Chapter 3

A Multiclass EEG Signal Classification Model Using Spatial Features and eXtream Gradient Boosting Algorithm

3.1. Introduction

In EEG-based MI classification, noise is an important factor that directly affects the global performance of the BCI model. Therefore, it is required to implement an effective noise handling technique to minimize the effect of noise from the samples in the early phase of BCI design. In this chapter, we show that an explicit noise-eliminating technique can precisely enhance the discrimination ability of a classifier when used in the identification of a specific mental-state pattern. In the pre-processing phase, we employ a convolution-based Savitzky-Golay filtering approach [57] to alleviate noise and outliers from the cognitive signals of dataset 1. This method fixes successive subsets of adjacent data points with a low-degree polynomial by the method of linear least squares. Originally, this method was popularized by Savitzky and Golay in the approximation of chemical reaction curves in analytical chemistry. This step suppresses irregular spikes present in the noisy EEG signals by increasing the precision of the data without distorting the signal tendency. Further, a Filter Bank Common Spatial Pattern (FBCSP) [58] was used to extract spatio-temporal features from the refined signals. FBCSP formed a large feature set with 11352 attributes. Therefore, Principal Component Analysis (PCA) was applied to filter only significant features with a 99.4% variance score. The extracted features are used in the classification process by the eXtreme Gradient Boosting (XgBoost) algorithm [59] with suitable node split criteria to manage optimal tree height. Finally, a five-fold cross-validation approach concluded the superior performance of XgBoost in terms of high classification accuracy compared to existing results. The Python code used in this chapter can be accessed from Link 1 given in the footer. The flow chart of the proposed methodology is shown in Figure 3.1.

---

[1] https://drive.google.com/drive/folders/1kivHVh5XG_jOm66gCiBitTIN12WaeJZ7?usp=sharing

