Selection of Learning Algorithm for Musical Tone Stimulated Wavelet De-Noised EEG Signal Classification

R.F. Navea[1] and E. Dadios[2]

[1] Electronics and Communications Engineering Department,
[2] Manufacturing Engineering and Management Department,
De La Salle University - Manila
roy.navea@dlsu.edu.ph

***Abstract*—The task of classifying EEG signals pose a challenge in the selection of which learning algorithm is best to provide higher classification accuracy. In this study, five well-known learning algorithms used in data mining were utilized. The task is to classify musical tone stimulated wavelet de-noised EEG signals. Classification tasks include whether the EEG signal is tone stimulated or not, and whether the EEG signal is stimulated by either the C, F or G tone. Results show higher correct classification instances (CCI) percentages and accuracies in the first classification task using the J48 decision tree as the learning algorithm. For the second classification task, the k-nn learning algorithm outruns the other classifiers but gave low accuracy and low correct classification percentage. The possibility of increasing the performance was explored by increasing the *k (number of neighbors)*. With the increment, its produced directly proportionate in accuracy and correct classification percentage within a certain value of *k*. A larger *k* value will reduce the accuracy and the correct classification percentages.**

***Index Terms*—Classifier; EEG Signals; Learning Algorithm; Musical Tones.**

I. Introduction

Machine learning is one of the growing fields of computational science especially with the increasing demand for data analytics. Digital data is increasing at a very fast rate due to the computer and/or mobile applications which enable users to contribute information and store them in a large capacity memory storage systems or data banks. Most machine intelligence algorithms were used in classification, segregation, prediction or detection of anything which interests the observer. Among the common machine learning algorithms are either in the form of a function or a decision tree.

The selection of machine a learning algorithm for a specific purpose is a challenging task given the different parameter settings for each algorithm. A change in parameter connotes a change in performance. A single algorithm may be chosen however, there could be other algorithms available to best serve a specific purpose.

In electroencephalogram (EEG) classification, the usual classification methods used are based on Naïve Bayes algorithm [1], [2], Artificial Neural Networks (ANN) [3]–[6], Support Vector Machines (SVM) [7]–[9], k-Nearest Neighbor (k-NN) [2] and J48 decision trees. Hence, in choosing classification algorithms, the type or nature of the EEG signal has to be considered. A classifier may be useful to one set of EEG signals but not to another set of EEG signals which are of different origin of stimulation.

The tonal response of the brain especially to musical tones has not been thoroughly researched. Most researches focus on the whole song [10]–[12] and not on its building blocks which are the tones. In this study, an attempt to classify musical tone stimulated and also wavelet de-noised EEG signals [13], [14], [4] was performed using five different classifiers as mentioned. The learning algorithms were simulated in WEKA [15], an open source platform for different machine learning algorithms.

II. Methods

The procedures for processing the EEG signals that lead to classification are shown and guided by the block diagram in Figure1. This process follows the Input-Process-Output (IPO) model. The EEG data set serves as the input. The process block includes the preprocessing procedures, the feature extraction and the classification tasks using five different classifiers. The output shows whether the EEG signals are tone stimulated or not and whether either the C, the F, or the G tone stimulated it.

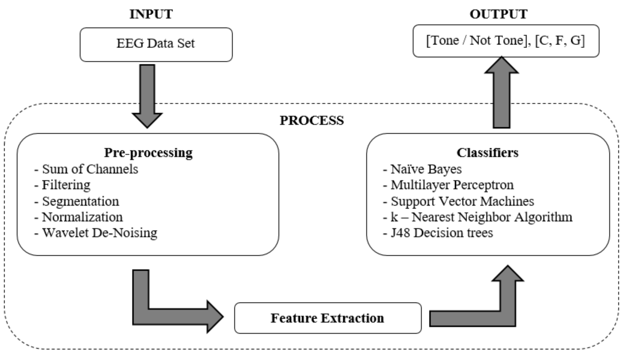

Figure 1: Classification Process Diagram

A. EEG Data Set

Musical tone stimulated EEG signals were obtained from 27 undergraduate students with age ranging from 18 to 21 years old while they are listening to an audio stimulus. The students sit in a comfortable chair with their eyes closed to minimize ocular artefacts. Audio stimulation was used and for optimal reception, a headphone was plugged into the ears of the respondents. The audio stimulation piece [13], [14], [4] is shown in Figure 2. The series of rests at the beginning establishes the baseline while the rests before the tones establish the secondary baseline. The notes represent the tones C, F and G.

