Self-Supervised Learning for Sleep Staging

Théo Moutakanni		Clément Bonnet

Supervisors: *Alexandre Gramfort*, *Nora Ouzir*

Contents

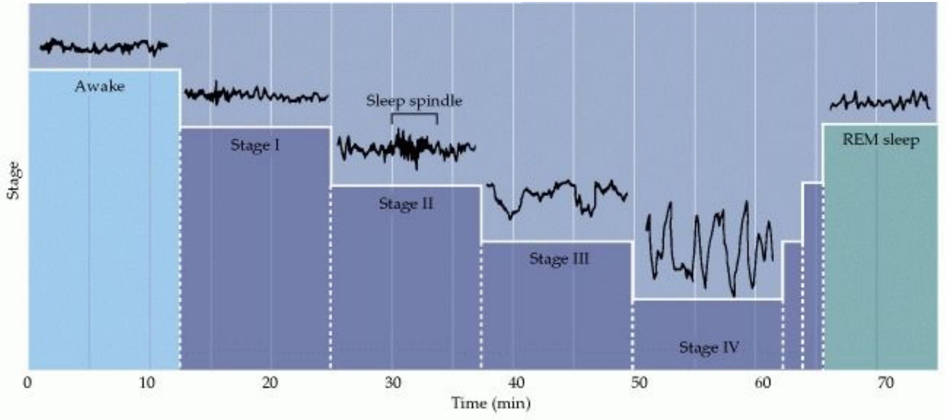
Introduction

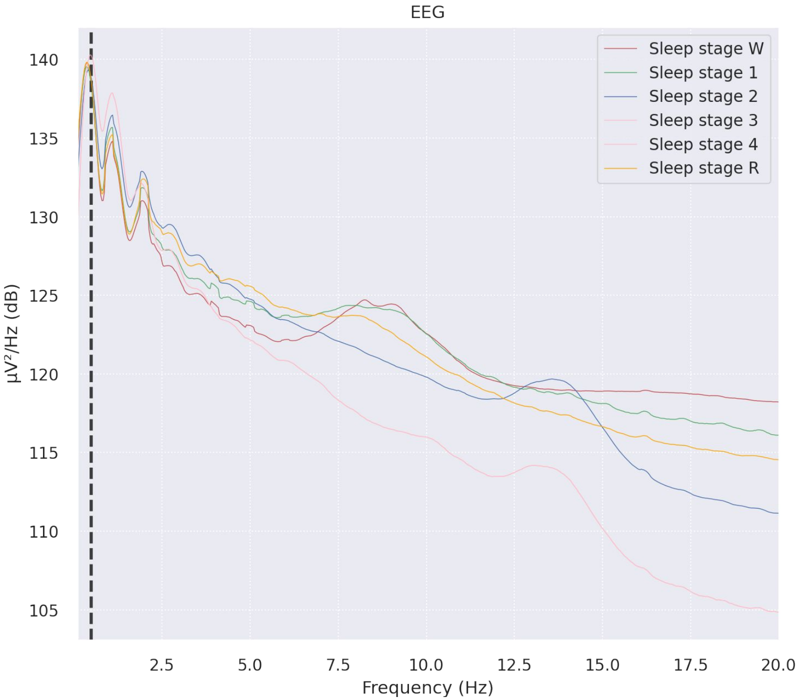
- Supervised learning
  - High performance
  - Requires lots of labeled data
  - Costly
- Unsupervised learning
  - Has access to lots of unlabeled data + some labeled data
  - Difficult
  - Not as good as supervised learning (but can be better if data >>>> 1)
- Self-supervised learning
  - Supervised learning objective from unlabeled data
  - Learn a representation to be provided to the downstream task as an inductive bias
  - Requires labels only for downstream task

Background

- Electroencephalograms & Sleep
- Self-Supervised Learning
- Application to Sleep Staging

Electroencephalograms





Awake: beta activity, high-frequency (15–60 Hz), low-amplitude activity (~30 μV).

Stage I: Theta waves (4–8 Hz) and increasing amplitude (50–100 μV).

Stage II: Spindles (10–15 Hz) oscillations (50–150 μV) during a few seconds.

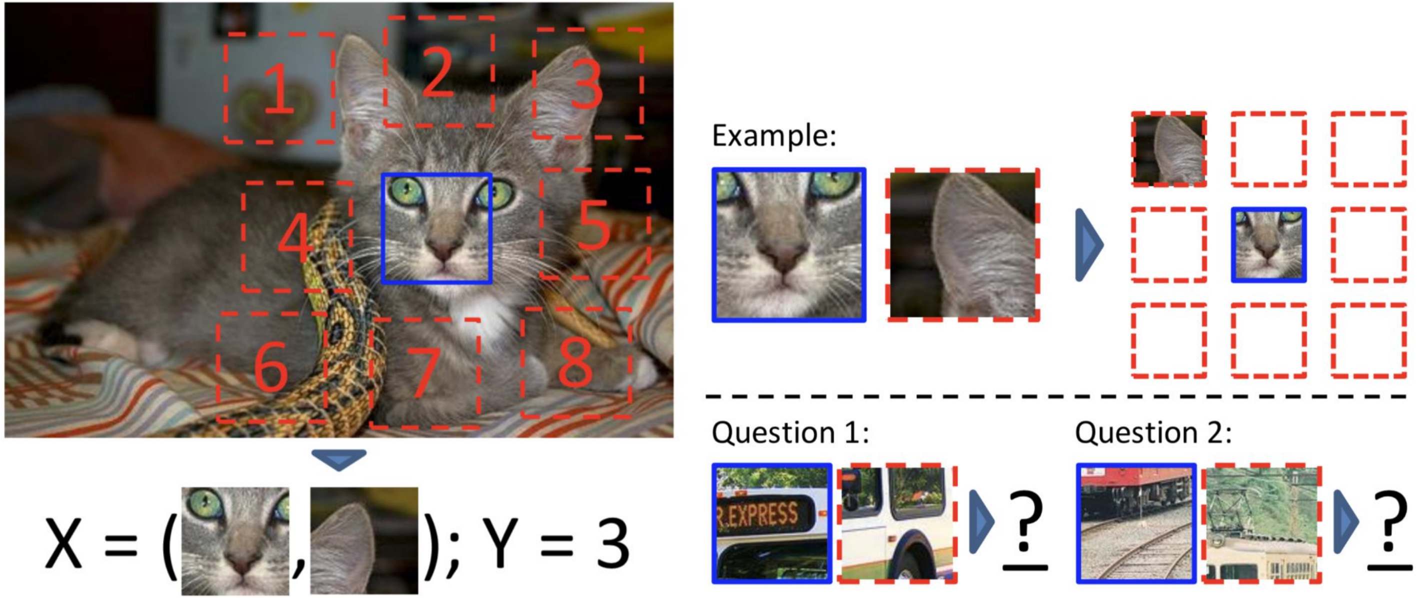
Stage III-IV: delta waves 0.5–4 Hz (100–150 μV).

REM: similar to the EEG activity of individuals who are awake

From Neuroscience. 2nd edition. Purves D, Augustine GJ, Fitzpatrick D, et al., 2001

Self-Supervised Learning

- Learn representations of the data and its intrinsic biases
- Train on a pretext task built from unlabeled data



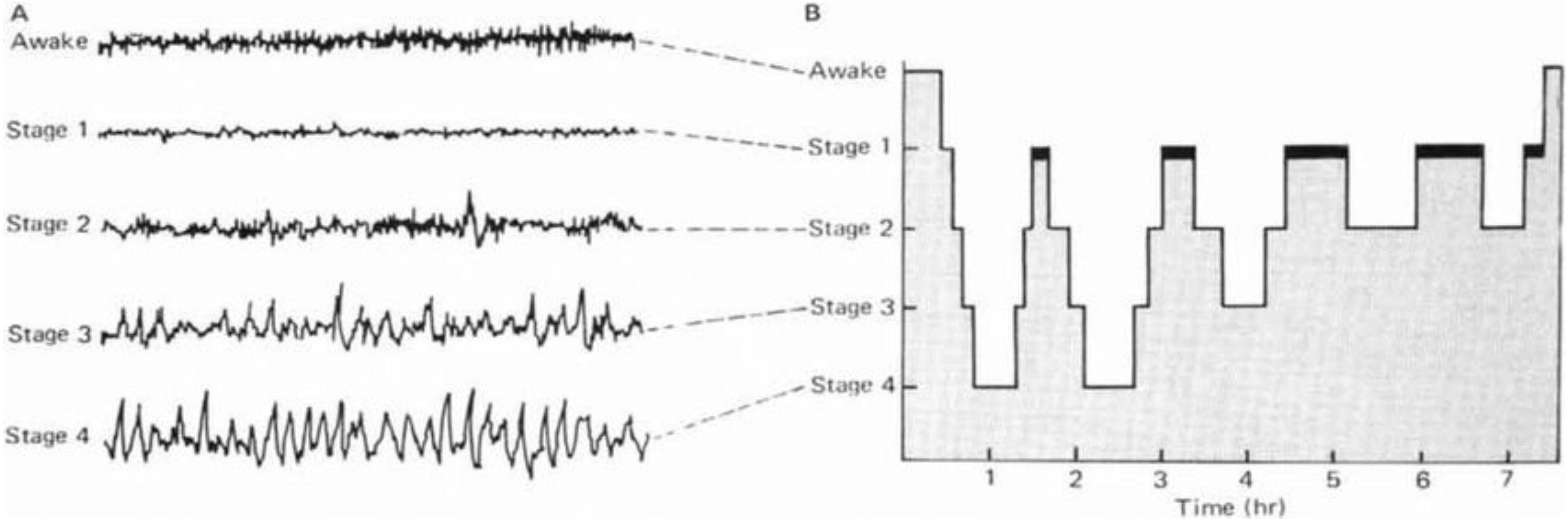
From Doersch et al., 2015

Related Works - Self-Supervised Learning

- Images
  - Prediction of positions of random patches selected from an image [Doersch et al., 2015]
  - Prediction of image rotation [Gidaris et al., 2018]
- Videos
  - Ordering of video frames [Misra et al., 2016]
- Time series
  - Classifying whether windows are sampled from the same temporal context [Banville et al., 2019]
  - Recovering relative position of sampled windows [Banville et al., 2020]
  - Recovering whether windows are chronologically ordered [Banville et al., 2020]

Application to Sleep Staging

- Downstream task: sleep staging classification
- Useful in low (labeled) data regime



From Wikipedia

Related Works - Sleep Stage Detection

- **H. Phan et al.**, Seqsleepnet: End-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging. *IEEE Transactions on Neural Systems and Rehabilitation Engineering,* **2019**.
- **Jens Stephansen et al.**, Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy. *Nature Communications,* **2018**.
- **Mathias Perslev et al.**, U-time: A fully convolutional network for time series segmentation applied to sleep staging, **2019**.
- **Antoine Guillot and Valentin Thorey**. Robustsleepnet: Transfer learning for automated sleep staging at scale, **2021**.