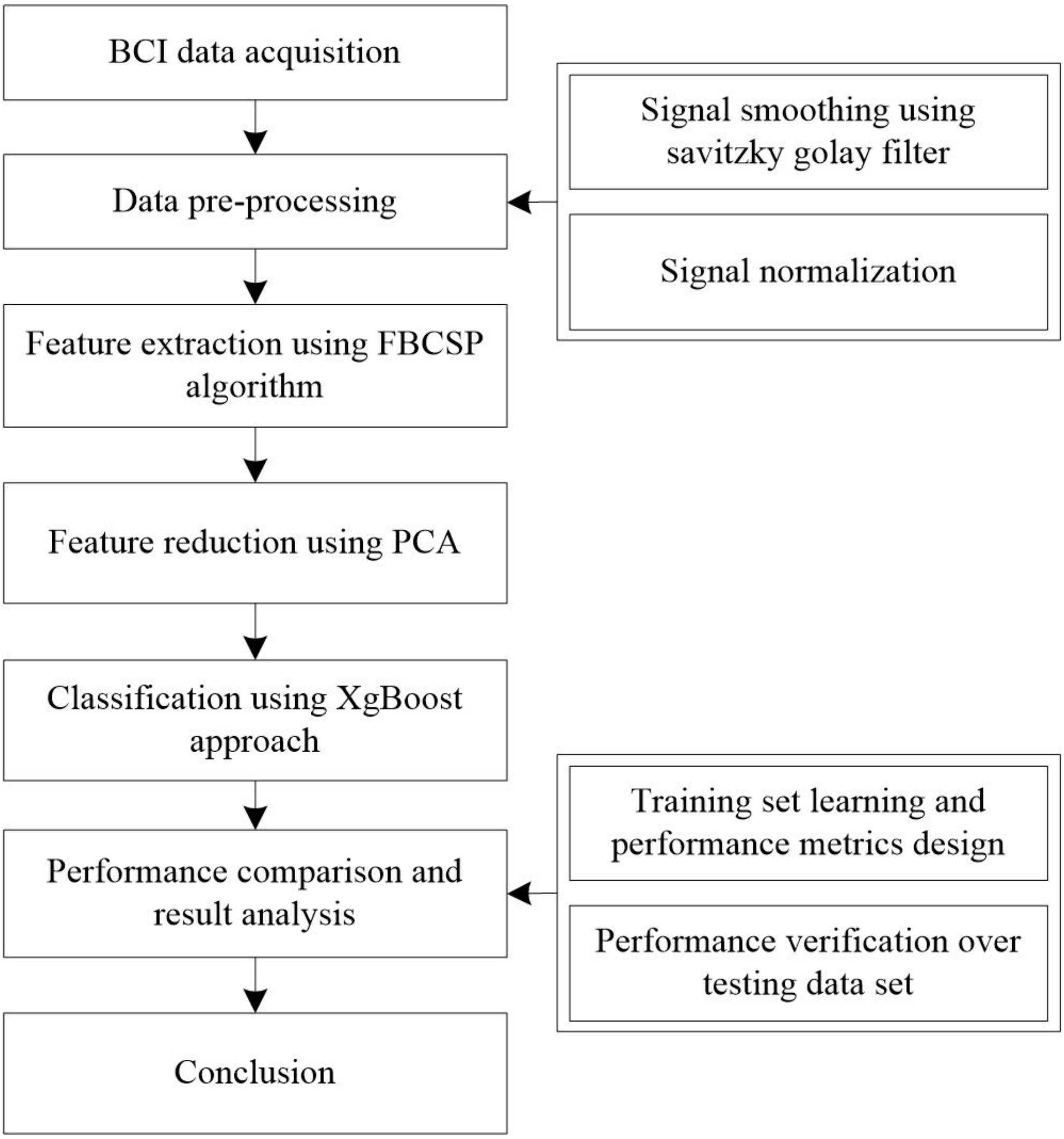


Figure 3.1. Flow diagram of the proposed classification model

3.2. Limitations of the Existing Literature

It is known that poor signal-to-noise Ratio (SNR) limits the performance of EEG-based BCI systems because of low-quality brain signals. The common sources of noise contamination in EEG signals are muscle, eye movement, blinking power lines, and interference with other devices. In addition, an inappropriate noise optimization method may disturb the topographical and functional properties of original signals which may provide unreliable classification results. Therefore, proper noise handling is an essential step to enhance the performance of the developed BCI model.

3.3. Our Contribution

The main contributions of the chapter can be summarised as follows:

1. Considering the limitations of EEG signals, the proposed study focussed on enhancing the signal quality without changing its intrinsic nature.

2. In this work, we used the Savitzky-Golay filtering approach for noise curtailment from the original brain signals.

3. We measured the effectiveness of the applied noise suppression method by minimizing the noise dispersion level in a given signal window.

4. We proposed a four-class classification model on a 22-channel dataset (dataset 1) to discriminate left, right, tongue, and foot movements based on neural signals.

5. We showed that the decision tree-based boosting method is effective in discriminating cognitive patterns from approximated EEG signals.

6. Our approach realized classification accuracy (88.80%) that is superior to
    I. 10% over Regularized common spatial patterns (78.01% accuracy)
    II. 15% over shift variance approximation (73.84% accuracy)
    III. 5% over the Riemannian approach (74.77% accuracy)

3.4. Proposed Methodology

In this phase, three sequential steps: (1) Signal preprocessing, (2) Feature extraction, and (3) Classification are performed to improve the signal quality and obtain the associated cognitive pattern. These steps are briefly discussed in the following subsections:

3.4.1. Signal Preprocessing

Improving the signal quality is an essential step in EEG signal analysis. The original signals are susceptible to noise and outliers hence prone to compute biased and unreliable results. In our work, we demonstrate multiple optimization methods such as (1) smooth filter, (2) median filter, and (3) Savitzky–Golay (SG) filter to smooth the unwanted fluctuations of the multi-channel data. We followed the findings of two research articles [60,61] that are based on the performance of the above-mentioned approaches in terms of (1) Effective smoothing without/ minimum distortion of the original wave shape, and (2) achieving high SNR. A visual inspection of smoothed data and original data showed the suitability of the SG filter over the remaining too. Moreover, the implementation of the SG filter on the channel-wise EEG signals was relatively easier. The performance of all the methods is compared in terms of reduction in noise dispersion level. Comparative studies [60, 61] conclude that the Savitzky Golay filter effectively reduces the dispersion level compared to other methods. **These were the main reasons to use the Savitzky- Golay technique in the preprocessing of motor-imagery signals.** This method employs least-squares fitting of the data by K^{th} order polynomial drawn over the data points. When the data points are equally spaced, an analytical solution to the least-squares equations can be found, in the form of a single set of "convolution coefficients" that can be applied to all data sub-sets, to give estimates of the smoothed signal. Another important parameter is convolution window size (m) which indicates the scope of convolution operation at a specific instance. Here, values of K and m are data-dependent and tuned based on minimum least square error. In our study, we obtained the best results for K= 3 and m =1800 points. Initially, the Savitzky Golay filter was applied then the dispersion level in raw input and smoothened signals was compared. Figure 3.2 shows noise distributions and the estimated dispersion in the training data samples of the first subject (A01) comprising all MI classes. Here, all the images on the left show noise level in raw data, and the ones on the right show a corresponding reduction. The noise dispersion level (Eq. 3.1) is higher in unoptimized signals than in optimized ones because of unwanted fluctuations.

