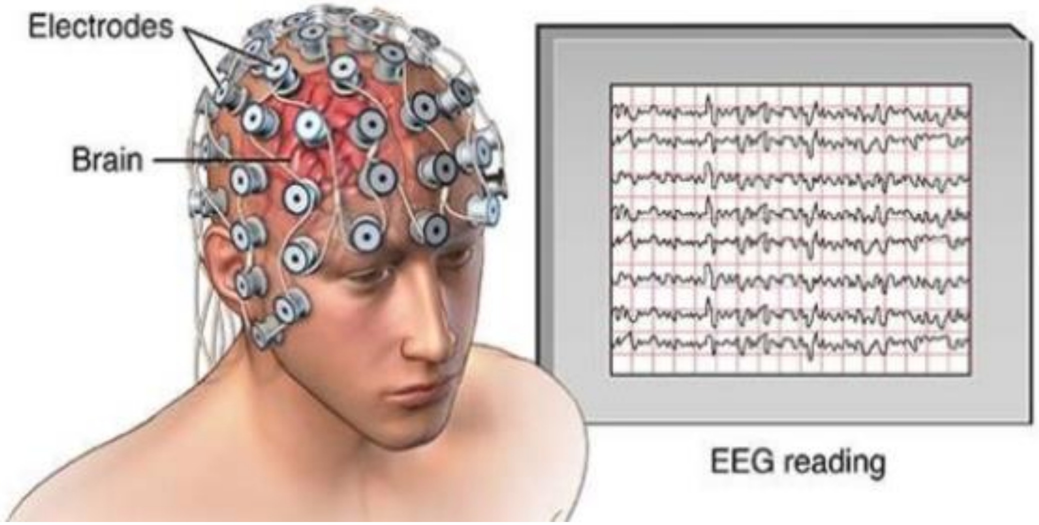
Brain2Object: Printing Your Mind from Brain Signals with Spatial Correlation Embedding

DongWook Kim
Soft Robotics and Bionics Laboratory
ME, Seoul National University
+Added Reproducing Experiment Results

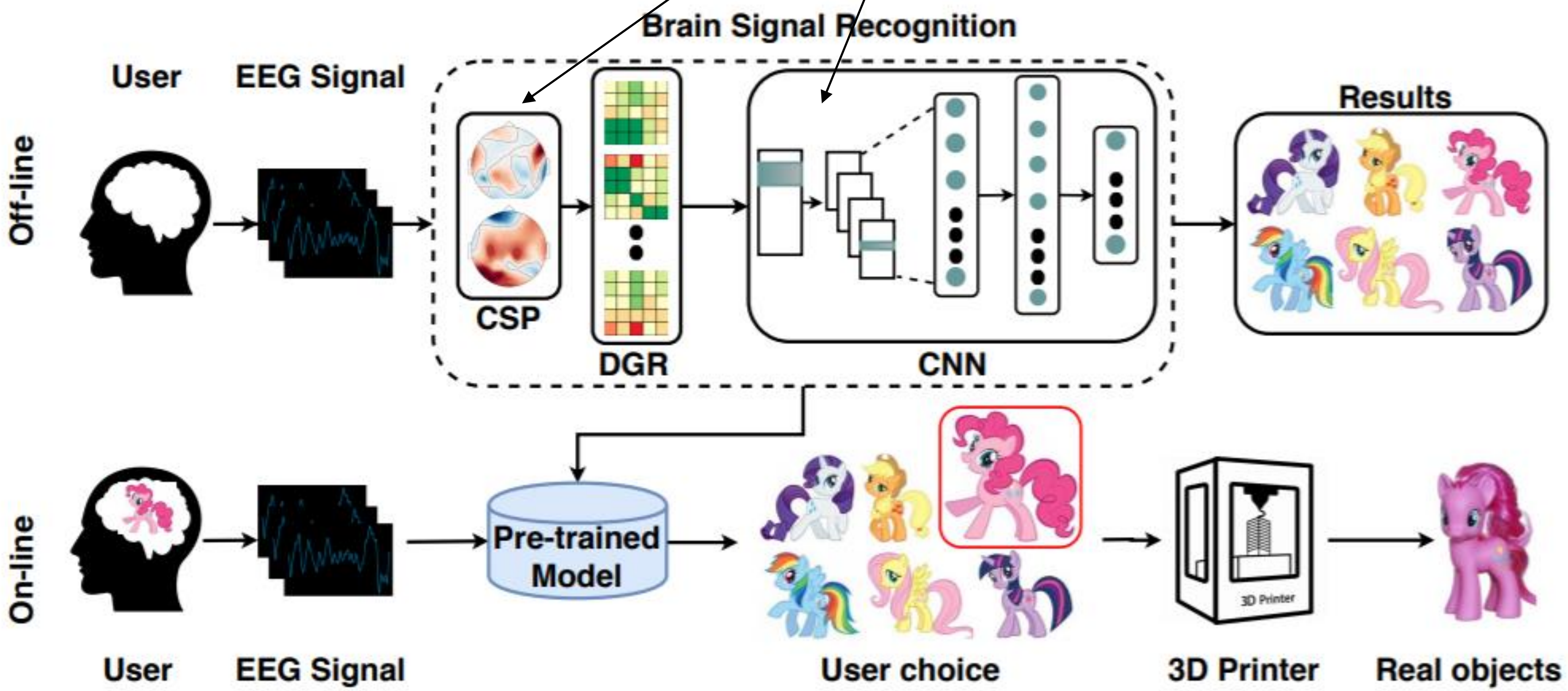
Graph Network for Brain Signals

- Electroencephalography (EEG)
  - Record the voltage of ionic current within the neurons of brains
- Learning the structured EEG signal using:
  - (1) Common Spatial Pattern (CSP)
  - (2) Graph Convolutional Network (GCN)
- Goal: predict which object is the human gazing
  - Application: Print the predicted object