Experiments

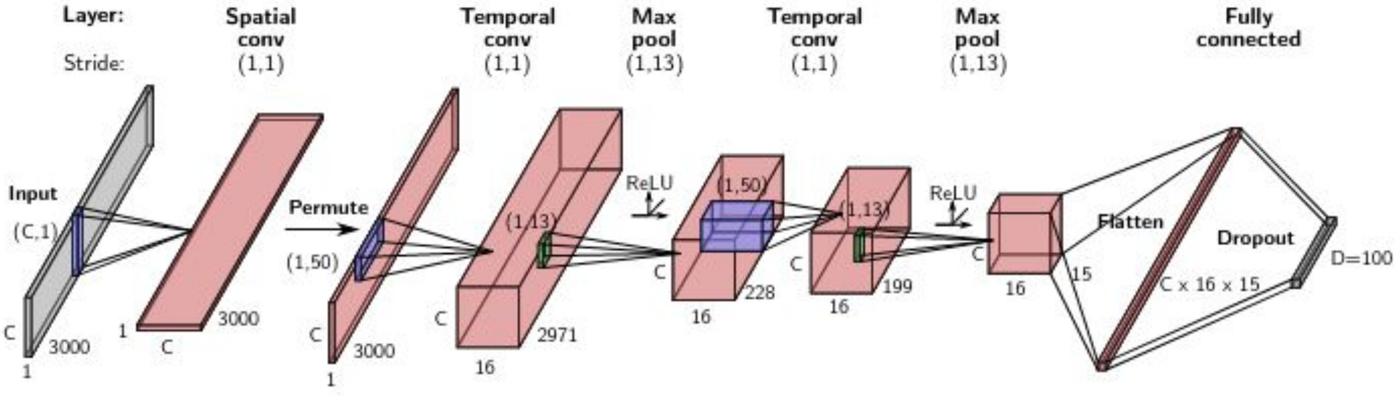
→ Reproduce results from *Uncovering the structure of clinical EEG signals with self-supervised learning, Banville et al., 2020.*

- Model Architecture
- Dataset
- Pretext Task (SSL)
- Downstream Task
- Pre-training Size
- Embeddings Analysis

Model Architecture

→ Modified StagerNet (Banville et al., 2020; Chambon et al., 2018)

- 3-layer CNN
- Relu
- Batchnorm
- Maxpool
- Dropout



From Banville et al., 2020

Dataset

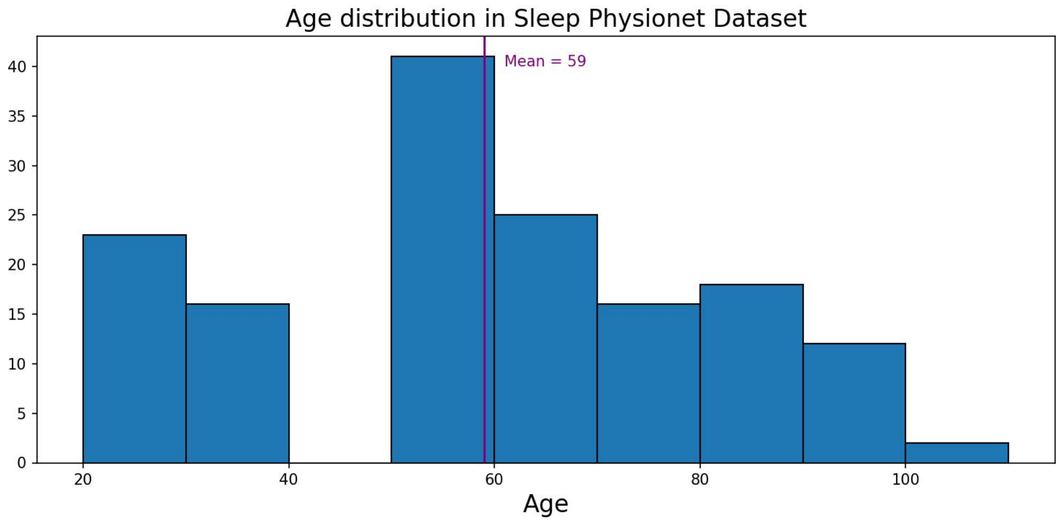
Sleep Physionet Dataset (Goldberget et al., 2000)

- 30-s windows
- 2 EEG channels
- 83 patients
- 2 nights per patient
- First 30-min wake removed

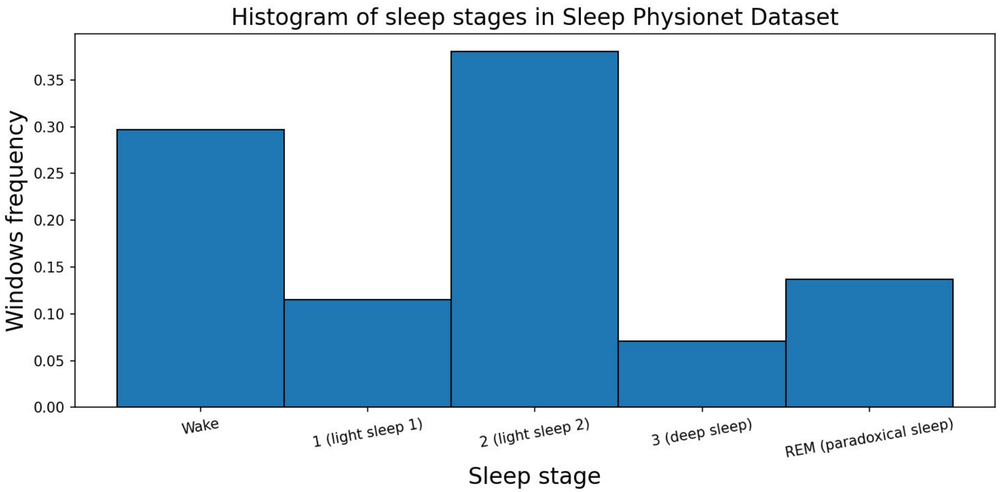
Dataset

Statistics:

- Age
  - Mean: 59
  - Std: 22



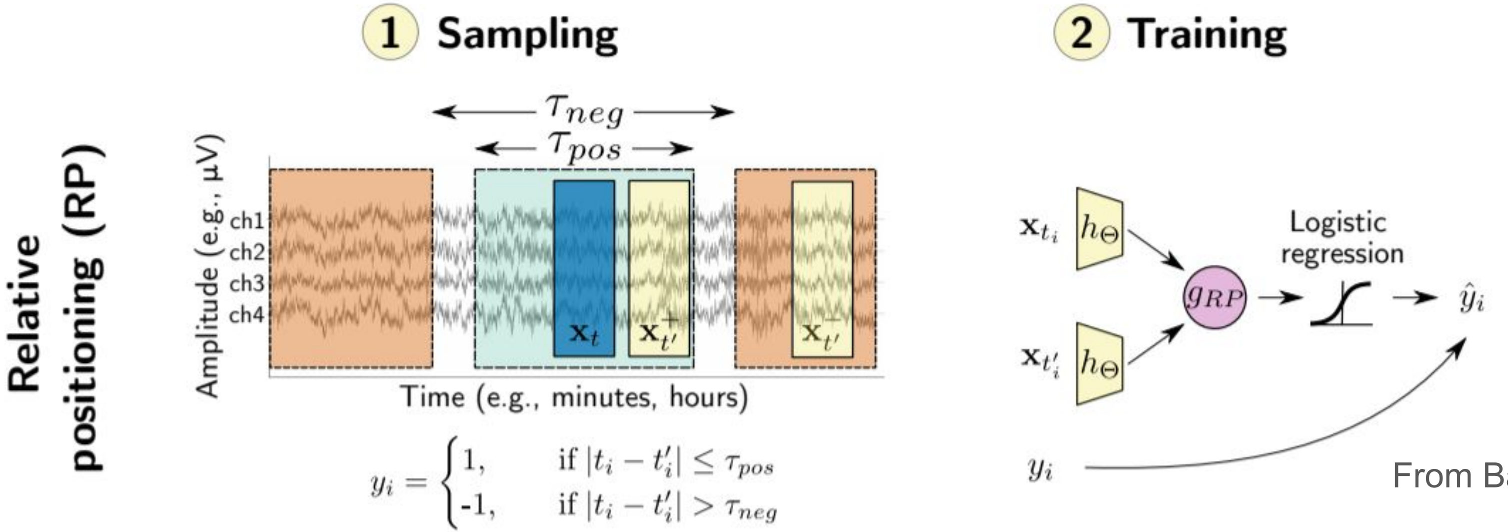
- Sleep stages



Pretext Task (SSL Pre-training)

