**Multivariate EEG Analysis for Schizophrenia Detection: Leveraging Entropy Measures and Machine Learning Techniques across Channel Configurations**

A Report Submitted

in Partial Fulfillment of the Requirements for the Degree of

### Bachelor of Technology

in

### Computer Science & Engineering

by

### Tushar Goyal (20214193)

### Utkarsh Mishra (20214226) Vasant Kr Sharma (20214312) Sayog Shendre (20214513)

Under the guidance of **Dr. Deepak Gupta Assistant professor**

****

**COMPUTER SCIENCE AND ENGINEERING DEPARTMENT** MOTILAL NEHRU NATIONAL INSTITUTE OF TECHNOLOGY ALLAHABAD PRAYAGRAJ

### May, 2025

**UNDERTAKING**

I declare that the work presented in this report titled “*Multivariate EEG Analysis for Schizophrenia Detection: Leveraging Entropy Measures and Machine Learning Tech- niques across Channel Configurations*”, submitted to the Computer Science and Engineering Department, Motilal Nehru National Institute of Technology Allahabad, Prayagraj, for the award of the ***Bachelor of Technology*** degree in ***Computer Science & Engineering*** , is my original work. I have not plagiarized or submitted the same work for the award of any other degree. In case this undertaking is found incorrect, I accept that my degree may be unconditionally withdrawn.

May, 2025

Allahabad

Tushar Goyal (20214193)

Utkarsh Mishra (20214226) Vasant kr Sharma (20214312)

Sayog Shendre (20214513)

**CERTIFICATE**

Certified that the work contained in the report titled “*Multivari- ate EEG Analysis for Schizophrenia Detection: Leveraging En- tropy Measures and Machine Learning Techniques across Chan- nel Configurations*”, by Tushar Goyal, Utkarsh Mishra, Vasant Kr Sharma, and Sayog shendre, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

(Dr. Deepak Gupta) Assistant Professor

Computer Science and Engineering Dept. M.N.N.I.T, Allahabad

May, 2025

**Preface**

Schizophrenia is a complicated neuropsychiatric illness diagnosed with difficulties because of its subjective nature and various symptoms such as hallucinations and social withdrawal. In this work, we have built a robust diagnostic model using EEG data for correct and early diagnosis. The EEG signals of healthy and patient subjects were preprocessed and divided into epochs; each epoch consisted of 500 samples and lasted two seconds at a sampling rate of 250 Hz. For consistency and further processing, 50 random epochs were selected from each signal.

Shannon entropy, Spectral entropy, Sample entropy, Approximate entropy and Tsallis entropy were some of the various entropy features extracted from these epochs, providing insight into the meaningful underlying neural patterns. These fea- tures will be used for training and evaluating the different machine learning models, namely Gradient Boosting, Random Vector Functional Link, Ensemble Deep Ran- dom Vector Functional Link, Extreme Learning Machines, and a newer type of ensemble learning called Twin Extreme Learning Machine (TELM).

By leveraging advanced computational techniques, this study aims to facilitate early and accurate diagnosis, paving the way for personalized treatment strategies. The ultimate goal is to improve patient outcomes and reduce the societal burden of schizophrenia.

**Acknowledgements**

We extend our heartfelt appreciation to all those who have contributed to the com- pletion of this project. In particular, we extend our gratitude to Dr. Deepak Gupta, our mentor and supervisor, whose unwavering support, guidance, and encourage- ment have been invaluable throughout this endeavor. We are grateful for the ample opportunities he has provided us to explore the content and dimensions of our work, and we consider it a privilege to have worked under his mentorship.

We also wish to express our sincere thanks to Prof. R. S Verma, Director of MNNIT Allahabad, Prayagraj, and Dr Mayank Pandey, Head of the Computer Science and Engineering Department, for their support and provision of the necessary facilities for the completion of this project.

Finally, we extend our appreciation to our friends and family for their unwavering support and invaluable advice throughout this journey. Their love, blessings, and encouragement have been a constant source of motivation, without which the com- pletion of this project would have been impossible.

**Contents**

|  |  |
| --- | --- |
| [**Preface**](#_bookmark0)  [**Acknowledgements**](#_bookmark1) | **iv**  **v** |
| [**1 Introduction**](#_bookmark2) | **1** |
| [1.1 Motivation](#_bookmark3) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 4 |
| [1.2 Objectives](#_bookmark4) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 4 |
| [**2 Related works**](#_bookmark5) | **6** |
| [2.1 Models](#_bookmark6) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 6 |
| [2.1.1 Gradient Boosting](#_bookmark7) . . . . . . . . . . . . . . . . . . . . . . . . | 6 |
| [2.1.2 Extreme Learning Machine](#_bookmark8) . . . . . . . . . . . . . . . . . . . . | 7 |
| [2.1.3 Twin Extreme Learning Machine](#_bookmark9) . . . . . . . . . . . . . . . . | 9 |
| [2.1.4 Random Vector Functional Link](#_bookmark10) . . . . . . . . . . . . . . . . . | 10 |
| [2.1.5 Ensemble Deep Random Vector Functional Link](#_bookmark11) . . . . . . . . | 12 |
| [**3 Proposed Methodology**](#_bookmark12) | **14** |
| [3.1 Data Collection](#_bookmark13) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 14 |
| [3.2 Working and Explanation](#_bookmark16) . . . . . . . . . . . . . . . . . . . . . . . . | 17 |
| [3.2.1 Data Preprocessing:](#_bookmark18) . . . . . . . . . . . . . . . . . . . . . . . . | 17 |
| [3.2.2 Feature Extraction](#_bookmark19) . . . . . . . . . . . . . . . . . . . . . . . . | 18 |
| [3.2.3 Feature Selection:](#_bookmark20) . . . . . . . . . . . . . . . . . . . . . . . . . | 19 |
| [3.2.4 Feature Vector Preparation](#_bookmark21) . . . . . . . . . . . . . . . . . . . | 21 |
| [3.2.5 Model Training:](#_bookmark22) . . . . . . . . . . . . . . . . . . . . . . . . . . | 22 |

[3.2.6 Result Analysis and Comparison:](#_bookmark23) 23

1. [Experimental Setup and Results Analysis](#_bookmark24) 24
   1. [Experimental setup](#_bookmark25) 24
   2. [System Requirements](#_bookmark27) 24
   3. [Technology Used](#_bookmark28) 25
   4. [Performance metrics](#_bookmark29) 26
   5. [Result Analysis](#_bookmark30) 27
   6. [All models Result Comparions](#_bookmark41) 33
2. [Conclusions and Future scope](#_bookmark45) 35

**Chapter 1 Introduction**

Mental disorders encompass a broad spectrum of conditions that impact an individ- ual’s thoughts, emotions, behavior, and overall mental well-being. Their severity can vary greatly from short-term issues like anxiety and depression to more persistent and complex conditions such as schizophrenia and bipolar disorder. These disorders can significantly hinder a person’s ability to function in daily life, affecting relation- ships, job performance, and even physical health. They are typically influenced by a combination of genetic, environmental, and psychological factors. However, with the right approach often involving therapy, medication, and strong support systems many mental health conditions can be effectively managed.

Schizophrenia is a serious mental disorder that affects key areas of functioning, including thoughts, awareness, emotions, and behavior. People with this condition often experience symptoms such as hallucinations (perceiving things that aren’t real), delusions (strong false beliefs), disorganized thinking, emotional flatness, and social withdrawal. Antipsychotic medications and psychotherapy can help manage these symptoms, but in cases where proper care is lacking, it may lead to severe outcomes like self-harm or suicide. The disorder impacts both mental and physical health, making diagnosis and treatment especially challenging.

This mental disorder often begins in adolescence or early adulthood and may worsen over time. It can affect many areas of life, including work, relationships,

physical health, self-esteem, and daily functioning. While the exact cause is un- known, it is believed to arise from a combination of genetic factors, emotional stress, environmental influences, and changes in brain chemistry. Together, these factors can significantly reduce a person’s overall quality of life.

Our study focuses on detecting schizophrenia using electroencephalogram (EEG) signals, which include data from multiple channels along with various entropy mea- surements. The EEG recordings, sourced from the RepOD dataset, include data from both healthy individuals and patients diagnosed with schizophrenia. These signals were captured over time using 19 electrodes placed on the scalp. By analyz- ing differences in EEG signal patterns across these electrodes, we aim to distinguish between healthy individuals and those exhibiting signs of schizophrenia.

Schizophrenia often develops during adolescence or early adulthood and tends to become more severe over time. It can deeply affect many aspects of a person’s life, including their career, relationships, physical health, self-esteem, and ability to carry out everyday activities. Although the exact cause is still not fully understood, it is widely believed to result from a combination of genetic factors, emotional stress, environmental influences, and changes in brain function. These combined factors can greatly reduce a person’s overall quality of life.

Hwang et al.[1] proposed in her study that auditory hallucinations (AH), com- monly seen in schizophrenia but also present in bipolar disorder, may be linked to specific patterns of brain connectivity. Using resting-state fMRI, she found that individuals with AH showed altered functional connectivity between the cerebellum and various brain regions, particularly in the temporal and frontal lobes. Her find- ings suggest that AH may have shared neural signatures across psychotic disorders, along with unique patterns within each condition.

To classify the extracted features, the authors used a support vector machine equipped with the Radial Fundamental Function (SVM-RBF) kernel, a popular machine learning classifier known to have the potential to efficiently process HD data. The SVM-RBF classifier was trained with 95 functions of the EEG signal. This approach yielded impressive results, with the model achieving accuracy and an F1 score of 93%. This shows a high level of performance in the distinction between

health and schizophrenia.

