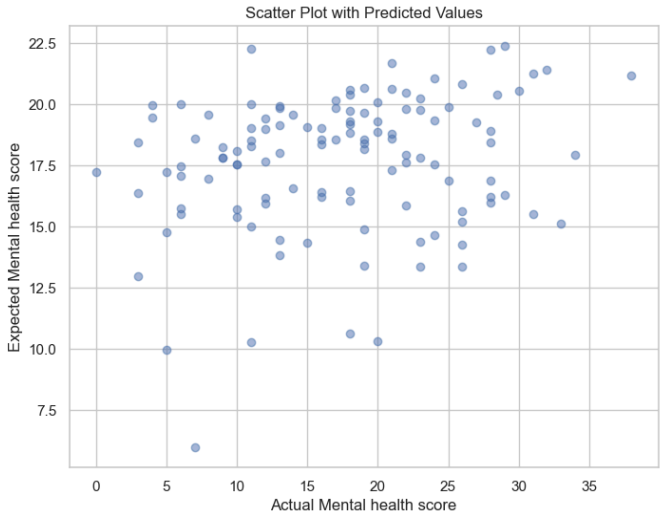


Figure 15: Scatter plot demonstrating the actual mental health score vs the predicted mental health score

This scatterplot shows that there is a low level of accuracy in the model as a higher level of accuracy would have less of a scatter between expected values and actual values.

To evaluate accuracy we also used measurements such as Mean Absolute Error (MAE) Mean Squared Error (MSE) Root Mean Squared Error (RMSE) and R-squared (R2). Low MAE, MSE

and RMSE usually means high accuracy. High R2 tells us the model has higher accuracy as well. The model scored poorly on all accounts.

```
Mean Absolute Error (MAE): 6.54016806722689
Mean Squared Error (MSE): 63.50981890756304
Root Mean Squared Error (RMSE): 7.9693047944951285
R-squared (R2): 0.024285343274595195
```

Figure 16: Evaluation metrics for Random Forest Regression

The model attempted to predict the mental health scores of people and it didn't perform very well. This model answered the question posed with the answer, "We cannot accurately predict an individual's mental health score based on music preferences" . This information tells us that the Random Forest Regression does not suit the data we have naturally or furthermore, the data has a low level of correlation between the input features and output features. The fitting of this model was all done without hyperparameters. To possibly optimise this model we would split our data into training, validation and testing and use the validation set to optimise the hyperparameters or also further remove unimportant features.

Naïve Bayes Classification

We used scikit-learn's accuracy_score and classification_report function to find the accuracy of our classifier function. The accuracy of the model is as follows:

```
Accuracy: 42.25352112676056%
Classification Report for Gaussian Naive Bayes Classifier
               precision    recall  f1-score   support

Zero Category       0.00      0.00      0.00         3
     Very Low       0.42      1.00      0.59        60
          Low       0.00      0.00      0.00        42
     Moderate       0.00      0.00      0.00        36
         High       0.00      0.00      0.00         1

     accuracy                           0.42       142
    macro avg       0.08      0.20      0.12       142
 weighted avg       0.18      0.42      0.25       142
```

Figure 17: Evaluation metrics for Naïve Bayes

The f1-score and the precision seemed to be very low, with the accuracy_score being middling at best. With the given results, we have come to the conclusion that the current model is incredibly inaccurate, and un-optimized. However, there exists potential for this code to be further optimised. For example, we could implement pipelines to ensure consistency, as well as use StandardScaler and SimpleImputer to make the data easier to read by the machine. We could also improve data pre-processing, such as by using methods like StandardScaler and SimpleImputer to make the data easier to read for the machine.

Multi-Layer Perceptron Classification

Similar to Naïve Bayes, we used scikit-learn's accuracy score and classification_report to retrieve metrics for our model. Below are the metrics for the Multi-Layer Perceptron Classifier:

```
MLP Classifier Testing Set Metrics:
Accuracy: 92.13483146067416%

MLP Classifier Validation Set Metrics:
Accuracy: 90.0%
```

```
Classification Report for MLP Classifier:
              precision    recall  f1-score   support

Zero Category       1.00      1.00      1.00         1
     Very Low       0.93      0.93      0.93        40
          Low       0.96      0.96      0.96        24
     Moderate       0.88      0.91      0.89        23
         High       0.00      0.00      0.00         1

     accuracy                           0.92        89
    macro avg       0.75      0.76      0.76        89
 weighted avg       0.91      0.92      0.92        89
```

Figure 18: Evaluation Metrics for Multi-Layer Perceptron Classification

Immediately obvious is that the accuracy of both classifiers is far higher than the accuracy of the Naïve Bayes Classifier, before optimization. The classifiers show exemplary performance, showing that they are a much better choice for predicting the mental health category of a person based on their musical preferences. However, there still exists plenty of potential for further optimization, such as through Hyperparameter Tuning, etc.

