Characteristic Input Patterns as Decision Boundaries for Explainable Rule Generation through Sensitivity and Distance-based Metrics

Duy Khuu

Research School of Computer Science
Australian National University
u6380923@anu.edu.au

Abstract. To determine if certain neurological signals can be used to predict features of stimuli a particular individual is currently exposed to, both a fully-connected and a convolutional neural network model were trained on sensor data collected from individuals whilst exposed to certain music stimuli to predict the genre of the song the individual was listening to while their brain activity was recorded. Characteristic inputs were generated for each music genre and two methods are proposed for explaining the model: 1. Sensitivity analysis in the form of causal index (impact on final result) to produce decision rules (using characteristic input values as boundaries) and 2. Calculating the distance from a given input to each characteristic input and classifying based on the nearest characteristic input's class. Both methods were able to predict 43.3% and 40.3% of their corresponding model's training predictions correctly, however the F1-score of the later method was over 0.20 higher, indicating a difference in understanding of all classes. A similar training prediction accuracy (40.8%) was obtained using the second method on a more complex CNN model, but caused the F1-score to drop from 0.631 to 0.274 – implying a poorer understanding compared to the fully-connected model used. While the resulting predictions perform better than random choice, they still fall far short of previous attempts to generate meaningful decision rules. The performance loss likely does not justify the increased explain-ability of the model. However, this may be due to the chosen techniques for resolving classes based on the input rules satisfied, along with the informative-ness of the dataset used. This suggests that the chosen techniques may provide increased performance given a better choice of resolution method and/or feature set, prompting further investigation.

Keywords: neurological analysis, neural activity prediction, neural networks, classification, audio stimuli, characteristic input, causal index, model interpretation, sensitivity analysis

1 Introduction

1.1 Motivation

Neurological analysis of brain activity can occur whilst partaking in an activity that involves either visual or audio-based stimuli, with neurologic feedback captured to determine the effect of each stimuli [1]. Results from such studies could be used to help determine when specific signals are emitted from the brain, which could aid projects that aim to reproduce certain behaviour. Examples of this include isolating particular visual cues or types of audio that can soothe individuals and calm them down; or isolating indicators of epileptic seizures to prevent situations that result in them triggering [2].

However, training neural network models to collect the required brain activity data can be expensive, both in time and resources. As of April 2021, the EPOC Flex Saline Sensor Kit Headset – a model which contains over 32 sensors - costs at least $1699 USD [3]. This can make data collection very expensive if a large amount of data is required in a short amount of time as multiple headsets would be needed to collect participant data in parallel. While cheaper models exist, such as the $299 USD EMOTIV Insight 5 Channel Mobile Brainwear [4], these models have far fewer sensors (5 in the case of the Brainwear), leading to less fidelity in the captured data.

In addition to this, due to the nature of neural network models it can be difficult to determine the effect of specific features and their relation to the output as model predictions are the result of complex relationships between many of the different inputs [5]. An understanding of specific decision boundaries or rules is vital in this scenario to determine which areas of the brain should be stimulated in order to produce an intended result. While it can be argued that further understanding is not required as long as the model performs well, this is not helpful if one is trying to reduce the feature space in order to reduce the cost of the training process (such as determining if all sensors currently being used are meaningful in the prediction process, or if a subset contained by a cheaper headset is enough).

Engelbrecht and Viktor [6] proposed a decision boundary detection method which uses the gradient effect of each input feature on the output as a scoring method to determine rules for specific class boundaries. However, this involves

computing the derivative of every output for each feature of all training inputs in the model. Not only can this lead to significant processing time to extract these rules, but the sheer number of generated rules can lead to very convoluted and complicated decision boundaries that defeat the purpose of coming up with them in the first place (breaking down the model into explainable decision rules).

