Health and Behavior: Next-Gen Health Monitoring Empowered by Python Programming and Deep Learning Applications

Alzheimer's Disease Detection via Machine Learning

Scholar : Ethan Zhu
Mentor : Dr. Behnaz Ghoraani
&Marjan Nassajpour Esfahani
Program : FAU Summer I-SENSE 2024

FLORIDA ATLANTIC UNIVERSITY
Institute for Sensing and Embedded Network Systems Engineering

Table of Contents

Existing Challenges

- EEG signals are highly non-linear and non-stationary, making them noisy and challenging to analyze.
- Limited availability of public datasets restricts the ability to develop and validate models
- lack of standardized international protocols, complicating consistent data collection and analysis
- Extracting significant features from EEG data is difficult

Objectives

1. Develop new algorithms to break down data for analysis
2. Use existing algorithms to classify patients using data
3. Validate existing methods for classifying patients
4. Visualization of decomposed data signals and features

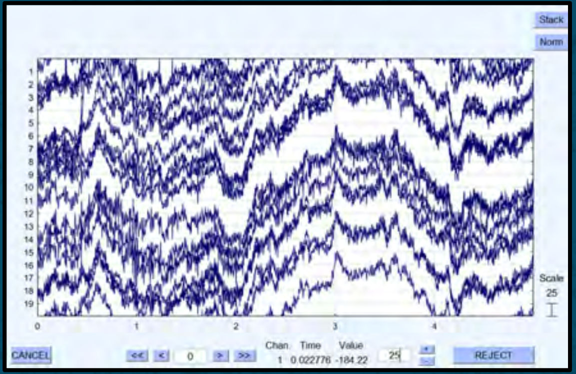
Introduce new data processing/analyzing techniques while using existing machine learning methods for sorting and classification

Introduction to Alzheimer's

- Characterized by permanent degradation in brain neurons
- 100% Fatality Rate
- 7th leading mortality rate in US
- Symptoms : Memory loss, disorientation, behavior change, personality change
- Active Methods rely on early detection to curb symptoms

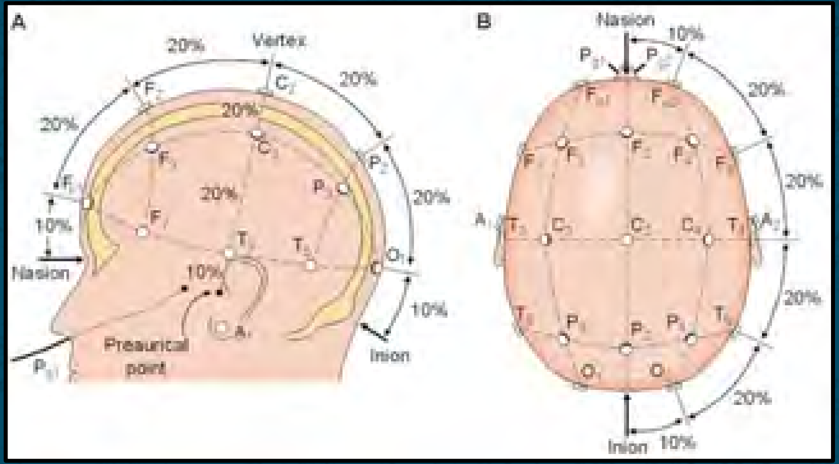
Table of Contents

Data Preprocessing 1

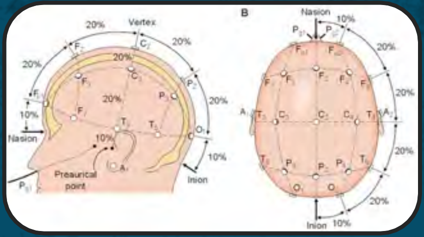


Electroencephalography (EEG):

- Measures electrical activity generated by neurons in the brain using electrodes placed on the scalp
- Postsynaptic potentials of pyramidal neurons
- High temporal resolution and non-invasive



Data Preprocessing 2



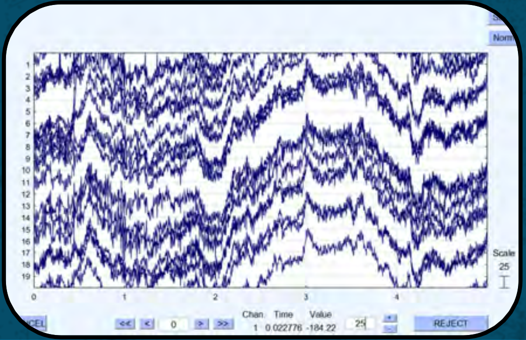
Step 1 Butterworth Band-Pass Filter (0.5-45 Hz)