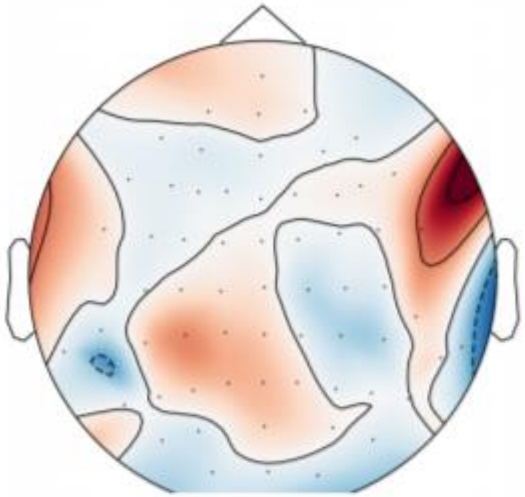
Brain Signals Recognition Schematics

- Spatial representation of EEG: (1) CSP (2) GCN



Common Spatial Pattern (CSP)

- EEG: 64 channels, sampling frequency 260Hz, K categories (each N_i)



- Step 1: Calculate the covariance matrix

Given time-domain signal: $E_i \in \mathbb{R}^{64\times 260}$

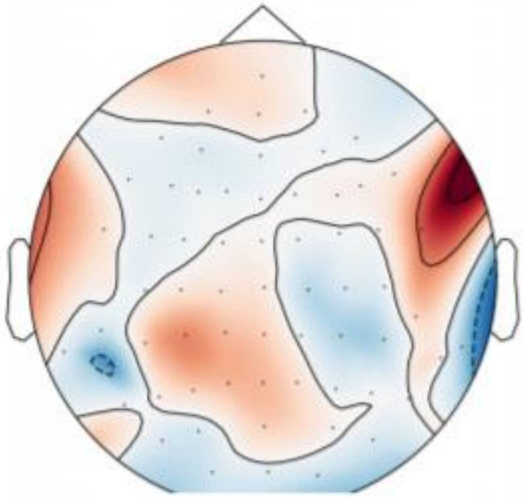
where i=1, 2, …, N is the number of samples

Calculate the covariance matrix as:

$$C_i = \frac{E_i E_i^T}{\mathrm{tr}(E_i E_i^T)}$$

Common Spatial Pattern (CSP)

- EEG: 64 channels, sampling frequency 260Hz, K categories (each N_i)



- Step 2: For each classification label, average them

$$\bar{C}_k = \frac{1}{N_k}\sum_{i=1}^{N_k} C_i$$

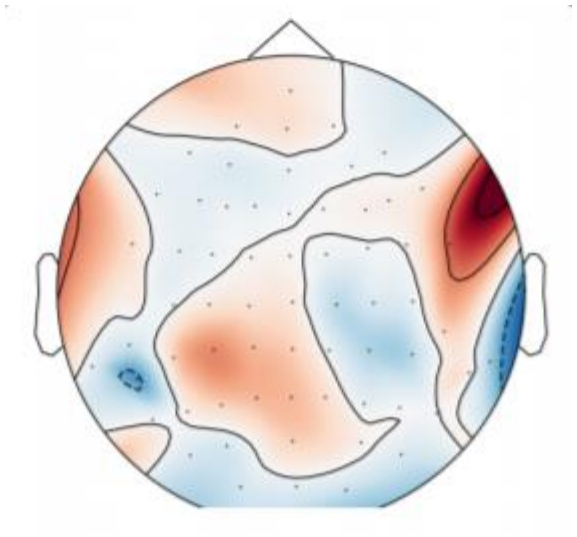
for each kth category.

The composite covariance matrix is

$$\bar{C} = \sum_{k=1}^{K} \bar{C}_k$$

Common Spatial Pattern (CSP)

- EEG: 64 channels, sampling frequency 260Hz, K categories (each N_i)



- Step 3: Decomposition and Whitening

$$\bar{C} = U\lambda U^T$$

$$S = \left(\sqrt{\lambda^{-1}}U\right)\bar{C}\left(\sqrt{\lambda^{-1}}U\right)^T$$

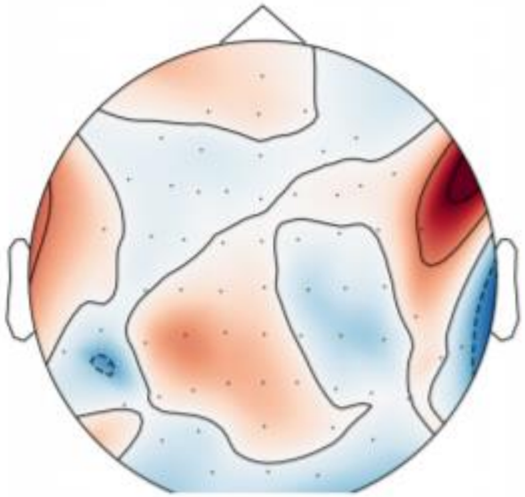
Applying this to each category:

$$S_i = \sum_{k=1}^{K} B\lambda_k B^T$$

where $B = PU$.