Figure 2: Audio Stimulus Piece

A 14-channel neuroheadset was used to pick-up the EEG signals from the respondents. The sampling rate is 128 samples per second. The data gathering sessions were done in a dimmed and acoustically prepared room. The EEG signals were transmitted from the neuroheadset to a laptop computer via Bluetooth. The raw EEG signals were saved in a *.csv* file and were processed in Matlab.

B. Preprocessing

The 14 channels were summed up to improve synchrony and similarity [16]. EEG signals have five distinct band of frequencies. These are the delta band with 0.5Hz – 4Hz, the theta band with 4Hz - 8Hz, the alpha band 8Hz – 13Hz, the beta band with 13Hz – 30Hz and the gamma band which is roughly greater than 30Hz. The raw EEG signals were initially filtered within the delta to gamma bands.

The audio stimulus has five segments namely, baseline, secondary baseline, C, F and G. Hence, the EEG signals were also segmented according to these segments. Table 1 [13], [14], [5] shows the segmentation according to its time stamp, period and number of samples. For the given sampling rate, a total of 29,184 samples were obtained for the whole duration of the audio stimulus. The baseline segment has 23,040 samples while the secondary baseline and the tones have 256 samples each. To describe the whole data set, there are 27 baseline segments, 324 secondary baseline segments, 162 C-tone segments, 81 segments for both F-tone and G-tone.

Table 1
Audio Stimulus Timing Table

Stimuli	Time Stamp	Period	No. of Samples	Sample Series
Baseline	0 - 3:00	180 secs	23040	1-23040
s-Baseline	3:01-3:02	2 secs	256	23041-23296
	3:05-3:06	2 secs	256	23553-23808
	3:09-3:10	2 secs	256	24065-24320
	3:13-3:14	2 secs	256	24577-24832
	3:17-3:18	2 secs	256	25089-25344
	3:21-3:22	2 secs	256	25601-25856
	3:25-3:26	2 secs	256	26113-26368
	3:29-3:30	2 secs	256	26625-26880
	3:33-3:34	2 secs	256	27137-27392
	3:37-3:38	2 secs	256	27649-27904
	3:41-3:42	2 secs	256	28161-28416
	3:45-3:46	2 secs	256	28673-28928
C	3:03-3:04	2 secs	256	23297-23552
	3:15-3:16	2 secs	256	24833-25088
	3:19-3:20	2 secs	256	25345-25600
	3:31-3:32	2 secs	256	26881-27136
	3:35-3:36	2 secs	256	27393-27648
	3:47-3:48	2 secs	256	28929-29184
F	3:07-3:08	2 secs	256	23809-24064
	3:23-3:24	2 secs	256	25857-26112
	3:39-3:40	2 secs	256	27905-28160
G	3:11-3:12	2 secs	256	24321-24576
	3:27-3:28	2 secs	256	26369-26624
	3:43-3:44	2 secs	256	28417-28672

A process of normalization was performed and wavelet-based de-noising [13], [17], [18] was implemented to remove other noise artefacts. The wavelet de-noising process utilized reverse biorthogonal (*'rbio'*) mother wavelets and the Rigorous Stein's Unbiased Risk Estimate ('*rigrsure*') thresholding method as in [19].