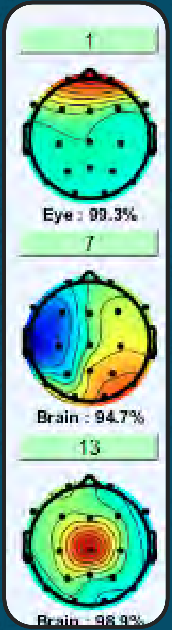
Step 2 Standardizes signal by using A1 and A2 as references

Step 3 Automatic artifact rejection technique (ASR)

Step 4 ICA Method - RunICA Method

Step 5 Signal Processing





Signal Processing

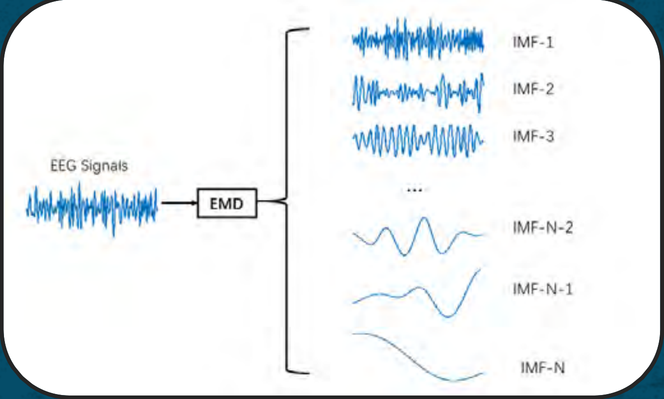
Empirical Mode Decomposition

~ Decompose non-linear and non-stationary signals into finite number of components
~ Sub-categories (EMD, EEMD, MEMD, NA-MEMD, etc)
~ Significant features capture without distorting time domain

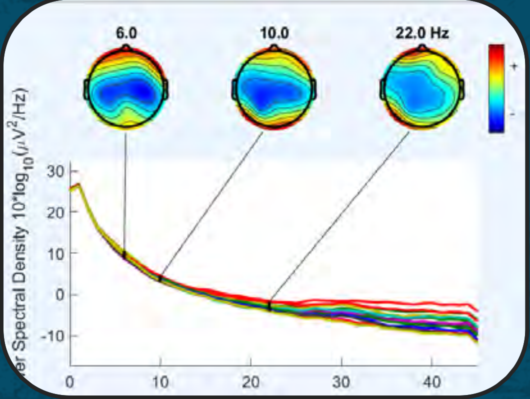
Power Spectral Density

~ Measures the signal's power over the frequency domain.
~ Used with RBP for bandpower extraction
~ Used for understanding energy and power distribution

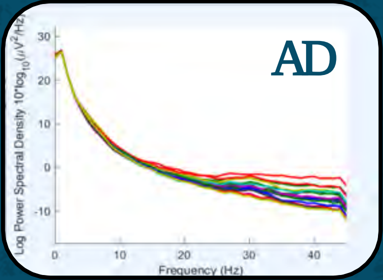
Visuals



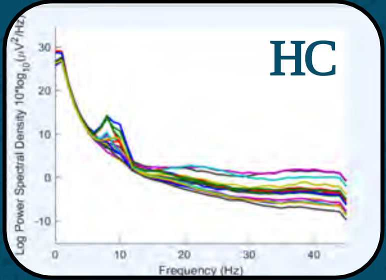
EEG Signals
EMD
IMF-1
IMF-2
IMF-3
...
IMF-N-2
IMF-N-1
IMF-N



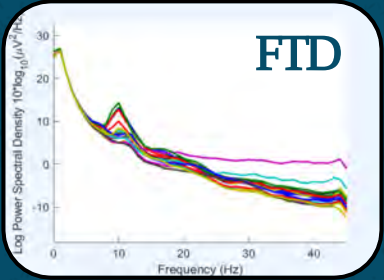
6.0
10.0
22.0 Hz
er Spectral Density 10*log$_{10}$(μV^2/Hz)