Instead of generating these decision rules for each training input, this project proposes to instead determine characteristic examples of each class, generated by taking the mean of the inputs in each predicted class of an already trained neural network model – a method used by Gedeon and Turner [7] which proved to be highly effective in generating useful rules for predicting final exam grades of students based on their previous marks. These representational inputs would be utilised with the Engelbrecht and Victor's decision boundary approach [6] to produce a much smaller rule set, with each rule ranked by its impact on the output (causal index). However, this may lead to situations where a particular input satisfies decision rules for multiple genres. Multiple approaches were proposed to resolve these conflicts: one Causal Index Score based (impact of a particular input on the final output) and the other solely based on the number of decision rules satisfied for each class. The results would be compared with a baseline neural network model to determine if the increased explain-ability of such rules is worth the trade-off in model performance, along with which genre resolution method performs best on the given data.

1.2 Problem Description

The aim of this project is to determine whether a neural network trained on a subset of music genre classification data can be reduced into a small number of decision rules, based on characteristic inputs of the dataset, that are both meaningful and accurate when compared to the original model.

Rahman, Gedeon, Caldwell and Jones [2] utilised a 14-channel headset to capture brain activity from 24 participants who were presented with audio-stimuli in the form of music from 3 different genres: Classical, Instrumental and Pop. The sensors captured data from 4 different locations in the brain (Pre-Frontal Lobe, Frontal Lobe, Temporal Lobe and Occipital Lobe). The resulting data was further transformed into multiple features including linear features such as mean, skewness and root mean square along with non-linear features such as Shannon entropy. This resulted in 26 features being extracted from each of the 14 sensors – leading a total of 364 features per song. While only the 150 most informative features were chosen, this was still a relatively large feature space spanning across many sensors in the different brain areas. These features were used to determine if the results from the sensors were enough to predict the genre of the song a particular individual listened to. The results were very strong, with their neural network model achieving over 94% accuracy and F-measure scores.

A heavily reduced subset of these features were used in this paper, containing data from only one of the sensors, which had its value recorded 4 times each second. Despite this, the task was the same – using the provided input features, is it possible to predict the genre of the song the individual was listening to. Two models were trained, a simple Fully-Connected Neural Network using various aggregations of the value of the sensor across the whole song as features (e.g. Mean, Standard Deviation, Integrated Signals) (Aggregate Feature dataset) and another Convolutional Neural Network based model, which was provided the actual values of the sensor at each timestamp (Temporal Feature dataset).

2 Methodology

2.1 Data pre-processing

2.1.1 Aggregate Feature dataset

The dataset used was a reduced version of the aforementioned musical stimuli dataset. Preliminary analysis was conducted by plotting the distribution of each feature – separated by music genre. In doing this it was found that there were significant outliers for the Fuzzy entropy feature – four of the inputs had a value of 65535, which was far greater than any of the other values for this feature (Figure 1), suggesting an error in the data. While aggregating these outlier values with the mean was considered, the dataset contained over 570 observations so the inputs with these values were dropped instead due to making up only a small portion of the original data.

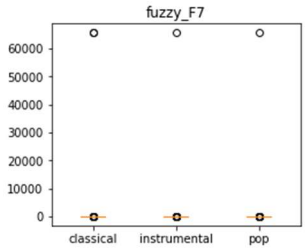

Fig. 1. Box plot of the distribution of the Fuzzy entropy feature, separated by music genre.

All other features did not appear to suffer from such severe outliers (See Appendix 1 for the full plots), so the data was min-max scaled between [0, 1] and music genre labels rescaled from {1, 2, 3} to {0, 1, 2}, where 0 = Classical, 1 = Instrumental and 2 = Pop.

The data was then split into training (75%) and testing sets (25%), with the training set further split into 10 folds for cross-validation of the model to produce a more accurate performance indicator of the model due to the reduced likelihood of overfitting due to the result of a favourable random split.

2.1.2 Temporal Feature dataset

Like the Aggregate Feature dataset, the values at each timestamp (all numerical) were compared to determine if there were any significant outliers. While there were no extreme values, there appeared to be some noise (such as the letter 's') at the end of columns. This was solved by truncating the dataset, as the noise did not appear while the song was playing. Additionally, the data was trimmed to the length of the shortest song (202 seconds) to prevent the model from trivially learning which song is playing by its ending time (if the data is zero-padded) due to the small number of songs (12) used in the experiment.

