IOP Publishing Biomed. Phys. Eng. Express 9 (2023) 045027 https://doi.org/10.1088/2057-1976/acdbd0

Biomedical Physics & Engineering Express

RECEIVED
6 March 2023

REVISED
27 March 2023

ACCEPTED FOR PUBLICATION
6 June 2023

PUBLISHED
16 June 2023

PAPER

A comparative study of machine learning methods for classifying ERP scalp distribution

Roya Salehzadeh[1,*], **Firat Soylu**[2] **and Nader Jalili**[1]

[1] Department of Mechanical Engineering, The University of Alabama, Tuscaloosa, AL 35487, United States of America
[2] Department of Educational Studies, The University of Alabama, Tuscaloosa, AL 35487, United States of America
[*] Author to whom any correspondence should be addressed.

E-mail: rsalehzadeh@crimson.ua.edu

Keywords: EEG signals, event-related potentials (ERPs), numerical cognition, finger numeral configurations, machine learning

Abstract

Objective. Machine learning (ML) methods are used in different fields for classification and regression purposes with different applications. These methods are also used with various non-invasive brain signals, including Electroencephalography (EEG) signals to detect some patterns in the brain signals. ML methods are considered critical tools for EEG analysis since could overcome some of the limitations in the traditional methods of EEG analysis such as Event-related potentials (ERPs) analysis. The goal of this paper was to apply ML classification methods on ERP scalp distribution to investigate the performance of these methods in identifying numerical information carried in different finger-numeral configurations (FNCs). FNCs in their three forms of montring, counting, and non-canonical counting are used for communication, counting, and doing arithmetic across the world between children and even adults. Studies have shown the relationship between perceptual and semantic processing of FNCs, and neural differences in visually identifying different types of FNCs. *Approach*. A publicly available 32-channel EEG dataset recorded for 38 participants while they were shown a picture of an FNC (i.e., three categories and four numbers of 1,2,3, and 4) was used. EEG data were pre-processed and ERP scalp distribution of different FNCs was classified across time by six ML methods, including support vector machine, linear discriminant analysis, naïve Bayes, decision tree, K-nearest neighbor, and neural network. The classification was conducted in two conditions: classifying all FNCs together (i.e., 12 classes) and classifying FNCs of each category separately (i.e., 4 classes). *Results*. The support vector machine had the highest classification accuracy for both conditions. For classifying all FNCs together, the K-nearest neighbor was the next in line; however, the neural network could retrieve numerical information from the FNCs for category-specific classification. *Significance*. The significance of this study is in exploring the application of multiple ML methods in recognizing numerical information contained in ERP scalp distribution of different finger-numeral configurations.

1. Introduction

Machine learning (ML) methods have gained significant attention across different fields, especially in the domain of classification and regression, due to their remarkable performance and diverse applications. In recent years, these methods have also been increasingly utilized with non-invasive brain signals, such as Electroencephalography (EEG), functional near-infrared spectroscopy (fNIRS), functional magnetic resonance imaging (fMRI), and magnetoencephalography (MEG) for a wide range of purposes, including mental disease diagnosis, medical image analysis, brain-computer interface (BCI), and classification tasks [1]. Among all non-invasive brain signals, EEG signals are popular ones that can be effectively analyzed using a range of ML methods. The ML methods include but are not limited to support vector machine (SVM), linear discriminant analysis (LDA), naive Bayes (NB), decision tree (DT), K-nearest neighbor (KNN), and neural network (NN) algorithms. Each technique has its unique strengths and limitations, making them more suitable for different types of EEG data analysis tasks [1–6].

SVM is a powerful ML algorithm that can effectively classify EEG data by constructing a hyperplane that separates the different classes [1]. SVM is used with EEG signals for emotion classification [7], emotion recognition [8], seizure detection in animals [9], visual comprehension [10], and facial recognition [11]. Several studies have compared the performance of different ML methods in analyzing EEG signals. SVM has consistently demonstrated superior accuracy compared to other techniques and is considered one of the most commonly used methods in the field [12]. LDA is another classification technique that finds the linear combination of features that best discriminates between classes. LDA is one of the commonly used methods with EEG signals for feature selection, seizure detection, motor imagery classification, mental workload detection [13], and ERP source time estimation using EEG signals. NB is a probabilistic ML algorithm that assumes features are independent and calculates the probability of the class given the features [1]. NB classifier is used with EEG signals for EEG classification [14], EEG autism detection [15], and emotion classification [16, 17]. KNN is an ML method that can be used for both classification and regression tasks. In KNN, the classification criterion is based on the distance between the known data point and its K neighbors [1]. KNN has been successfully applied in various EEG data analysis tasks, including emotion recognition and seizure detection. Its ability to handle both classification and regression tasks makes it a versatile tool for analyzing EEG signals and has contributed to its widespread use in the field of neurosciences [18]. DT is a popular ML algorithm used for analyzing EEG signals in various applications. DTs operate by making choices using the characteristics of the data. They begin at the root of the tree and proceed by asking a series of questions about the features of the data, with each question leading to a child node that provides an answer to the question. DTs have been successfully applied in various EEG data analysis tasks, such as epilepsy seizure prediction [19], classification of the severity of trachea stenosis from EEG signals [20], and arrhythmia detection [21]. NNs are a widely used ML algorithm that can be used for supervised classification and regression tasks. This approach actually mimics the way our brain is structured and just like our brain, in this method, each layer receives input signals from the previous layer, it then performs calculations on these signals before passing them on as output signals to the next layer. NNs have been successfully applied in several EEG data analysis tasks, including seizure detection, epilepsy detection [22], EEG classification [23], motor imagery classification [24], and many others.