Model Optimization

Multiple Linear Regression

The optimisation of machine learning models is crucial for enhancing their predictive performance . To optimise our multiple linear regression model for predicting mental health score , we did two regularisation techniques , Lasso and Elastic Net regression . Regularisation improves our model to work on unseen data by ignoring the less important features , minimises the validation loss and tries to improve the fit of the model.It also avoids overfitting by adding a penalty to the model with high variance.

In the process of optimising this model, we used a grid search technique to systematically search through a range of hyperparameters so we can find the optimal values. For Lasso, we focused on the regularisation parameter (alpha) which controls the strength of the penalty applied to the coefficients.

For Elastic Net, it combines the strengths of Lasso and Ridge regression techniques. It keeps the feature selection quality from the lasso penalty as well as the effectiveness of the ridge penalty. By adjusting its parameters, Elastic Net finds the right balance between these penalties, making the model more accurate and stable for predicting outcomes.

Due to these optimisation techniques, we were able to enhance our performance metrics but only by a small margin. Both Lasso and Elastic Net regression models demonstrated a higher R-squared value when compared to the unoptimised model.

Figure 19: Evaluation metrics for Lasso

```
Lasso - Mean Squared Error: 63.91864586613501
Elastic Net - Mean Squared Error: 62.535714237376
Elastic Net - R-squared: 0.04945435122559794
```

Figure 20: Evaluation metrics for Elastic Net

Random Forest Regression

To optimise this model, we tried to tune the hyperparameters to the optimal ones for the data. The hyperparameters we were concerned with were 'bootstrap', 'max_depth', 'max_features', 'min_samples_leaf', 'min_samples_split' and 'n_estimators'. We then tuned them by using grid search to find the best combination of hyperparameters. In doing this we found out the optimal parameters were ['bootstrap': True, 'max_depth': None, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 10, 'n_estimators': 50]. This change caused an increase in accuracy with regard to all previously mentioned accuracy measures as seen below in fig 23

In another attempt to increase performance and accuracy, we explored feature importance and achieved the graph in fig 21. This graph showed us that the most important features in our X input were Age, Hours per day, BPM and Fav genre. With this, we removed some of the insignificant features such as 'Foreign languages', 'Frequency [Gospel]', 'Frequency [K pop]', 'Music effects' and 'Frequency [R&B]', alongside others. After testing the accuracy metrics after this change, it was found that there was a difference between these results and the result of the regressor after hyperparameter optimization. Further after applying hyperparameter optimization to this model, the accuracy metrics actually did not improve and worsened. Once again the answer achieved was “We cannot accurately predict an individual's mental health score based on music preferences".