Common Spatial Pattern (CSP)

- EEG: 64 channels, sampling frequency 260Hz, K categories (each N_i)



- Step 4: Optimization

$$w^* = \text{argmax} \frac{wS_iw^T}{\sum_{j\neq i} wS_jw^T}$$

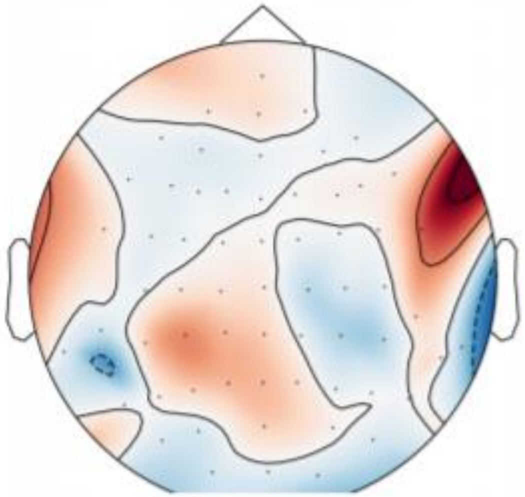
- Rayleigh Quotient Optimization

$$R(M,x) = \frac{x^*Mx}{x^*x}$$

$$\boldsymbol{S}_i\boldsymbol{w} = \lambda \sum_{j\neq i} \boldsymbol{S}_j\boldsymbol{w}$$

Common Spatial Pattern (CSP)

- EEG: 64 channels, sampling frequency 260Hz, K categories (each N_i)



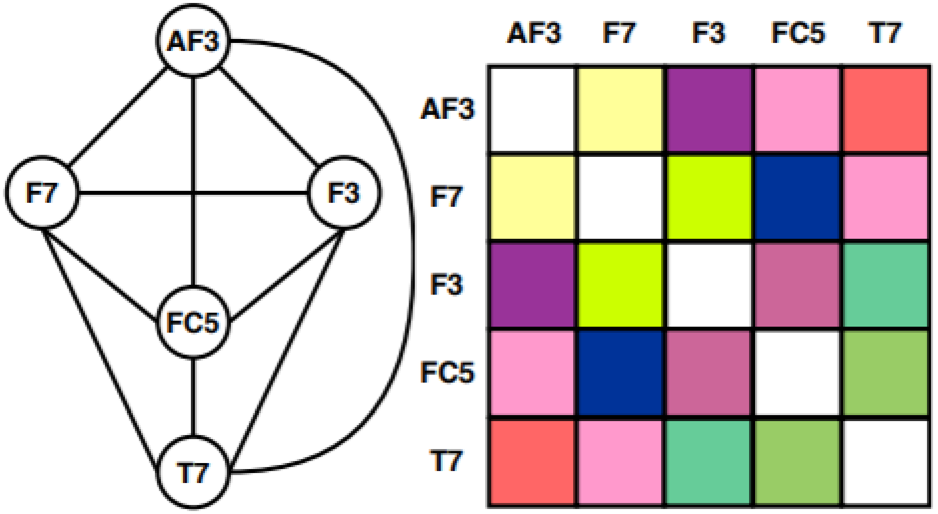
- Step 4: Optimization

$$w^* = \operatorname{argmax} \frac{wS_iw^T}{\sum_{j\neq i} wS_jw^T}$$

Step 5: Signal Filtering

$$E_F = wE$$

Convolutional Graph Representation **(GCN)**



(a) Graph (b) Adjacency matrix

Fig. 2: Example of a complete weighted undirected graph with 5 vertices and the corresponding adjacency matrix. The five vertices are reading from Frontal (F) and Temporal (T) lobes of human brain. The adjacency matrix is symmetric matrices, in which the colors denote the connection weights.

Convolutional Graph Representation (GCN)

- Consider the brain network as a graph, where each vertex is a channel

$$\bar{E} = w^{*}E \longrightarrow E' = (\mathcal{A} + I)\bar{E}$$

Adjacency Matrix (Trainable Variable)

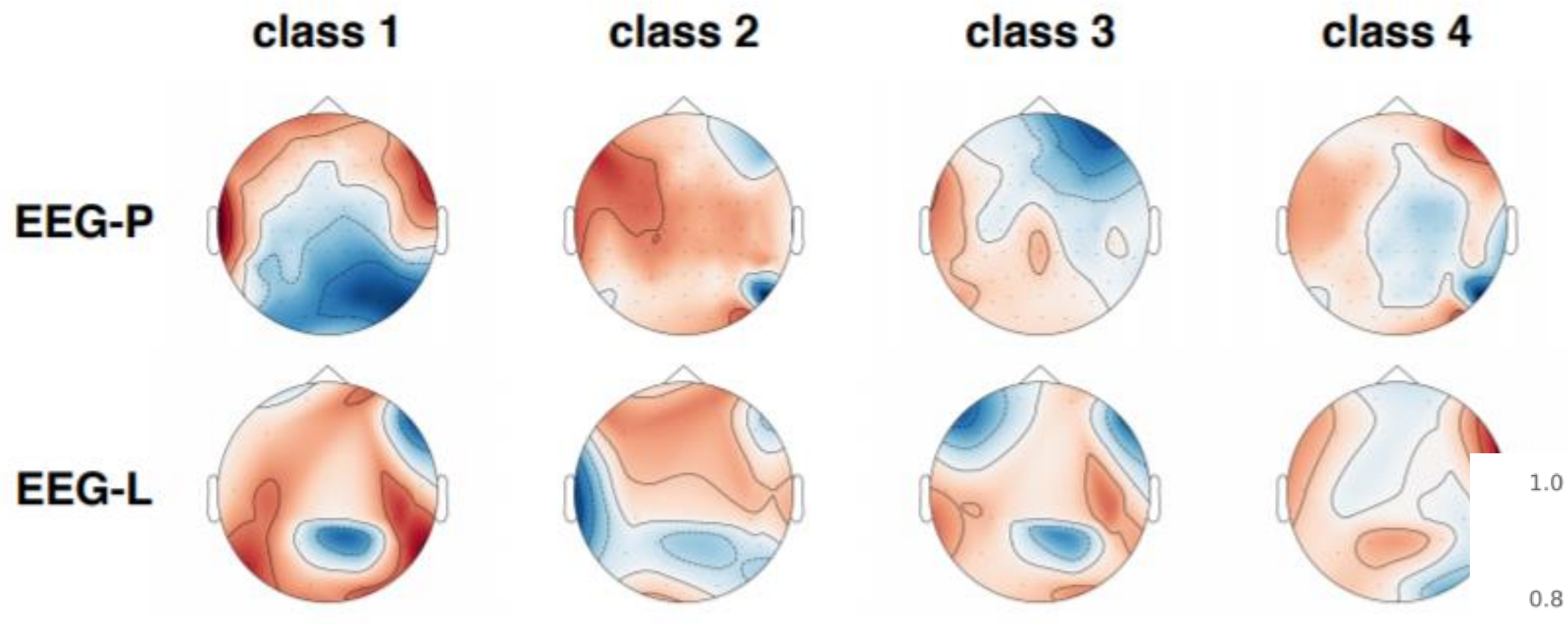
$$\boldsymbol{x}' = tanh(\sum_{i}\sum_{j}\boldsymbol{f}_{ij} * \boldsymbol{x}_{ij})$$