Contrastive task

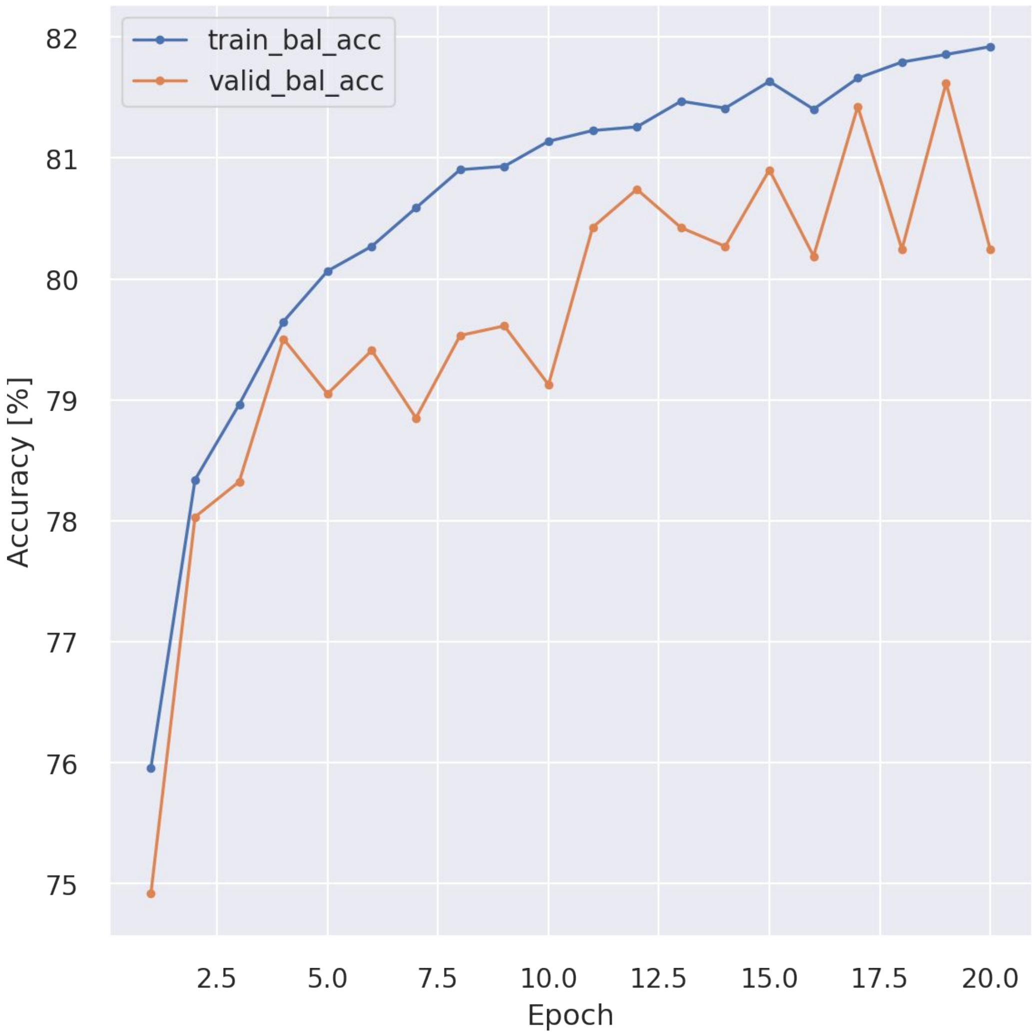
- Sample positive / negative pairs
- Tau_pos = 10 windows / Tau_neg = 100 windows



From Banville et al., 2020

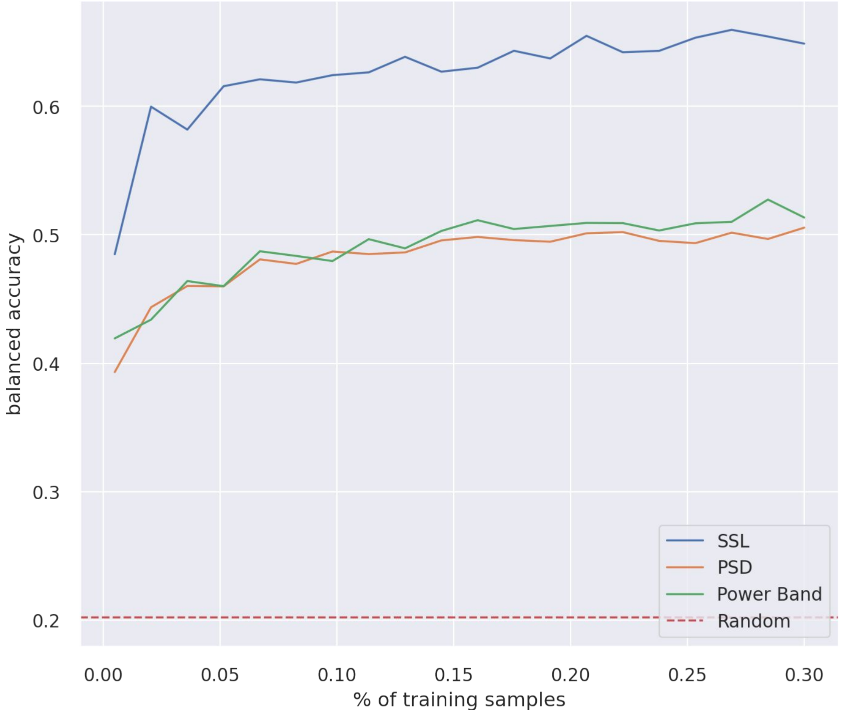
Pretext Task - Training

- Trains successfully in a few epochs
- Reaches 81% validation accuracy

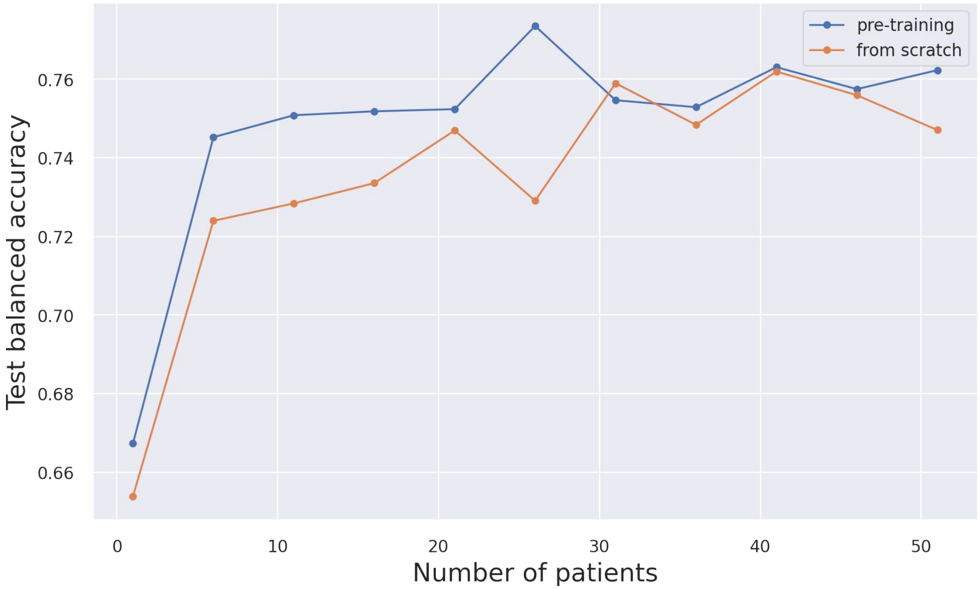


Proof Of Concept - Sleep Stages

SSL compared to baseline with a k-NN trained on the embeddings (k=10, cross validated)

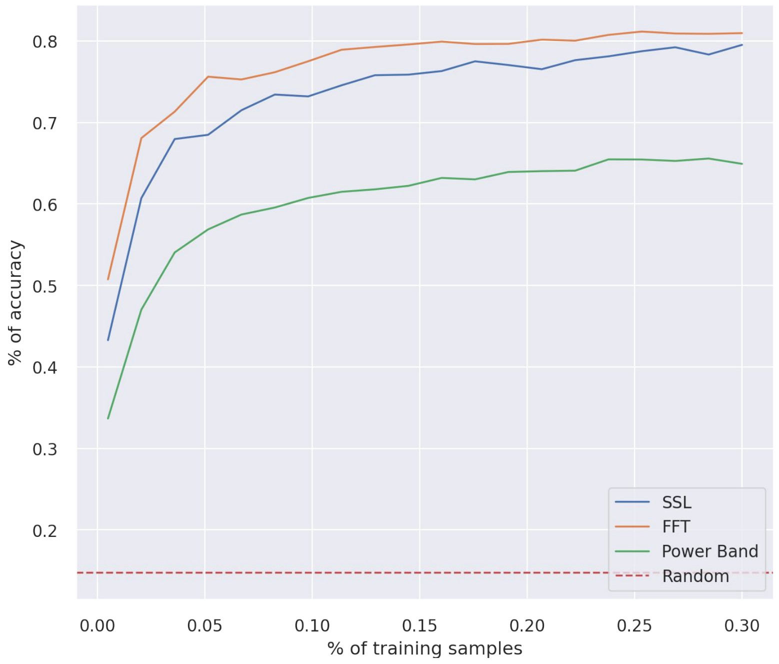


Impact of pre-training with 70% unlabeled data and then fine-tuning with a MLP VS training from scratch

