Classification Model for Parkinson's disease using EEG Signals

Name : Shatarupa Bepari,
Roll Number : 22210042

Indian Institute of Information Technology Gandhinagar
beparishatarupa@iitgn.ac.in

1 Introduction

Parkinson's disease (PD) is a progressive disorder that affects the nervous system and the parts of the body controlled by the nerves and the symptoms start very slowly. The first symptoms may be barely noticeable tremor in just one hand. Early detection of (PD) is very important in clinical diagnosis for preventing disease development in future. The paper presents efficient discrete wavelet transform (DWT)-based methods for detecting PD from health control (HC) in two cases, namely, off- and on-medication. The EEG signals are pre-processed to remove major artifacts before being decomposed into several EEG sub-bands (approximate and details) using discrete wavelet transform. The features are then extracted from the wavelet packet-derived reconstructed signals using different entropy measures, namely, log energy entropy, Shannon entropy, threshold entropy, sure entropy, and norm entropy. Then KNN machine learning techniques are investigated to classify the resulting PD/HC features.The aim of the present study is to present uncomplicated feature extraction and classification methods while maintaining high classification accuracy and validating them using two open-source datasets (the UNM and SanDiego datasets).With the SanDiego dataset, the classification results of off-medication PD versus HC are 99.89, 99.87, and 99.91 for accuracy, sensitivity, and specificity, respectively, using the combination of DWT+TShEn and KNN classifier. Using the same combination, the results of on-medication PD versus HC are 94.21, 93.33, and 95 percentage respectively.

2 Methods

2.1 Data Description

In this study we intend to use an open-source EEG dataset is used to test the proposed approaches. The University of San Diego in California provided the dataset, for simplicity, we will refer this dataset as SanDiego Datasets. The data has been recorded from two groups of individuals. The first group contains EEGs of 16 healthy individuals, while the second group contains EEGs of 15 PD patients, taken both on and off medication. The right-handedness, gender, age, and cognition of the PD patients were remarkably similar to those of the HC, as evaluated by the Mini-Mental State Exam (MMSE) and the North American Adult Reading Test (NAART). The healthy subjects only volunteered once. At a sampling frequency of 512

Hz, EEG data was captured for at least 3 min in a 32-channel Biosemi active EEG system. More details can be fond on Openneuo [4].

2.2 Data Pre-processing

The EEG signals are split into segments with a size of 'ch×T', where 'ch' is the number of channels and 'T' is the segment length in seconds. The choice of the segmentation time interval 'T' will be selected based on the length of recording of each dataset.

A detailed pre-processing steps have been discussed in the below points:

1. The EEG raw data is loaded and filtered to remove unwanted frequencies i.e., low frequency is set to 0.5 Hz and high frequency to 50 HZ. This is done using the mne.filter() function in MNE-Python. And notch filtration has also been done.

2. Then the EEG data is referenced to a common reference electrode or re-referenced to an average. This is done using mne.set-eeg-reference() function.

3. Epochs are created from the raw EEG data using a fixed-length sliding window approach of duration 1.0 seconds across the entire duration of the raw data. Overlapped data issues are also removed.

4.Artifacts like eye blinks and movements, muscular activity, electrical noise, and other noise artefacts are also removed using Independent Component Analysis or ICA, mne.preprocessing.ICA, in this case 20 components were used.

5. The filtered data is then concatenated and has been converted to a pandas dataframe to our further with our future steps.

Please refer to the code snippet below. It shows a python code that is implemented to pre-process the EEG data.

```python
%%capture
raw_list = []
for raw_data in raw_data:
    raw = mne.io.read_raw_bdf(raw_data, preload=True)
    raw.set_eeg_reference()
    raw.filter(l_freq=0.5, h_freq =50)
    events = mne.make_fixed_length_events(raw, duration=1.0)
    epochs = mne.Epochs(raw, events=events, tmin=0.0, tmax=1.0, baseline=None)
    ica = mne.preprocessing.ICA(n_components=20, random_state=42)
    ica.fit(raw)
    ica.apply(raw)
    raw_list.append(raw)

# concatenate the raw data from all files
raw_all = mne.concatenate_raws(raw_list)
```

Fig. 1. Code:Data Pre-Processing.

As we can see the raw EEG data before data pre-processing from (see Fig. 2) the figure it is very concentrated and overlapping at many bands. So we continued to remove the interference and noise caused by the electrodes and magnetic fields.

(Fig. 3) shows the pre-processed EEG data. It is then converted to a pandas data frame. Please refer to the GitHub repository [5] to get an incite of the entire process in details.

2.3 Wavelet decomposition/reconstruction

The discrete wavelet transform (DWT) has the ability to analyze the features of a signal in the time and frequency domains by decomposing it into a number of mutually orthogonal components using a single function called the mother wavelet41. The particular choice of mother function is crucial for

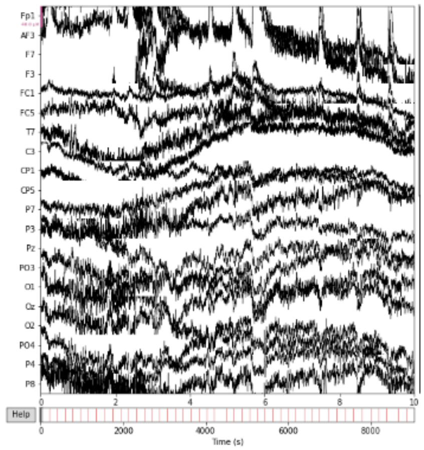

Fig. 2. EEG Data Before Pre-Processing.

