# Uncovering the structure of clinical EEG signals with self-supervised learning

Hubert Banville, Omar Chehab, Aapo Hyvärinen, Densi-Alexander Engemann and Alexandre Gramfort

Romane Barra
Grégoire Béchade

8 Janvier 2025

# Objectives



Analyze ElectroencEphaloGraphy (**EEG**) signals by using Self-Supervised Learning (**SSL**) methods.

- Reduce expensive manual annotations.
- Models trained to identify structure in the unlabeled data
- Two application problems :
  - **Sleep staging** : classify sleep stages using short EEG windows.
  - **Pathology detection** : identify pathological EEG patterns.

# Computational approach and models



Optimization based on stochastic gradient descent (SGD) with backpropagation.

1. **Preprocessing** : filtering, downsampling and segmenting the data into windows.
2. **Training** : CNNs on SSL **pretext tasks**.
3. **Evaluation** : learned embeddings tested on supervised tasks (logistic regression).

# The Pretext Task : three SSL methods



SSL principle : learn "latent structure" in the data.
Relative Positioning (**RP**) : predicts whether two EEG windows are close in time or not.

- Assumption : close windows in time are likely to share similarities.
- + simple and computationally efficient.
- - limited in simple tasks (e.g. sleep staging) : cannot apply to models with abrupt changes in time.

# The Pretext Task : three SSL methods



Temporal Shuffling (**TS**) : predicts if three EEG windows are in temporal order or shuffled.

- Assumption : EEG signals follow a temporal progression.
- $+$ efficient for finding transitions between stages.
- $+$ more robust than **RP** : evaluation of a triplet (and not a pair).
- - limited in simple tasks (e.g. sleep staging) : cannot apply to models with abrupt changes in time.
- - less computationally efficient than RP.

# The Pretext Task : three SSL methods

Contrastive Predictive Coding (**CPC**) : predicts the next EEG window given a sequence of previous windows.

- + learn long-term dependencies and more complex structures.
- + uses an autoregressive model.
- + genezalise on complex downstream tasks (e.g. pathology detection).
- - requires larger datasets.
- - less computationally efficient.

# The Pretext Task : three SSL methods

SSL methods are test-dependent.

- **RP** and **TS** perform well with short-term temporal dependencies (e.g. sleep staging).
- **CPC** excels in long-term temporal understanding (e.g. pathology detection).

SSL tasks outperform supervised methods when there are few labeled data.

# The data

36 EEG of 60 seconds, sampling rate = 516 Hz, 19 channels.
Task : detecting the EEG of a patient performing mental arithmetic tasks.
Each EEG is split in 124 segments of 1000 points.
Available here.

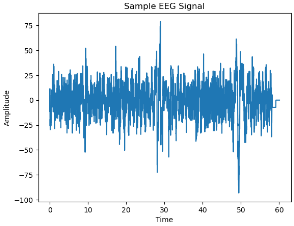


Figure – Example of EEG

# Definition of the task



Objective : training a neural network that takes as an input a segment of 1000 points and outputs the probability for the patient to be currently performing a mental arithmetic task.

Structure :

1. A first CNN to extract features from the EEG (vector in $\mathbb{R}^{100}$).
2. A fully connected layer that outputs a probability to be in class 1.

# Features Extractor

Based on StagerNet from [1].

1. 4 convolutional layers, padding $= 1$, kernel size $= 3$.
2. Dropout, relu activation function and max pool at each layer.
3. Last layer : fully connected.

# Pretraining



Pretraining of the feature extractor.
Intuition : the features of two patterns from the same EEG should be the same if they are close in time.
Training on 8928 pairs of subsequences close or not.
Training on 100 epochs with BCE loss.

# Fine tuning



We added a fully connected layer and trained only the weights of the last layer on 200 epochs with BCE Loss on the dataset of EEG on the classification task.
On the other hand, we trained the whole network not pretrained on the same dataset.
Key points :

- Clever initialization of the weights for the fully connected layer.
- Normalization of the EEG essential.

# Results

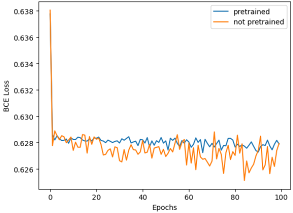


Figure – Loss of the models during training

| Model | Test loss | F1 score |
| --- | --- | --- |
| Pretrained | 0.62 | 0.81 |
| Not pretrained | 0.66 | 0.79 |

Table – Results of the models

# Conclusion

- Pretraining enables to get better results, especially when facing a small dataset.
- Other models could be tested, and different pretraining tasks could be tried.

[1] Hubert BANVILLE et al. "Uncovering the structure of clinical EEG signals with self-supervised learning". In : *Journal of Neural Engineering* 18.4 (2021), p. 046020.