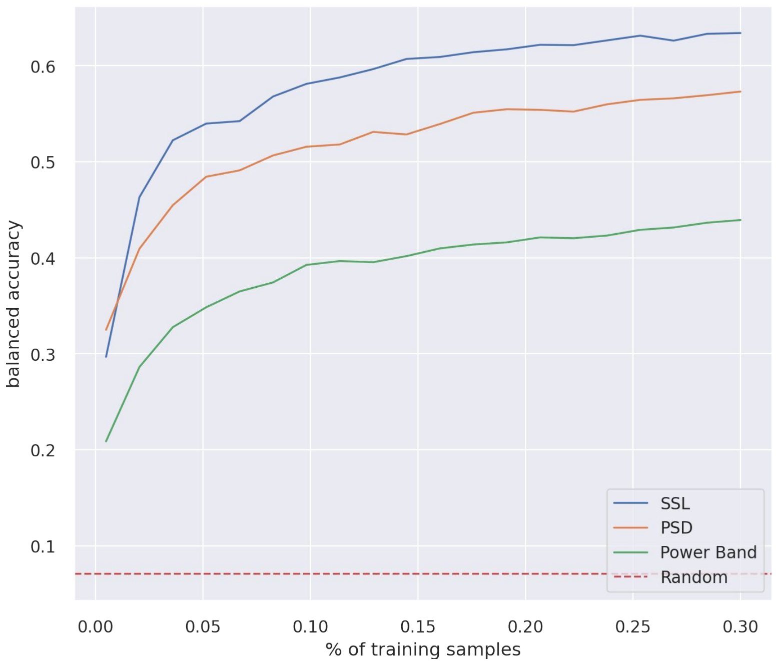


Proof Of Concept - Learned Biases

Patient age

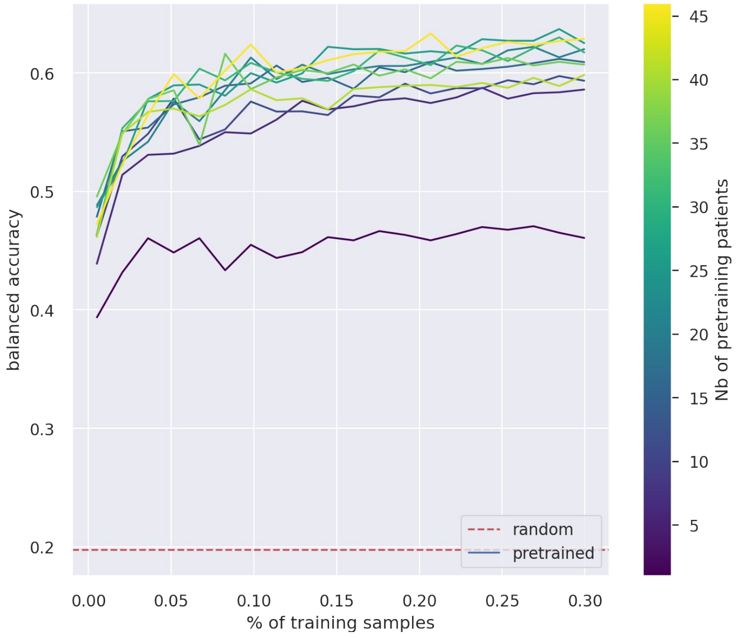


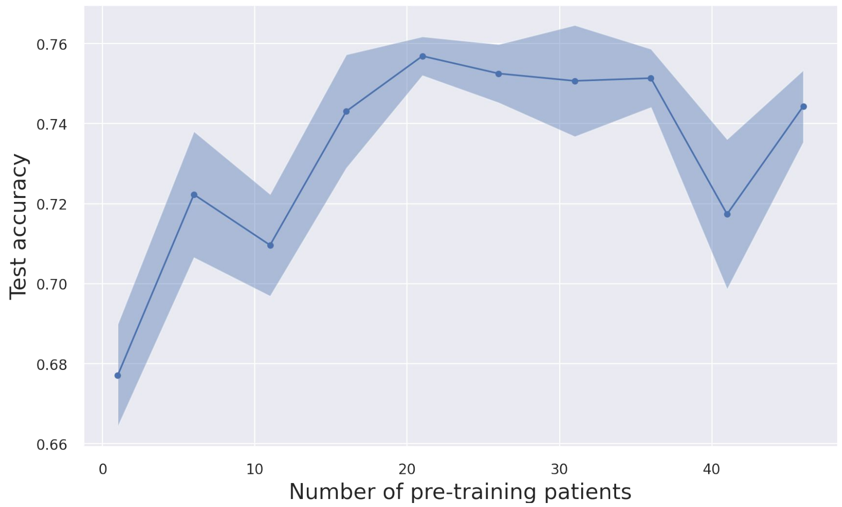
Patient ID



Pre-training Size - Sleep Stages

- Downstream task

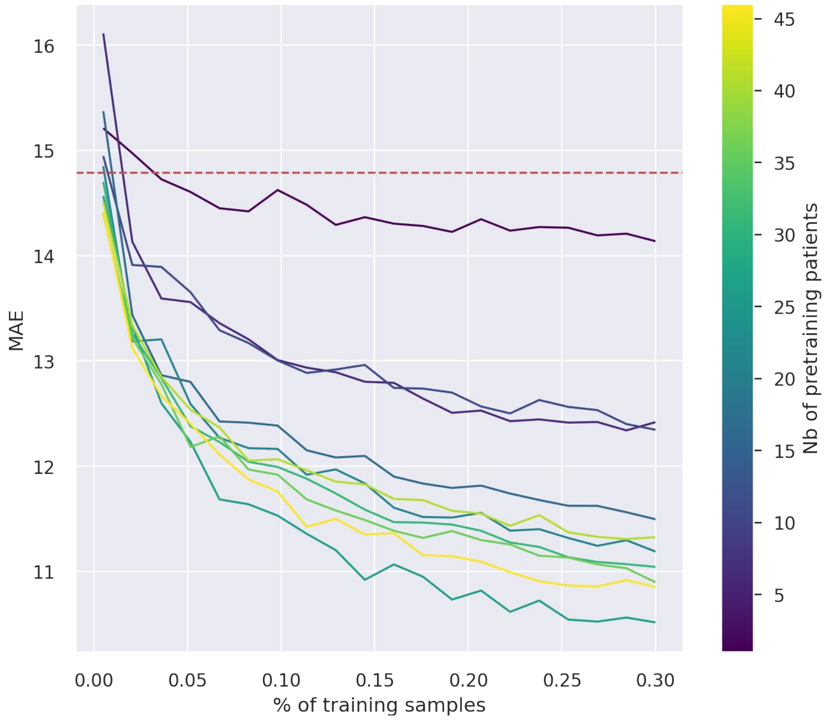




Increasing sets of pre-training patients
(folds not independent)

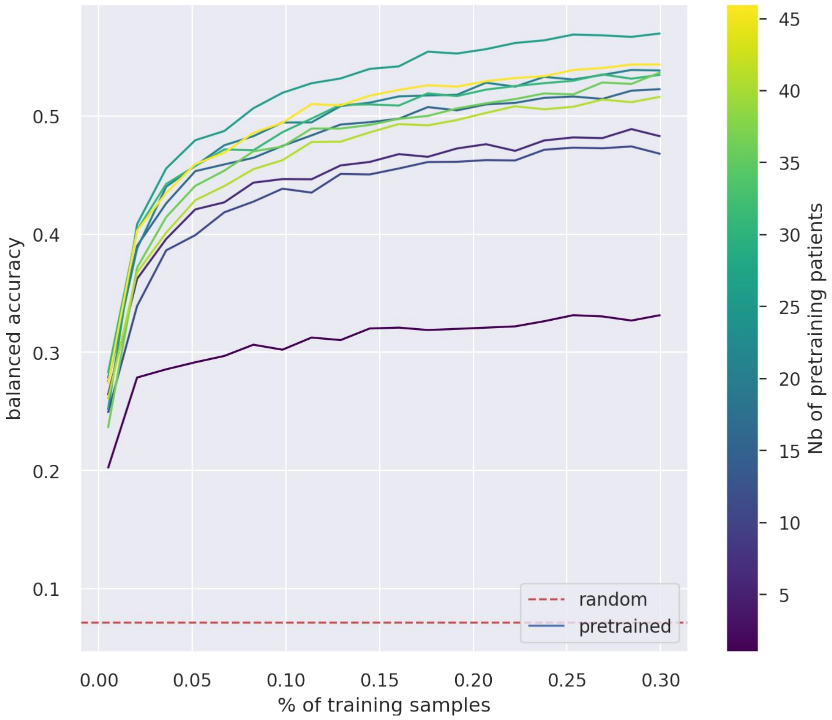
Pre-training Size - Learned Unwanted Features