The properties used in this study were carefully cut based on two important aspects. Channels (scalp position where EEG signals are recorded) and entropy measurements. This branching allowed the model to take into account the spatial distribution of EEG signals across 19 channels and the complexity of the signals recorded by different entropy roads. By combining these factors, same as Krishnan et al.[2], it may improve classifier capabilities, recognize patterns in EEG data, and create comprehensive functional rates that contribute to a more accurate diagnosis of schizophrenia.

Boutros et al. [3] proposed a study reviewing EEG spectral abnormalities as a potential diagnostic tool for schizophrenia. Their meta-analysis found consistent increases in slow brain rhythms in schizophrenia patients, even when unmedicated. However, they noted that further research with standardized criteria and diagnostic accuracy data is needed before EEG can be used clinically for diagnosis. Tarci et al. [4] also shows the same.

Verma et al. [5] conducted a comprehensive review of machine learning tech- niques to diagnose schizophrenia. They investigated a variety of methods, including machine learning classifiers, artificial neuronal networks, deep learning models, and applications in previous research. This paper aims to identify research gaps to de- velop more accurate diagnostic models and treatments for schizophrenia.

Jafari et al. [6] conducted a study of schizophrenia that included significant cognitive, emotional and behavioral changes. In the diagnosis of schizophrenia re- searchers, they used machine learning and deep learning approaches, as well as artificial intelligence.

Tanveer et al. [7] examined various techniques for extracting properties and selection and presented papers identifying the correct functionality rate. This pa- per includes a comprehensive review of several studies focusing on the detection of schizophrenia using EEG signals and highlighting the use of machine learning and deep learning for checking.

Oh and Vienesh et al.[8] and Kutepov [9] also done shows in there research of deep Convolutional Neural Network Model for Automated Diagnosis of Schizophrenia

Using EEG Signals. It shows how DL is helpful in schizophrenia diagnose.

Harmah et al.[10] research in easuring the non-linear directed information flow in schizophrenia by multivariate transfer entropy, shows that how entropy can be a valuable feature for schizophrenia detection. Zahar A et al.[11] research in schizophre- nia detection from EEG signals using multivariate empirical mode decomposition.

* 1. **Motivation**

Detecting schizophrenia remains one of the toughest challenges in mental health care. Right now, doctors mostly rely on observing behavior and ask- ing patients about their experiences that can be highly subjective and open to misunderstanding. This often leads to delayed diagnoses or even misdiagnoses, making it harder for people to get the help they need when it matters most.To improve this process, we’re exploring the use of electroencephalogram (EEG) technology, which records the brain’s electrical activity in real time. EEG can detect changes in brain function that may signal the early stages of schizophre- nia long before traditional symptoms become obvious. By tracing this data, we hope to develop tools that support earlier, more accurate diagnoses. Our goal is to bring together neuroscience, technology, and clinical insight to create practical solutions that improve patient care. Ultimately, we want to reduce the impact of schizophrenia on individuals and their families by enabling faster interventions and better treatment outcomes. Kannathal et al. [12] shows how entropy can be a useful with EEG signals so we are proceeding with that.

* 1. **Objectives**

This report explores how EEG technology can be used to build more accu- rate and reliable tools for diagnosing schizophrenia. While earlier studies have looked into different ways of processing EEG signals, our focus is on under- standing how the number of EEG channels and the way we analyze their data affects the accuracy of diagnosis. By testing different channel combinations

and using entropy based methods, we aim to find the setup that delivers the best results. Abasolo et al [13] also shows different Entropy analysis of the EEG activity in SCZ disease patients.The ultimate goal is to make it easier to detect schizophrenia early, so people can get the right support and treatment sooner. In the long run, we hope this work can lead to better care and a better quality of life for those living with the disorder.

**Chapter 2 Related works**

* 1. **Models**
     1. **Gradient Boosting**

Gradient Boosting [16] is a powerful, monitored learning technology used for clas- sification and regression tasks introduced by Jerome Friedman in the late 90s. In a gradient hub, the main goal is to build a powerful predictive model by sequen- tially combining several weak learners, usually several weak learners who determine the tree. All subsequent learners are trained to correct the errors in the previous ensemble and gradually improve the performance of the model at each step. This approach helps to record complex patterns in your data and make them suitable for a variety of practical problems. This model is created with regard to stages by repeatedly adding new functions to reduce residual errors. The general format of the N-character combination boost model is:

*N*

Σ

*TN* (*x*) = *γnhn*(*x*) (1)

*n*=1

Here *hn*(*x*) is the weak learner in the *n*-th stage, and *γn* is a weight that scales its contribution to the overall model.

At each stage, the algorithm computes the negative gradient of the loss function with respect to the current model prediction:

*rim*

= − *∂L*(*yi, T* (*xi*)) for *i* = 1*,* 2*, . . . , M* (2)

*∂T* (*xi*)

These residues act as pseudo-labels, and new learners *hn*(*x*) are trained to predict them. The model is then updated by adding new learners to the optimal step size.

*Tn*(*x*) = *Tn*−1(*x*) + *γnhn*(*x*) (3)

In this framework, *T* (*x*) is the overall prediction function, and *L*(*y, T* (*x*)) is the loss function that measures the difference between actual and predicted values.

Real-world data records often contain nonlinear relationships, and gradient boost negotiations effectively handle this by learning a tree structure that divides the data based on distinctive values. In contrast to linear models, no explicit transformation into a higher dimensional space is required. However, due to the growing number of learners, models are susceptible to overadjudication, and normalization techniques such as contraction, partial scan, and tree procurement may be required to maintain generalized performance and reduce computing load.

* + 1. **Extreme Learning Machine**

ELM [17] is an ML technology used for both classification and regression tasks known for their simplicity and speed. It is based on a single-layer feedforward neural network (SLFN). In contrast to traditional networks, hidden layer weights are first started and remain determined during training. Only the starting layer weights are learned, eliminating the need for complex iterative optimization algorithms that are typically required in traditional neuronal networks. The exit functions of the ELM of a generalized SLFN can be expressed as follows:

*L*

Σ

*f* (**x**) = *G*(**w***i, bi,* **x**)*βi* = **h**(**x**) · ***β****,* (4)

*i*=1

where ***β*** represents the output weight vector connecting the hidden layer to the output layer. The activation function *G*(**w***i, bi,* **x**) is parameterized by **w***i*, the in- put weight vector, and *bi*, the bias of the *i*-th hidden node. The term **h**(**x**) =

[*G*(**w**1*, b*1*,* **x**)*, . . . , G*(**w***L, bL,* **x**)]*T* defines the nonlinear feature mapping from the in- put **x** to the hidden layer.

ELM aims to minimize both the output weight norm and the training error. This optimization problem is defined as:

*N*

Σ

min |***β*** · **h**(**x***i*) − *ti*| and min ***β *** *,* (5)

*i*=1

where *ti* is the target output corresponding to the *i*-th training sample.

If the training error is reduced to zero, i.e.,

*N*

Σ

|***β*** · **h**(**x***i*) − *ti*| = 0*,* (6)

*i*=1

the solution for ***β*** is computed as:

***β*** = **H**†**T***,* (7)

where **T** is the target output matrix, and **H**† is the Moore-Penrose generalized inverse of the hidden layer output matrix **H** = [**h**(**x**1)*, . . . ,* **h**(**x***N* )]*T* .

To address potential overfitting, one can relax the zero training error condition, introducing slack variables *ξi*. The modified optimization problem becomes:

subject to:

1

min

2

*N*

 ***β *** 2 + *C ξi,* (8)

Σ

*i*=1

*ti*(***β*** · **h**(**x***i*)) ≥ 1 − *ξi,*

*ξi* ≥ 0*, i* = 1*,* 2*, . . . , N,*

where *C >* 0 is a regularization parameter controlling the trade-off between the margin and training error, and *ξi* represents the error for the *i*-th sample.

The decision function for this formulation of ELM is:

*f* (**x**) = sign

*Ns*

*s*=1

Σ

*αstsK*ELM(**x***,* **x***s*)!

*,* (9)

where *αs* are the Lagrange multipliers, *ts* are the target outputs, and *K*ELM(**x***,* **x***s*) is the kernel function for ELM.

Because of its efficiency and simplicity, ELM is suitable for tasks with large data records. It was used successfully in areas such as image recognition, language processing, time series prediction, and medical diagnosis. Healthcare allows ELM to analyze EEG signals. This provides a quick and effective way to identify dis- eases such as schizophrenia by identifying patterns of brain activity without the mathematical complexity associated with more traditional deep learning models.

* + 1. **Twin Extreme Learning Machine**

TELM [18] is an advanced version of traditional ELM, improving the generalization function of the model, especially when there are challenges such as domain shifts and limited data. Telm integrates Learning with ELM to use knowledge from the source domain and the target domain. The Twin Extreme Learning Machine (TELM) model solves the following several square programming problems (QPPS):

1



min

**Uw **2 + *c* **e**⊤*ξ*

subject to

**w**1*,ξ*1 2

1 1 2 1

−**Vw**1 + *ξ*1 ≥ **e**2*, ξ*1 ≥ 0 (10)

and

1



min

**Vw **2 + *c* **e**⊤*η*

**w** *,η* 2

2 2 1 2

2 2

subject to

**U***w*2 + ***η*** ≥ **e**1*,* ***η***2 ≥ 0 (11) Here, *c*1 *>* 0 and *c*2 *>* 0 are user-defined parameters.we derive their corresponding Wolfe dual problems, which are given by: The dual form of TELM is given by:

subject to

2

max **e**⊤***α*** −

***α*** 2

1 ***α***⊤**V** **U**⊤**U** + *ϵ***I** −1 **V**⊤***α*** (12)

0 ≤ ***α*** ≤ *c*1**e**2*,*

and

subject to

max **e**⊤***γ*** −

***γ*** 1

1 ***γ***⊤**U** **V**⊤**V** + *ϵ***I** −1 **U**⊤***γ***

0 ≤ ***γ*** ≤ *c*2**e**1*.* (13)

2

Here, ***α*** and ***γ*** are vectors of Lagrange multipliers. After obtaining the optimal values of ***α*** and ***γ***, the decision variables are calculated as:

and

1

***β*** = − **U**⊤**U** + *ϵ***I** −1 **V**⊤***α****,* (14)

***β*** = − **V**⊤**V** + *ϵ***I** −1 **U**⊤***γ****.* (15)

2

For classifying an unseen test data point **x** ∈ R*n*, we use the decision rule:

*f* (**x**) = arg min *dk*(**x**) = arg min |*h*(**x**)***β****k*| *.* (16)

*k*=1*,*2 *k*=1*,*2

The main advantage of TELM is that it can use existing knowledge from related domains and reduce the dependency on large labeled data records in the target domain. Telm maintains the benefits of ELM efficiency and fast training, and uses transfer learning to improve accuracy. As a result, TELM can be used in areas such as medical diagnosis, image detection, and language processing, particularly in situations where labeled data is limited or difficult to purchase.