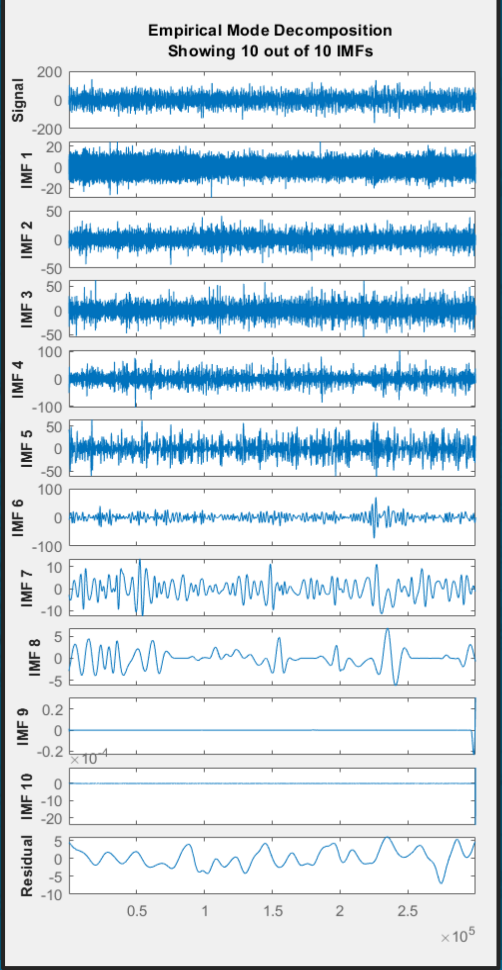
AD
Log Power Spectral Density 10*log$_{10}$(μV^2/Hz)
Frequency (Hz)



HC
Log Power Spectral Density 10*log$_{10}$(μV^2/Hz)
Frequency (Hz)



FTD
Log Power Spectral Density 10*log$_{10}$(μV^2/Hz)
Frequency (Hz)



Empirical Mode Decomposition
Showing 10 out of 10 IMFs
Signal
IMF 1
IMF 2
IMF 3
IMF 4
IMF 5
IMF 6
IMF 7
IMF 8
IMF 9
IMF 10
Residual

Table of Contents

Data Overview

~ 19 scalp electrodes and 2 reference electrodes

		AD	FTD	HC
Recorded Time	Total (min)	485.5	276.5	402
	Range (min)	[5.1, 21.3]	[7.9, 16.9]	[12.5,16.5]
Age	Average (yrs)	66.4	63.6	67.9
	SD (yrs)	7.9	8.2	5.4
MMSE	Average (yrs)	17.75	22.17	30
	SD (yrs)	4.5	8.22	0

	participant	sex	age	category	MMSE
0	sub-002	F	78	A	22
1	sub-003	M	70	A	14
2	sub-004	F	67	A	20
3	sub-005	M	70	A	22
4	sub-006	F	61	A	14
5	sub-007	F	79	A	20

Data and Feature Abstraction

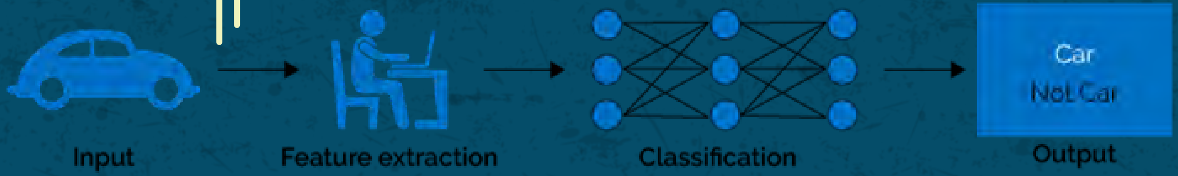
Step 1 PSD-bandpower-extraction (alpha, beta)

Step 2 Multivariate Empirical Mode Decomposition

Step 3 Feature Extraction (see next slide)

Step 4 Machine Learning Models (SVM, XGBoost, R-forest)

Step 5 Cross Validation (LOSO, K-fold)



Features Extracted

1. LBP - normalize bandpower
2. Norm - overall power
3. Energy - intensity
4. HFD - complexity
5. KFD - irregularity and self-similarity
6. LZC - randomness via distinct patterns
7. MF - dominant frequency
8. HP (AMC) - temporal properties

```
def extraction_2(data, count):
    feature_extract = np.full( shape: (19, count, 8),  fill_value: None)
    data = data.transpose(1, 0, 2)
    for i in range(data.shape[0]):
        for j in range(data.shape[1]):
            putt = data[i][j]
            feature_extract[i][j][0] = LBP(putt)
            feature_extract[i][j][1] = Norm(putt)
            feature_extract[i][j][2] = Energy(putt)
            feature_extract[i][j][3] = HFD(putt)
            feature_extract[i][j][4] = KFD(putt)
            feature_extract[i][j][5] = LZC(putt)
            feature_extract[i][j][6] = MF(putt)
            feature_extract[i][j][7] = HP(putt)
            print("successful through")
        print(f"{j} IMF extracted")
    print(f"{i} channel extracted")
    return feature_extract
```

Machine Learning

Support vector machine (SVM)
XGBoost (XGB)
Random Forest (RF)