$$\text{Dispersion level in signal } (X) = \Sigma|x_i - E\ (X)| \qquad (3.1)$$

Where X is a set that consists of 'n' signal amplitudes, E (X) represents the mean signal amplitude value, and x_i shows signal instance at index 'i'. In Table 3.1, we show the statistical details of optimized EEG signals for DS1 in terms of dispersion level in original EEG signals (D_o), dispersion level in optimized EEG signals (D_{op}), and enhancement factor (EF) where

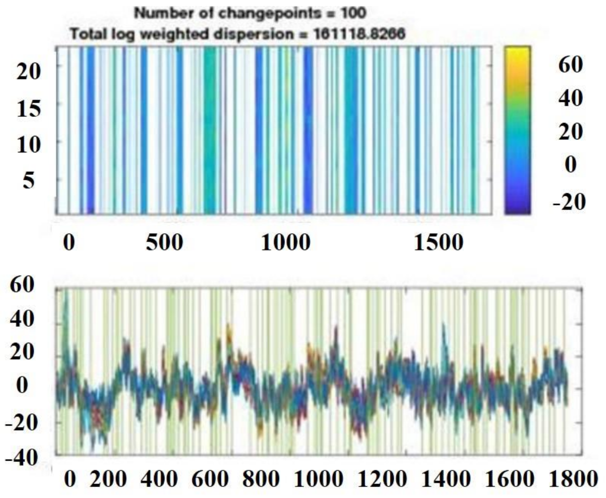
$$\text{Enhancement factor (EF)} = \frac{D_{op}}{D_o} \tag{3.2}$$

From Table 3.1, it can be noticed that filtering effectively improves the SNR of input signals corresponding to all MI activities. The sequence of the motor imagery tasks in terms of enhancement factor is tongue > left hand > feet > right hand. The optimized signals are further normalized in a fixed interval of ([0:1]) such that each instance value must fall in the defined data range. This step is required for multiple reasons such as: (1) to reduce the impact of outliers, (2) to deal with biased data, and (3) to improve the interpretability of the results. Several normalized methods: (1) Z-score, (2) max absolute, (3) unit vector, and (4) Min-Max are available to transform data into a common scale and each of them has its advantages and disadvantages. For example: **z-score normalization can also amplify the effect of noise and max absolute, and unit vector normalization can distort the original scale of the data.** Moreover, their implementation is application-specific and depends on data quality. In our experiment, we selected Min-Max normalization because of a few advantages it has over other methods such as (1) Ease to implement, (2) Interpretability (doesn't alter the distribution of original data), (3) robustness to outliers, and (4) speed (fast and efficient technique that can be applied to large datasets) [62].

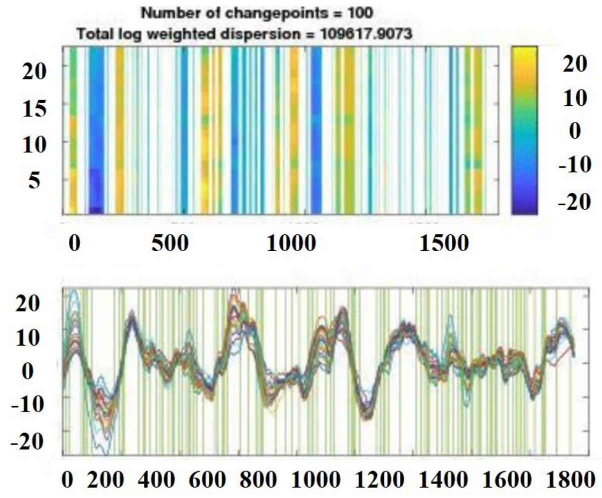
To do it, min and max values of instances from each EEG sample are extracted, and then each value of the data sample is scaled down using Eq. 3.

$$x' = \frac{\mathrm{x} - \min(\mathrm{x})}{\max(\mathrm{x}) - \min(\mathrm{x})} \tag{3.3}$$