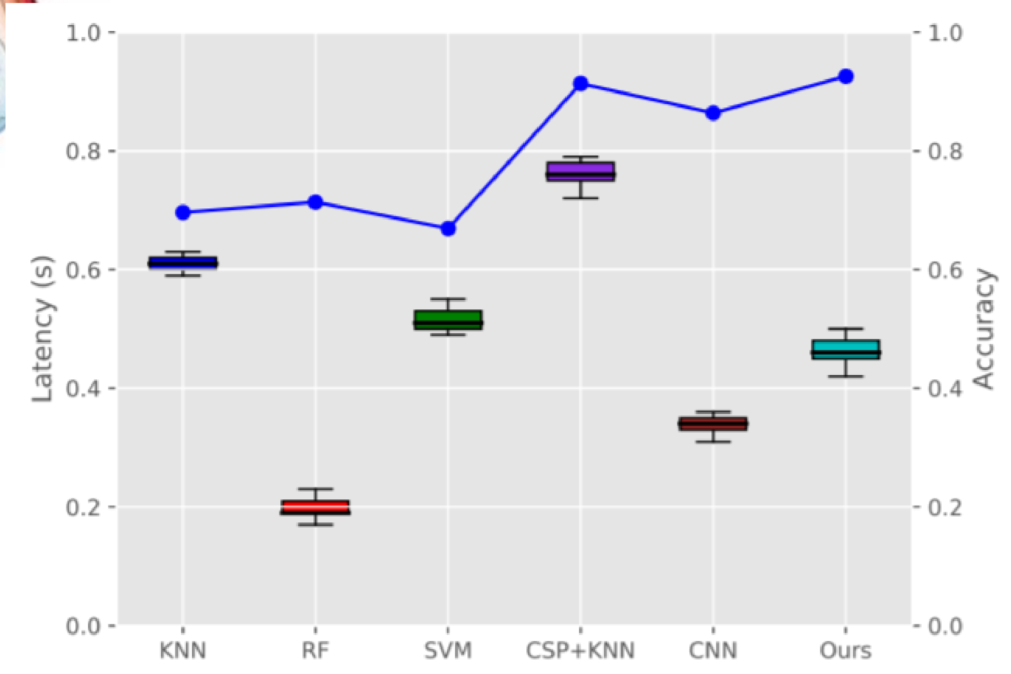
$$\boldsymbol{E}^{\boldsymbol{h}+\boldsymbol{1}} = softmax(\bar{\boldsymbol{w}}\boldsymbol{E}^{\boldsymbol{h}} + \bar{\boldsymbol{b}})$$

$$loss = -\sum_{k=1}^{K}\boldsymbol{y}_{k}log(p_{k})$$

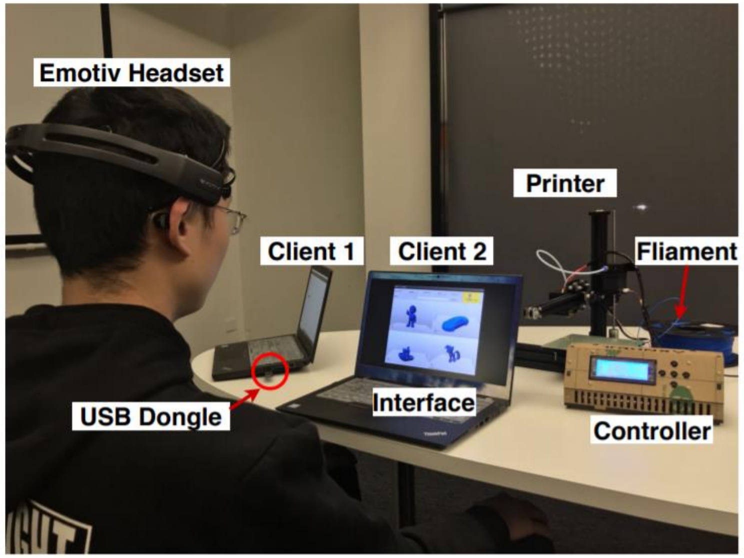
- Note that the Adjacency matrix is “Trainable”
- Dynamic Graph Network