Since data was captured 4 times each second and each song was truncated to 202 seconds, this lead to 808 features for each pattern (an individual listening to a particular song). Convolutional Neural Networks imply that there should be some sort of natural ordering to the data (e.g. images shaped width x height), so each entry was reshaped to a 4 x 202 matrix where each row represents the signals captured over one second of the song.

Tthe data was min-max scaled between [0, 1] and the same genre labels appended as the Aggregate Feature dataset. The data was split into training (70%), validation (15%) and testing (15%) sets, using a stratified sampling method to ensure a balanced number of classes in each set.

2.2 Baseline Neural Network

To obtain a baseline of the performance of the Aggregate Features, a fully-connected neural network model with one hidden layer was trained on the training set. Both the number of neurons in the hidden layer, along with learning rate were treated as hyperparameters to be tuned. This was done by experimenting with various combinations of each hyperparameter (hidden neurons ∈ {26, 30, 40, 50, 75, 100, 125, 150, 175, 200, 250, 300}, learning rate ∈ {0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.05, 0.01, 0.005}) and recording performance metrics after 1000 epochs. Both Stochastic Gradient Descent [8] and Adam [9] optimisers were used, with Adam significantly outperforming SGD in accuracy.

Hidden neurons	Learning Rate	Accuracy	Precision	Recall	F1
30	0.1	0.668	0.380	0.381	0.368
50	0.1	0.668	0.371	0.376	0.348
40	0.1	0.661	0.359	0.361	0.346
26	0.1	0.651	0.351	0.348	0.339
26	0.05	0.618	0.372	0.380	0.364

Fig. 2. Best 5 performing hyperparameter combinations sorted by average **validation accuracy** across all folds

Hidden neurons	Learning Rate	Accuracy	Precision	Recall	F1
30	0.05	0.562	0.406	0.408	0.380
30	0.2	0.568	0.403	0.404	0.379
100	0.1	0.575	0.403	0.418	0.378
40	0.05	0.587	0.396	0.389	0.378
40	0.2	0.578	0.387	0.399	0.377

Fig. 3. Best 5 performing hyperparameter combinations sorted by average **validation F1-score** across all folds

Taking into account both the accuracy and F1-scores of each hyperparameter combination (Figures 2 and 3), it was clear that a hidden layer size of 30 neurons produced the best performance in both categories. A learning rate of 0.1 also performed very well on small hidden layer sizes. The combination of 30 hidden neurons and a learning rate of 0.1 resulted in the best performance in the accuracy metric and falling only slightly behind the best performers in F1-score, leading to these being chosen as the final hyperparameters for the model.

Evaluating the model with the chosen hyperparameters on the testing data produced an accuracy of 60.1% and F1-score of 0.394.

2.3 Characteristic Inputs and Causal Index Decision Rules

To determine characteristic inputs for each of the three genres, the final network was used to predict the classes of each of the patterns in the training data. Each pattern was grouped by its predicted class and the mean of the inputs of each group were taken to create characteristic inputs to represent each genre.

The impact of feature f for input x_i on final output $y_i(c)$ is given by the derivative $\frac{\partial y_i(c)}{\partial x_i(f)}$. These were calculated for every feature of the characteristic inputs for the corresponding output classes (genres) they represent. The absolute values of each derivative (referred to from now on as causal index score) were used to order the rules based on their impact on the final decision.

If a particular input feature is chosen to be used as a decision rule:

- $\frac{\partial y_i(c)}{\partial x_i(f)} > 0$: Rule is $x_j(f) \geq x_i(f) \rightarrow y_j(c) = y_i(c)$
- $\frac{\partial y_i(c)}{\partial x_i(f)} < 0$: Rule is $x_j(f) < x_i(f) \rightarrow y_j(c) = y_i(c)$

Since this is a multi-class classification problem, the resulting rules were adjusted to account for overlapping intervals. For example if $feature_1 \geq 50$ implies classical music, but $feature_1$ >= 75 also implies instrumental music, the former rule will be adjusted to $50 \leq feature_1 < 75$ to imply classical music.