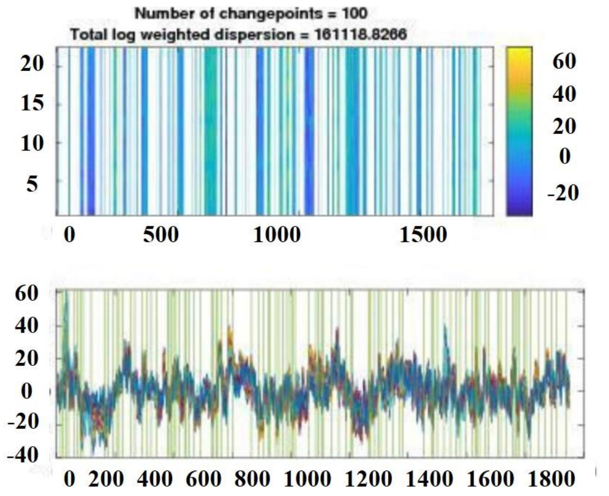
where x is a real signal instance at a time, 't' and x' is the scaled value of the same instance at a time 't'.



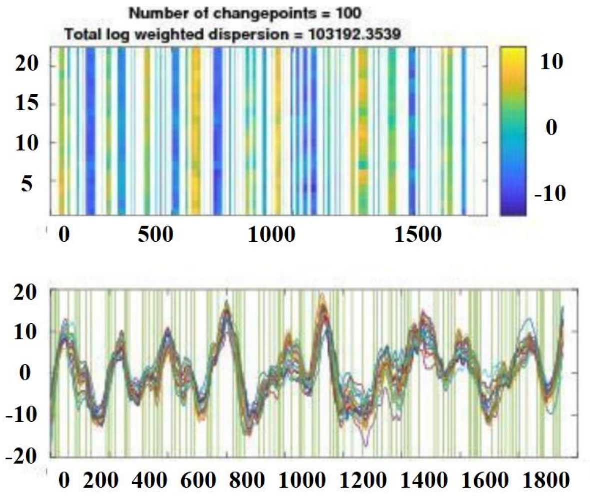
(A). Training sample_Left-hand (Before optimization)



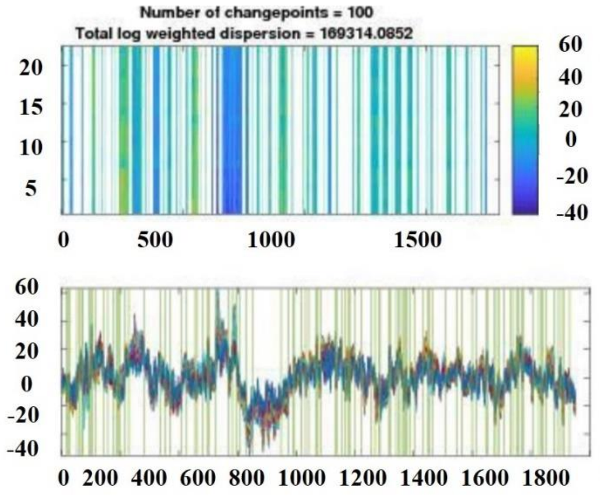
(B). Training sample_Left-hand (After optimization)



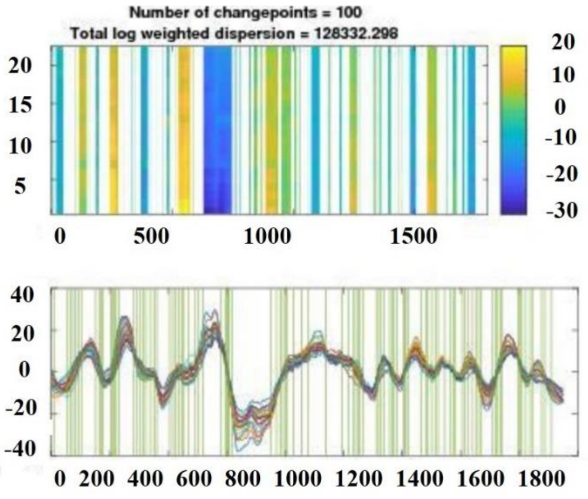
(C). Training sample_Right-hand (Before optimization)