This study makes a valuable contribution to the field of numerical cognition by utilizing ML methods to decode the ERP scalp distribution of finger numeral configurations (FNCs) over time. The goal of the study was to identify the numerical information carried out in FNCs and evaluate the performance of each ML method. FNCs are categorized as montring (M), canonical counting (C), and non-canonical counting (NC). Across cultures, fingers have been utilized to communicate numerical information, count, and perform arithmetic operations [25]. Finger-based interactions also play a crucial role in early development, providing access to fundamental mathematical concepts such as one-to-one correspondence and whole-part relations [26]. Various forms of finger processing interact with the neural processes underlying numerical cognition, and finger-counting strategies are commonly used by children across different cultures and even by adults when solving mathematical and arithmetical problems [27, 28]. Finger representation provides a natural and embodied expression of numerical quantity that aids in numerical cognition [29]. Therefore, understanding the influence of fingers on numerical cognition has been a vital area of research.

One research field in numerical cognition is studying the differences in the perceptual and semantic processing of FNCs. ERP analysis is a commonly used method for this purpose [26], but it has limitations such as the need to identify commonalities across scalp distribution [30]. In response to the limitations of the traditional methods including ERP analysis, innovative approaches such as mass univariate and multivariate techniques have been introduced to the field of numerical cognition research. These novel methods offer the potential to overcome challenges and open up exciting new avenues of analysis in this important field of study. Among these novel methods, the use of ML has attracted lots of attention where they could be used to analyze the ERP scalp distribution of FNCs over time to investigate the neural mechanisms underlying numerical cognition. By decoding the numerical information carried in FNCs and evaluating the performance of different ML methods, this study contributes to our understanding of how the brain processes numerical information and the potential role of finger-based interactions in numerical cognition. This paper utilized six ML methods to analyze a publicly available EEG dataset collected from 38 participants in an FNC experiment [31]. SVM, LDA, NB, DT, KNN, and NN were the MML methods used in this study which were chosen after conducting a literature review [1]. This study builds upon a previous study that focused on using the SVM method for classifying ERP signals of FNCs on the same dataset [32].

The remaining paper is structured as follows: section 2 introduced the methods; results were provided in section 3. Section 4 provided the discussion, and the conclusion was provided in section 5.

2. Methods

2.1. Participants

The original experiment had thirty-eight adult participants (20 female, M = 19.68 years, SD = 1.84) [5]. All participants were native English-speaking undergraduate

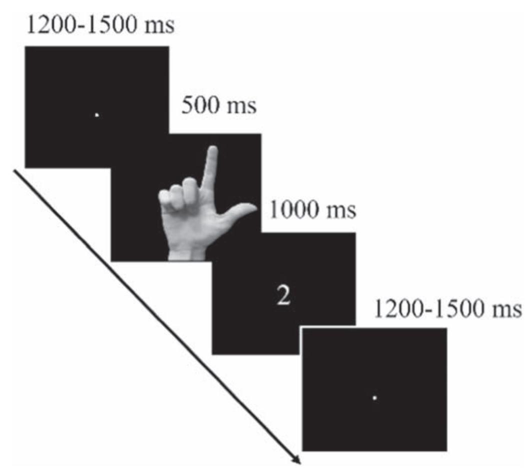

Figure 1. Stimulus presentation order for each trial.

students, with no history of neurological illness and normal or corrected-to-normal vision.

2.2. Stimuli and experimental procedures

The original EEG signals were collected in an experiment in which participants were shown a picture of an FNC for 500 ms followed by an Arabic numeral for 1000 ms. Then, participants were asked to press one of the two buttons on a Logitech F310 game controller to identify if the Arabic numeral showed the same numerical magnitude as the FNC or not. The inter-trial interval was 1200 ms, with a 300 ms additional jitter (total ITI varying 1200–1500 ms) [26]. In total, 24 pictures consisted of three types of FNCs (i.e., M, C, NC), and four numbers of one to four separately for the left and right hands were shown to the participants. In a previous study [26], no significant differences were found in the EEG signals between FNCs performed by the left and right hands. Therefore, the data from both hands were combined, resulting in 12 distinct categories representing the three FNC types and four different numbers. The experiment was composed of 10 blocks, and each block had 96 trials. Each trial included a set of 12 unique FNC configurations that were randomly sequenced within each block. The experiment had a total of 960 trials, with 80 trials assigned to each of the 12 FNC configurations. The order of stimuli was shown in figure 1.

2.3. Data acquisition and pre-processing

A BrainVision 32-Channel ActiChamp system, with Easy Cap recording, Ag/AgCl electrodes, and an international 10–20 system for electrode locations has been used for collecting raw EEG data. The recording sampling rate was 500 Hz, and the recording reference electrode was defined as electrode Cz. In this study, data pre-processing was completed in MATLAB using various features of EEGLAB [33] and ERPLAB [34] plugins. The data-pre-processing had several steps, including re-referencing, filtering, epoching, artifact detection, and artifact removal. First, the Cz was added back to the data electrodes and the EEG data was re-referenced to the average reference. Then, a 0.1 Hz (half-amplitude cutoff) high-pass and an 80 Hz (half-amplitude cutoff) low pass IIRButterworth filter (24 dB/octave) was applied, and the data were resampled at 250 Hz.

The EEG signals epoching was done between 500 ms before the stimulus onset (FNC presentation) to 1500 ms after (stimulus offset) and then the 500 ms pre-stimulus baseline was used for correcting epochs. Eye blinks and eye movements were detected using a moving window peak-to-peak threshold algorithm (threshold 60 μV, window size 80 ms, window step 20 ms) and a step-like artifacts algorithm (threshold 50 μV, window size 200 ms, window step 100 ms); respectively. In the end, epochs containing artifacts (i.e., automatic and visual inspection) were excluded (20.92% of trials, SD = 21.45). In addition, for the sake of this study, only the epochs that preceded a correct response (Arabic numeral validated correctly) were included in the analysis (79.08% of trials). In the next step, epoched data were filtered by applying a 6 Hz (half-amplitude cutoff) low pass IIRButterworth filter (24 dB/octave). The 6 Hz low-pass filter was selected to avoid overlapping between alpha-power and ERPs and to increase the signal-to-noise ratio for perceptual processing [35]. Figure 2 represents the 12 FNCs and the associated topography (Scalp Distribution) of ERPs for Each FNC, at 500 ms, averaged across participants.