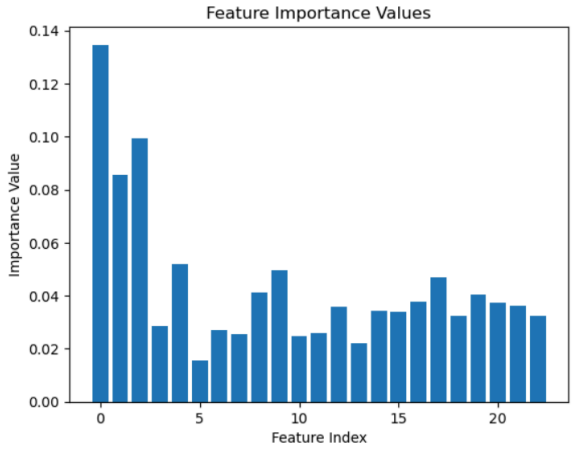


Figure 21: Bar graph of feature number vs importance

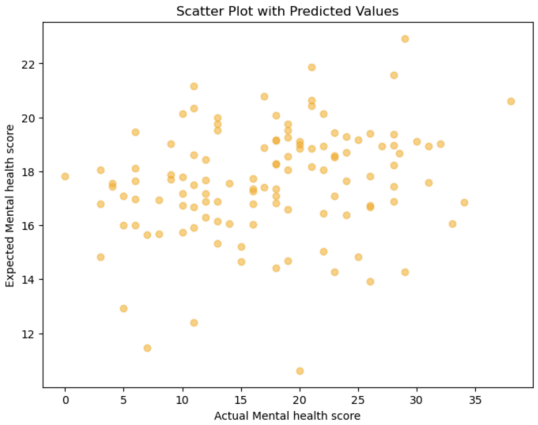


Figure 22: Scatter plot of actual mental health score vs the predicted mental health score

```
Mean Absolute Error (MAE): 6.368316591999006
Mean Squared Error (MSE): 61.46860980926818
Root Mean Squared Error (RMSE): 7.840191949771905
R-squared (R2): 0.055644866398827486
```

Figure 23: Evaluation Metrics for Random Forest Regression after hyperparameter tuning

Naïve Bayes Classification

To optimise the Naïve Bayes learning model, the first task was to encode categorical features into a numerical format, which is easier to read for a machine. Then, the dataset was split into a training, testing, and validation set. This allowed greater testing, to see how the data would react to completely unseen data a second time. This reduces the chance of false results or incorrect accuracies.

Finally, a pipeline system was implemented, ensuring that the data in the training, testing, and validation sets are all given the same preprocessing steps. The dataset is run through the SimpleImputer and StandardScaler functions using a pipeline, ensuring consistency across the training, testing, and validation set. The use of StandardScaler also helps make the data far easier to read by the machine.

This optimization has greatly increased the accuracy of the model. The results of the optimised model for the testing and the validation sets are as follows:

```
Naive Bayes Testing Set Metrics:
Accuracy: 86.51685393258427%
```

Figure 24: Accuracy for Naive Bayes on testing set

```
Naive Bayes Validation Set Metrics:
Accuracy: 87.77777777777777%
```

Figure 25: Accuracy for Naive Bayes on validation set

```
Classification Report for Gaussian Naive Bayes Classifier:
              precision    recall  f1-score   support

Zero Category       1.00      1.00      1.00         1
     Very low       0.89      0.80      0.84        40
          Low       0.77      0.96      0.85        24
     Moderate       0.95      0.87      0.91        23
         High       1.00      1.00      1.00         1

     accuracy                           0.87        89
    macro avg       0.92      0.93      0.92        89
 weighted avg       0.87      0.87      0.87        89
```

Table comparing actual value to predicted value

Figure 26: Evaluation metrics for Gaussian Naive Bayes

Multi-Layer Perceptron Classification

The first area that could use optimization was hyperparameters. Using scikit-learn's GridSearchCV, we were able to iterate through and find the best possible combination of hyperparameters for the program, which are then printed out for debug purposes. Pipelines from the Naïve Bayes implementation were re-used in this implementation, ensuring consistent data-preprocessing across all sets. With these changes, we can see that the performance of the MLP Classifier has increased, seemingly marginally, only due to the already exemplary performance of the model earlier.

```
MLP Classifier Testing Set Metrics:
Accuracy: 95.50561797752809%
```

Figure 27: Accuracy for MLP Classifier on testing set

```
MLP Classifier Validation Set Metrics:
Accuracy: 96.66666666666667%
```

Figure 28: Accuracy for MLP Classifier on validation set

```
Classification Report for MLP Classifier:
                precision    recall  f1-score   support

 Zero Category       1.00      1.00      1.00         1
      Very Low       0.95      0.97      0.96        40
           Low       1.00      0.96      0.98        24
      Moderate       0.92      0.96      0.94        23
          High       0.00      0.00      0.00         1

      accuracy                           0.96        89
     macro avg       0.77      0.78      0.78        89
  weighted avg       0.95      0.96      0.95        89
```

Figure 29: Evaluation Metrics for MLP Classifier

Insights Extraction

Our goal was to predict an individual's mental health score based on their music taste preferences. However, the evidence demonstrates that we cannot reliably predict mental health rankings from music taste in this dataset alone. Mental health is a very complex challenge. There must be more important factors that play in the outcome of a person's mental health. These factors may be different features that are not contained in this dataset. Since we do not have access to this information, we cannot precisely predict what mental health an individual has. However, we can predict whether an individual has a very low, low, moderate or high mental health score. This categorical prediction can still be very useful in many cases and serves as a more universal unit.