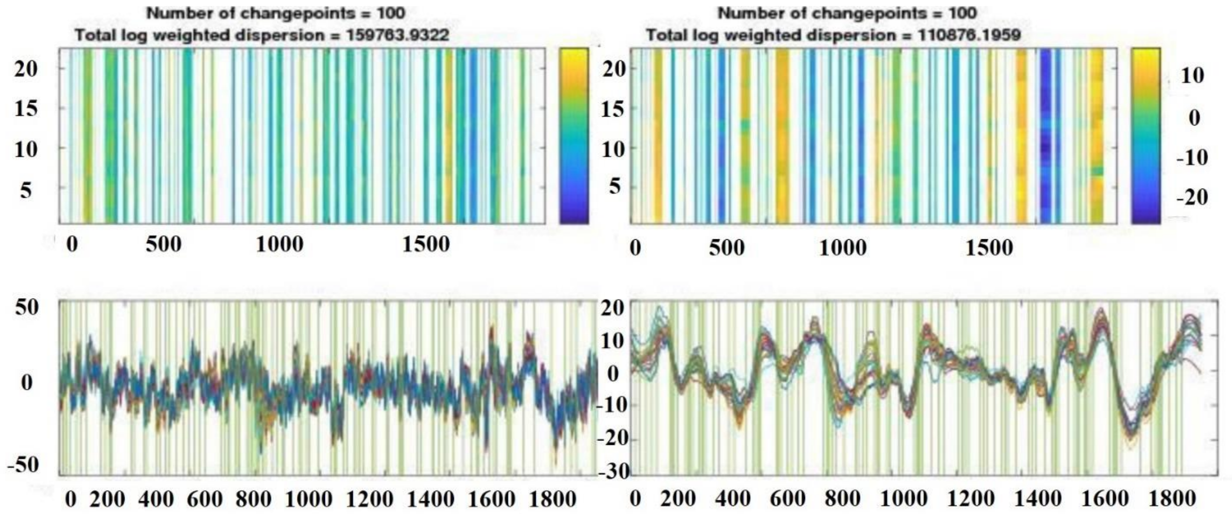
(D). Training sample_Right-hand (After optimization)



(E). Training sample_Tongue (Before optimization)



(F). Training sample_Tongue (After optimization)



(G). Training sample_Feet (Before optimization) (H). Training sample_Feet (After optimization)

Figure 3.2. Performance enhancement in training data samples of dataset 1 using the Savitzky-Golay filter for subject A01. All four windows corresponding to MI classes show weighted dispersion after Savitzky-Golay smoothing the corresponding EEG signal band. A complete window of 1800 data points is considered a benchmark and decomposed into ten equal-sized sub-windows. The color bar shown on the left side of the signal window indicates the distribution of color noise dispersed in the entire window. In each sub-window, the weighted dispersion distribution is also shown.

Table 3.1. MI class-wise dispersion level score in the original and optimized EEG signals (dataset 1)

Index	MI class	Dispersion level in original EEG signals (Do)	Dispersion level in optimized EEG signals (Dop)	Enhancement factor (EF)
1	Left hand	161118.82	109617.90	0.31
2	Right hand	169314.08	128332.29	0.24
3	Tongue	161118.82	103192.35	0.35
4	Feet	159763.93	110876.19	0.30

3.4.2. Feature Extraction

The feature extraction phase is a crucial step in multi-class EEG signal classification. In this paper, we have applied a spatial-temporal-based subject-specific algorithm termed Filter Bank Common Spatial Pattern (FBCSP) for multi-trial BCI EEG dataset classification. One of the main reasons to choose FBCSP is to maximize the relative variance between the pair of values in the Mutual Information States. Another criterion for the selection of FBCSP is its ability to discriminate subject-specific EEG signal spectrum. In several demonstrations, FBCSP has shown noticeable performance gain as compared to the state-of-the-art models. Structurally, the FBCSP algorithm comprises 4 steps to select spatial-temporal features from the EEG spectrum. In phase 1, the EEG channel is decomposed into equidistant signal chunks using the Chebyshev Type II filter bank. In the second phase, decomposed signal chunks are linearly transformed into the following feature vector (Equation 3.4).

$$X = [cf_1, cf_2, \dots, cf_k] \tag{3.4}$$

where $cf_i \in \mathbb{R}^{2m}$ represents m pairs of CSP features for bandpass-filtered EEG measurements. In the third phase, features are selected based on Mutual Information of Best Individual Feature (MIBIF). This algorithm sorts initial k features in decreasing order by considering mutual information of features and then sorted features are used in the classification process. The working of FBCSP is shown in Figure 3.3.