2.4. General procedure for classification ERP scalp distribution of FNCs across time

The overarching goal of this paper was to classify the ERP scalp distribution of different FNCs using multiple ML methods to compare their performance in detecting FNCs' differences. The general classification procedure implemented in this paper was proposed by Bae and Luke in 2018 for ERP decoding of spatial attention and working memory which was a subject-based classification across the desired time range [35]. Figure 3 represents the general classification procedure which was completed in 9 steps described in more detail below:

Step 1: The first step in the classification process involved loading the processed EEG data for each participant separately.

Step 2: To ensure that the analysis was not biased by any systematic ordering of the data, the processed EEG data were randomly shuffled.

Step 3: The objective of the classification approach was to decode the numerical information contained in the brain signals over time. To achieve this, the data was decoded for specific time points within the time range of −500 ms to 1500 ms. To obtain a comprehensive result over time, 100 time points were extracted every 20 ms.

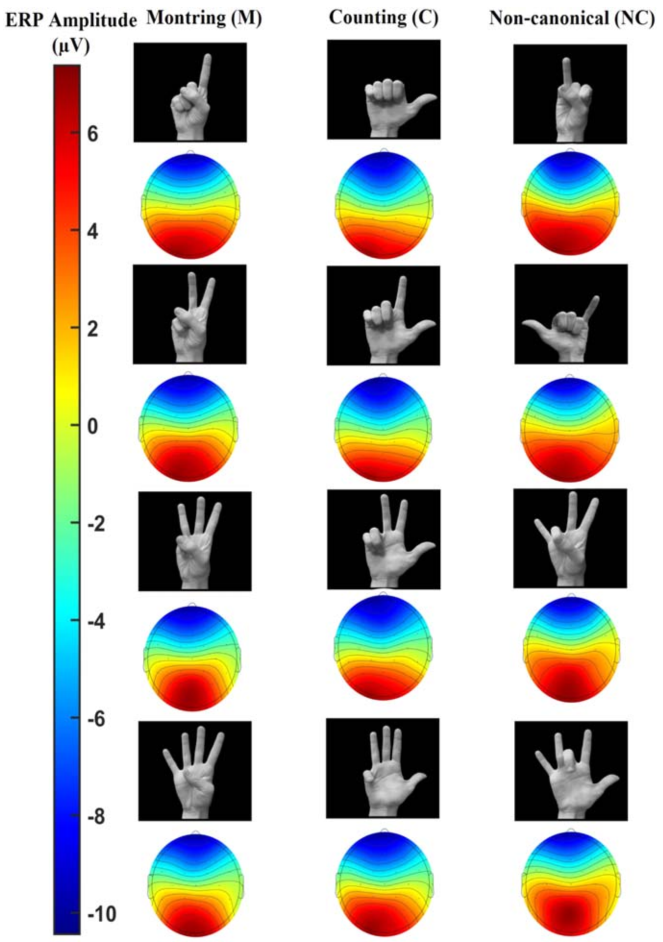

Figure 2. 12 FNCs and the associated topography (Scalp Distribution) of ERPs for Each FNC, at 500 ms, averaged across participants.

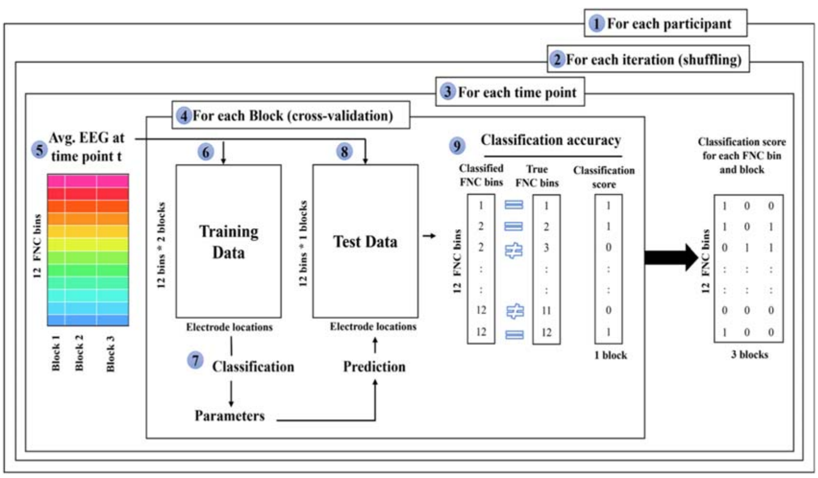

Figure 3. Procedure of ERP decoding analysis by Steven J Luck, Gi-Yeul Bae, and Aaron M Simmons.

Step 4: Threefold cross-validation was implemented by dividing the EEG data into three separate blocks.

Step 5: At each time point, the ERP signals of all classes were computed by taking the average of all trials across all electrodes. These ERP signals were considered as the selected features for the classification.

Step 6: Two-thirds of the ERP signals at each time point were used as input to train the classifier.

Step 7: Once the classifier was trained, its parameters were saved for future use.

Step 8: The trained classifier was then used to predict the labels of the remaining one-third of the data that were not used for training.

Table 1. Detailed information about the ML methods and parameters.

Classifier	MATLAB function	Detail
SVM	*fitcecoc()*	One. versus all, linear kernel
LDA	*fitcecoc()*	One. versus all, linear discriminant, Kernel: normal
NB	*fitcecoc()*	One. versus all
KNN	*fitcecoc()*	One. versus all, k = 12 for decoding all FNCs and K = 4 for category-specific decoding
DT	*fitcecoc()*	One. versus all
NN	*fitcnet()*	Input: [1, 32], FC1: [10, 32], FC2: [4, 10] or [10, 12], output: [1,1]

Step 9: The predicted labels were compared with the true labels to calculate the classification score. To ensure that the data in all three blocks were used as the test dataset, steps 5–9 were repeated three times for each specific time point. As a result, three blocks of classification scores were calculated for each time point, which was averaged to obtain the final score.