2.4 Decision Rule Selection and Class Resolution

Different selection methods were evaluated to further refine the generated decision rules into a smaller subset for use. In addition to the 75 decision generated rules (25 features for each of the 3 characteristic inputs), a reduced ruleset of only the top 50 rules (ranked by causal index score) was also evaluated. This number was chosen as the top 25 rules consisted of all the rules generated for characteristic input 2 (Pop). To determine if this skewed the predictions, another ruleset was generated that selected the top 15 rules for each characteristic input – ensuring a balanced representation of each genre in the rule set.

Another consideration to take into account is which class should be predicted if a particular input satisfies rules from multiple genres. Two separate Class Resolution methods were evaluated with the first being choosing the genre that the majority of the input's satisfied rules come from (Rules Satisfied) – however this may lead to bias if the ruleset contains a large number of rules from one class. To try and mitigate this, another scoring system (Causal Index Sum) was also assessed where each input has a score for all three genres. If a rule is satisfied for a particular genre, that genre's score increases by the square root of the causal index score of that rule. The final prediction is the genre with the highest score. The choice to square root the scores was done to reduce the impact of one very highly ranked feature for one genre if many of the other features match rules for a different genre.

2.5. CNN Neural Network

In addition to images, convolutional techniques have been successful in classifying time series data such as Becker et. Al [10], who were able to achieve 92.53% prediction accuracy on a 10-class classification problem on an audio frequency signal dataset.

A baseline model was also established using the processed Temporal Features. Recall that each pattern is a 4 x 202 matrix where each row represents the signals captured over one second. The pipeline for the convolutional network is as follows: convolution layer with 2x2 kernel and 3 output channels, Max Pool with a filter and stride of 3, second convolution layer with 3x1 kernel and 6 output channels and padding of 1 above and below. The output of the second convolution layer was then fed into a fully-connected Neural Network with 2 hidden layers, with the first hidden layer's size being a hyperparameter, the second hidden layer containing 10 neurons and the output layer containing 3 neurons (representing each music genre).

Hyperparameter tuning was done by comparing performance across combinations of number of epochs $\in \{50, 100, 150, 250\}$, hidden neurons in first hidden layer $\in \{15, 50, 100, 150, 250, 300\}$, learning rate $\in \{0.0003, 0.0005, 0.001, 0.005, 0.01, 0.05\}$ and activation function (sigmoid or tanh).

Epochs	Hidden neurons	Learning Rate	Activation Function	F1	Accuracy
250	50	0.005	sigmoid	0.494048	0.517241
250	150	0.001	tanh	0.483529	0.482759
150	250	0.005	tanh	0.482456	0.482759
200	300	0.001	sigmoid	0.45749	0.482759
200	250	0.005	tanh	0.448729	0.448276

Fig. 4. Best 5 performing hyperparameter combinations sorted by average **validation F1**

The selected model was trained with 250 epochs, contained 50 hidden neurons in the first hidden layer using a learning rate of 0.005 and the sigmoid activation function. It was chosen due to performing the best in both accuracy and F1-score on the validation set. Evaluating the final model with these hyperparameters on the testing set produced an accuracy of 62.07% and F1-score of 0.618. While the accuracy was only slightly higher than the Baseline Neural Network model, the F1-score was over 0.20 higher, suggesting a better representation of classes in the model predictions.

3 Results and Discussion

3.1. Causal Index Resolution Methods

The fully-connected neural network was trained on the best performing hyperparameters found in 2.2 and evaluated on the held out testing data. Characteristic inputs of this model were then generated from its predictions and decision rules extracted from the causal index scores mentioned in 2.3. The resulting rules were then split into multiple rule sets, which were used to predict the genres for both the training and testing sets used for the full-connected neural network. Results were compared to the predictions of the baseline neural network Recall that the "balanced" rule set contains the top 15 rules for each genre (ranked by causal index).