C. Feature Extraction

Ten statistical features were obtained from the power spectrum vector, P(x), of the EEG segments [4]. These features are defined as follows:

- Mean (M_x) – arithmetic average of all the scores in P(x) with length *x*.

$$M_x = [P(1) + P(2) + P(3) + \ldots + P(x)] / x \qquad (1)$$

- Median – a point in P(x) at which 50% of the scores fall below and 50% of the scores fall above. The median is the $[(x + 1) / 2]^{th}$ value in ranked distribution.
- Mode – the most frequently appearing score or group of scores appearing in P(x). It is also the most common value.
- Standard Distribution (SD) – a quantitative measure defining the extent to which scores are dispersed throughout P(x) in relation to the arithmetic mean.

$$SD = \sqrt{[\Sigma \{P(x) - M_x\}^2] / (x - 1)} \qquad (2)$$

- Variance (V) – the square of the standard distribution.

$$V = (SD)^2 \qquad (3)$$

- Range (R) – the difference between the highest score and the lowest score.

$$R = P(x)_{max} - P(x)_{min} \qquad (4)$$

- Mean Absolute Deviation (MAD) – it defines the mean distance between each data points in P(x) and its average value.

$$MAD = \Sigma |P(x) - M_x| / x \qquad (5)$$

- Interquartile Range – this is a variability measure obtained by dividing the rank-ordered P(x) into quarters, called quartiles.
- Skewness (Sk) – this is a third-order statistical measure that defines the degree of the slanting symmetry or departure from the symmetry of P(x).

$$Sk = \frac{\sum_{x=1}^{x}(P(x) - M_x)^3}{(x-1)SD^3} \qquad (6)$$

- Kurtosis (Kt) – this is a fourth-order statistical measure that defines the degree of peakedness of P(x).

$$Kt = \frac{\sum_{x=1}^{x}(P(x) - M_x)^4}{(x-1)SD^4} \qquad (7)$$

These features were fed into the machine learning tool for classification tasks.

III. CLASSIFIERS

Five well-known classifiers were employed in this study namely, Naïve Bayes (NB), Multilayer Perceptron, Support Vector Machines, k-Nearest Neighbor (k-nn) and the J48 decision tree.

A. Naïve Bayes Classifier

Naïve Bayes (NB) algorithm can be used as a predictor and a classifier. This tool is anchored on Bayes' theorem that describes how a probability of a given event can be determined given the probability of another event.

In Bayes' theorem, the posterior probability of a target class ***m*** given the predictor or attribute ***n***, P(m|n), the prior probability of the predictor or attribute ***n***, P(n), the posterior probability of the predictor or attribute ***n*** given the class ***m***, P(n|m), and the prior probability of the class, P(m), are computed. Hence,

$$P(m|n) = [P(n|m)\ P(m)]\ /\ P(n) \tag{8}$$

and

$$P(m|n) = P(n_1|m)\ P(n_2|m)\ \ldots\ P(n_j|m)\ P(m) \tag{9}$$

NB assumes class conditional independence. That is, the effect of the value of a predictor (n) on a given class (m) is independent of the values of the other predictors. This model is useful for very large data sets. It is easy to build and usually outruns other classifications methods [1].

B. Multilayer Perceptron

The perceptron is a part of an artificial neural network that returns 0 or 1 according to the value of a linear function of its inputs. It is composed of weights, biases, a summation processor and an activation function. A perceptron takes a weighted sum of inputs and outputs. If the predicted output is the same as the target output, then the performance is said to be satisfactory and no changes in weights are needed. However, if the output does not match the target output, then the weights change to reduce the difference between the predicted output and the target output [20], [21]. The perceptron weights adjustment is defined by

$$\Delta W = (\eta)\ (d)\ (x) \tag{10}$$

where $\boldsymbol{\Delta W}$ is the change in weights, $\boldsymbol{\eta}$ is the learning rate, usually less than 1, ***d*** is the difference between the predicted output and the expected output, and ***x*** is the input data.

A multilayer perceptron has one or more hidden layers. Commonly, single input layer, hidden layer and output layer is used. In the hidden layer, the inputs and weights work with the activation function for any node either as a weighted sum or a transfer function. The output from the hidden layer nodes is used work with an activation function for an output node. The structure is like a passing forward processed data from the input layer, to the hidden layer, then to the output layer. The inputs are propagated by taking the sum of all the weighted inputs and then the output is computed using the sigmoid function.

To adjust the weights of inputs at the output layer, the back propagation (BP) principle was used by exploiting the output error. In BP, the error at previous layer can be calculated and use it to adjust the weights arriving at that point. The process of weight adjustments can be done through any number of layers by using a differentiable sigmoid as the non-linear transfer function.