Step 2 for each participant was repeated 10 times (iteration = 10), which means that all steps 3–9 were repeated for each participant 10 times. This resulted in a 4D classification score with the dimensions of the number of time points* the number of iterations* the number of cross-validation blocks* and the number of classes. The final classification accuracy was calculated by taking an average of all participants. It should be noted that since the goal was to classify ERP scalp distribution of FNCs, all electrodes were used for classification (Fp1, Fz, F7, FT9, FC5, FC1, C3, T7, TP9, CP5, CP1, Pz, P3, P7, O1, Oz, O2, P4, P8, TP10, CP6, CP2, C4, T8, FT10, FC6, FC2, F4, F8, Fp2, Cz).

Matlab *fitcecoc()* function was used for implementing five of the classifiers (i.e., SVM, LDA, NB, KNN, and DT). This function fits multiclass models for the classifiers combined with error-correcting output codes (ECOC). The ECOC model combines the results of multiple binary classifications to solve multiclass categorization problems. Moreover, the MATLAB *fitcnet()* function was used to train an NN which trains a feedforward, fully connected neural network for classification. The NN in this study had an input layer, a fully connected layer with 10 neurons, a Relu activation function, another fully connected layer with 10 neurons, and then a subsequent 'SoftMax' activation function that produces the network's output, namely classification scores and predicted labels. Detailed information about all ML methods were represented in table 1. It should be noted that all the classification methods were implemented in two conditions: (1) classifying all FNCs together and (2) classifying category-specific FNCs. The classification steps for these two conditions were identical, except for all FNCs together there were twelve classes and for category-specific classification, there were four classes to predict.

2.5. Performance evaluation

Four metrics of overall accuracy, precision, recall, and F-score were chosen to evaluate the performance of all implemented classification methods. The followings are the selected performance metrics:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

In the following formulas, TP (true positive) represents the number of FNCs correctly classified as belonging to a certain class, while FN (false negative) represents the number of FNCs incorrectly classified as not belonging to this class. Similarly, TN (true negative) represents the number of FNCs not belonging to a given class that are correctly classified as such, while FP (false positive) represents the number of FNCs incorrectly classified as belonging to this class. In addition to these measures, confusion matrices were obtained that provided detailed information about the predictability of each of the classes.

3. Results

3.1. Classification results of ERP scalp distribution for all FNCs

Figure 4(a) compares the results of implemented ML methods' classification accuracy for the condition where all FNCs (i.e., 12 classes) were considered. The chance level accuracy, in this case, was 8.30% (1/12), meaning that if the scalp distribution contains no information about the FNCs, the classification accuracy should be around 8.30%. As shown in figure 4(a), SVM had the highest classification accuracy compared to the other methods, 27% at 460 ms. KNN was next with the highest classification accuracy after SVM, 24% at 460 ms. The range of classification accuracy of each of the classifiers across subjects for the time range of −500 ms to 1500 ms was also shown in figure 4(b).

Moreover, confusion matrices for the two classifiers with the highest classification accuracy were presented in figure 5 to compare the decodability of each of the FNCs. In the confusion matrices, the vertical axis typically represents the true classes or stimuli, while the horizontal axis represents the predicted classes. The cell values are normalized by the total number of

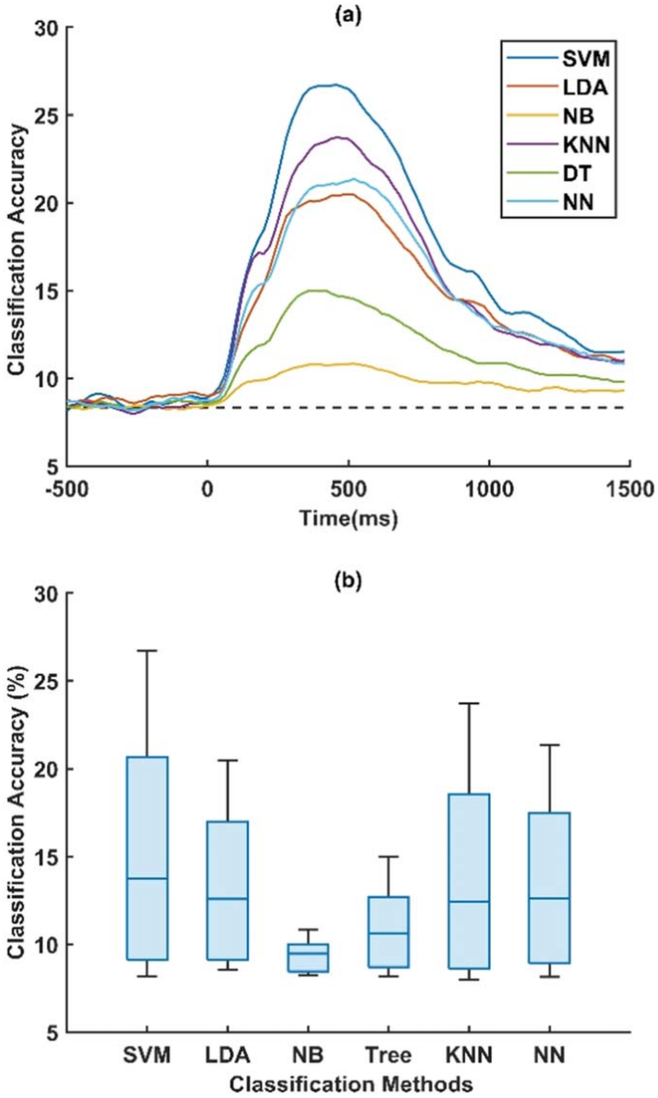

Figure 4. (a) Across time classification accuracy and, (b) Box-chart of ML methods for ERP scalp distribution of all FNCs.

observations with the same true class. The diagonal values indicate the probability of correct prediction for each class, also known as the true positive rate. The values outside of the diagonal in each row represent the false positive rate, which refers to the instances where other classes were incorrectly classified as the true class in that row. The values outside of the diagonal in each column represent the false negative rate, which refers to the instances where the true class was misclassified as other classes. As presented in figure 5, the FNC of NC of number four had the highest true positive rate (19.1%) for SVM, and the FNC of NC of number two had the highest true positive rate of 17.6% for KNN.