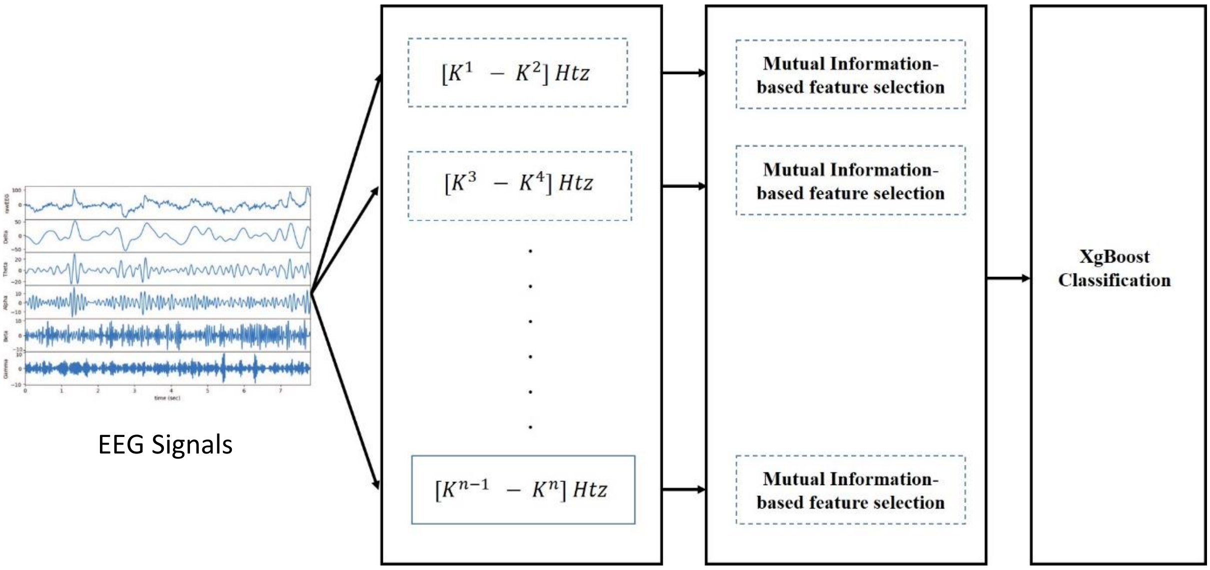


Figure 3.3. A generalized framework of the FBCSP approach

3.4.3. Classification

In the classification step, we classify observations by merging weak learning-based features into a single strong feature in an iterative manner. This process provided a scalable and end-to-end decision tree-based boosting approach i.e. called the XGBoost approach. The XgBoost approach is implemented to resolve this four-class (left hand, right hand, tongue, and feet) classification problem. This approach is also called an ensemble method because it employs a synergy that involves multiple varieties of classifiers to obtain better predictive performance than that which could be achieved by any of the single classifiers. These approaches are very popular and have been successfully used to reduce variance, and bias and achieve improvement in prediction.

The complexity of the proposed methodology mainly depends upon three steps: (1) Preprocessing, (2) Feature extraction, and (3) Classification. In the preprocessing step, the execution complexity of the Savitzky-Golay filter depends on the length of the filter window (n) and the order of the polynomial (m). Experimentally, the execution time can be represented as an order of $O(n^2 * m)$. In our study, we used a 2^{nd} order SG filter so its time will be $O(n^2)$. In the second step, the performance of MEMD is dominated by the calculation of the Hilbert transform which results in a time complexity of $O(n^m * d)$, where n is the number of data points, m is the number of IMFs, and d is the number of dimensions. In the current study, we computed 11 IMFs in the three-dimensional subspace. Therefore, the time complexity of this step will be $O(n^{11})$. Finally, the time complexity of the XgBoost classifier will be given by $O(tdxlog(n))$ where t is the number of trees in the model, d is the maximum depth of each tree, x is the number of non-missing entries in the training data, n is the total number of data points in the training data. In this way, the overall complexity of the proposed approach is the sum of all the steps given by $(O(n^2) + O(n^{11}) + O(tdxlog(n))$ which is an order of $O(n^{11})$. In Python, we have used a scikit-learn [63] library to implement all the classification-related tasks. For classification with the XgBoost, we fixed some of the performance parameters viz. (1) for a small sample set iteration, the learning rate was typically set to 0.01 (default value is 0.3), (2) overfitting was prevented by minimizing sampling frequency (5 Hz) and (3) The optimal depth of the decision tree was found to lie between 3 to 7 but a better accuracy was obtained for depth value 3. To demonstrate, we have split the subject data into four test suits (Training: Testing) as (1) 60: 40, (2)70:30, (3) 75:25, and (4) (80:20) but optimal results are obtained for the 3rd test suit. Nested fivefold cross-validation [64] for the training set is estimated by dividing training data into five equal samples. Each fifth sample was used for validation and the remaining four were used for learning purposes. A permuted sample approach is used iteratively for the entire validation scheme. We have estimated the confusion matrix for individual participants (Table 3.2) and have calculated performance metrics (classification accuracy, precision, recall, F1-score) in Table 3.3.