* + 1. **Random Vector Functional Link**

RVFL [19] models (random vector functional links) are important machine learning techniques for classification and regression problems. This works by first showing the input data in a higher dimensional space via pre-determined and unadjusted random projections during training. This projection helps to reveal more complex patterns within the data by expanding its dimensions. After this transformation, nonlinear activation functions such as sigmoid, tan, or relu are applied to the projection data, allowing the model to record complex relationships.

The Random Vector Functional Link (RVFL) network is a robust single-layer feedforward network (SLFN) that eliminates the need for iterative weight tuning by initializing weights and biases randomly. Let ***θ*** = (*θ*1*, θ*2*, . . . , θP* )*T* , where *P* = *L*+*n*, represent the weight vector connecting to the output layer. Here, *L* is the number of enhancement nodes (hidden layer nodes).

The weight vector for the enhancement layer is given as ***ϕ****l* = (*ϕl*1*, ϕl*2*, . . . , ϕlm*)*T* , and *b* is the bias introduced in the enhancement nodes. The activation function *g*(·*,* ·*,* ·) produces the output for the *l*-th enhancement node with respect to the *i*-th training sample as:

*ql*(**x***i*) = *g*(***ϕ****l, bl,* **x***i*)*,*

where *l* = 1*,* 2*, . . . , L* and *i* = 1*,* 2*, . . . , m*.

The Hessian matrix **H**, which represents the transformation from input to en- hancement nodes, is defined as:

*q*1(**x**1) · · · *qL*(**x**1)

 

 

.

.

. .

**H** =  . . .  *.*





*q*1(**x***m*) · · · *qL*(**x***m*)

The regularized optimization problem for RVFL is expressed as:

min **y** − **T*θ *** 2 + *C * ***θ *** 2*,* (17) where **T** = [**H X**], and *C >* 0 is the regularization parameter.

By performing partial differentiation with respect to ***θ***, the solution in the primal space is derived as:

***θ*** = (**T***T* **T** + *C***I**)−1**T***T* **y***,* (18)

where **I** is the identity matrix.

For a new input sample **x**, the RVFL classifier is defined as:

*f* (**x**) = sign([**q**(**x**) **x**]***θ***)*,* (19)

where **q**(**x**) = [*q*1(**x**) *q*2(**x**) · · · *qL*(**x**)]. The classifier *f* (**x**) separates the classes with a hyperplane in the feature space.

The nonlinear extension of this framework follows a similar approach. Unlike conventional neural networks, which learn weights across all layers, the RVFL model only adjusts the weights between the transformed features and the output layer using linear regression. This simplifies the training process and makes the model computationally efficient. Given its ability to handle high-dimensional and nonlinear data, the RVFL model is well-suited for tasks such as EEG signal analysis, where it can identify patterns associated with conditions like schizophrenia. Its simplicity and effectiveness make it a promising tool for improving diagnostic accuracy and facilitating timely medical interventions.

* + 1. **Ensemble Deep Random Vector Functional Link**

Ensemble Deep Random Vector Functional Link (edRVFL) [20] is an advanced ex- tension of the traditional Random Vector Functional Link network, designed for robust and efficient supervised learning. It leverages randomization and deep ar- chitecture while addressing key limitations in scalability and representation power. In edRVFL, the core idea is to stack multiple RVFL layers such that each layer receives not only the input data but also the concatenated outputs of all preced- ing layers. This deep stacking enriches the representational capacity of the model without backpropagation.

Each layer in edRVFL contains randomly initialized weights and biases, which remain fixed during training. Only the output weights are computed analytically using ridge regression, greatly reducing computational overhead. Formally, the pre- diction function of the m-th layer in the edRVFL structure is given by:

**Z***m* = *σ*(**XW***m* + **b***m*) (20)

**H***m* = [**X***,* **Z**1*,* **Z**2*, . . . ,* **Z***m*] (21)

Here, **Z***m* is the non-linear activation of the current layer, **H***m* is the concatenated feature matrix up to layer *m*, and **W***m*, **b***m* are the randomly generated weights and biases, respectively.

***β*** = (**H**⊤ **H***m* + *λ***I**)−1**H**⊤ **Y** (22)

*m m m*

This layer-wise learning process avoids gradient descent and instead computes direct solutions, promoting fast convergence. The ensemble of all layers’ predictions is aggregated (e.g., via averaging or majority voting) to generate the final output.

edRVFL shines in scenarios demanding fast learning and minimal tuning. Its ar- chitecture encourages diversity across layers, enhancing generalization. Moreover, it offers robustness to overfitting, especially in small-data regimes, and can be adapted for both classification and regression tasks.

**Chapter 3**

**Proposed Methodology**

* 1. **Data Collection**

This project focuses on the detection of schizophrenia (SCZ) using EEG signals us- ing data related to RepoD [21]. This provides a verified set of 28 EEG signals from 14 people with delusional schizophrenia and healthy patients. Data were acquired including F7, F4, T3, O2, C4, F3, P3, T5, T4, PZ, CZ, FP2, C3, T6, F8, P4, FP1,

O1, Fz using a typical 10-20 EEG assembly system with 19 electrodes. The signal was recorded at a sampling rate of 250 Hz. This data record aims to identify patterns of EEG signals that distinguish between positive schizophrenia and healthy people by analyzing variations in brain activity on various electrodes. Data were collected under different conditions to provide a deeper understanding of how these signals change with different mental states and how these changes indicate schizophrenia. Preparing data such as filtering, artifact removal, epoching and other data is suitable for cleaning and machine learning of very important signals. As shown in Figure [1,](#_bookmark14) the EEG signals for a healthy patient exhibit more regular patterns, while the un- healthy patient (Figure [2)](#_bookmark15) shows irregularities across several frontal, temporal, and parietal channels.