In addition to evaluating the classification accuracy, precision, recall, and F-score were also selected as performance metrics for each classifier to allow for comparisons. These metrics were calculated for each individual class across all subjects, and an overall value for each metric was determined by taking the average across all classes. Table 2 summarized the calculated values for the average accuracy, precision, recall, and the F-score of all six ML methods for all FNCs. The SVM model for classifying all FNCs had a precision ratio of 15.57% and a recall ratio of 15.46%.

3.2. Classification results of ERP scalp distribution for categories of M, C, and NC

The selected ML methods were used for category-specific classification where classification was done for each category separately considering four numbers (1, 2, 3, 4) as four classes. Results were provided in figure 6 for three categories of M, C, and NC, and the chance level accuracy was 25% (1/4). The chance level accuracy of 25% means that if the scalp distribution contained no information about the FNCs, the classification accuracy should be around 25%. As shown in figure 6(a), for all category-specific classifications, SVM was the classifier with the highest prediction accuracy. SVM had the highest classification accuracy of 42% for FNC of M at 340 ms, 51% for FNC of C at 420 ms, and 54% for FNC of NC at 600 ms. In addition to SVM, the NN model had a classification accuracy

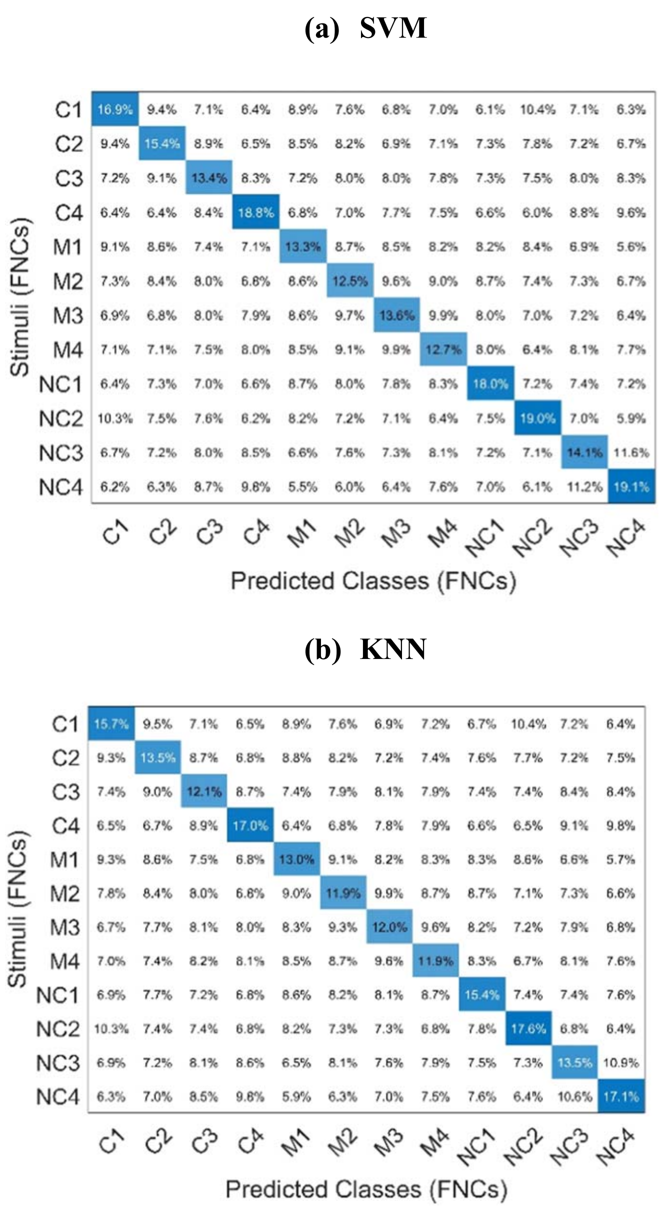

Figure 5. Confusion matrices of implemented (a) SVM, and (b) KNN for all FNCs.

Table 2. Performance measures of ML classifiers for all FNCs in percentage.

Classifier	Avg. Accuracy	Precision	Recall	F-Score
SVM	15.46	15.57	15.46	15.51
LDA	13.45	13.53	13.45	13.49
NB	9.46	10.96	9.46	10.15
KNN	14.13	14.22	14.13	14.18
DT	10.98	11.13	10.98	11.06
NN	13.59	13.62	13.59	13.60

noticeably higher than the chance level accuracy (i.e., 25%) for all three categories. The highest classification accuracy was 40% for MC at 340 ms, 49% for C at 440 ms, and 52% for NC at 600 ms.

Figure 6(b) provided detailed information about the performance of each of the ML methods, including the lowest classification accuracy, highest classification accuracy, and median classification accuracy. For example, for the classification of FNC M, SVM had the lowest accuracy of 24.04%, the highest accuracy of 42%, and the median value for the classification accuracy was 31.31%.

The category-specific confusion matrices of two classifiers of SVM and NN were obtained and shown in figure 7. According to the results, for the M category, the highest true positive rate belonged to montring the number one for both SVM and NN, 33.50% and 32.20%. Also, counting the number four had the true positive rate of 40.70% and 37.80% for the SVM and NN; respectively. For the NC category, SVM had the highest true positive rate of 41.20% for non-canonical counting of number two and NN showed the highest true positive rate for the same number (40.0%).