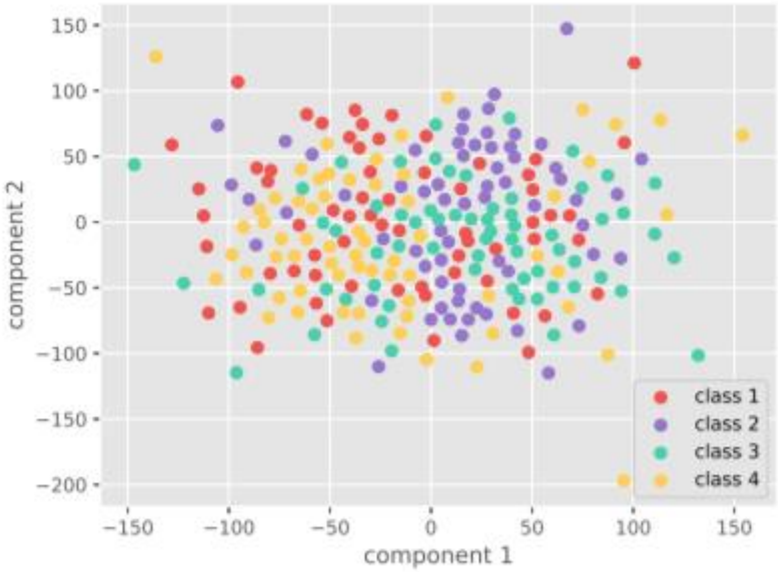
Experimental Results



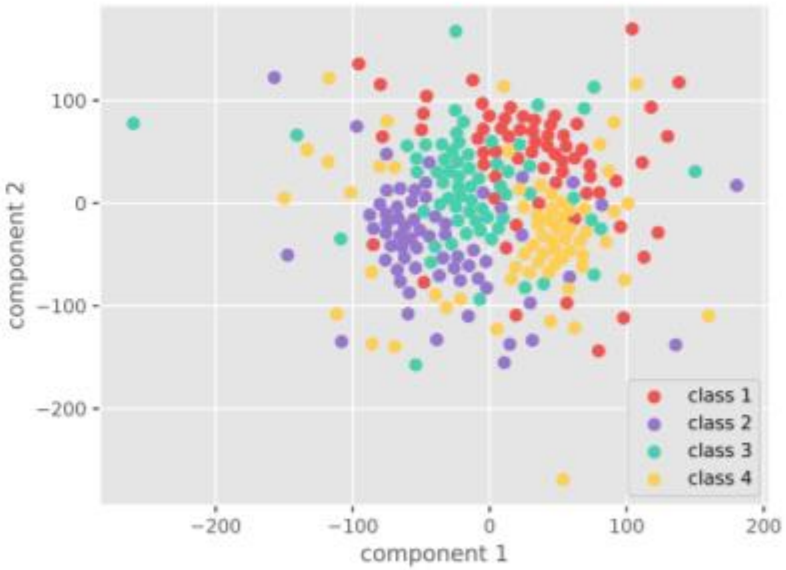


Experimental Results





(a) EEG-P raw data



(b) EEG-P feature

Reproducing Experiment

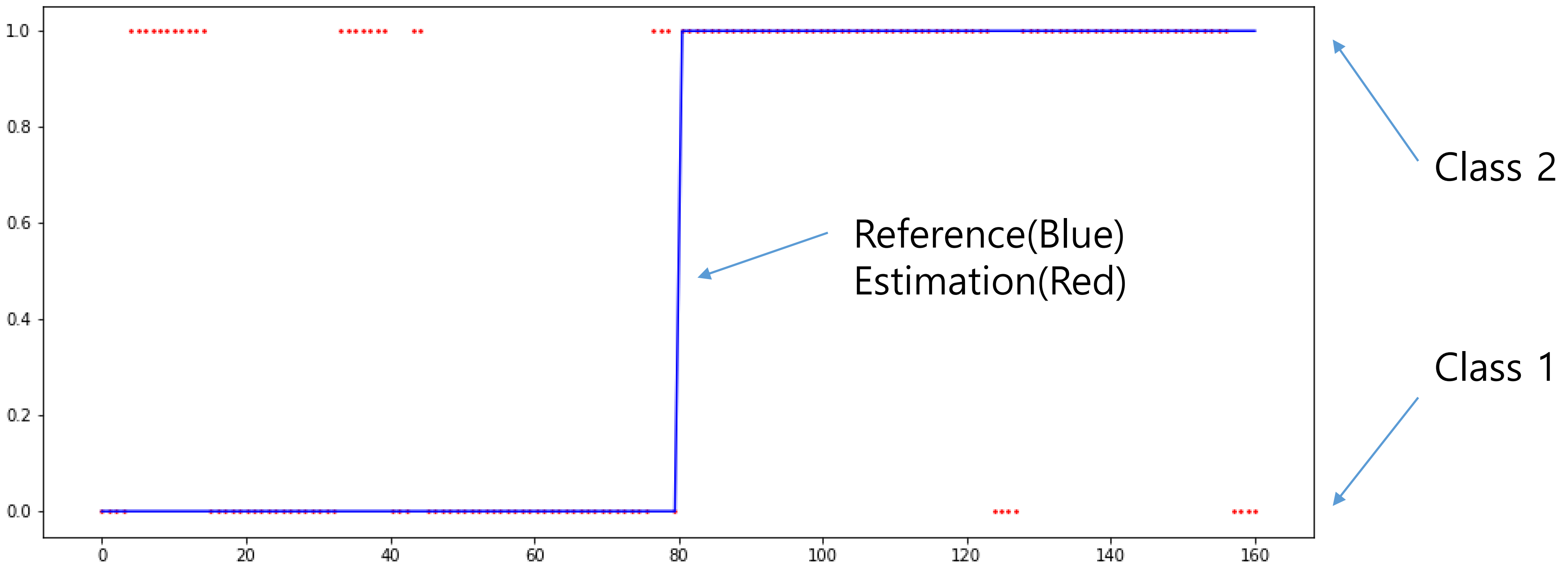
- Reproducing 실험으로, 분당서울대병원과 co-work 실험에서 fNIRS라는 장비를 사용해 얻은 뇌파 데이터를 사용하였습니다.
- 뇌파 신호는 뇌졸중 환자가 손을 움켜쥐는 동작 (class 1)과 팔을 들었다가 내리는 동작 (class 2)을 얻은 후에, 두 동작을 잘 구분할 수 있는지 여부를 평가합니다.
- Github에 Reproducing code와 뇌파 데이터 샘플 (mat 파일)이 있습니다.
- 뇌파 신호를 CSP를 이용해 training set의 feature를 추출한 후, Graph Neural Network를 사용해 피쳐를 학습해서 test set에서 class 1과 class 2를 구분할 수 있도록 합니다.
- Jupyter notebook에서 작성되었으며, Tensorflow 1.13버젼을 사용했습니다.
- 총 5명의 환자 데이터에 대해 분석을 진행하였습니다.