Patient age



Age distribution: 59 +/- 22

Patient ID



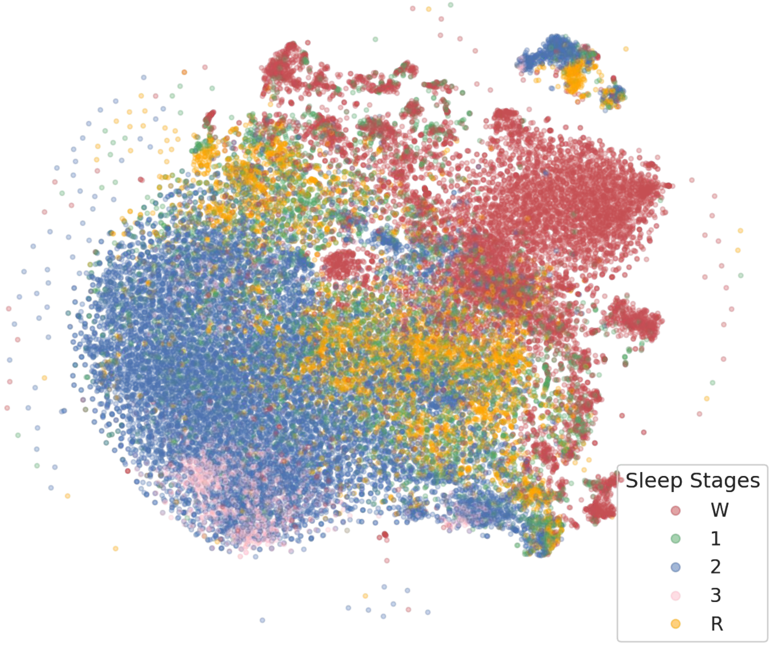
Test on 14 patients

Embeddings Analysis

PSD Embeddings Analysis

PCA

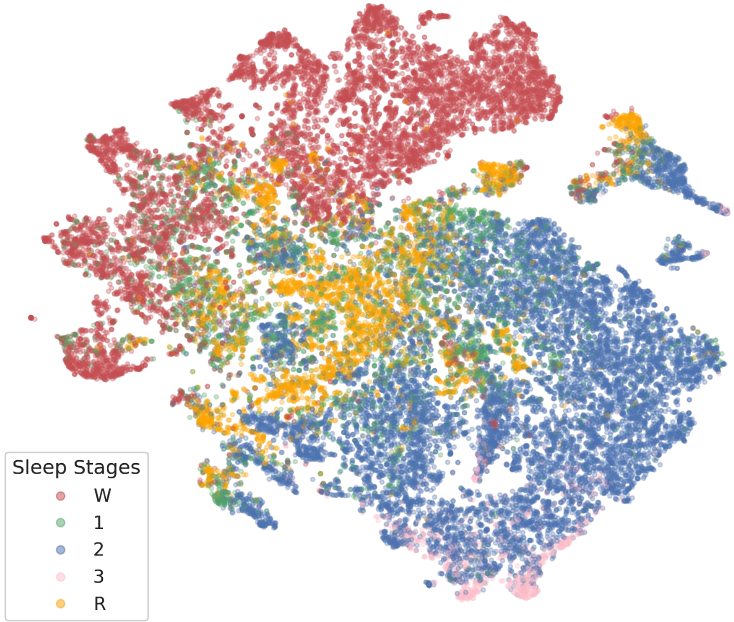
T-SNE



SSL Embeddings Analysis - Sleep Stages

PCA

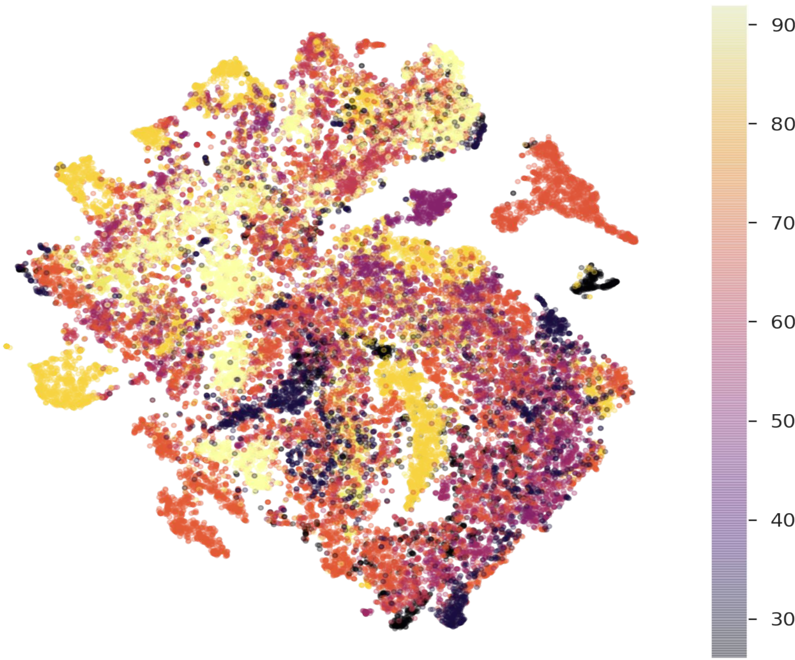
T-SNE



SSL Embeddings Analysis - Patient Age

PCA

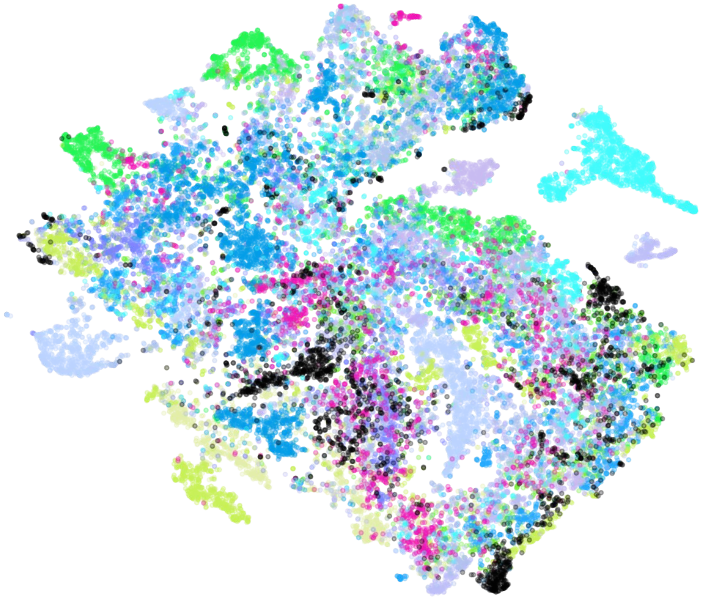
T-SNE



SSL Embeddings Analysis - Patient ID

PCA

T-SNE

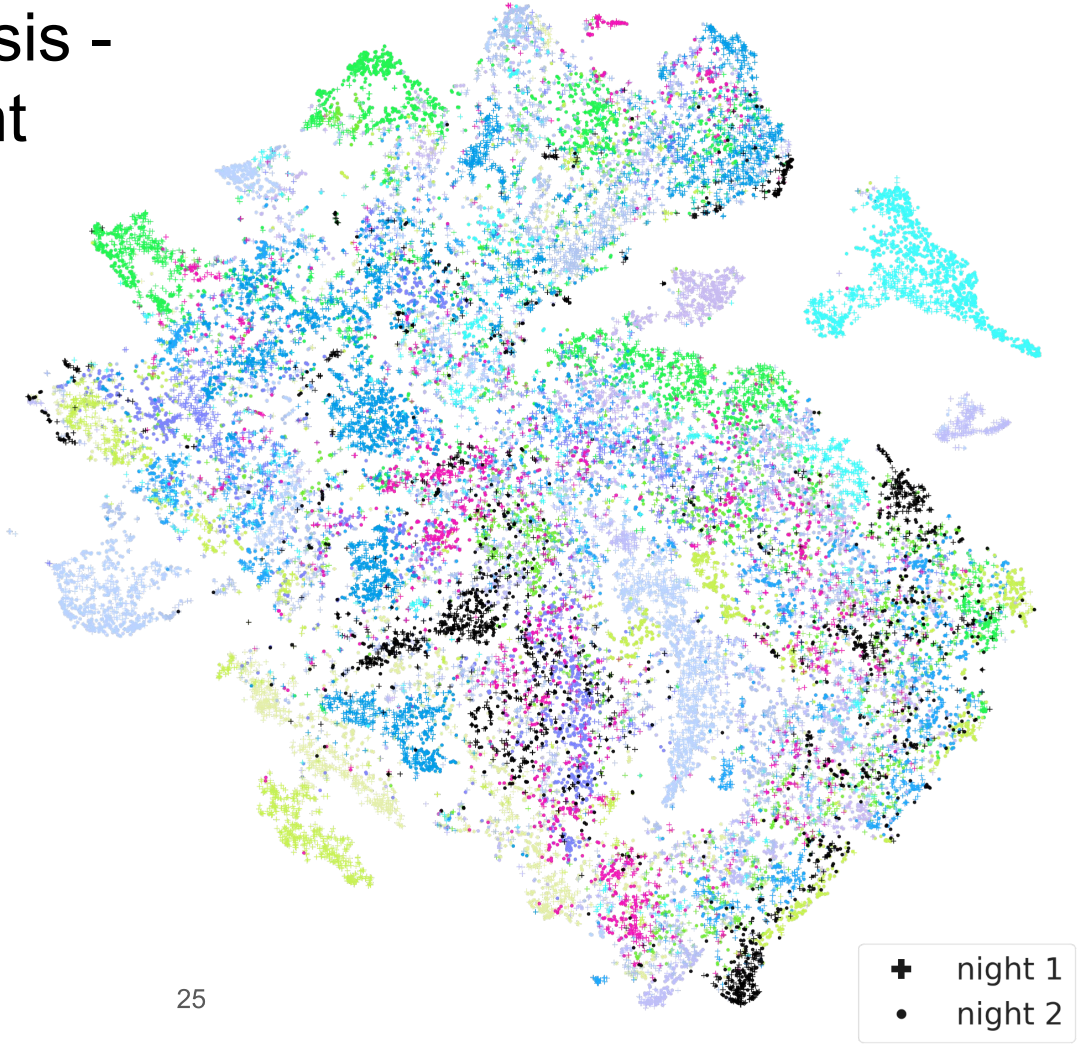


SSL Embeddings Analysis - Patient ID + patient night

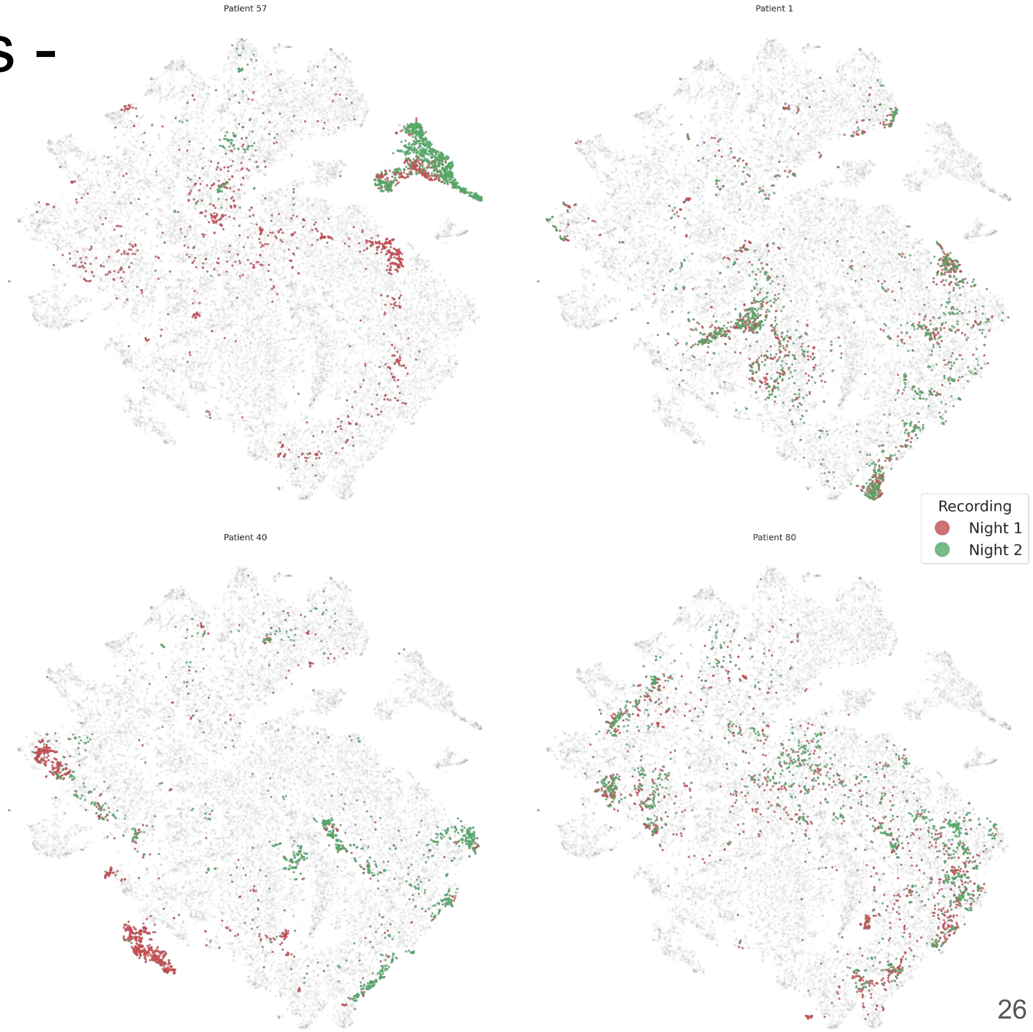
A lot of patient are clustered alone !