C. Support Vector Machines

Support Vector Machines (SVM) are a function bounding classifiers that can differentiate classes in the training data by finding the hyperplane that optimally sets boundaries between the classes. In a two-class scenario, it draws the widest channel, or street, between the two classes. The classes are labeled +1 (for positive examples) and -1 (for negative examples) [8], [22].

The intuition behind SVM is that points or instances are like vectors. Every point becomes a vector of the input variables of the features. SVM finds the two closest points from the two classes that support the best separating boundary and draws a line connecting them. The best separating line is the line that bisects and is normal to the connecting line [23].

A separating (decision) hyperplane can be defined in terms of an intercept, ***p*** and a normal vector, $\vec{\boldsymbol{V}}$. To identify which among the hyperplanes should be chosen, the intercept term ***p*** has to be specified and all points $\vec{\boldsymbol{n}}$ on the hyperplane should satisfy

$$\vec{V}^T\vec{n} = -p \tag{11}$$

as the hyperplane is perpendicular to the normal vector. The training dataset is defined as D = $\{(\vec{n}_i, y_i)\}$, a pair of a point and a class label corresponding to it. Now the linear classifier becomes

$$f(n) = sign\ (\vec{V}^T\vec{n} + p) \tag{12}$$

If the sign is positive then the input belongs to the positive class, otherwise, it belongs to the negative class.

D. k – Nearest Neighbor Algorithm

The k-nearest neighbors (k-nn) is an algorithm that creates new cases from a pool of available cases according to their similarity. The ***k*** variable indicates the number of closest or nearest neighbors to be considered. If k is unity, then it is indicative of the single nearest neighbor.

Optimizing the value of ***k*** is a challenging task. Hence, a heuristic approach is possible in order to determine a ***k*** value with the highest correct classification results. The k-value can be tested using cross-validation. A high ***k*** usually gives better results but that is not always the case. In this study, k=1 and k=3 were initially used. High values of ***k*** are also explored.

E. J48 Decision Tree

J48 is an algorithm used to generate a decision tree according to the attributes of a given data set. This decision tree is based on the C4.5 algorithm. C4.5 is a decision tree algorithm developed by J.R. Quinlan [24] which is an extension of his earlier Iterative Dichotomiser 3 (ID3) algorithm. Decision trees are statistical classification models which resemble a graphical tree structure with two or more decision nodes and leaf nodes brought by breaking data sets into smaller subsets.

In Quinlan's ID3, decision trees are built based on entropy and information gains. The algorithm reiterates as it looks at all possible branches using a top-down, greedy search with no backtracking. It tries all possible branches and it chooses the best one. The J48 algorithm works with missing values, decision tree pruning, continuous attribute value ranges, and derivation of rules to name a few. This algorithm is implemented in an open source Java program and can be simulated in the WEKA platform [25].

Classification tasks performed in this study were based on whether the EEG signal was tone stimulated or not and whether the EEG signal is stimulated by either the C, F or G tones. The correct classification percentages for each classifier were obtained and together with their respective accuracies in terms of the F-measure. The kappa statistics were also provided to show the measure of how close is the classification results to the expected results. Confusion matrices were also made available to understand what has transcribed during classification.

IV. Results and Discussion

A. Tone or Not Tone Classification

The first classification procedure is whether the EEG signal is tone stimulated or not. Tone stimulated means that the EEG signal is either stimulated by any of the C, F or G tones. Not tone stimulated refers to either baseline or secondary baseline.

In Table 2, the correct classification instances (CCI) percentage and the kappa of the five classifiers are shown. It is evident that the J48 classifier has the highest CCI percentage with 79.26% and a kappa value of 0.5302 which falls in the fair to good category according to Fleiss' range of kappa values [26]. In [27], this kappa value falls in the moderate level. This means that the classified results match the expected results moderately.

The accuracy of the classifiers is shown in Table 3. The J48 classifier has the highest classification accuracy in both tone and not tone classification.

The confusion matrix in Table 4 shows that 27 instances are correctly classified as not tone and 27 instances are classified as tone but they are not tone. There are 80 instances of tone which are correctly classified and 1 instance of tone but classified as not tone.