Tables 3–5 also presented the various performance measures obtained for the ML methods for M, C, and

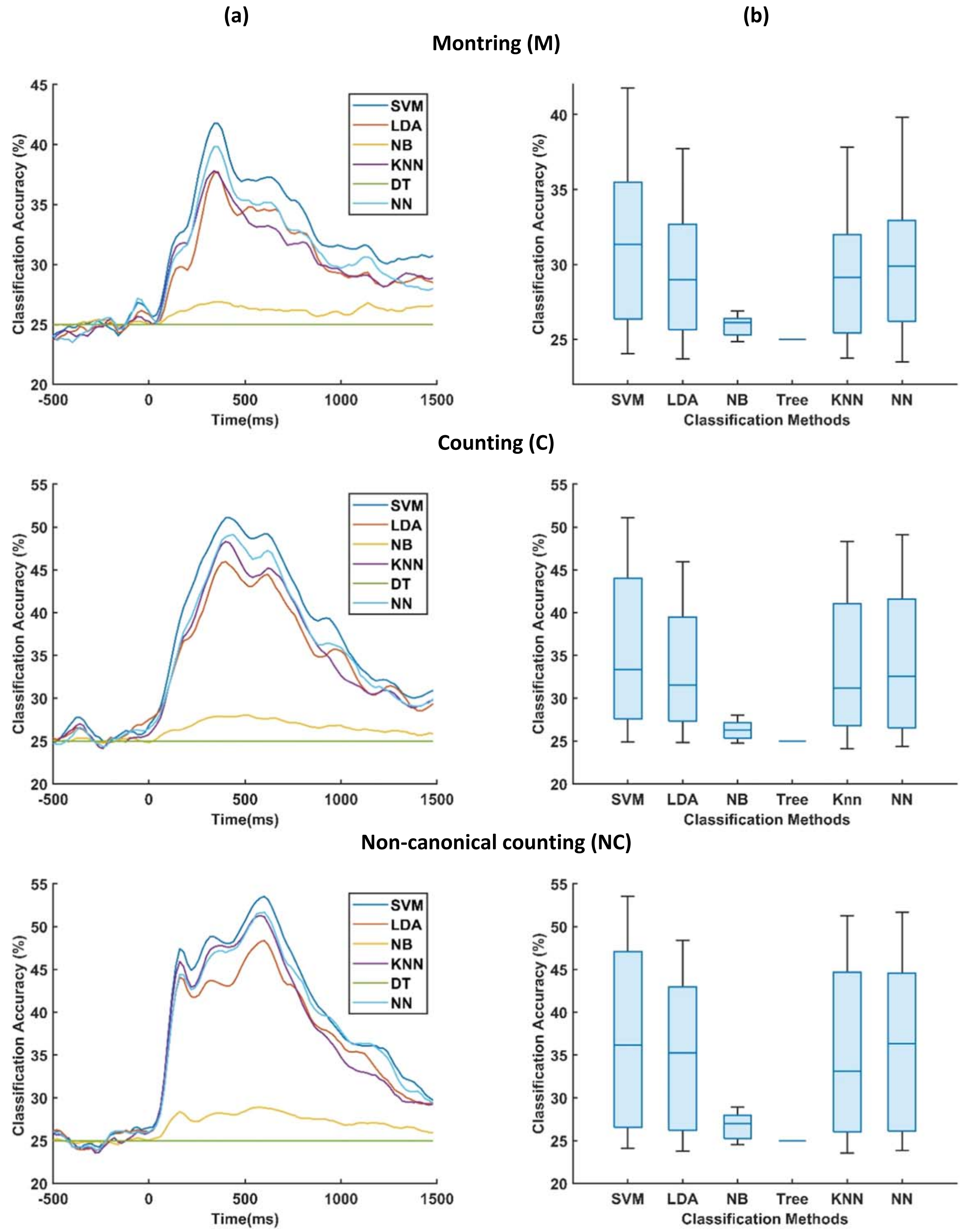

Figure 6. (a) Across time classification accuracy and, (b) Box-chart of ML methods for ERP scalp distribution classification for FNCs of M, C, and NC.

NC. The SVM model classified the test ERP of the scalp distribution with an average precision ratio and a recall rate of 31.38% for the category of M. This model had a precision ratio of 36.08% and a recall ratio of 36.02% for the C category; a precision ratio of 37.38% and recall ratio of 37.29% for the NC category. The average precision ratio and recall rate for the NN model, the second model with the highest classification accuracy, are 30.08% for M; 34.58%, and 34.54% for C; and 36.45% and 36.40% for NC.

4. Discussion

In this study, different ML models were implemented for the classification of ERP scalp distribution of FNCs. The EEG signals belonged to three categories of montring, counting, and non-canonical counting for the numbers1,2,3, and 4. The raw signals were pre-processed (i.e., re-referencing, filtering, epoching, first, artifact detection, and artifact removal) using a customized MATLAB script. ERP scalp distribution

Figure 7. Confusion matrices of implemented (a) SVM, and (b) NN for all three categories of M, C, and NC.

Table 3. Performance measures of ML classifiers for M in percentage.

Classifier	Avg. accuracy	Precision	Recall	F-Score
SVM	31.38	31.38	31.38	31.38
LDA	29.44	29.45	29.44	29.45
NB	25.93	27.65	25.93	26.76
KNN	29.52	29.53	29.52	29.53
DT	25.00	—	25.00	—
NN	30.08	30.08	30.08	30.08

Table 4. Performance measures of ML classifiers for C in percentage.

Classifier	Avg. accuracy	Precision	Recall	F-Score
SVM	36.02	36.08	36.02	36.05
LDA	33.60	33.62	33.60	33.61
NB	26.33	29.40	26.33	52.20
KNN	33.84	33.93	33.84	33.88
DT	25.00	—	25.00	—
NN	34.54	34.58	34.54	34.56

Table 5. Performance measures of ML classifiers for NC in percentage.

Classifier	Avg. accuracy	Precision	Recall	F-Score
SVM	37.29	37.38	37.29	37.34
LDA	35.02	35.03	35.02	35.03
NB	26.75	31.19	26.75	28.80
KNN	35.5	35.64	35.56	35.60
DT	25.00	—	25.00	—
NN	36.40	36.45	36.40	36.42

was calculated from the processed EEG signals and then used as input for classification purposes. The classification procedure was a subject-based and time-specific approach proposed by Bae and Luck [35]. The applied ML methods for classifying FNCs were SVM, LDA, NB, KNN, and DT implemented by MATLAB *fitecoc()* function along with an NN classifier implemented by MATLAB *fitnet()*.