Table 3.2. Individual confusion matrix details for all subjects.

Confusion Matrix (A01)	[67 4 0 0] [0 70 0 0] [2 0 67 0] [3 5 0 63]
Confusion Matrix (A02)	[64 7 0 0] [1 70 0 0] [3 14 49 3] [0 20 0 52]
Confusion Matrix (A03)	[63 0 4 0] [0 69 1 0] [0 0 66 2] [0 0 4 64]
Confusion Matrix (A04)	[50 1 4 4] [0 55 1 1] [0 0 55 4] [0 0 0 53]
Confusion Matrix (A05)	[66 0 3 1] [10 43 9 3] [2 0 70 0] [2 1 8 58]
Confusion Matrix (A06)	[51 2 0 0] [0 55 0 0] [0 6 46 2] [1 2 0 50]
Confusion Matrix (A07)	[60 11 0 0] [0 69 0 0] [3 8 55 5] [0 6 0 60]
Confusion Matrix (A08)	[63 3 0 0] [7 37 13 11] [4 0 47 18] [0 0 7 61]
Confusion Matrix (A09)	[52 0 2 11] [1 44 1 19] [0 0 69 0] [0 0 0 65]

Table 3.3. The performance of our model is represented in terms of Precision, Recall, and F1 Score for individual subjects.

Subject	Precision	Recall	F1-Score
Sub 1	0.95	0.95	0.95
Sub 2	0.88	0.83	0.83
Sub 3	0.96	0.96	0.96
Sub 4	0.94	0.93	0.93
Sub 5	0.88	0.86	0.86
Sub 6	0.95	0.94	0.94
Sub 7	0.90	0.88	0.88
Sub 8	0.79	0.77	0.76
Sub 9	0.91	0.87	0.87
Average	0.90	0.88	0.88

3.5. Performance Comparison with Baseline Models

The performance of FBCSP with the XGBoost classifier model is substantially higher than the different state-of-the-art models. To compare classification results with existing approaches, we have considered two performance parameters i.e. (1) Accuracy measurement and (2) Robustness of the proposed model. These parameters confirm the superiority of the proposed work over some of the listed research. Performance comparison (Table 3.4) in accuracy level shows that the proposed model shows better results as compared to Lotto et al. (2010) [65] where a regularized Common Spatial Pattern (CSP) algorithm was used to overcome noise influence. One probable reason for this performance enhancement is the early state smoothing of noise level before applying the feature detection approach.

In a similar work, Raza et al. [66] explored intelligence in the feature extraction step by introducing covariance shift and measured the likelihood distribution of input data and the conditional distribution of results over noisy EEG data. This approach performed well in multiclass MI classification but failed to provide a marginal accuracy level in the dataset where training and testing samples overlap (overlapping problem). This problem is a common challenge in those classification problems where the observation count is much less as compared to the feature count (same as in the BCI dataset). In a different comparison [67], feature extraction was based on the symmetric positive definite (SPD) matrix of input data. However, this translation suffers a true curvature structure [68] problem which more often reduces the performance of a classifier.

To analyze the robustness (Table 3.5) of the proposed classification model, we have compared it with the BCI competition winner results. To obtain interrater reliability of estimated features during the experiment, we have approximated Cohen's kappa value for individual subject data. In the BCI winner 1 result [69], FBCSP was applied to extract features, and an average kappa value was estimated as 0.57 while the runner-up implemented CSP on bandpass filtered data (between 8 and 30 Hz) and computed log variance of the best eight components as the features and then classified by LDA and Bayesian classifiers and computed average Kappa 0.52 [70]. The third group applied bandpass filtering between 8 and 25 Hz and used a recursive channel elimination for the channel selection [71]. They, thereafter extracted the CSP feature using an ensemble multi-class classifier using three SVM classifiers and computed a mean Kappa value of 0.31. The fourth group applied CSP on spectrally filtered neural time-series prediction pre-processing (NTSPP) signals at the pre-processing stage and used the log variance of each filtered channel with a one-second sliding window as features and then the best classifier among two variants of support vector machine (SVM) and three variants of LDA was chosen for each subject individually for classification purposes and calculated mean Kappa value 0.30 [72]. In the last approach, authors implemented CSP with SVM as two against two (Class 1 and 2 with Class 3 and 4) classification approaches and validated the results with 2 fold cross-validation method [73].