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split( *arrays: all_features, all_labe
print(f"Train Shape: X_train: {X_train.shape}, y_train: {y_train.shape}")
print(f"Test Shape: X_test: {X_test.shape}, y_test: {y_test.shape}")

# Standardize the features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Handle class imbalance using SMOTE
smote = SMOTE(random_state=42)
X_train, y_train = smote.fit_resample(X_train, y_train)

# Hyperparameter tuning
param_grid = {'C': [0.1, 1], 'gamma': [0.1, 0.01], 'kernel': ['linear', 'rbf']}
grid = GridSearchCV(SVC(), param_grid, refit=True, verbose=2, cv=5)
grid.fit(X_train, y_train)
```

Cross Validation

K-fold Validation

- K=10 (applied ML)
- K=n (LOSO)

```
Feature: HP_Mob | Class 1 vs Class 2, Feature Shape: (10868,), Labels Shape: (10868,)
Train Shape: X_train: (8694, 1), y_train: (8694,)
Test Shape: X_test: (2174, 1), y_test: (2174,)
Feature: HP_Mob | Class 1 vs Class 2
Accuracy: 0.6471941122355106
Classification Report:
              precision    recall  f1-score   support

           0       0.67      0.72      0.69      1206
           1       0.62      0.55      0.58       968
```

Table of Contents

Findings/Conclusion

	HC v. AD		HC v. FTD		AD v. FTD	
	Acc.	**F1**	**Acc.**	**F1**	**Acc.**	**F1**
SVM	56%	71%	57%	71%	62%	75%
RF	65%	65%	68%	72%	63%	77%
XGBoost	65%	66%	66%	72%	62%	77%

Findings/Conclusion

Channel-Sample	Original (s)	New (s)	SpeedUp
5-500	2.50	2.82	11% ▼
10-1000	5.27	3.43	54% ▲
15-5000	22.64	12.56	80% ▲
20-10000	86.83	31.69	174% ▲

Summary of Conclusion

Machine Learning

- Testing accuracy of 68% on small data
- XGBoost provides good accuracy with faster execution.
- RF delivers the best accuracy but with higher time overhead

NA-MEMD

- New MEMD method is exponentially faster than alternative
- Prototype for NA_MEMD with expansive noise options

Future Improvements

Preprocessing

Alternative algorithms and artifact removal

Bandpower

Further bandpower decomposition [0-4Hz], [13-45Hz]

Features

More features focusing on multivariate relationships

ML + Classifier

Implement neural networks (CNN, RNN, etc)

Thank you!

Questions?

Credit and References

1. Li Z, Zhang L, Zhang F, Gu R, Peng W, Hu L. Demystifying signal processing techniques to extract resting-state EEG features for psychologists. Brain Science Advances. 2020;6(3):189-209. doi:10.26599/BSA.2020.9050019

2. AlSharabi, K., Salamah, Y. B., Aljalal, M., Abdurraqeeb, A. M., & Alturki, F. A. (2023). EEG-based clinical decision support system for Alzheimer's disorders diagnosis using EMD and deep learning techniques. *Frontiers in Human Neuroscience*, *17*. https://doi.org/10.3389/fnhum.2023.1190203

3. Miltiadous, A., Tzimourta, K. D., Afrantou, T., Ioannidis, P., Grigoriadis, N., Tsalikakis, D. G., Angelidis, P., Tsipouras, M. G., Glavas, E., Giannakeas, N., & Tzallas, A. T. (2023). A Dataset of Scalp EEG Recordings of Alzheimer's Disease, Frontotemporal Dementia and Healthy Subjects from Routine EEG. *Data*, *8*(6), 95. https://doi.org/10.3390/data8060095

4. Zhang Y, Wang G, Li Z, et al. Matlab Open Source Code: Noise-Assisted Multivariate Empirical Mode Decomposition Based Causal Decomposition for Causality Inference of Bivariate Time Series. *Front Neuroinform*. 2022;16:851645. Published 2022 Jun 16. doi:10.3389/fninf.2022.851645

5. Miltiadous A, Tzimourta KD, Giannakeas N, Tsipouras MG, Afrantou T, Ioannidis P, Tzallas AT. Alzheimer's Disease and Frontotemporal Dementia: A Robust Classification Method of EEG Signals and a Comparison of Validation Methods. Diagnostics (Basel). 2021 Aug 9;11(8):1437. doi: 10.3390/diagnostics11081437. PMID: 34441371; PMCID: PMC8391578.

6. Zhu E, Health and Behavior: Next-Gen Health Monitoring Empowered by Python Programming and Deep Learning Applications.(2024). GitHub Repository https://github.com/PiethonProgram/NA-MEMD