Table 2
CCI for Tone / Not tone Classification

Classifier	CCI	Kappa
Naive Bayes	77.78%	0.5000
Multilayer Perceptron (30n)	77.78%	0.5000
Support Vector Machines	78.52%	0.5151
k-nn (k=1)	68.15%	0.3260
k-nn (k=3)	68.89%	0.3312
J48	***79.26%***	***0.5302***

Table 3
F-measure for Tone / Not tone Classification

Classifier	F-measure		Weighted Ave.
	Not Tone	Tone	
Naive Bayes	0.643	0.839	0.741
Multilayer Perceptron (30n)	0.643	0.839	0.741
Support Vector Machines	0.651	0.845	0.748
k-nn (k=1)	0.583	0.743	0.663
k-nn (k=3)	0.571	0.756	0.664
J48	***0.659***	***0.851***	***0.755***

Table 4
Confusion Matrix for the J48 Classifier

Classifier	Confusion Matrix			
	a	**b**		<<< = classified as
J48	27	27	|	a = Not Tone
	1	80	|	b = Tone

B. C, F and G Tone Classification

The next classification task is to determine which tone (C, F or G) stimulates the EEG signal. Correct classification results are shown in Table 5. Among the five classifiers, the k-nn (k=3) stood out with 29.63%. This is a low percentage in terms of classification. The C, F and G segments could have very similar features which made it hard for the classifiers to distinguish them from one another. A poor value of kappa is also observable [26].

With low correct classification results, the accuracy also shows low values. Among the given classifiers, the k-nn (k=3) still had the highest accuracy.

Table 7 shows the confusion matrix for the k-nn (k=3) classifier. It is noticeable that the correctly classified instances are not that far from the incorrectly classified instances. Hence, this reflected the low accuracy of the classifier giving only a weighted average of 0.291.

Table 5
CCI for C, F and G Classification

Classifier	CCI	Kappa
Naive Bayes	16.05%	-0.2593
Multilayer Perceptron (30n)	28.40%	-0.0741
Support Vector Machines	20.99%	-0.1852
k-nn (k=1)	23.46%	-0.1481
k-nn (k=3)	***29.63%***	***-0.0556***
J48	25.93%	-0.1111

Table 6
F-measure for C, F and G Classification

Classifier	F-measure			Weighted Ave.
	C	F	G	
Naive Bayes	0.048	0.328	0.038	0.138
Multilayer Perceptron (30n)	0.346	0.377	0.049	0.257
Support Vector Machines	0.182	0.235	0.214	0.21
k-nn (k=1)	0.222	0.276	0.200	0.233
k-nn (k=3)	***0.349***	***0.255***	***0.269***	***0.291***
J48	0.281	0.259	0.235	0.258

Table 7
Confusion Matrix for the k-nn (k=3) Classifier

Classifier	Confusion Matrix				
	a	**b**	**c**		<<< = classified as
k-nn (k=3)	11	6	10	|	a = C
	13	6	8	|	b = F
	12	8	7	|	c = G

The possibility of increasing the number of neighbors was explored in order to increase the correct classification percentage of the k-nn classifier. In Table 8, it can be seen that the CCI increases as the value of $\boldsymbol{k}$ are increased. The increase is observable up to a value of $\boldsymbol{k}$ equal to 15 only. After which, the CCI falls back to a CCI close to when k = 1. Kappa values also increase but were not able to go beyond 0.4 to get into the good to a fair level of kappa according to Fleiss' kappa table [26]. In addition, if the results were based on Landis' and Koch's kappa table [27], the obtained kappa value indicates a slight match between the classified results versus the expected results.

The data in Table 8 is graphed in Figure 3. It can be seen that the trend of CCI increases and decreases before and after reaching k = 15, respectively. The pattern of the kappa value follows the same trend as that of the CCI. However, it is noticeable that the rms error continually decreases but somehow stays at 0.4722 with more than a hundred value of

k. The result is indicative that there is a certain value of ***k*** to optimize CCI results. A Higher number of neighbors is not always a guarantee to increase correct classification rates.