3.6. Limitations of the Study

From Table 3.4, it is clear that the proposed classification model outperforms different baseline methods. However, our study has the following three limitations:

I. The proposed model suffers from high computational complexity because of parameter tuning in the XgBoost classifier.

II. A grid-search approach is always a time-consuming procedure to determine the node-splitting point and optimal height of the decision tree.

III. The proposed classification model may realize poor classification accuracy when used with default tree parameters therefore, a proper parameter tuning procedure is required to achieve good classification accuracy.

Table 3.4. Performance comparison of the proposed study with state-of-the-art models

Subject	Applied approach	Ref [65]	Ref [66]	Hersche et al., 2018 (Riemannian and SVM performance) [67]					
				Riemannian (Kernel 1)	Riemannian (Kernel 2)	Riemannian (Kernel 3)	SVM (Linear Kernel)	SVM (RBF Kernel)	SVM (Poly. Kernel)
Sub 1	95.02	88.89	90.28	91.81	90.75	84.70	86.83	85.41	83.27
Sub 2	83.04	51.39	54.17	51.59	47.70	48.76	57.24	57.24	49.47
Sub 3	95.97	96.53	93.75	83.52	85.35	84.25	86.45	80.95	77.66
Sub 4	93.42	70.14	64.58	73.25	63.16	58.33	61.40	62.28	56.58
Sub 5	85.87	54.86	57.64	63.41	67.39	62.32	61.23	67.03	60.14
Sub 6	93.95	71.53	65.28	58.60	58.60	56.74	50.70	50.23	44.65
Sub 7	88.09	81.25	62.50	86.64	89.89	81.59	92.42	86.28	70.40
Sub 8	76.75	93.75	90.97	81.55	85.24	79.34	87.82	87.45	86.72
Sub 9	87.12	93.75	85.42	82.58	80.30	80.68	79.17	85.98	79.92
Avg.	88.80	78.01	73.84	74.77	74.27	70.75	73.69	73.65	67.65

Table 3.5. Robustness measurement (Cohen's kappa value) between the proposed approach and state-of-the-art models

Subject	Proposed Method	BCI 1 Ref. [69]	BCI 2 Ref. [70]	BCI 3 Ref. [71]	BCI 4 Ref. [72]	BCI 5 Ref. [73]
Sub 1	0.711	0.68	0.69	0.38	0.46	0.41
Sub 2	0.565	0.42	0.34	0.18	0.25	0.17
Sub 3	0.746	0.75	0.71	0.48	0.65	0.39
Sub 4	0.682	0.48	0.44	0.33	0.31	0.25
Sub 5	0.494	0.40	0.16	0.07	0.12	0.06
Sub 6	0.319	0.27	0.21	0.14	0.07	0.16
Sub 7	0.727	0.77	0.66	0.29	0.00	0.34
Sub 8	0.757	0.75	0.73	0.49	0.46	0.45
Sub 9	0.774	0.61	0.69	0.44	0.42	0.37
Average	0.641	0.57	0.52	0.31	0.30	0.29

3.7. Conclusion & Future Scope

These enhanced results show some advanced tradeoffs of the FBCSP algorithm with the XGBoost classifier. It estimates the significance of the intelligent node-splitting approach in signal classification. The strong node splitting criterion regulates the flow of EEG information in a deterministic and hierarchical manner so a decision tree with a fast Boosting algorithm can also be used to solve the multi-class classification problem. This work also demonstrates FBCSP and XG Boost as an effective package for non-stationary EEG data that processes spatial features in a more sophisticated way. Statistical analysis shows that this package is efficient in capturing the non-deterministic behavior of cognitive processes.

In existing research articles, authors have ignored the significance of noise reduction but current work emphasizes noise minimization as a prerequisite step in the signal pre-processing phase. In the future, a chaotic feature extraction algorithm such as a stochastic Petri-net-based transition model [74] can be used to better discriminate MI activities because it can capture variance levels of the action potential of different neurons. Higher-level optimization functions such as randomized smoothing techniques and convex smoothing can also be used to approximate EEG fluctuations. In another optimization, node splitting criteria can be designed keeping in mind the sensitive EEG data points.