But the network nether saw them…

Must be some bias somewhere



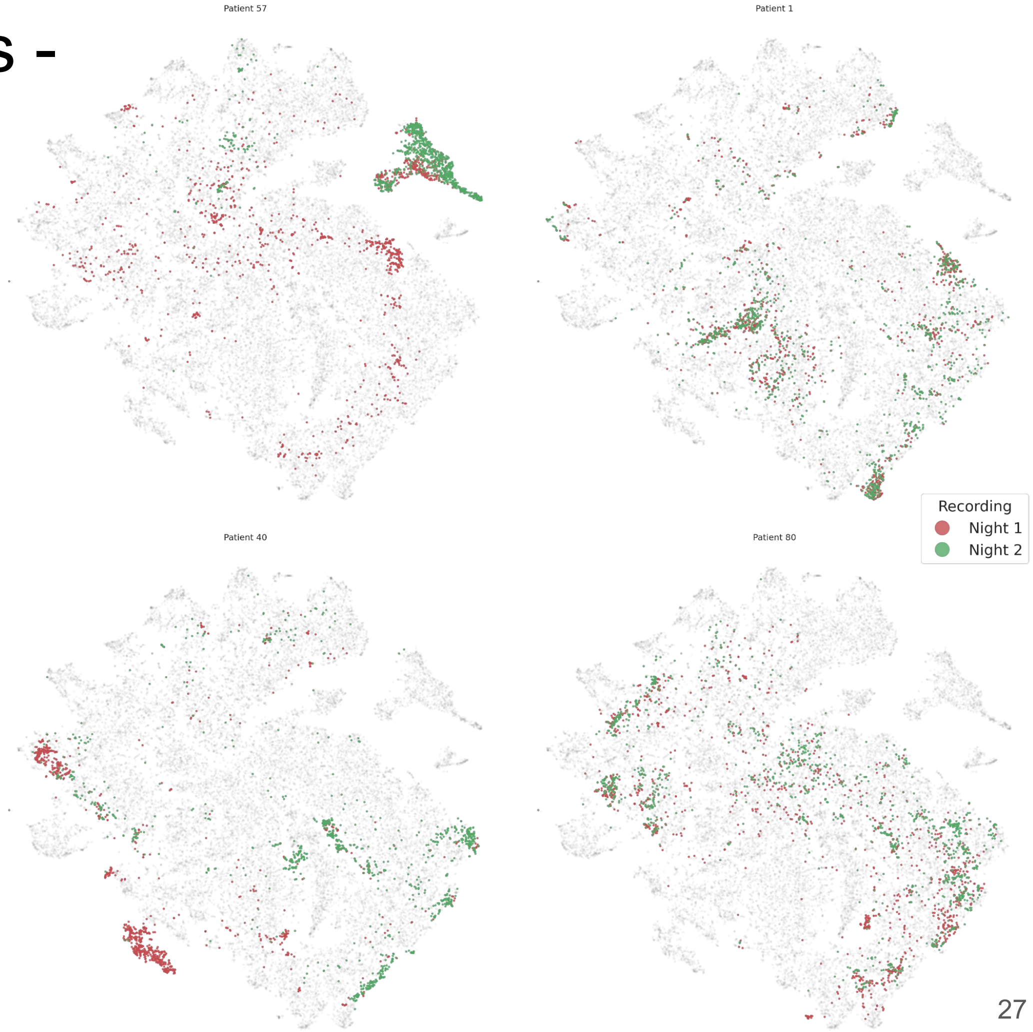
SSL Embeddings Analysis - Patient ID + patient night



SSL Embeddings Analysis - Patient ID + patient night

Spread out → less bias

Tightly clustered → more bias



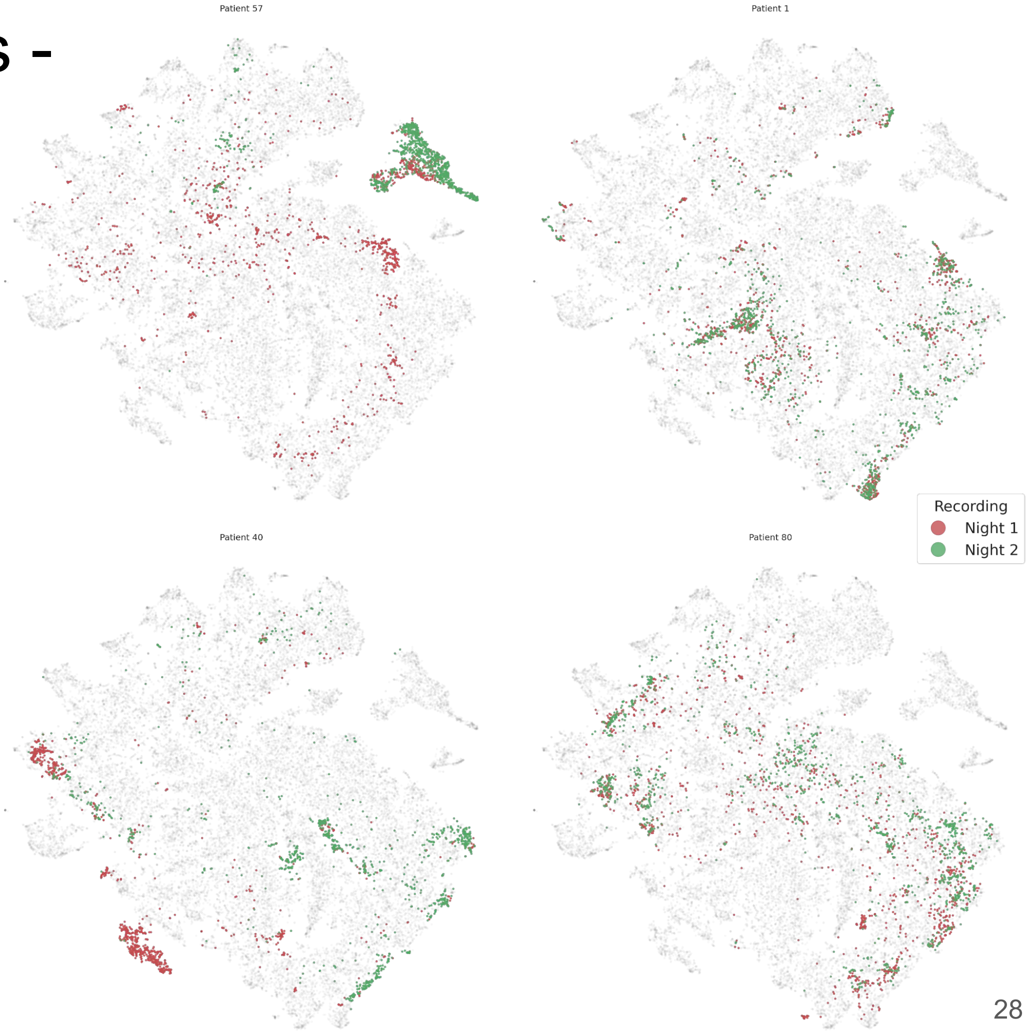
SSL Embeddings Analysis - Patient ID + patient night

Spread out → less bias

Tightly clustered → more bias

Observations :

- If night 1 is tightly clustered, then night 2 is also likely to be tightly clustered.
- If night 1 is spread out, night 2 may also be spread out.



SSL Embeddings Analysis - Patient ID + patient night

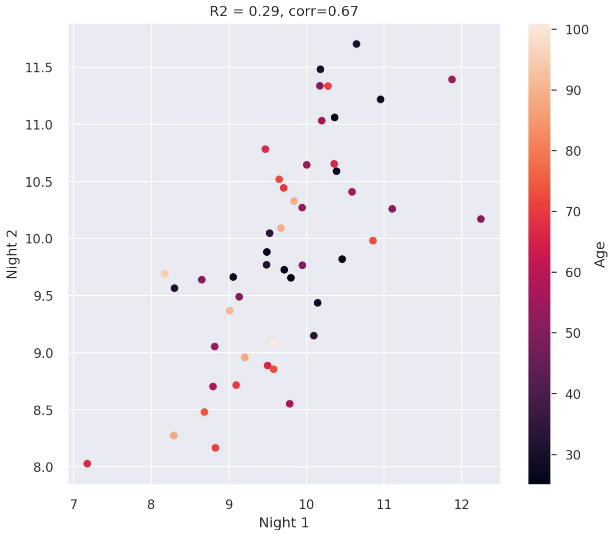
To get a quantitative metric of the spreading :

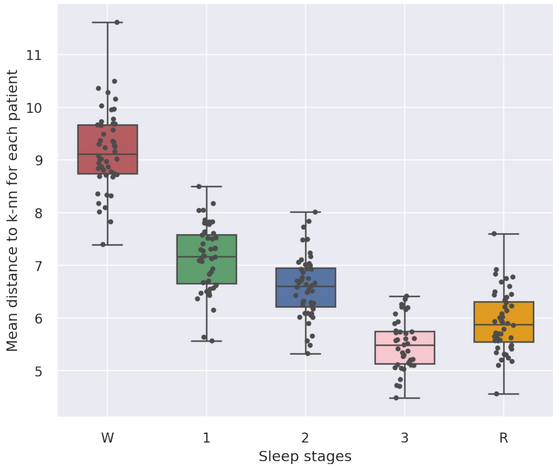
1. Take only certain points (filter on patient, night, sleep stage...).
2. Compute the distance to k-NN for each point (k=20% of # points).
3. Aggregate by patient, night and/or sleep stage.