Table 8
k search for the k-nn Classifier

k	CCI	kappa	rms error
1	23.46%	-0.1481	0.7003
3	29.63%	-0.0556	0.5638
5	34.57%	0.0185	0.5273
7	37.04%	0.0556	0.5049
9	27.16%	-0.0926	0.5026
11	37.04%	0.0556	0.4949
13	34.57%	0.0185	0.4915
15	***37.04%***	***0.0556***	***0.4875***
21	35.80%	0.037	0.4816
31	20.99%	-0.1852	0.4855
45	18.52%	-0.2222	0.4789
75	25.93%	-0.1111	0.4722
101	25.93%	-0.1111	0.4722

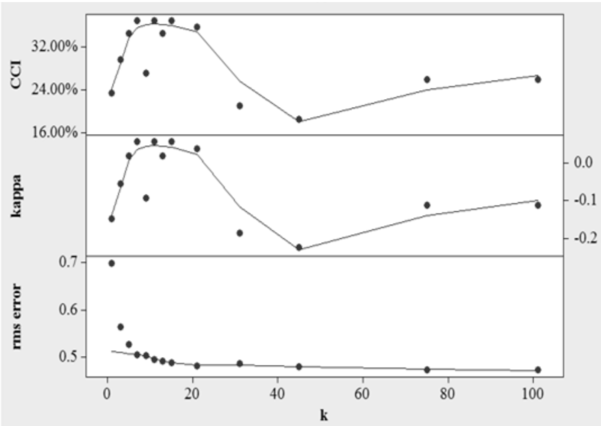

Figure 3: Matrix Plot of CCI, kappa, rms error vs. *k*

V. Conclusion

EEG classification is a challenging task because of the nature by which the EEG signals were obtained. The selection of classifier has to go to a process of comparing their performance in terms of correct classification and accuracy. There is no single best classifier that is applicable to all EEG classification tasks.

In this study, the classifier that best suits the tone or not tone classification is the J48 algorithm and with the C, F and G classification task, the k-nn (k=3) was able to outrun the other four classifiers. However, the accuracy is not that high and the possibility of increasing it was explored by increasing the number of neighbors, ***k***. The highest classification obtained was 37.04% when k = 15.

The high classification results for tone or not tone classification task is possibly due to the increase in energy once the brain is stimulated. The difference between the baseline signals and the tone stimulated signals is highly 'seen' by the classifier.

On the other hand, for tone stimulated signals, there is a possibility that the power spectrum of the C, F and G segments are similar giving a result of good to a fair level of classification.

Since the EEG signals here are analyzed within the delta to gamma band, it is recommended to perform a sub-band classification task with a classifier grid search for optimal results.

References

[1] S. Ashok and G. Purushotaman, "DWT based Epileptic Seizure Detection from EEG Signals using Naïve Bayes/k-NN Classifiers," *IEEE Access*, vol. 3536, no. c, pp. 1–1, 2016.

[2] S. K. Hadjidimitriou and L. J. Hadjileontiadis, "Toward an EEG-based recognition of music liking using time-frequency analysis," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 12, pp. 3498–3510, 2012.

[3] M. P. Paulraj, S. Bin Yaccob, A. H. Bin, and K. Subramaniam, "Eeg Based Hearing Threshold Determination Using Artifical Neural Networks," no. October, pp. 268–270, 2012.

[4] R. F. Navea and E. Dadios, "Classification of tone stimulated EEG signals using independent components and power spectrum vectors," in *8th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2015*, 2016.

[5] R. F. Navea and E. Dadios, "Classification of tone stimulated EEG signals using independent components and power spectrum vectors," in *2015 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2015, no. December, pp. 1–5.

[6] J. Kim, B. Şen, and et al, "Sleep stage classification based on EEG hilbert-huang transform," *Conf. Proc. ... Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE Eng. Med. Biol. Soc. Annu. Conf.*, vol. 2014, no. 3, pp. 1–6, 2014.

[7] C. Guerrero-Mosquera, M. Verleysen, and A. N. Vazquez, "EEG feature selection using mutual information and support vector machine: A comparative analysis.," *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, vol. 2010, pp. 4946–4949, 2010.

[8] L. Ke and J. Shen, "Classification of EEG signals by ICA and OVR-CSP," *Proc. - 2010 3rd Int. Congr. Image Signal Process. CISP 2010*, vol. 6, no. d, pp. 2980–2984, 2010.