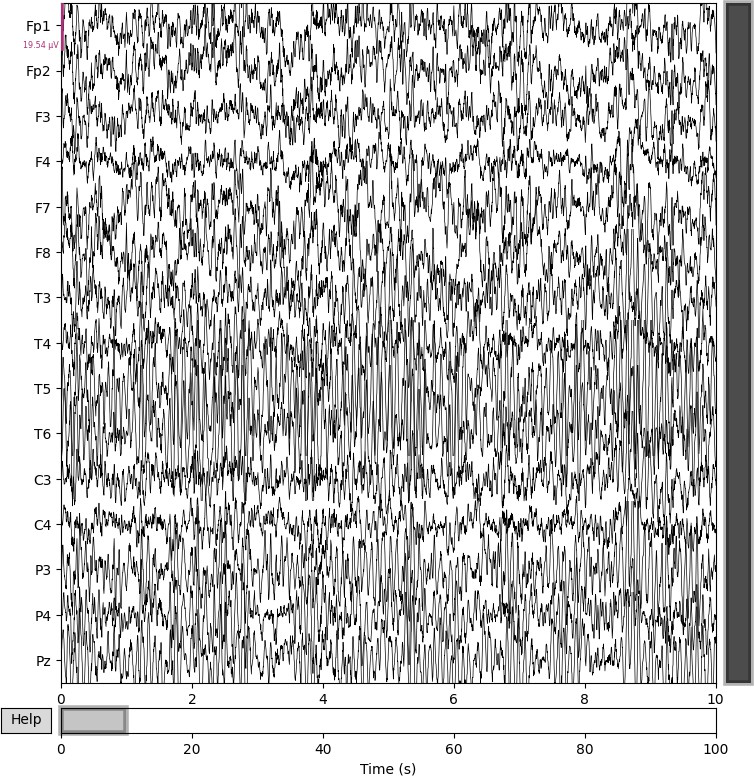


Figure 1: Healthy schizophrenia patient EEG signal on 16 different electrodes

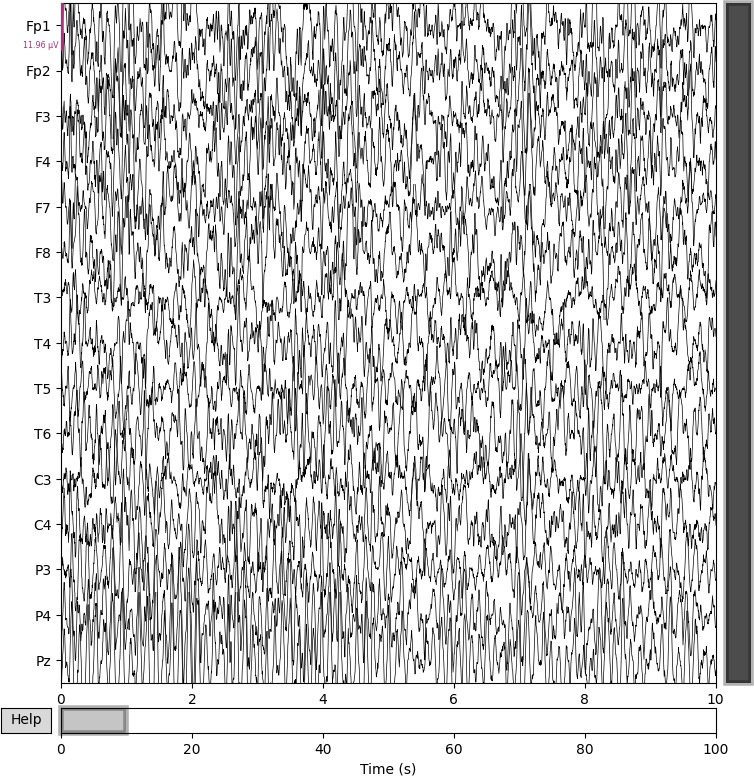


Figure 2: Unhealthy schizophrenia patient EEG signal on 16 different electrodes

* 1. **Working and Explanation**

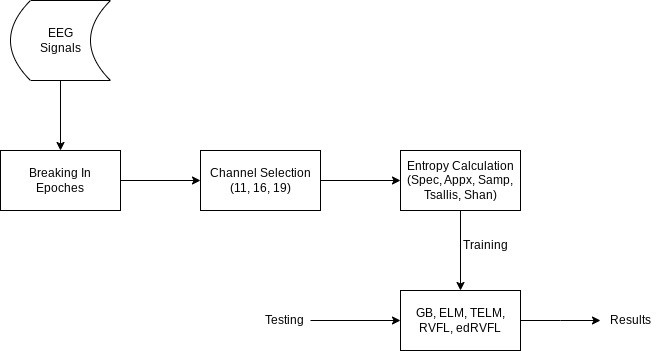
****

Figure 3: Flowchart of implemented models based on entropy The above Figure [3](#_bookmark17) shows the flow of our schizophrenia detection model.

* + 1. **Data Preprocessing:**

While working on .edf files (European data format), first read the EEG signals using a Python library like mne. From these files, we extract important metadata such as EEG signal length, number of channels (electrodes), and sample rate. Analysis shows that EEG signals differ from record to record. To address this, segment the data into ERAS. This makes up each epoch for 2 seconds and contains 500 samples (based on a sampling rate of 250 Hz). After segmenting the complete signal with the epoch, select 50 epochs as test samples from each signal and each electrode for further analysis. The EEG electrodes are shown in specific rags as follows:

* + - * **Frontal Lobe**: Fp1, Fp2, F3, F4, F7, F8, Fz
      * **Temporal Lobe**: T3 (T7), T4 (T8), T5, T6
      * **Parietal Lobe**: P3, P4, P7, P8, Pz
      * **Occipital Lobe**: O1, O2, Oz
      * **Central (Motor Cortex)**: C3, C4, Cz

We create three datasets based on combinations of these lobes to analyze their respective contributions to schizophrenia classification.

* + 1. **Feature Extraction**

After breaking each EEG signal into **50 epochs** of **2 seconds** each, the next step involves the extraction of entropy-based features. For every epoch, we compute five distinct entropy measures for each selected electrode:

* + - * Shannon Entropy
      * Sample Entropy
      * Approximate Entropy
      * Spectral Entropy
      * Tsallis Entropy

This entropy characteristic records various aspects of signal complexity and ran- domness. This is extremely important when proving neurological abnormalities such as schizophrenia.

In our study, special attention is given to the **frontal**, **temporal**, and **parietal lobes**, as these brain regions have shown significant involvement in schizophrenia according to multiple neurophysiological studies.

* + - * The **frontal lobe** is associated with cognitive functions such as decision- making, attention, and executive management that are usually impaired in people with schizophrenia.
      * The **temporal lobe** is essential for auditory processing and language under- standing, and is often associated with symptoms such as auditory hallucina- tions.
      * The **parietal lobe** played a role in sensory integration and spatial orienta- tion, with abnormalities in this area associated with unorganized behavior and perception.

Focusing on these lobes helps in identifying the most discriminative brain regions for schizophrenia diagnosis and enhances the interpretability of the EEG-based clas- sification model.

* + 1. **Feature Selection:**

Before running any model, choosing the right features is one of the most important steps especially when working with EEG signals. These brain signals contain a lot of information, but not all of it is useful for diagnosis. That’s why it’s important to focus on the features that really matter the ones that show noticeable differences between healthy individuals and those affected by schizophrenia. By narrowing it down to the most meaningful data, we can help the model make more accurate and reliable predictions. Below are some of the functions used:

**Spectral Entropy**: This explains the complexity of the signal’s frequency con- tent. We show how energy spreads by calculating the service distribution across frequencies and the use of entropy equations. High spectral entropy indicates com- plex signals with energy distributed over many frequencies, whereas low entropy indicates simpler signals with the most energy in tight frequency bands. The spec- tral entropy *H*spec is given:

*N*

Σ

*H*spec = − *Q*(*fi*) log(*Q*(*fi*)) (23)

*i*=1

where:*Q*(*fi*) is the normalized PSD at the frequency *fi*, The number of frequency bins in the spectrum is represented by *N* .

**Approximate Entropy**: Approximate entropy is a useful instrument for as- sessing the regularity or complexity of a time series containing EEG signals. By calculating the APEN, you can quantify the extent to which predictable or irregular brain activity is likely to help you recognize abnormalities such as schizophrenia.

Higher Apen values indicate more irregular brain activity, which may be a charac- teristic of schizophrenia, whereas lower values indicate more normal or normal brain function.

ApEn(*p, s, T* ) =

1

*T* − *p*

*T* −*p*

ln

Σ

*j*=1

*Dj*(*p, s*)

*Dj*(*p* + 1*, s*)

(24)

where *p* is the embedding dimension, *s* is the tolerance, *T* is the length of the time series. *Dj*(*p, s*) counts the number of vector pairs of length *p* within tolerance *s* at the *j*-th position, while *Dj*(*p* + 1*, s*) counts similar vector pairs of length *p* + 1. These quantities are used to calculate the logarithmic ratio that quantifies the system’s complexity.

**Sample Entropy**: Metrics called rehearsal tropics are used to quantify how complex or unpredictable time series such as EEG signals are. It is particularly useful for the analysis of physiological signals such as EEG. This makes the goal often constitutes by assessing irregularities or variations in signals that may indi- cate specific conditions, such as schizophrenia. Abtsch entropy is closely related to approximate entropy (Apen). It provides a more consistent and reliable estimate, especially for short time series. By calculating the likelihood that an equivalent pattern will be discovered within a particular time frame, SAMP measures the pre- dictability or degree of random numbers in a time series, and thus quantifies its regularity. The definition of a sampling ropy (SE) is as follows:

SE(*k, ϵ, L*) = − ln  *C*(*k, ϵ*) (25)

*C*(*k* + 1*, ϵ*)

where *L* is the number of data points in the time series, *ϵ* is the tolerance, and *k* is the embedding dimension (usually 2 or 3). While *C*(*k* + 1*, ϵ*) counts comparable vector pairs of length *k* + 1 within the same tolerance, *C*(*k, ϵ*) indicates the number of similar vector pairs of length *k* within tolerance *ϵ*.

**Tsallis Entropy**: System complexity or destruction can be measured using Tsallis entropy, which represents the generalization of Shannon entropy. It was founded in 1988 by physicist Constantino Zaris as part of a study on unoptimized statistical mechanics that characterize systems with fractal structures, long-term

memory, and long-term interactions. Tsallis entropy can be applied to a variety of fields such as information theory, statistical mechanics, and neuroscience. Even when nonlinear and unbalanced systems are used, this is especially useful: B. B. EEG signals in brain activity analysis.

The order’s Tsallis entropy *α* for a distribution of probabilities {*rj*} is given by:

*Tα* =

1

*α* − 1

*M*

1 − *α*

*r*

Σ

!

*j*

*j*=1

(26)

The Tsallis entropy of order *α*, denoted as *Tα*, is defined in terms of the proba- bility *rj* of the *j*-th event, where *j* ranges over all possible events or states. The total number of possible events or states is represented by *M* , and the order parameter *α*, which satisfies *α* ≥ 1, has a significant impact on how the entropy is formed.

**Shannon Entropy**: This measures uncertainty or randomness within the signal.

Calculate the benefits of each symbol in the signal, taking into account the possibility that each symbol (voltage level, intensity value) will occur. Higher entropy indicates a more unpredictable signal with no larger range of values and no dominant symbols. A low entropy indicates a predictable signal with few clear values.

For a discrete random variable *Y* with probability distribution {*q*1*, q*2*, . . . , qM* },

the Shannon entropy *G*(*Y* ) is defined as follows:

*M*

Σ

*G*(*Y* ) = − *qj* log(*qj*) (27)

*j*=1

where *M* is the number of alternative occurrences or states, and *qj* is the prob- ability of the *j*-th event or result.