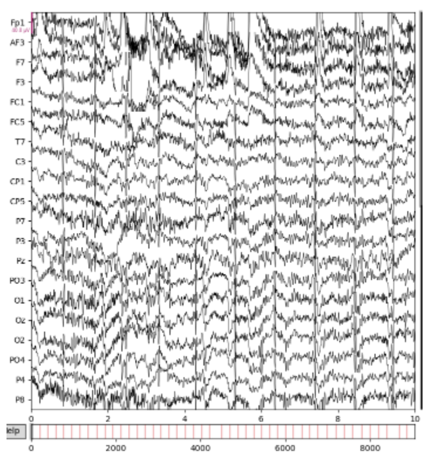

Fig. 3. EEG Data After Pre-Processing.

signal analysis. Low pass and high pass filters are used in first level decomposition to produce the signal's representation as approximation (A1) and detail (D1) coefficients. The coefficients with higher frequencies have good time resolution but poor frequency resolution, while the ones with the lowest frequencies have good frequency resolution but poor time resolution. Therefore, it proposes reconstructing wavelet packet (WP) signals from the decomposed ones (approximate and details) to increase the signals' time resolution at low frequencies.

2.4 Feature Extraction

In this study, features are extracted by measuring the amount of randomness and uncertainty through nonlinear methods such as entropy. Here, instead of directly computing entropy measures from EEG signals it is computed from each reconstructed WP signal to form the feature vectors.

2.5 Classification and problem formulation

In this study, the performance of several classification techniques to differentiate the PD features from the HC ones: LR, LDA, RF, SVM, and KNN and to use Fuzzy Rough Nearest Neighbor (FRNN) machine learning model which is proposed as much better model to give more efficient results as we proceed. This is to compare between them and to determine which one provide the best results.

3 Performance Evaluation

The metrics are used to evaluate the performance of the developed models are classification accuracy, sensitivity, specificity, F-score, and receiver operating characteristic (ROC) curve.

3.1 Classification Accuracy

Here, TP=True Positives, TN=True Negatives, FP=False Positives, and FN=False Negatives CA = TP + TN$\frac{}{TP+TN+FP+FN\times 10}$

3.2 Sensitivity

Sensitivity = TP$\frac{}{TP+FN\times 100\%}$

3.3 Specificity

Specificity = TN$\frac{}{TN+FP\times 100\%}$

3.4 F-Score

F - score = 2 $\times \frac{Precision\times Sensitivity}{Precision+Sensitivity} \times$ 100%

3.5 Precision

Precision = TP$\frac{}{TP+FP\times 100}$

3.6 ROC Curve

TPR as (Sensitivity) and FPR (1 - Sensitivity)

Please refer to the below image for a detailed method (see Fig. 4).

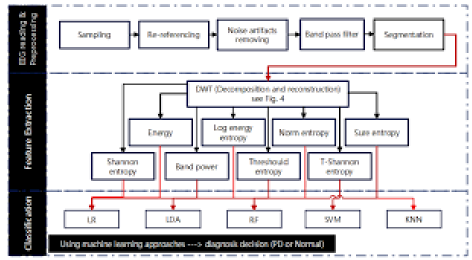

Fig. 4. Block diagram of the proposed PD DWT-based classification methods to show diagonisis of PD or Normal.

4 What we aim to do?

In previous works, several machine learning-based strategies like Decision Tress, Random Forest, K-nearest Neighbor (KNN) were introduced to investigate and interpret EEG signals for the purpose of their accurate classification. In our study we propose to use Fuzzy Rough Nearest Neighbor (FRNN) machine learning model which is proposed as much better model to give more efficient results. Therefore, using Fuzzy classifiers to check if this can give the highest classification accuracy scores, with improved sensitivity and specificity percentages over traditional methods machine learning methods. Performance will be evaluated on the same metrics to compare both the models.

5 Dataset Used

The public dataset used to verify the proposed methods: The SanDiego dataset (31 subjects, 93 min) https://openneuro.org/datasets/ds002778/versions/1.0.2

For citations of references, we prefer

the use of square brackets and consecutive numbers. Citations using labels or the author/year convention are also acceptable. The following bibliography provides a sample reference list with entries for journal articles [1], [2], [3], and a homepage [4]. Multiple citations are grouped [1–3], [1–4, **?**,**?**,**?**].

References

1. Majid Aljalal, Saeed A. Aldosari, Khalil AlSharabi Fahd A. Alturki, Marta Molinas. Detection of Parkinson's disease from EEG signals using discrete wavelet transform, different entropy measures, and machine learning techniques. Article number: 22547 (2022) https://www.nature.com/articles/s41598-022-26644-7data-availability.
2. Han, C. X., Wang, J., Yi, G. S. Che, Y. Q. Investigation of EEG abnormalities in the early stage of Parkinson's disease. Cogn. Neurodyn.**7**(4), 351–359 (2013).https://link.springer.com/article/10.1007/s11571-013-9247-z.
3. Aayesha, Muhammad Bilal Qureshi, Muhammad Afzaal, Muhammad Shuaib Qureshi Muhammad Fayaz. Machine learning-based EEG signals classification model for epileptic seizure detection. 80, 17849–17877 (2021). https://link.springer.com/article/10.1007/s11042-021-10597-6.
4. OpenNero Dataset Link, https://openneuro.org/datasets/ds002778/versions/1.0.2
5. Github Repository containing the colab notebook link, https://github.com/Shatarupa21/Computational-Neuroscience-Project
6. YouTube Video that has contributed to my knowledge, https://www.youtube.com/watchv=$_BdBJOOqMeslist = PLtGXgNsNHqPTgP9wyR8pmy2EuM2ZGHU5Z$
7. Springer Lecture PDF, https://www.overleaf.com/project/63ec72a4876599d249c3a402
8. LNCS Homepage,http://www.springer.com/lncs. Last accessed 15 Feb 2023