[9] L. Z. L. Zhang, G. L. G. Liu, and Y. W. Y. Wu, "Wavelet and Common Spatial Pattern for EEG signal feature extraction and classification," *Comput. Mechatronics, Control Electron. Eng. (CMCE), 2010 Int. Conf.*, vol. 5, no. 4, pp. 243–246, 2010.

[10] M. Casey, J. Thompson, O. Kang, R. Raizada, and T. Wheatley, "Population codes representing musical timbre for high-level fMRI categorization of music genres," 2012.

[11] S. M. Park and K. B. Sim, "A study on the analysis of auditory cortex active status by music genre: Drawing on EEG," *Proc. - 2011 8th Int. Conf. Fuzzy Syst. Knowl. Discov. FSKD 2011*, vol. 3, pp. 1916–1919, 2011.

[12] M. G. Quiñones, A. Kassabian, and E. Boschi, *Ubiquitous musics: The everyday sounds that we don't always notice*. Farnham: Ashgate Publishing, Ltd., 2013.

[13] R. F. Navea and E. Dadios, "Classification of Wavelet-denoised Musical Tone Stimulated EEG Signals using Artificial Neural Networks," in *IEEE TENCON*, 2016.

[14] R. F. Navea and E. Dadios, "Beta/Alpha power ratio and alpha asymmetry characterization of EEG signals due to musical tone stimulation," in *Project Einstein 2015*, 2015.

[15] E. Frank, M. Hall, and I. Witten, "The WEKA Workbench. Online Appendix for 'Data Mining: Practical Machine Learning Tools and Techniques', Morgan Kaufmann, 4th Edition." 2016.

[16] T. Collura, *Technical Foundations of Neurofeedback*. Abingdon: Routledge, 2014.

[17] E. Estrada, H. Nazeran, G. Sierra, F. Ebrahimi, and S. K. Setarehdan, "Wavelet-based EEG denoising for automatic sleep stage classification," *CONIELECOMP 2011 - 21st Int. Conf. Electron. Commun. Comput. Proc.*, pp. 295–298, 2011.

[18] N. K. Al-Qazzaz, S. Ali, S. A. Ahmad, M. S. Islam, and M. I. Ariff, "Selection of mother wavelets thresholding methods in denoising multi-channel EEG signals during working memory task," *IECBES 2014, Conf. Proc. - 2014 IEEE Conf. Biomed. Eng. Sci. "Miri, Where Eng. Med. Biol. Humanit. Meet,"* no. December, pp. 214–219, 2015.

[19] R. F. Navea and E. Dadios, "Selection of Mother Wavelet Function and Thresholding Method for Musical Tone Stimulated EEG Signal De-noising and Classification," in *9th AUN/SEED-Net Conference on Electrical and Electronics Engineering*, 2016.

[20] A. Subasi and E. Erçelebi, "Classification of EEG signals using neural network and logistic regression," *Comput. Methods Programs Biomed.*, vol. 78, no. 2, pp. 87–99, 2005.

[21] V. B. Semwal, M. Raj, and G. C. Nandi, "Biometric gait identification based on a multilayer perceptron," *Robot. Auton. Syst.*, vol. 65, pp. 65–75, 2015.

[22] I. Belakhdar, W. Kaaniche, R. Djmel, and B. Ouni, "A Comparison Between ANN and SVM Classifier for Drowsiness Detection Based on

Single EEG Channel," pp. 443–446, 2016.
[23] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. Mclachlan, A. Ng, B. Liu, P. S. Yu, Z. Z. Michael, S. David, and J. H. Dan, *Top 10 algorithms in data mining*. 2008.
[24] J. R. Quinlan, "Induction of Decision Trees," pp. 81–106, 2007.
[25] G. Kaur, "Improved J48 Classification Algorithm for the Prediction of Diabetes," vol. 98, no. 22, pp. 13–17, 2014.
[26] J. Fleiss, *Statistical Methods for Rates and Proportions, 2nd Ed.* New York: John Wiley, 1981.
[27] J. Landis and G. Koch, "The measurement of observer agreement for categorical data.," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.