* + 1. **Feature Vector Preparation**

We have already extracted entropy-based features from different EEG signal seg- ments. Before training the classification models, it is essential to organize these features into a consistent format. Our approach involves training on data from **28 subjects**, with **50 randomly selected epochs** from each EEG signal. For each epoch, entropy features are computed from selected electrodes. We prepare **three**

**distinct datasets** based on different combinations of EEG electrodes corresponding to specific brain lobes:

* + - * **Dataset 1 (All 19 Channels)**: Entropy features are extracted from all 19 EEG electrodes. For each epoch, 5 entropy values are computed per electrode, resulting in a feature vector of length 95 (19 × 5). Thus, the final dataset shape becomes:

28 × 50 = **1400** samples*,* **95 features** ⇒ **1400** × **95**

* + - * **Dataset 2 (Frontal, Temporal, Parietal – 16 Channels)**: Feature ex- traction is limited to 16 electrodes associated with the frontal, temporal, and parietal lobes. Each epoch yields a feature vector of length 80 (16 × 5), pro- ducing a dataset of shape:

**1400** × **80**

* + - * **Dataset 3 (Frontal and Temporal – 11 Channels)**: Only 11 electrodes from the frontal and temporal lobes are considered. Each epoch yields a feature vector of length 55 (11 × 5), resulting in:

**1400** × **55**

This separation into three datasets allows us to study the discriminative power of different brain regions in schizophrenia detection. Each dataset is used inde- pendently to train and evaluate machine learning models, enabling a comparative analysis across brain lobes.

* + 1. **Model Training:**

We test our model’s accuracy using five distinct models: Gradient Boost (GB), Ex- treme Learning Machine (ELM), Twin Extreme Learning Machine (TELM), Ran- dom Vector Functional Link (RVFL) and Ensemble Deep Random Vector Functional Link (edRVFL). With the training of these model, we also need to tune our models with different situations for best result according to our need.

## Result Analysis and Comparison:

Once results are obtained from different machine learning models applied to the schizophrenia dataset, a detailed comparison is essential to identify which model performs best for diagnosis. This involves evaluating each model using key perfor- mance metrics such as accuracy, precision, recall, and F1-score. In medical diagnos- tics, sensitivity and specificity are especially important, as they reflect how well the model can correctly identify both healthy individuals and those with schizophrenia. It’s also important to consider how efficiently each model runs—factors like training time, prediction speed, and resource usage matter, especially if the model is to be used in real clinical settings. To confirm that the differences in performance aren’t just due to chance, statistical tests can be used to validate the results. In the end, the goal is to find a model that not only delivers accurate and reliable predictions but is also practical enough to be used in real-world diagnosis.

**Chapter 4**

**Experimental Setup and Results Analysis**

# Experimental setup

For experiment setup, we have used dataset from RepOD in the form of edf files. As represented in figure 4 [4](#_bookmark26)



Figure 4: Experimental setup

# System Requirements

* + - 8GB RAM
    - GPU t4x2
    - Corei5

# Technology Used

Several technologies were used in the project to detect schizophrenia, including:

1. **NumPy:** With the support of large multidimensional arrays and matrices, as well as various mathematical functions, numpy is being created for numerical calculations.
2. **Python:** Python, a programming language celebrated for simplicity and read- ability, was introduced in Rossum in 1991. Rossum quickly received its ver- satility and intuitive understanding of its syntax. Its feature is readability, resembling pseudocode, allowing for quick understanding and reduced error rates. Python is an interpreted language, which dynamically type code design lines through lines by interpreters that allow for flexible and concise code-Sans to explicit type declarations.
3. **Pandas:** It provides high-level data structures and tools for data analysis and manipulation, especially when working with tabular data such as CSV files. Provides dataframe objects for efficient organization and manipulation of data.
4. **MNE:** MNE offers a wide range of methods for extracting functions from EEG data related to mechanical learning tasks. MNE includes powerful visu- alization tools for the study and visualization of neuronal data, including: B. Interactive plot function for EEG data.
5. **Pyedflib:** This is a Python library developed to read and write European Data Formats (EDF) and its derivative files. EDF is a standard file format commonly used to store biological signals, particularly in the field of electro- physiology, such as EEG data.
6. **Scikit-learn:** Scikit-Learn is a robust library for machine learning that pro- vides effective and easy-to-use data mining and analytics capabilities. It offers a variety of techniques for issues such as classification, regression, clustering, model selection, and dimension reduction.
7. **Tensorflow:** Google creates open source Tensorflow frames for machine learn- ing targeted at deep learning applications, and is used to create and train machine learning models.
8. **SciPy:** The open source Python library Scipy, short for ”Scientific Python,” expands Numpy and offers a variety of scientific computer tools and numerical methods. It is widely used in fields such as statistics, mathematics, engineer- ing, physics and more.

# Performance metrics

After training is done, we are evaluating our model on their accuracy to predict the labels.

**Accuracy**: In machine learning, accuracy serves as a parameter that evaluates how a model can effectively identify a class or result of a particular input. The percentage of cases correctly predicted in all examples in this data record is the measurement method. However, this is not always the best metric, especially when working with unbalanced datasets. It is usually advantageous to evaluate the performance of some models and use accuracy along with other metrics to select the best model for your particular task.

ACC =

TP + TN

TP + TN + FN + FP

(28)

**Precision:** It is the proportion of actual positive predictions among all of the model’s positive predictions. This metric evaluates the correctness of positive pre- dictions, showing how many of the instances identified as positive are truly relevant.

TP

Precision =

TP + FP

(29)

**Recall:** What the model perceived was the percentage of actual positive cases. Measures the ability of a model to correctly identify all related instances from a data record.

Recall =

TP

TP + FN

(30)

**F1 Score:** Both consider how well the model accurately identifies positive cases (precision) and records all positive cases in the data record (recall). A higher F1 point count provides a thorough assessment of the overall effectiveness of the model, indicating that it works well in terms of accuracy and recall.

The following formula can be used to determine the F1 score:

*F* 1 = 2 ×

Precision × Recall Precision + Recall

(31)

# Result Analysis

**Gradient Boosting:** Table [1](#_bookmark31) presents the performance of a Gradient Boosting model across different k-fold cross-validation settings (5 to 10 folds) and input chan- nel numbers (11, 16, 19). The results, likely representing accuracy, show variations based on these configurations. The highest performance (96.84%) was observed with 10-fold cross-validation using 16 channels, suggesting this combination yielded the best model performance in this evaluation.

Table 1: Result summary of Gradient Boosting

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N | k-Fold | 11 Channels | 16 Channels | 19 Channels |
| 1 | 5 fold | 92.43% | 95.29% | 95.86% |
| 2 | 6 fold | 92.00% | 95.43% | 96.07% |
| 3 | 7 fold | 92.57% | 95.64% | 95.71% |
| 4 | 8 fold | 93.62% | 96.54% | 95.71% |
| 5 | 9 fold | 93.33% | 96.54% | 96.07% |
| 6 | 10 fold | 93.41% | **96.84%** | 96.11% |

The performance of the Gradient Boosting model is further illustrated by the ROC curve shown in Figure [5,](#_bookmark32) which demonstrates its strong discriminatory ability. **Extreme Learning Machine:** Table [2](#_bookmark33) presents the performance results of an Extreme Learning Machine model across different configurations. The table shows the outcomes, likely accuracy, obtained by varying the k-fold cross-validation (from

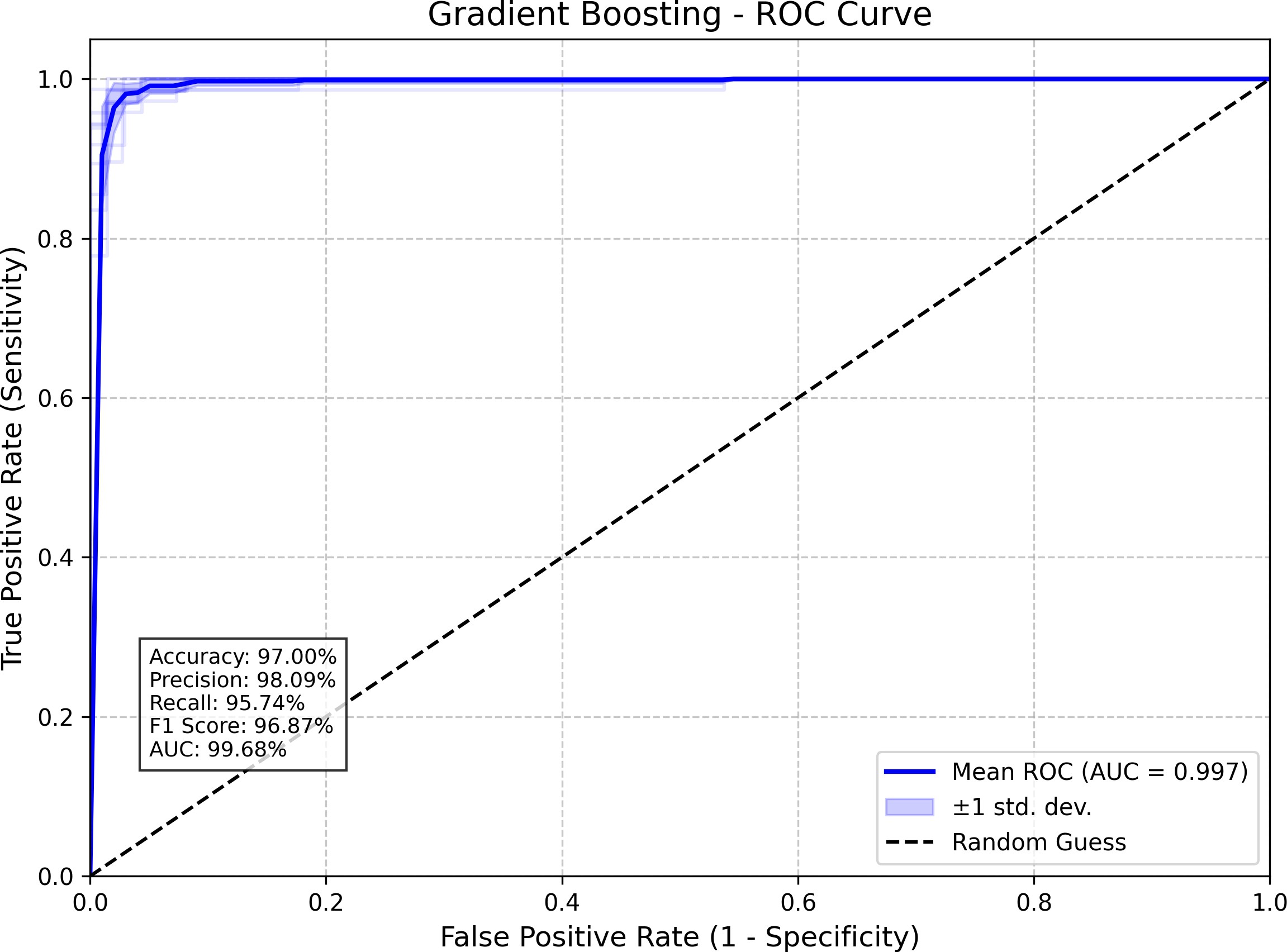


Figure 5: Gradient Boosting ROC Curve

5 to 10 folds) and the number of input channels (11, 16, and 19). These results illustrate how different experimental setups affect the model’s performance, with the highest performance of 92.43% achieved using 10-fold cross-validation with 16 channels.