According to the classification accuracy results for all FNCs, SVM was recognized as the most reliable classification method among the implemented methods (i.e., LDA, KNN, NB, DT, NN), with the highest classification accuracy of 27% achieved 460 ms after showing the stimuli. Although the classification was done for the time range of −500 ms (i.e., 500 ms before showing the stimuli) to 1500 ms, the time range of 200 ms to 700 ms was considered critical in this paper since the processing of numerical information included in FNCs happen at this time range [26, 32]. Accordingly, having the highest classification accuracy for SVM at 460 ms was predictable and proves the reliability of the classification method [32]. Also, according to the results, all the ML methods had their highest classification at this specific time range which was a key time for most of the classifiers This was the time that numerical information processing happened [32]. The value of other performance measures, including precision, recall, and F-score was presented for the SVM and the other ML methods. These measures had close values to each other which means that they had the same true and false positive rates. In addition, confusion matrices were also provided for classifying all FNCs, and the highest true positive rate in both SVM and KNN was obtained for NC of numbers two and four. According to these results, it could be concluded that the NC category was the most detectable compared to the other two categories of M and C.

The classification accuracy of all category-specific classifiers was higher compared to the condition that all FNCs were classified together. The classifiers could detect the numerical information present in FNCs in a better way in category-specific classification. This could be explained in this way that category-specific classification eliminates FNCs hard to distinguish from the data which resulted in improving the classification accuracy. Moreover, the two categories of C and NC had the highest classification accuracy in general compared to the M, revealing that less distinguishable processing takes place for montring because there is not much effort needed to process montring compared to the two other FNCs (i.e., C, and NC) [32, 36]. Also, non-canonical finger gestures were more predictable than the FNCs of C since the classification accuracy for almost all the classifiers in this category was in a higher range of threshold. In addition, SVM and then NN had the highest classification accuracy at the time range of 200 ms to 700 ms where the processing of numerical information happens for all three categories, but DT didn't have a good performance at all because the classification accuracy was the same as chance level accuracy (25%). One possible reason for the difference in the performance of DT for the two conditions could be the distribution of the data within the categories. When the classifier is trained using the one versus-all approach with 12 classes, the model has access to more data from all 12 classes, which can improve its ability to distinguish between them. On the other hand, when the classifier is trained using 4 classes within a specific category, the model has access to less data for each class, which can make it harder for the model to accurately distinguish between them. It's also possible that the choice of learner could play a role in the performance difference. Different learners have different strengths and weaknesses, and the decision tree learner may be better suited to handling the 12-class problem compared to the 4-class problem. Moreover, according to the results of precision and F-score reported in tables 3–5, DT has predicted all instances to belong to a single which may be due to insufficient data. Based on the category-specific confusion matrices, M of number one, C of number four, and NC of number two had the highest true positive rate and were more predictable/decodable compared to the other numbers in each category. According to the confusion matrices obtained for both conditions, it was concluded that the NC of number two has the highest possibility of correct prediction.

It is important to acknowledge that the ML methods implemented in this study had certain limitations. Specifically, the parameters of the models were chosen as the default values, which may not have been optimal for the given dataset. However, the results obtained from this study suggest that it is possible to reliably detect numerical information in EEG signals provided by FNCs. Moving forward, we plan to address these limitations by tuning the parameters of the models in future work. By doing so, we aim to improve the performance of the classifiers and gain a better understanding of the underlying patterns in the EEG signals.

5. Conclusion

In this study, the ERP scalp distribution of FNCs was classified using the ML methods of SVM, LDA, NB, KNN, DT, and NN. A comparative study was carried out to assess the performance of the ML method, using a public database with EEG data from 38 subjects. The results showed that SVM had the highest classification accuracy among all the methods for classifying all FNCs together and for category-specific classification. In addition, the results indicated that KNN and NN were also effective in exploiting the information present in the ERP scalp distribution of FNCs. The future work would be to the further testing performance of the ML methods by changing the ECOC classifier or the learners' parameters such as changing the kernel function for SVM and NB, changing the discriminate type for LDA, adjusting the minimum leaf size, minimum parent size, and prediction selection for DT, or retraining the NN with more layers and neurons in the layers (i.e., change the network design). Additionally, the other future direction would be applying deep learning classifiers such as deep neural networks, recurrent neural networks, and convolutional neural networks to the dataset to capture the differences between FNCs. Furthermore, in this study, the ERP scalp distribution of FNCs was used as an input for the classification which means that all the EEG channels collecting signals were used. However, the most effective EEG channels in distinguishing FNCs could be detected first (i.e., channel selection) to improve classification accuracy.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://doi.org/10.7910/DVN/BNNSRG.

ORCID iDs

Roya Salehzadeh https://orcid.org/0000-0002-6522-7615

References

[1] Hosseini M-P, Hosseini A and Ahi K 2021 A review on machine learning for EEG signal processing in bioengineering *IEEE Rev. Biomed. Eng.* **14** 204–18

[2] Saeidi M, Karwowski W, Farahani F V, Fiok K, Taiar R, Hancock P A and Al-Juaid A 2021 Neural decoding of EEG signals with machine learning: a systematic review *Brain Sciences* **11** 1525

[3] Lotte F, Bougrain L, Cichocki A, Clerc M, Congedo M, Rakotomamonjy A and Yger F 2018 A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update *J. Neural Eng.* **15** 031005

[4] Tahernezhad-Javazm F, Azimirad V and Shoaran M 2018 A review and experimental study on the application of classifiers and evolutionary algorithms in EEG-based brain–machine interface systems *J. Neural Eng.* **15** 021007

[5] Abiri R, Borhani S, Sellers E W, Jiang Y and Zhao X 2019 A comprehensive review of EEG-based brain–computer interface paradigms *J. Neural Eng.* **16** 011001

[6] Al-Fraiji S S and Al-Shammary D 2022 Survey for electroencephalography EEG signal classification approaches *Mobile Computing and Sustainable Informatics Lecture Notes on Data Engineering and Communications Technologies* vol 68 ed S Shakya, R Bestak, R Palanisamy and K A Kamel (Singapore: Springer Singapore) 199–214