    - 이 5명의 환자는 동일한 기저질환 및 손상부위를 지닌 환자입니다.
- CSP -> GNN(총 2개 layer, 각각 hidden feature의 차원 3, 5) -> FCN(1개 layer)
- Code: github.com/mochacoco/GCN_Project

Reproducing Experiment: Result

- 36번 환자의 Task 구분 결과입니다.
- 구분확률: 129/160 = 80.6%
- 일반 ANN을 사용했을 경우 구분확률: 71.3%

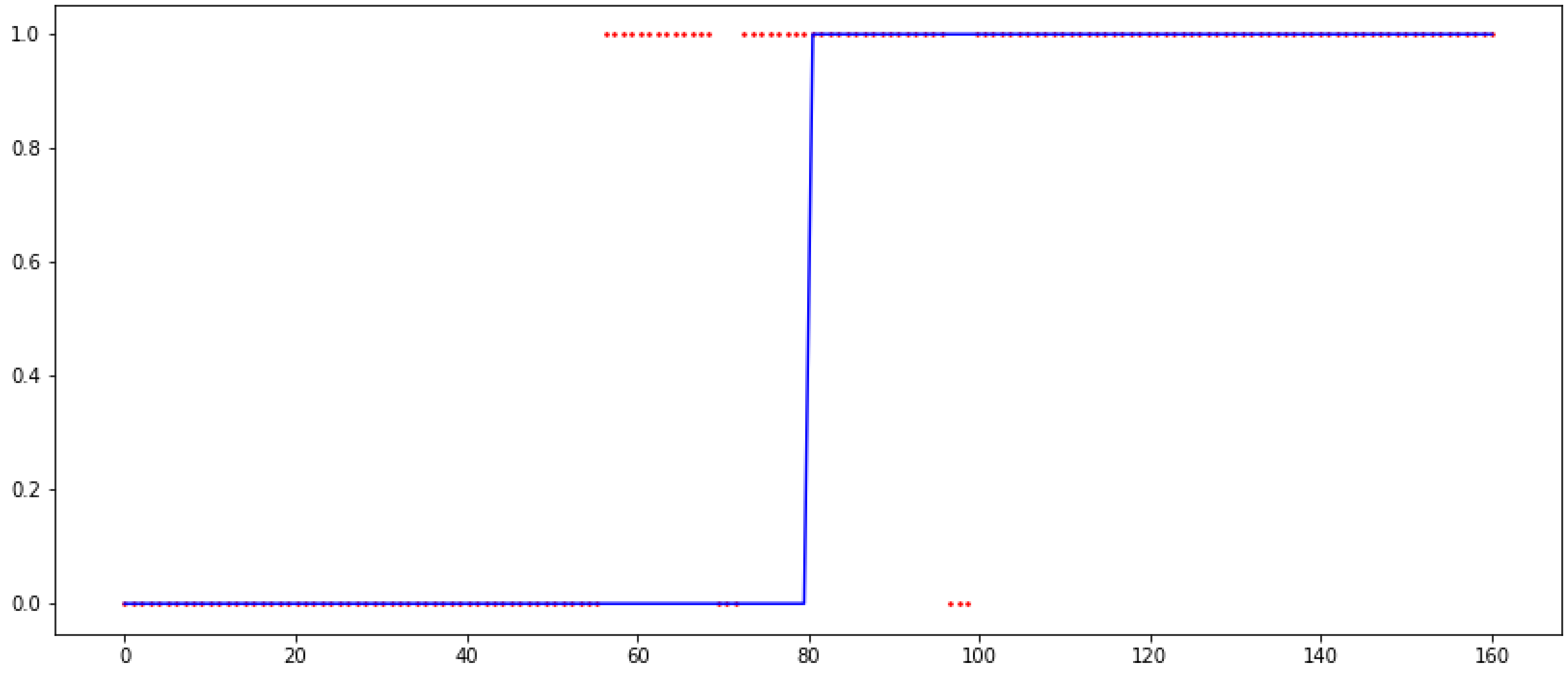
Out [408]: 129



Reproducing Experiment: Result

- 37번 환자의 Task 구분 결과입니다.
- 구분확률: 136/160 = 85.0%
- 일반 ANN을 사용했을 경우 구분확률: 74.4%