Table 2: Result summary of Extreme Learning Machine

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N | k-Fold | 11 Channels | 16 Channels | 19 Channels |
| 1 | 5 fold | 89.21% | 91.71% | 90.64% |
| 2 | 6 fold | 89.79% | 91.50% | 91.71% |
| 3 | 7 fold | 89.86% | 92.07% | 91.36% |
| 4 | 8 fold | 88.79% | 91.93% | 91.36% |
| 5 | 9 fold | 88.79% | 92.07% | 91.87% |
| 6 | 10 fold | 90.00% | **92.43%** | 92.33% |

The performance of the ELM model is further illustrated by the ROC curve shown in Figure [6,](#_bookmark34) which demonstrates its strong discriminatory ability.

**Twin Extreme Learning Machine:** Table [3](#_bookmark35) summarizes the performance of a Twin Extreme Learning Machine model under various conditions. It displays

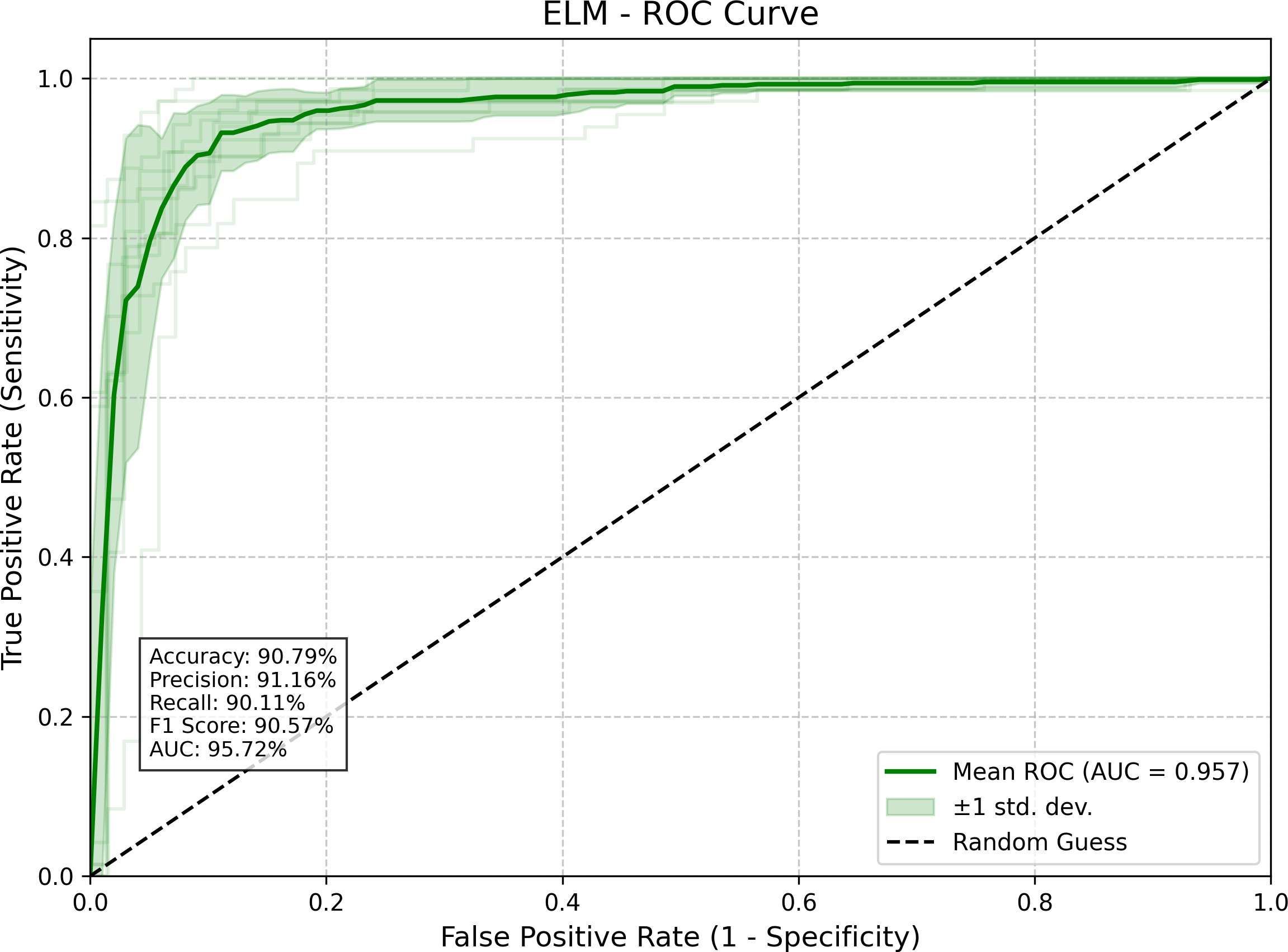


Figure 6: ELM ROC curve

the results, likely accuracy, achieved with different k-fold cross-validation settings (5 to 10 folds) and input channel numbers (11, 16, and 19). The table highlights how these experimental parameters influence the model’s performance, with the peak performance of 94.86% observed when using 10-fold cross-validation with 19 channels.

Table 3: Result summary of Twin Extreme Learning Machine

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N | k-Fold | 11 Channels | 16 Channels | 19 Channels |
| 1 | 5 fold | 90.79% | 93.14% | 94.43% |
| 2 | 6 fold | 91.14% | 93.43% | 94.46% |
| 3 | 7 fold | 91.15% | 93.50% | 94.46% |
| 4 | 8 fold | 91.71% | 93.51% | 94.57% |
| 5 | 9 fold | 91.71% | 93.86% | 94.64% |
| 6 | 10 fold | 91.71% | 94.50% | **94.86%** |

The performance of the TELM model is further illustrated by the ROC curve shown in Figure [7,](#_bookmark36) which demonstrates its strong discriminatory ability.