[7] Jalilifard A, Pizzolato E B and Islam M K 2016 Emotion classification using single-channel scalp-EEG recording *2016 38th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)* 845–9

[8] Zhuang N, Zeng Y, Yang K, Zhang C, Tong L and Yan B 2018 Investigating patterns for self-induced emotion recognition from EEG signals *Sensors* **18** 841

[9] Nandan M, Talathi S S, Myers S, Ditto W L, Khargonekar P P and Carney P R 2010 Support vector machines for seizure detection in an animal model of chronic epilepsy *J. Neural Eng.* **7** 036001

[10] Ford L K, Borneman J D, Krebs J, Malaia E A and Ames B P 2021 Classification of visual comprehension based on EEG data using sparse optimal scoring *J. Neural Eng.* **18** 026025

[11] Hosseini M-P, Tran T X, Pompili D, Elisevich K and Soltanian-Zadeh H 2020 Multimodal data analysis of epileptic EEG and rs-fMRI via deep learning and edge computing *Artif. Intell. Med.* **104** 101813

[12] Quitadamo L R, Cavrini F, Sbernini L, Riillo F, Bianchi L, Seri S and Saggio G 2017 Support vector machines to detect physiological patterns for EEG and EMG-based human–computer interaction: a review *J. Neural Eng.* **14** 011001

[13] Bagheri M and Power S D 2020 EEG-based detection of mental workload level and stress: the effect of variation in each state on classification of the other *J. Neural Eng.* **17** 056015

[14] Amin H U, Mumtaz W, Subhani A R, Saad M N M and Malik A S 2017 Classification of EEG signals based on pattern recognition approach *Frontiers in Computational Neuroscience* **11** 103

[15] Fan J, Wade J W, Bian D, Key A P, Warren Z E, Mion L C and Sarkar N 2015 A Step towards EEG-based brain computer interface for autism intervention *37th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)* 3767–70

[16] Anon *Emotion Classification Using EEG Signals | Proc. of the 2018 2nd Int. Conf. on Computer Science and Artificial Intelligence*

[17] Oktavia N Y, Wibawa A D, Pane E S and Purnomo M H 2019 Human emotion classification based on EEG signals using naïve bayes method *International Seminar on Application for Technology of Information and Communication (iSemantic)* 319–24

[18] Ekhlasi A, Ahmadi H, Molavi A, Nia M S and Nasrabadi A M 2021 EEG signal analysis during Ishihara's test in subjects with normal vision and color vision deficiency *Biomed. Phys. Eng. Express* **7** 025008

[19] Xu X, Lin M and Xu T 2022 Epilepsy seizures prediction based on nonlinear features of EEG signal and gradient boosting decision tree *International Journal of Environmental Research and Public Health* **19** 11326

[20] Singer G, Ratnovsky A and Naftali S 2021 Classification of severity of trachea stenosis from EEG signals using ordinal decision-tree based algorithms and ensemble-based ordinal and non-ordinal algorithms *Expert Syst. Appl.* **173** 114707

[21] Latif G, Al Anezi F Y, Zikria M and Alghazo J 2020 EEG-ECG signals classification for arrhythmia detection using decision trees *Fourth Int. Conf. on Inventive Systems and Control (ICISC)* 192–6

[22] Türk Ö and Özerdem M S 2019 Epilepsy detection by using scalogram based convolutional neural network from EEG signals *Brain Sciences* **9** 115

[23] Xu G, Shen X, Chen S, Zong Y, Zhang C, Yue H, Liu M, Chen F and Che W 2019 A deep transfer convolutional neural network framework for EEG signal classification *IEEE Access* **7** 112767–76

[24] Zhang C, Kim Y-K and Eskandarian A 2021 EEG-inception: an accurate and robust end-to-end neural network for EEG-based motor imagery classification *J. Neural Eng.* **18** 046014

[25] Göbel S M, Shaki S and Fischer M H 2011 The cultural number line: a review of cultural and linguistic influences on the

development of number processing *Journal of Cross-Cultural Psychology* **42** 543–65

[26] Soylu F, Rivera B, Anchan M and Shannon N 2019 ERP differences in processing canonical and noncanonical finger-numeral configurations *Neurosci. Lett.* **705** 74–9

[27] van den Berg F C G, de Weerd P and Jonkman L M 2021 Electrophysiological evidence for internalized representations of canonical finger-number gestures and their facilitating effects on adults' math verification performance *Sci. Rep.* **11** 11776

[28] Badets A, Pesenti M and Olivier E 2010 Response–effect compatibility of finger–numeral configurations in arithmetical context *The Quarterly Journal of Experimental Psychology* **63** 16–22

[29] Sixtus E, Lindemann O and Fischer M H 2020 Stimulating numbers: signatures of finger counting in numerosity processing *Psychological Research* **84** 152–67

[30] Grootswagers T, Wardle S G and Carlson T A 2017 Decoding dynamic brain patterns from evoked responses: a tutorial on multivariate pattern analysis applied to time series neuroimaging data *Journal of Cognitive Neuroscience* **29** 677–97

[31] Soylu F 2019 ublic dataset: ERP differences in processing canonical and noncanonical finger-numeral configurations *Harvard Dataverse* (https://doi.org/10.7910/DVN/BNNSRG)

[32] Salehzadeh R, Rivera B, Man K, Jalili N and Soylu F 2023 EEG decoding of finger numeral configurations with machine learning *Journal of Numerical Cognition* **9** 206–21

[33] Delorme A and Makeig S 2004 EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis *J. Neurosci. Methods* **134** 9–21

[34] Lopez-Calderon J and Luck S J 2014 ERPLAB: an open-source toolbox for the analysis of event-related potentials *Frontiers in Human Neuroscience* **8** 213

[35] Bae G-Y and Luck S J 2018 Dissociable decoding of spatial attention and working memory from EEG oscillations and sustained potentials *J. Neurosci.* **38** 409–22

[36] Van den Berg F C G, De Weerd P and Jonkman L M 2022 Canonical finger-numeral configurations facilitate the processing of Arabic numerals in adults: an event-related potential study *Neuropsychologia* **170** 108214