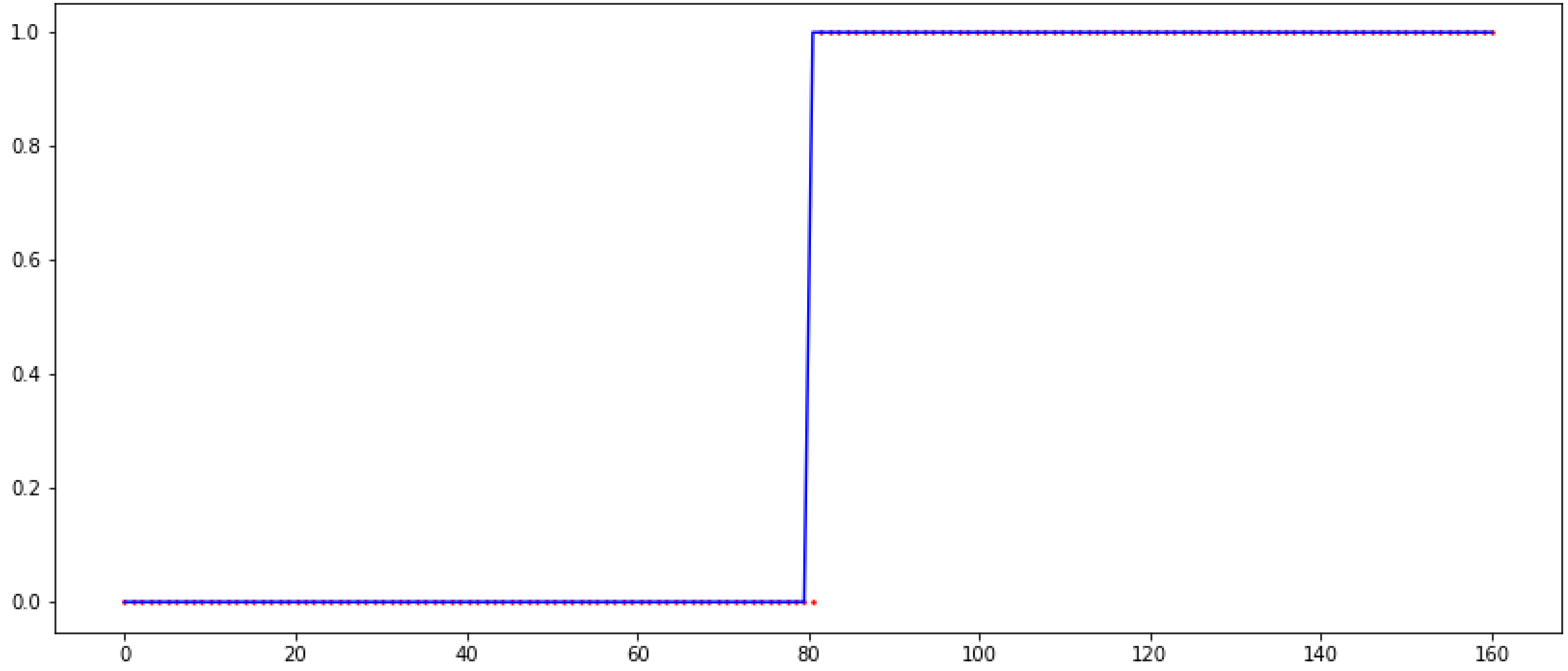
Out [415]: 136



Reproducing Experiment: Result

- 42번 환자의 Task 구분 결과입니다.
- 구분확률: 159/160 = 99.4%
- 일반 ANN을 사용했을 경우 구분확률: 58.3%

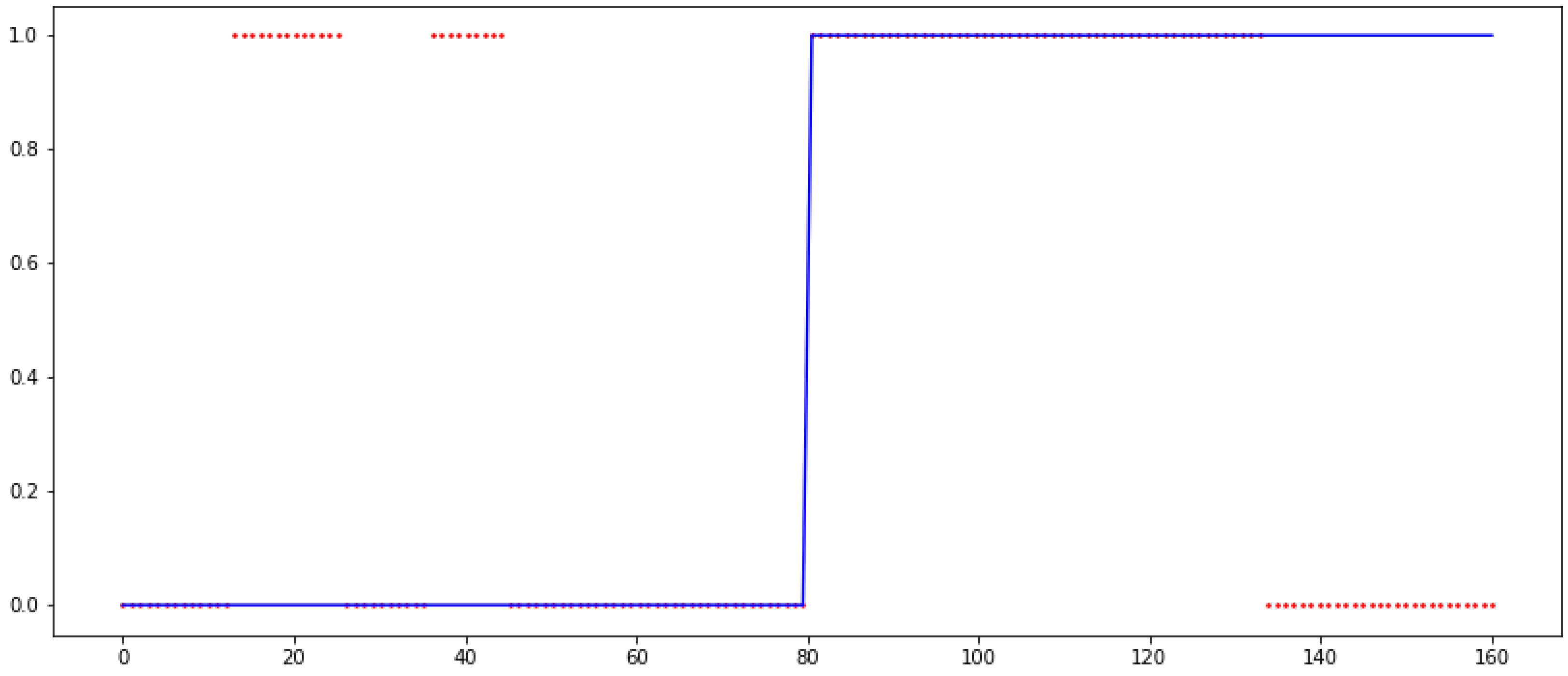
Out [444]: 159



Reproducing Experiment: Result

- 44번 환자의 Task 구분 결과입니다.
- 구분확률: 111/160 = 69.4%
- 일반 ANN을 사용했을 경우 구분확률: 50.0%

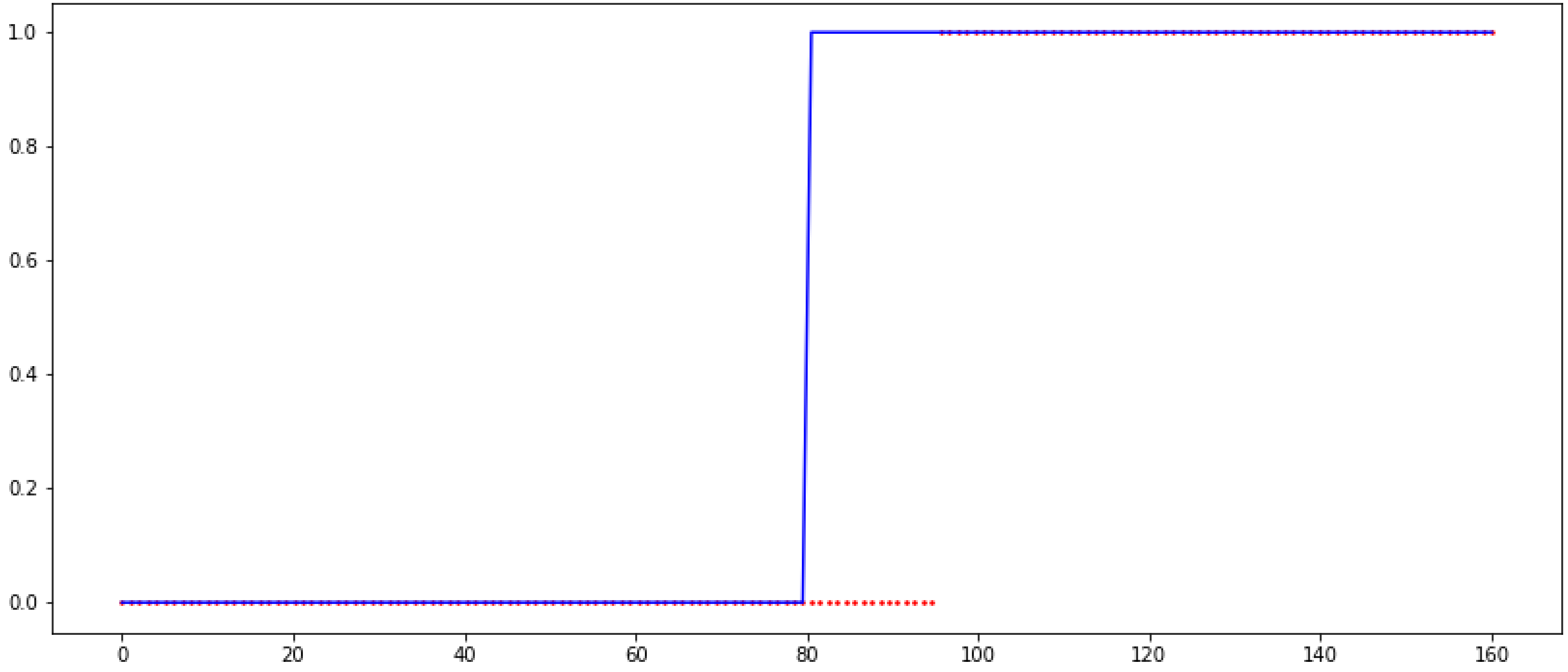
Out [460]: 111



Reproducing Experiment: Result

- 46번 환자의 Task 구분 결과입니다.
- 구분확률: 145/160 = 90.6%
- 일반 ANN을 사용했을 경우 구분확률: 80.4%

Out [470] : 145



Reproducing Experiment - Discussion

- fNIRS 신호 차원(56)에 비해 신호의 길이가 짧아서 (총 800) 학습을 진행할 때의 불안정성이 존재함. (매 학습 시마다 정확도 편차가 존재)
- 신호가 time series라는 것을 고려해서 Recurrency를 고려한 GNN을 학습한다면 더 좋은 결과를 얻을 수 있을 것이라고 기대함.
- GNN의 adjacency matrix를 변수로 두었을 때 정확도를 높일 수 있는 알고리즘을 사용하면 더 좋은 결과를 얻을 수 있을 것이라고 기대함.
- 일반 ANN을 사용하였을 경우보다 정확도가 상승함.