**Random Vector Functional Link:** Table [4](#_bookmark37) summarizes the performance of

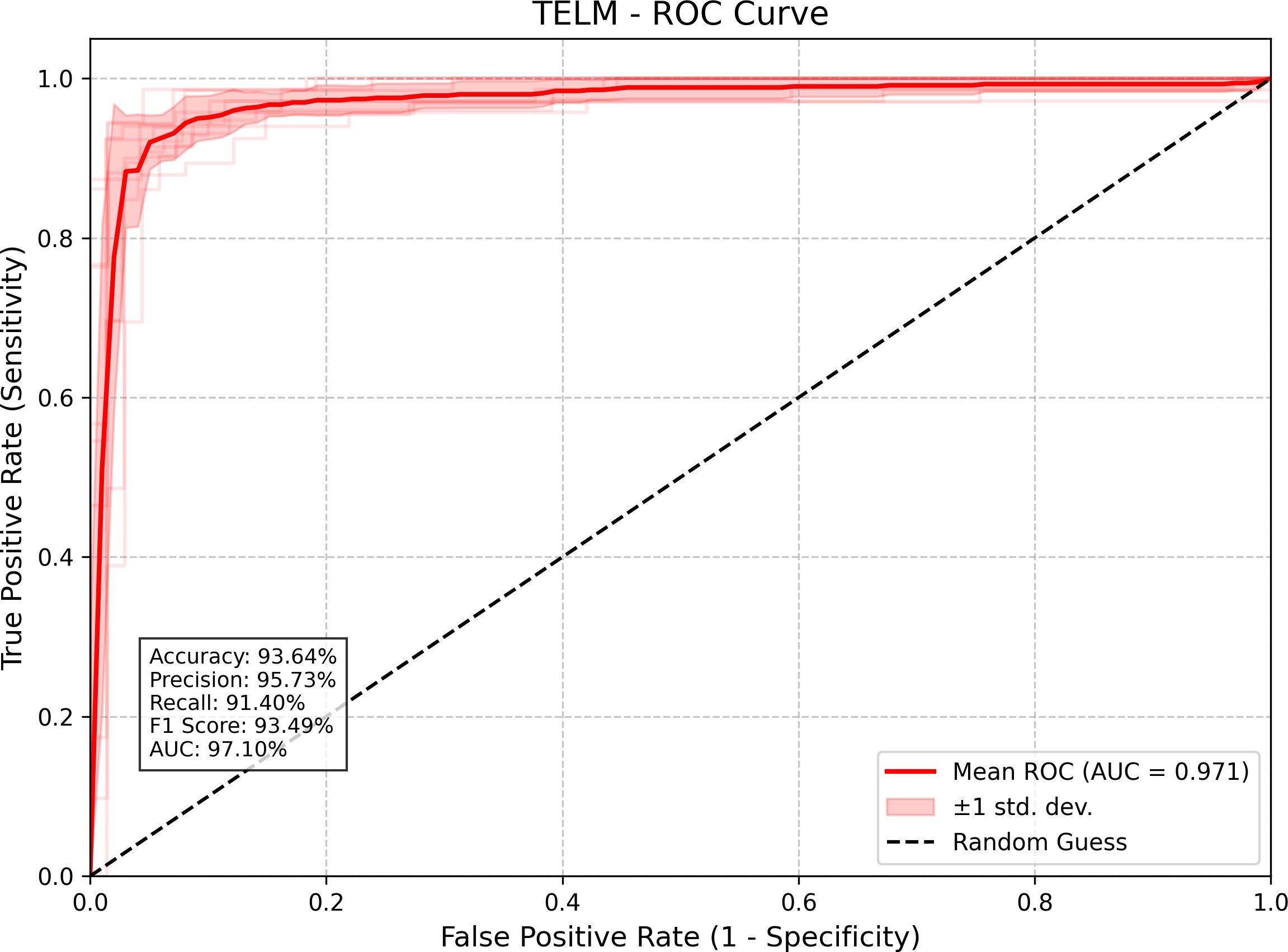


Figure 7: TELM ROC curve

a Random Vector Functional Link network under different experimental conditions. It displays the results, likely accuracy, achieved with varying k-fold cross-validation (5 to 10 folds) and input channel numbers (11, 16, and 19). The table illustrates how these parameters impact the network’s performance, with the highest result of 94.89% observed when using 10-fold cross-validation with 19 channels.

Table 4: Result summary of Random Vector Functional Link

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N | k-Fold | 11 Channels | 16 Channels | 19 Channels |
| 1 | 5 fold | 90.79% | 93.14% | 94.43% |
| 2 | 6 fold | 91.14% | 93.43% | 94.36% |
| 3 | 7 fold | 91.15% | 93.50% | 94.07% |
| 4 | 8 fold | 91.71% | 93.51% | 94.57% |
| 5 | 9 fold | 91.71% | 93.86% | 94.64% |
| 6 | 10 fold | 91.71% | 94.50% | **94.89%** |

The performance of the RVFL model is further illustrated by the ROC curve shown in Figure [8,](#_bookmark38) which demonstrates its strong discriminatory ability.

**Ensemble Deep Random Vector Functional Link:** Table [5](#_bookmark40) presents the

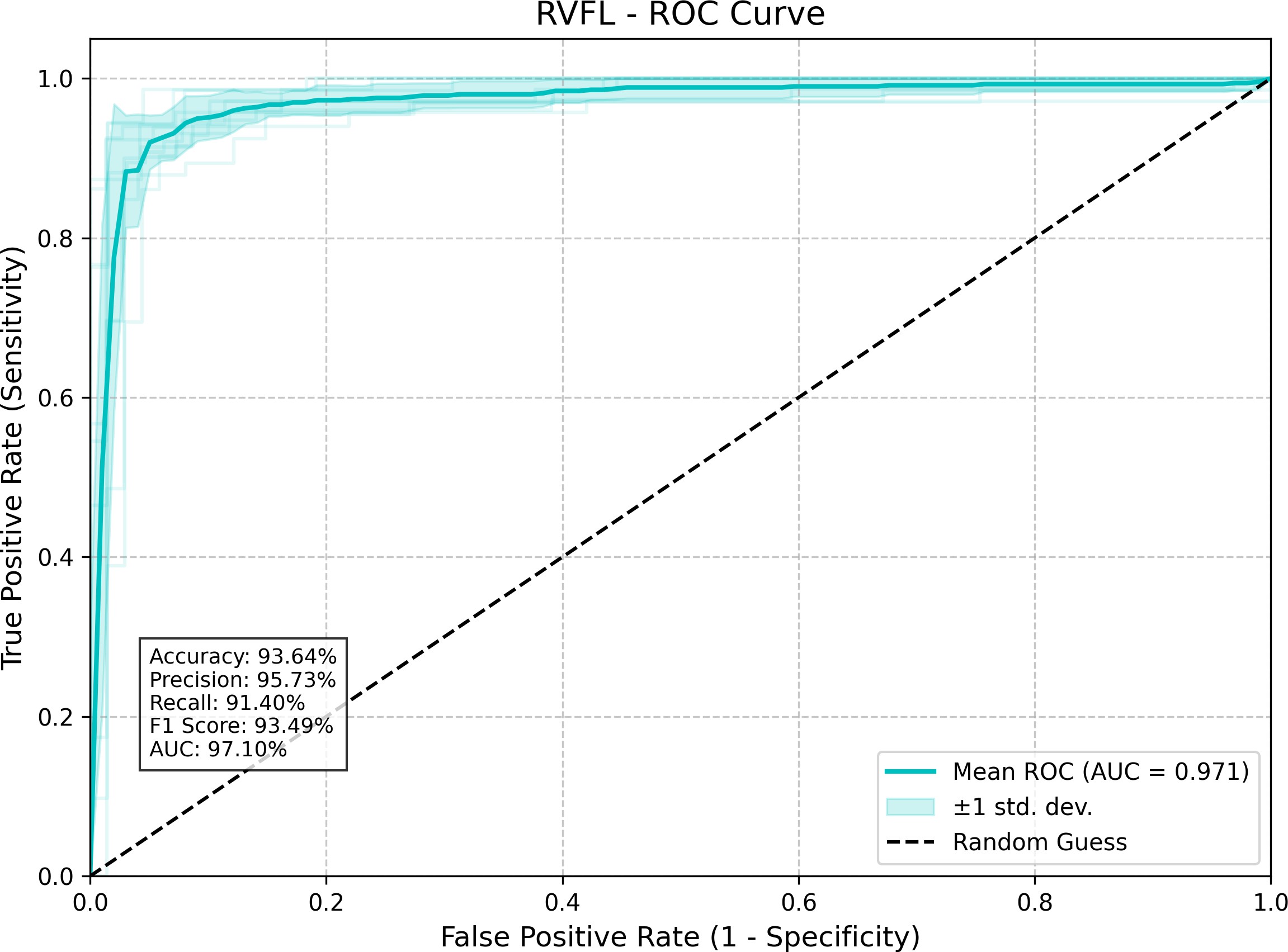


Figure 8: RVFL ROC curve

performance summary of an Enhanced Deep Random Vector Functional Link net- work across different configurations. The table shows the results, likely accuracy, obtained by testing various k-fold cross-validation settings (from 5 to 10 folds) and different numbers of input channels (11, 16, and 19). These results demonstrate how these experimental parameters influence the network’s performance, with the highest performance of 94.86% achieved using 10-fold cross-validation with 19 channels.

The performance of the edRVFL model is further illustrated by the ROC curve shown in Figure [9,](#_bookmark39) which demonstrates its strong discriminatory ability.

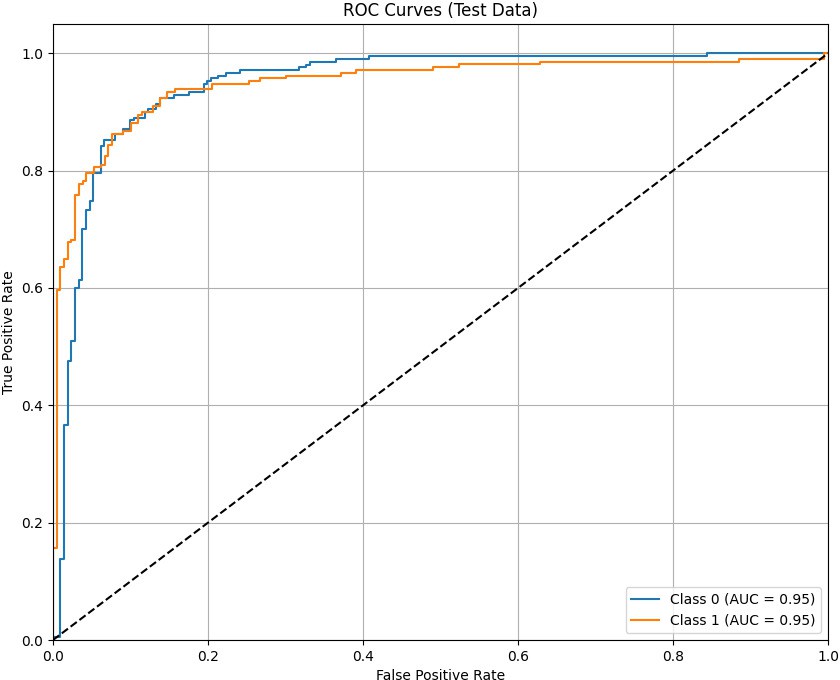


Figure 9: edRVFL ROC curve

Table 5: Result summary of Enhanced Deep Random Vector Functional Link

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | 11 Channels | 16 Channels | 19 Channels |
| 1 | 93.82% | **97.38%** | 96.67% |
| 2 | 93.82% | 97.38% | 96.67% |
| 3 | 93.82% | 97.38% | 96.67% |
| 4 | 93.82% | 97.38% | 96.67% |
| 5 | 92.39% | 97.15% | 96.44% |
| 6 | 92.39% | 97.15% | 96.44% |
| 7 | 92.39% | 97.15% | 96.44% |
| 8 | 92.39% | 97.15% | 96.44% |
| 9 | 92.87% | 97.15% | 96.44% |
| 10 | 92.87% | 97.15% | 96.44% |

# All models Result Comparions

The performance comparison across five models using three different channel sets highlights variations in accuracy and generalization, indicating model-specific strengths depending on the channel configuration.