SSL Embeddings Analysis - Patient ID + patient night

To get a quantitative metric of the spreading :

1. Take only certain points (filter on patient, night, sleep stage...).
2. Compute the distance to k-NN for each point (k=20% of # points).
3. Aggregate by patient, night and/or sleep stage.





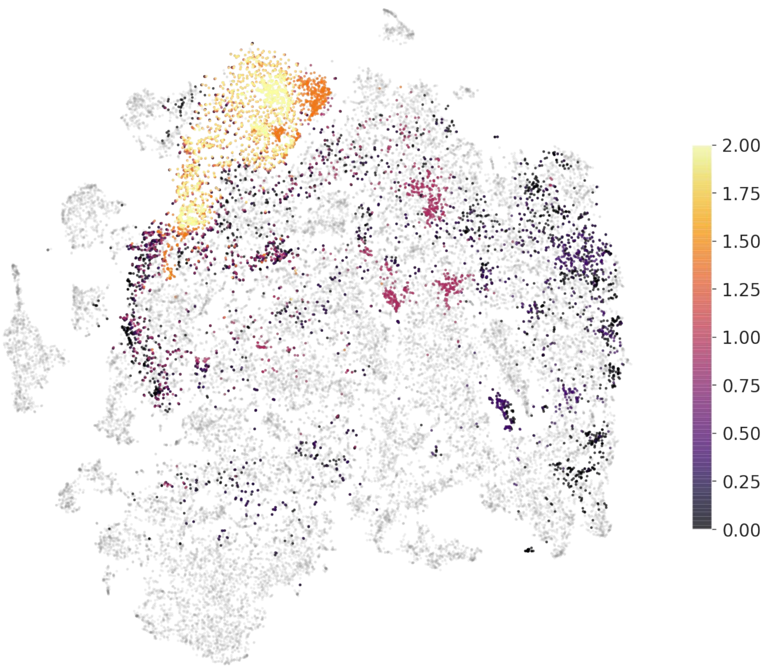
Night 1 and Night 2 correlated (67%)

SSL Embeddings Analysis - Adding controlled noise

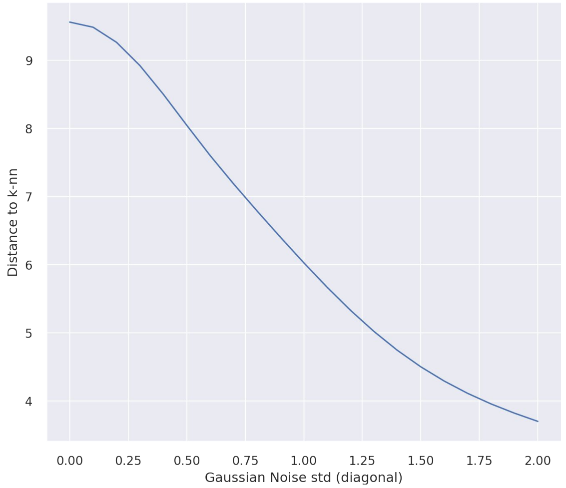
Add a Gaussian noise with std in [0, 2] and fixed seed.

Embed it using the pre-trained encoder to see the evolution of the spreading.

Clustering when adding more and more noise



Spreading when adding noise

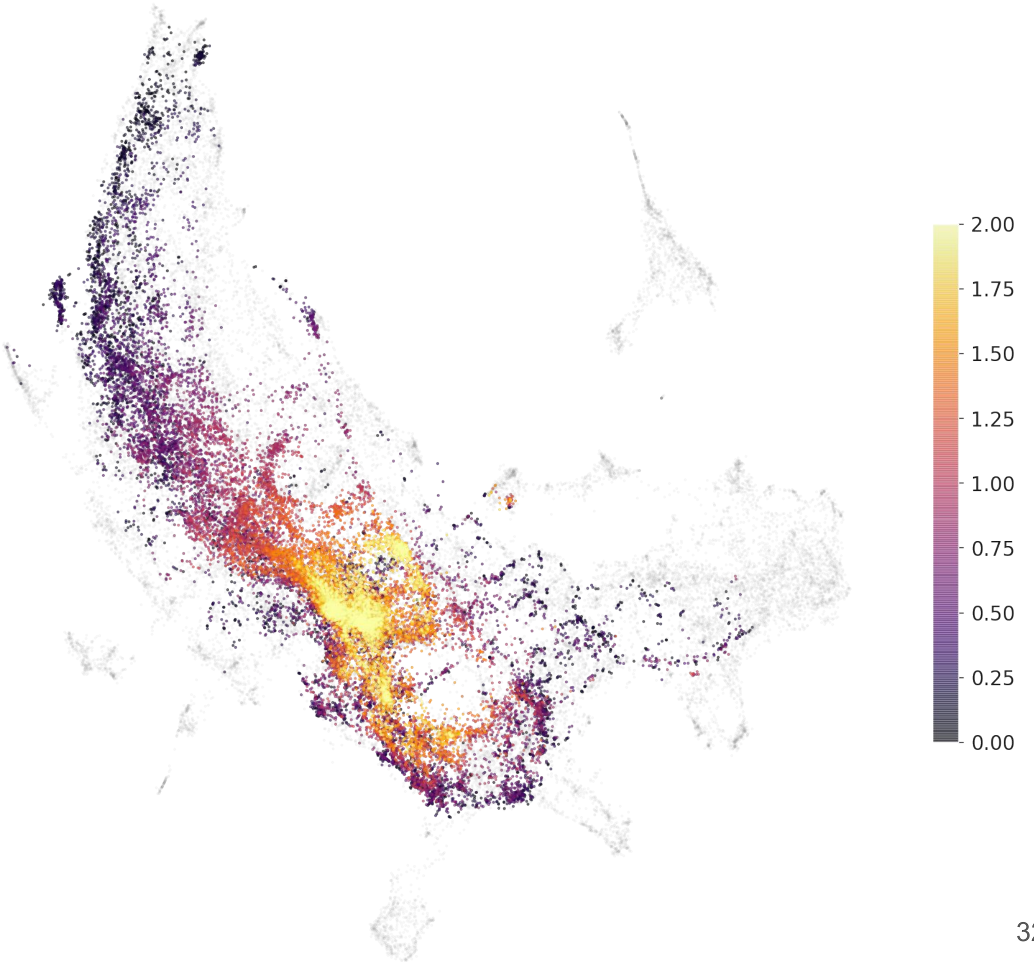


SSL Embeddings Analysis - Adding controlled noise

Add a Gaussian noise with std in [0, 2] and fixed seed.

Embed it using the pre-trained encoder to see the evolution of the spreading.

UMAP of embeddings when adding more and more noise



Conclusion

- Self-supervised learning → pre-train for sleep stage classification
- Main issue → patient-specific features, biases, clusters of patients
- Study of the pre-training size: diminishing returns to scale, plateau
- Limited robustness of the learned embeddings
- SSL may help perform well on EEG data in low-data regimes

→ Analyze biases provided by pretext task

Opening - What To Do Next ?

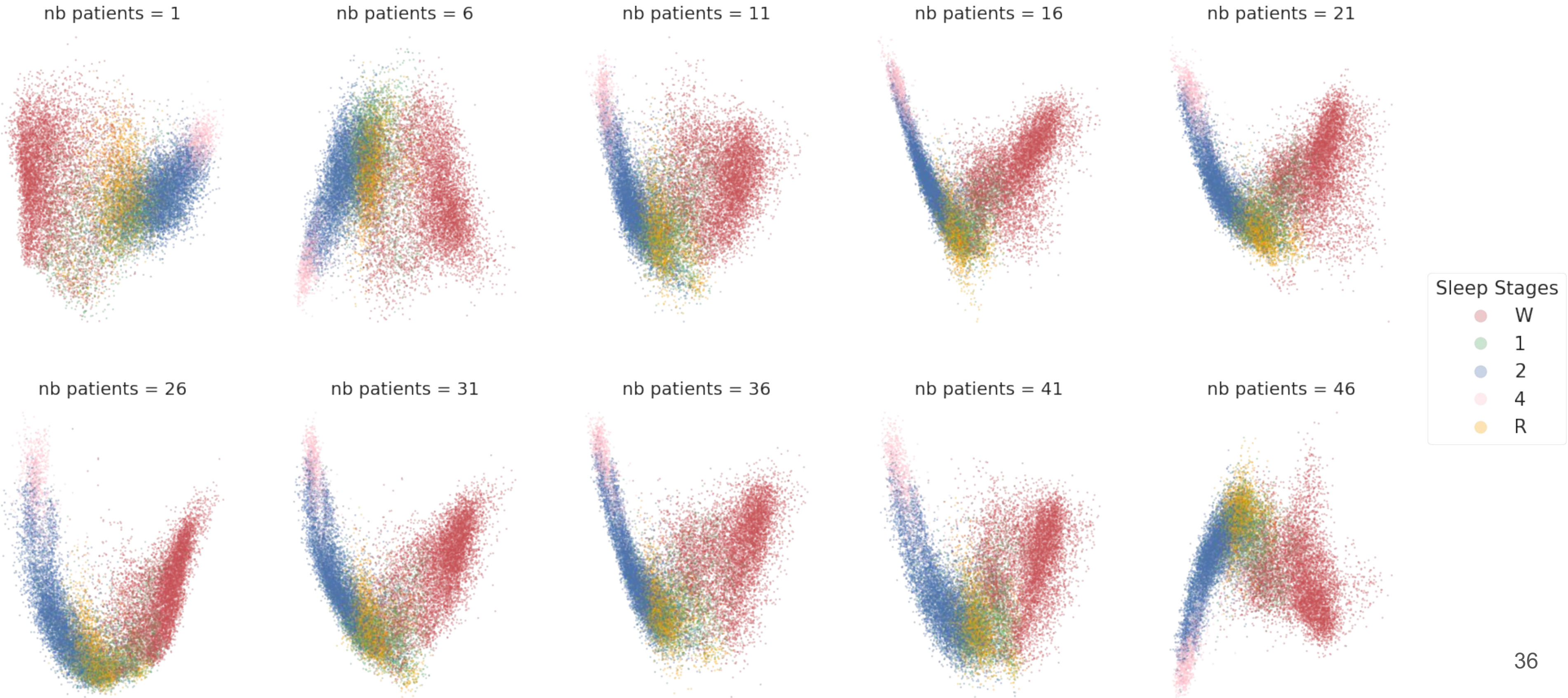
- Find a way to pre-train RNN along the encoder (as in SoTA architectures)
- Find new data augmentations to reduce patient-specific bias
- Study learned biases from a dataset to another or from different cohorts