Rule Set	Resolution Method	Accuracy	F1
Baseline NN		**0.603**	**0.603**
All rules	Rules Satisfied	0.433	0.387
Top 50	Rules Satisfied	0.37	0.292
Balanced	Rules Satisfied	0.356	0.327
All rules	Causal Index Sum	0.291	0.150
Balanced	Causal Index Sum	0.291	0.150
Top 50	Causal Index Sum	0.291	0.150

Fig. 5. Rule Set and Class Resolution combination results on **training set**, sorted by accuracy

Rule Set	Resolution Method	Accuracy	F1
Baseline NN		**0.601**	**0.394**
Balanced	Rules Satisfied	0.419	0.401
All rules	Rules Satisfied	0.384	0.352
Top 50	Rules Satisfied	0.349	0.266
All rules	Causal Index Sum	0.300	0.154
Balanced	Causal Index Sum	0.300	0.154
Top 50	Causal Index Sum	0.300	0.154

Fig. 6. Rule Set and Class Resolution combination results on **test set**, sorted by accuracy

Looking at Figures 5 and 6, Rules Satisfied outperforms Causal Index Sum in both metrics on both the training and testing set of data. Interestingly, all Causal Index Sum predictions resulted in the exact same accuracy and F1-scores. Upon further inspection this was due to returning the same predictions for all possible inputs. This is likely due to the fact that the rules with the highest causal index were all based on the characteristic for pop music (Figure 7). This meant that the predictions would be skewed towards this genre, as satisfying pop feature rules is weighted much higher than other genres. This suggests that the Causal Index Sum scoring method is not suitable and further investigation needs to be done on how to augment the causal scores (such as fixed constants for each causal index rank as opposite to adding the square root of the actual causal index score) in order to produce better decisions based on the given decision rules.

Class	Feature	Sign	Value	**Causal Index**
Pop	var_F7	>=	0.049719	**2.01069**
Pop	ssi_F7	>=	0.048989	**1.97976**
Pop	iqr_F7	<	0.076183	**1.52344**
Pop	log_F7	>=	0.049178	**1.43621**
Pop	var_F7.1	>=	0.047259	**1.34974**
Pop	skw_F7	<	0.494721	**1.11648**
Pop	fuzzy_F7	<	0.096297	**0.816108**
Pop	dasdv_F7	<	0.098267	**0.781684**
Pop	aac_F7	>=	0.081239	**0.68325**
Pop	sum_F7	<	0.087935	**0.645168**

Fig. 7 Top 10 decision rules sorted by causal index score

Amongst the Rules Satisfied predictions, the All Rules dataset was able to perform relatively well on both the training and test sets, achieving above a 38% accuracy and 0.38 F1-score on each – which is clearly above the threshold of random chance (33%). The Balanced Rule Set was able to achieve the best performance on the test set, implying that ensuring equal representation of rules could be meaningful. This entails that the generated ruleset is somewhat informative over random choice, however the trade-off in accuracy is quite high compared to the increase explain-ability gained from the

rules. In comparison to Gedeon and Turner's final decision model, which was able to recreate the training predictions with 94% accuracy [7], the results produced were much worse.

3.2. Characteristic Input Distance

Based on the observations made in 3.1, it is clear that in their current state the Causal Index Sum resolution method is not very informative. Additionally, the "All rules" rule set produced the best overall performance on both the training and testing set. This suggests that when using the rules satisfied method, the causal index is not required to be calculated as using all generated rules produces the best performance.

Therefore, a simpler metric was proposed where each input is classified by its distance to each of the characteristic inputs, with total distance calculated as the sum of the square differences between each feature of the input and characteristic input. Each input is then categorised as the same class as the characteristic input it is "closest" to.

Comparing the results of the predictions made by this metric to the predictions made by the baseline neural network model produced similar accuracy results to the top models in 3.2. However, this method also produced much larger F1-scores (seen in Figure 8) suggesting that this metric is able to capture more of the variability between different classes rather than only being able to recall one or two of them well.

Model	Data Set	Accuracy	F1
Baseline NN	Train	0.403	0.631
	Test	0.412	0.576
CNN	Train	0.408	0.274
	Test	0.250	0.318

Fig. 8. Results of the Characteristic Input Distance predictions on both models

To determine if this assumption holds for a more complex model, characteristic inputs were also calculated for each class based on the results of the convolutional model, which produced better predictions than the baseline model (as seen in 2.5). While the training accuracy of the Characteristic Input Distance predictions was similar to that of the baseline model, the test accuracy and F1-scores for both data sets were far lower. This indicates that while the metric can explain a similar amount of predictions as the baseline model, the representation of correct predictions is much poorer.