Table 6: Model Performance Metrics using 90% train and 10% test across 11 EEG Channels

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Serial No.** | **Model Name** | **Accuracy** | **Precision Score** | **Recall** | **F1 Score** |
| 1 | GB | 93.71% | 0.95 | 0.92 | 0.93 |
| 2 | ELM | 87.71% | 0.90 | 0.86 | 0.87 |
| 3 | TELM | 88.86% | 0.92 | 0.86 | 0.88 |
| 4 | RVFL | 88.86% | 0.92 | 0.86 | 0.88 |
| 5 | EDRVFL | 93.82% | 0.94 | 0.94 | 0.94 |

As shown in Table [6,](#_bookmark42) the Ensemble Deep RVFL (EDRVFL) model achieved the highest performance among all tested models using 11 EEG channels, with an accuracy of 93.82%, precision of 0.94, recall of 0.94, and F1 score of 0.94.

Table 7: Model Performance Metrics using 90% train and 10% test across 16 EEG Channels

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Serial No.** | **Model Name** | **Accuracy** | **Precision Score** | **Recall** | **F1 Score** |
| 1 | GB | 96.79% | 0.97 | 0.96 | 0.97 |
| 2 | ELM | 90.5% | 0.92 | 0.89 | 0.90 |
| 3 | TELM | 92.79% | 0.94 | 0.91 | 0.93 |
| 4 | RVFL | 92.79% | 0.94 | 0.91 | 0.93 |
| 5 | EDRVFL | 97.39% | 0.97 | 0.97 | 0.97 |

As shown in Table [7,](#_bookmark43) the Ensemble Deep RVFL (EDRVFL) model achieved the highest performance among all tested models using 11 EEG channels, with an accuracy of 97.39%, precision of 0.97, recall of 0.97, and F1 score of 0.97.

Table 8: Model Performance Metrics using 90% train and 10% test across 19 EEG Channels

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Serial No.** | **Model Name** | **Accuracy** | **Precision Score** | **Recall** | **F1 Score** |
| 1 | GB | 97% | 0.98 | 0.96 | 0.97 |
| 2 | ELM | 90.79% | 0.91 | 0.90 | 0.91 |
| 3 | TELM | 93.64% | 0.96 | 0.91 | 0.93 |
| 4 | RVFL | 93.64% | 0.96 | 0.91 | 0.93 |
| 5 | EDRVFL | 96.67% | 0.97 | 0.97 | 0.97 |

As shown in Table [8,](#_bookmark44) the Gradient Boosting (GB) model achieved the highest performance among all tested models using 11 EEG channels, with an accuracy of 97%, precision of 0.98, recall of 0.96, and F1 score of 0.97.

**Chapter 5**

**Conclusions and Future scope**

Based on the comparative analysis across the three electrode configurations pre- sented in the tables (11, 16, and 19 channels), several key conclusions emerge re- garding machine learning-based schizophrenia detection using EEG. Firstly, there is a notable overall improvement in detection accuracy as the number of channels increases from 11 (Frontal and Temporal) to 16 (adding Parietal), highlighting the valuable contribution of parietal lobe activity. While the performance gain from 16 to 19 channels (adding Occipital) is less dramatic or shifts slightly in ranking, it indicates that the optimal set of discriminative features is either predominantly captured by 16 channels or distributed across all 19. Consistently across all three configurations, Gradient boosting and edRVFL proved to be the most effective meth- ods, regularly achieving accuracies above 93%, and peaking at 97.39% (edRVFL, 16 channels) and 97% (Gradient boosting, 19 channels). Other methods like ELM, TELM, and RVFL showed consistently lower, albeit still reasonable, performance. These results underscore the potential of Gradient boosting and edRVFL algorithms and emphasize that electrode selection, particularly the inclusion of parietal chan- nels, is crucial for achieving high accuracy in the early detection of schizophrenia from EEG signals.

**References**

1. Melissa Hwang, Youkyung Roh, Jessica Talero, Bruce Cohen, Justin Baker , ”Auditory hallucinations across the psychosis spectrum: Evidence of dyscon- nectivity involving cerebellar and temporal lobe regions”.
2. Palani Thanaraj Krishnan and Alex Noel Joseph Raj and Parvathavarthini Balasubramanian and Yuanzhu Chen *“Schizophrenia detection using Multi- variateEmpirical Mode Decomposition and entropy measures from multichan- nel EEG signal”* Biocybernetics and Biomedical Engineering,Volume 40, Issue 3,2020, Pages 1124-1139,ISSN 0208-5216
3. Nash N. Boutros, Cynthia Arfken, Silvana Galderisi, Joshua Warrick , Garrett Pratt and William Iacono, ”The status of spectral EEG abnormality as a di- agnostic test for schizophrenia”. https://doi.org/10.1016/j.schres.2007.11.020
4. B. Tasci, G. Tasci, S. Dogan, and T. Tuncer, “A novel ternary pattern-based automatic psychiatric disorders classification using ECG signals,” *Cogn. Neu- rodyn.*, vol. 8, 2022, doi: 10.1007/s11571-022-09918-8.
5. S. Verma, T. Goel, M. Tanveer, W. Ding, R. Sharma, and R. Murugan, “ML techniques for the SCZ diagnosis: a review and future research directions and diagnosis,” *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 5, pp. 4795–4807, 2023, doi: 10.1007/s12652-023-04536-6.
6. M. Jafari, D. Sadeghi, A. Shoeibi, H. Alinejad-rokny, and A. Beheshti, “AI- Driven SCZ Diagnosis with the help of EEG Signals: A Review from 2002- 2023,” 2023.
7. M. Tanveer, J. Jangir, M. A. Ganaie, I. Beheshti, M. Tabish, and N. Chhabra, “Detection of SCZ: An Evaluation,” *IEEE J. Biomed. Heal. Informatics*, vol. 27, no. 3, pp. 1185–1192, 2023, doi: 10.1109/JBHI.2022.3168357.
8. Oh SL, Vicnesh J, Ciaccio EJ, Yuvaraj R, Acharya UR. Deep CNN model for automated dectection of schizophrenia using EEG signals. *Appl Sci* 2019;9(14)

:2870. <http://dx.doi.org/10.3390/app9142870>

1. Kutepov IE, Dobriyan VV, Zhigalov MV, Stepanov MF, Krysko AV, Yakovleva TV, et al. EEG analysis in patients with schizophrenia based on Lyapunov exponents. *Inform Med Unlocked* 2020;18:100289. <http://dx.doi.org/10.1016>

/j.imu.2020.100289

1. Harmah DJ, Li C, Li F, Liao Y, Wang J, Ayedh WMA, et al. Measuring the non-linear directed information flow in schizophrenia by multivariate transfer entropy. *Front Comput Neurosci*

2020;13:85. <http://dx.doi.org/10.3389/fncom.2019.00085>

1. Zahra A, Kanwal N, ur Rehman N, Ehsan S, McDonald-Maier KD. Seizure detection from EEG signals using multivariate empirical mode decomposition. *Comput Biol Med* 2017;88:132–41. [http://dx.doi.org/10.1016/j.compbiomed.](http://dx.doi.org/10.1016/j.compbiomed) 2017.07.010
2. Kannathal N, Choo ML, Acharya UR, Sadasivan P. Entropies for detection of epilepsy in EEG. *Comput Methods ProgrBiomed* 2005;80(3):187–94. [http://dx.doi.org/10.1016/j.cmpb.2005.06.012.](http://dx.doi.org/10.1016/j.cmpb.2005.06.012)
3. Aba´solo D, Hornero R, Espino P, A´lvarez D, Poza J. different Entropy analysis of the EEG activity in SCZ disease patients. *Physiol Meas* 2006;27(3):241–53. [http://dx.doi.org/10.1088/0967-3334/27/3/003.](http://dx.doi.org/10.1088/0967-3334/27/3/003)
4. Liang Z, Wang Y, Sun X, Li D, Voss LJ, Sleigh JW, Hagihira S, Li X. EEG en- tropy measures in anesthesia. *Front Comput Neurosci 9.* doi:10.3389/fncom.2015. 00016.
5. M, Rehman B, Rizvi A, et al. Schizophrenia. [Updated 2024 Feb 23]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2024 Jan-. Available from: h[ttps://www.ncbi.nlm.nih.gov/books/NBK539864/](http://www.ncbi.nlm.nih.gov/books/NBK539864/)
6. A. B. Desai, D. R. Gangodkar, K. Pant and B. Pant, ”Harnessing the Potential of Light Gradient Boosting Machine for Accurate Diagnosis of Schizophrenia from EEG Signals,” 2024 14th International Conference on Cloud Computing,

Data Science Engineering (Confluence), Noida, India, 2024, pp. 568-574, doi: 10.1109/Confluence60223.2024.10463450.

1. Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, Extreme learning ma- chine: Theory and applications, Neurocomputing, Volume 70, Issues 1–3, 2006, Pages 489-501, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2005.12.126
2. Yihe Wan, Shiji Song, Gao Huang, Shuang Li, Twin extreme learning machines for pattern classification, Neurocomputing, Volume 260, 2017, Pages 235-244,

ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2017.04.036.

1. S.A. Varaprasad, Tripti Goel, M. Tanveer, R. Murugan, An effective diagnosis of schizophrenia using kernel ridge regression-based optimized RVFL classi- fier, Applied Soft Computing, Volume 157, 2024, 111457, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2024.111457.
2. M. Hu, J. Herng Chion, P. N. Suganthan and R. K. Katuwal, ”Ensemble Deep Random Vector Functional Link Neural Network for Regression,” in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 53, no. 5, pp. 2604-2615, May 2023, doi: 10.1109/TSMC.2022.3213628.
3. Olejarczyk E, Jernajczyk W (2017) Graph-based analysis of brain connectivity in schizophrenia. PLoS ONE 12(11): e0188629. https://doi.org/10.1371/ journal.pone.0188629