Design Choices

Data Structures

Data structures play a significant role in machine learning and data analytics. The most significant data structures we used were decision trees, pandas data frames and pipelines.

Decision Trees

Decision trees were used in the Random Forest model used. This data structure was used because they are perfect for the purpose. Each tree has a certain condition, for example ‘Hours per day’ > 4. If this condition is fulfilled, the decision taken is the one to the left. Random Forest makes use of this by using many decision trees to decide what output should be obtained. This allows multiple factors (feature in this case) to be considered before deciding on such an output. The hierarchical structure is also great for considering features with more importance first. Hence, decision trees are very well fitting for Random Forest Regression and more generally, our project.

Pandas Data Frames

A DataFrame is a data structure that organises data into a 2-dimensional table of rows and columns, much like a spreadsheet. We chose this data structure because they are a flexible and intuitive way of storing and working with data. It made our hard-to-comprehend csv of data entries into a clean table that is perfect to use. Data frames are also compatible with almost every machine learning task and every machine learning library, making it the perfect choice for our purpose. These are better than the use of numpy arrays due to better visual aspect and the ability to use functions like “head” and “describe”

Pipelines

The Scikit-Learn library introduces a new Data Structure, the pipeline. It is used to link a series of processing steps into one singular “Pipeline” object. Typically, it follows the structure of “Transformer_1, Transformer_2, Transformer_3……Transformer_N, Estimator”. This data structure was implemented in the Naïve Bayes and Multi-Layer Perceptron learning model, such that there were two Transformers, scikit-learn’s SimpleImputer and StandardScaler, followed by the Gaussian Naïve Bayes/MLP classifier. This gives a concrete, consistent and streamlined machine learning workflow, ensuring the same steps are run across the testing, training, and validation sets. This code also has applications in other Machine Learning models as it is possible to fit hyperparameter tuning into the pipeline. Since Gaussian Naïve Bayes does not need hyperparameter tuning, however, such steps were not implemented. Since it comes with the scikit-learn library, pipelines are compatible with every machine learning model, making it ideal for machine learning workflows.

Assumptions

No assumptions were made throughout the course of this project. The design brief covered all aspects of this project, which was followed by all group members.

Bibliography

- American Music Therapy Association. (n.d.). About Music Therapy & AMTA. Retrieved March 11, 2024, from https://www.musictherapy.org/about/
- Russell E. Hilliard, The Effects of Music Therapy on the Quality and Length of Life of People Diagnosed with Terminal Cancer, *Journal of Music Therapy*, Volume 40, Issue 2, Summer 2003, Pages 113–137, https://doi.org/10.1093/jmt/40.2.113
- Jacobsen, J., (2023). Mental Health Statistics 2024 — Quick Facts & Statistics About Mental Health. [Online] Available at: https://www.ehproject.org/mental-health/mental-health-statistics-2023 (Accessed: March 13, 2024).
- Webb, G.I., Keogh, E. and Miikkulainen, R., 2010. Naïve Bayes. *Encyclopaedia of machine learning*, *15*(1), pp.713-714.
- Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011.
- Scikit-Learn. sklearn.preprocessing.StandardScaler. Retrieved March 17th, 2024 from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html
- Bhadauriya, R. (2021) *Lasso ,Ridge & Elastic net regression: A complete understanding (2021)*, *Medium*. Available at: https://medium.com/@creatrohit9/lasso-ridge-elastic-net-regression-a-complete-understanding-2021-b335d9e8ca3 (Accessed: 16 March 2024).
- *Learn how multiple linear regression works in minutes* (2023) *Dataaspirant*. Available at: https://dataaspirant.com/multiple-linear-regression/ (Accessed: 10 March 2024).
- Scikit-Learn. Neural Network models. Retrieved March 19th, 2024 from https://scikit-learn.org/stable/modules/neural_networks_supervised.html
- *What is lasso regression?* (no date) *IBM*. Available at: https://www.ibm.com/topics/lasso-regression (Accessed: 14 March 2024).