4 Conclusion and Future Work

In this paper, characteristic inputs were created for each genre in the training dataset and used to create explainable decision rules to be used for more transparent and understandable classification of brain activity in order to predict music genre. While the Rules Satisfied Class Resolution method is able to reproduce the results of a neural network that is hyper-parameter tuned on the classification task with better results than random choice, the resulting metrics leave a lot to be desired. The resulting decision results can presently be used to understand the impact of the neurological area measured by the sensor whose features are used in the prediction, which at this time suggests that further features are required for the task.

Based on the results of the Causal Index methods, a simpler method was proposed using Characteristic Input Distance. This has the advantage of not requiring causal indexes to be computed as every feature is used in the calculation of the resulting prediction. This model was able to achieve similar results as the more calculation intensive Causal Index and Class Resolution methods on the baseline neural network, along with a similar training prediction accuracy of 40.8% on the more complex Convolution Neural Network model. However, the poor test prediction and F1-scores suggest that it is not able to capture the full complexity of these models – implying that a more refined Class Resolution method would be required to produce better results.

To improve on this, enhanced Class Resolution methods for choosing the final genre class based on the satisfied decision rules should be investigated. Direct Causal Index Score was not useful for predictions due to varying in scale and if causal index is to be used in future a different weighting method should be applied. Furthermore, it is not clear if the results of the techniques applied in this paper suffer from using an uninformative feature set. The test performance of

the baseline neural network model was around 60.1% accuracy and 0.394 in F1-score. On the contrary, Rahman, Gedeon, Caldwell and Jones were able to achieve over a 94% accuracy and F1-score (coincidentally the same number as Gedeon and Turner) on the full set of sensor data (14 sensors compared to the 1 sensor used in this paper) [2]. Applying these techniques to a greater subset of features in the original data that better represent the final output would allow for more clarity about the impact of the chosen Class Resolution methods without needing to worry about poor features leading to complex decision boundaries that do not best represent the problem being explained.

References

[1] F. Alturki, K. AlSharabi, A. Abdurraqeeb and M. Aljalal, "EEG Signal Analysis for Diagnosing Neurological Disorders Using Discrete Wavelet Transform and Intelligent Techniques," *Sensors,* vol. 20, no. 9, p. 2505, 2020.

[2] J. Rahman, T. Gedeon, S. Caldwell and R. Jones, "Brain Melody Informatics: Analysing Effects of Music on Brainwave Patterns," in *International Joint Conference on Neural Networks*, 2020.

[3] Emotiv, "EPOC Flex Saline Sensor Kit," 2021. [Online]. Available: https://www.emotiv.com/product/epoc-flex-saline-sensor-kit/. [Accessed 2021].

[4] Emotiv, "EMOTIV Insight 5 Channel Mobile Brainwear," 2021. [Online]. Available: https://www.emotiv.com/product/emotiv-insight-5-channel-mobile-eeg/. [Accessed 2021].

[5] D. Castelvecchi, "Can we open the black box of AI?," *Nature,* vol. 538, no. 7623, pp. 20-23, 2016.

[6] A. Engelbrecht and H. Viktor, "Rule Improvement Through Decision Boundary," in *International Work-Conference on Artificial Neural Networks*, Berlin, 1999.

[7] T. Gedeon and H. Turner, "Explaining student grades predicted by a neural network," in *1993 International Joint Conference on Neural Networks*, Nagoya, 1993.

[8] L. Bottou, "Large-Scale Machine Learning with Stochastic Gradient Descent," in *Proceedings of COMPSTAT'2010*, Paris, 2010.

[9] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference for Learning Representations*, San Diego, 2015.

[10] Soren Becker; Marcel Ackermann; Sebastian Lapuschkin; Klaus-Robert Muller; Wojciech Samek, "Interpreting and Explaining Deep Neural Networks for Classification of Audio Signals," *arXiv preprint arXiv:1807.03418,* 2018.

Appendix 1: Box plots for each feature, separated by genre