References

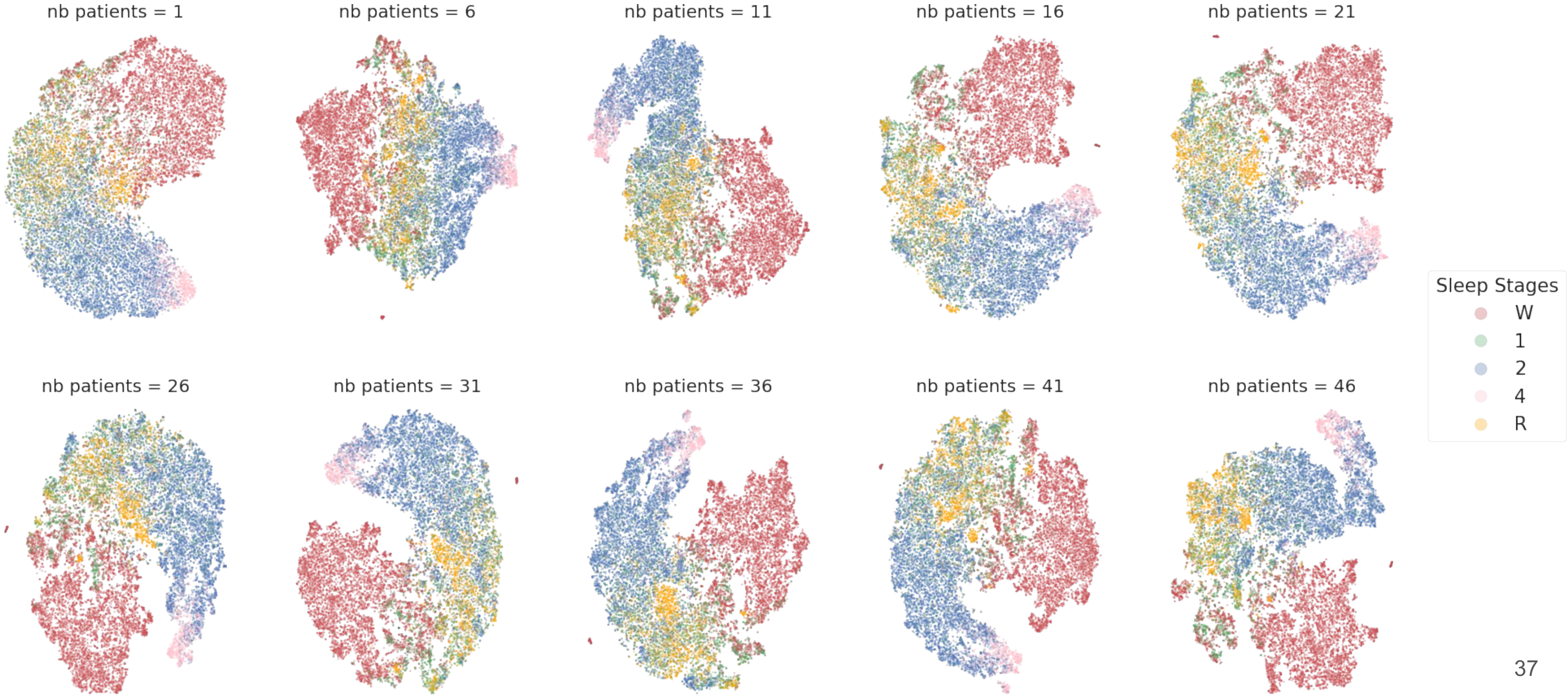
Code shared at: *https://github.com/TheoMoutakanni/TCC-EEG*

[1] Lilian Weng. Self-supervised representation learning. lilianweng.github.io/lil-log, 2019.
[2] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database, 2009.
[3] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction, 2016.
[4] Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification, 2016.
[5] Hubert Banville, Isabela Albuquerque, Aapo Hyvärinen, Graeme Moffat, Denis-Alexander Engemann, and Alexandre Gramfort. Self-supervised representation learning from electroencephalography signals, 2019.
[6] Hubert Banville, Omar Chehab, Aapo Hyvarinen, Denis Engemann, and Alexandre Gramfort. Uncovering the structure of clinical eeg signals with self-supervised learning, 2020.
[7] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. De Vos. Seqsleepnet: End-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging, 2019.
[8] Jens Stephansen, Alexander Olesen, Mads Olsen, Aditya Ambati, Eileen Leary, Hyatt Moore,Oscar Carrillo, Ling Lin, Fang Han, Han Yan, Yun Sun, Yves Dauvilliers, Sabine Scholz, Lucie Barateau, Birgit Högl, Ambra Stefani, Seung Hong, Tae Kim, Fabio Pizza, and Emmanuel Mignot. Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy, 2018.
[9] Mathias Perslev, Michael Hejselbak Jensen, Sune Darkner, Poul Jørgen Jennum, and ChristianIgel. U-time: A fully convolutional network for time series segmentation applied to sleep staging, 2019.
[10] Antoine Guillot and Valentin Thorey. Robustsleepnet: Transfer learning for automated sleep staging at scale, 2021.
[11] S. Chambon, M. N. Galtier, and A. Gramfort. Domain adaptation with optimal transport improves eeg sleep stage classifiers, 2018.
[12] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks, 2016.
[13] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q.Weinberger, 2014.
[14] Mohsen Ghafoorian, Alireza Mehrtash, Tina Kapur, Nico Karssemeijer, Elena Marchiori, Mehran Pesteie, Charles R. G. Guttmann, Frank-Erik de Leeuw, Clare M. Tempany, Bramvan Ginneken, Andriy Fedorov, Purang Abolmaesumi, Bram Platel, and William M. Wells. Transfer learning for domain adaptation in mri: Application in brain lesion segmentation, 2017.
[15] Ary L. Goldberger, Luis A. N. Amaral, Leon Glass, Jeffrey M. Hausdorff, Plamen Ch. Ivanov,Roger G. Mark, Joseph E. Mietus, George B. Moody, Chung-Kang Peng, and H. Eugene Stanley. Physiobank, physiotoolkit, and physionet, 2000.

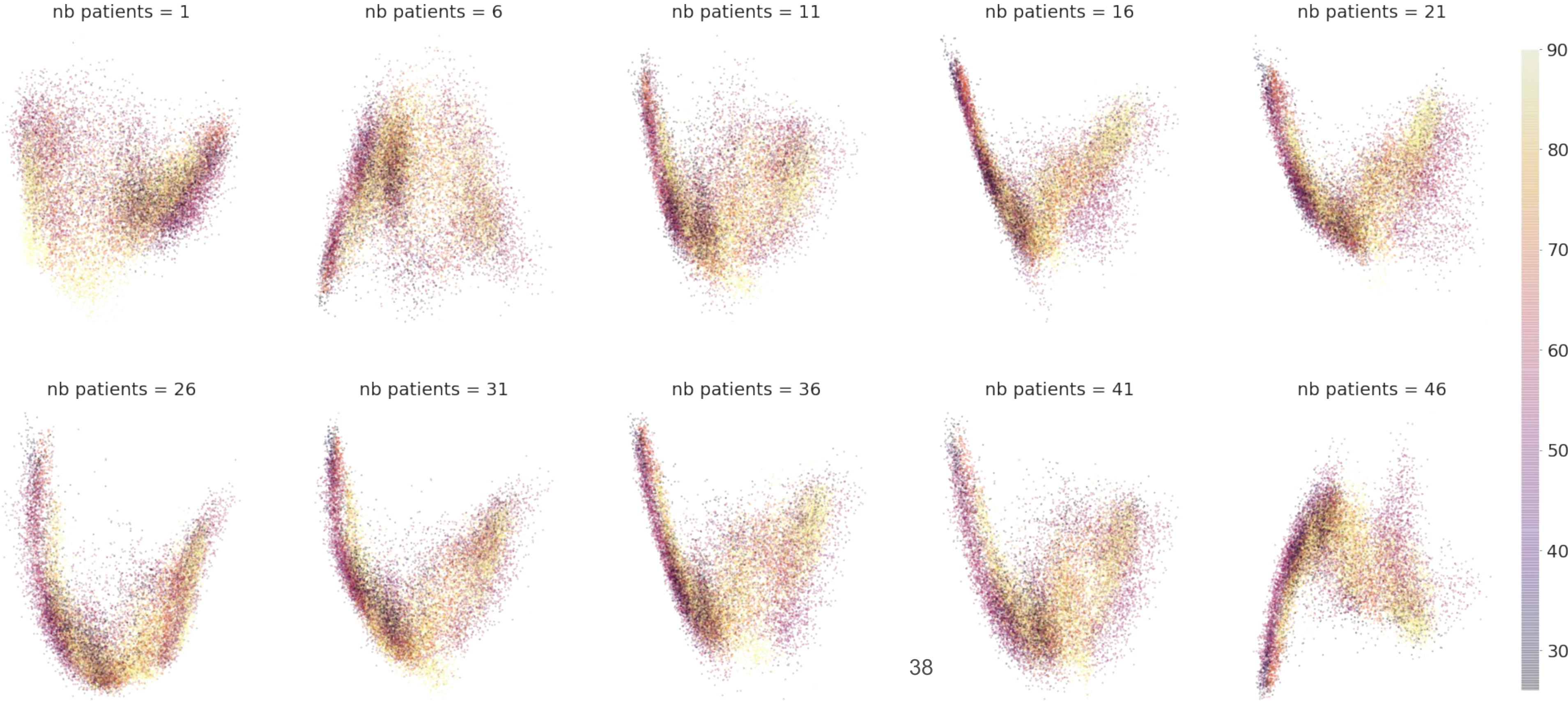
Evolution of Sleep Stages Embeddings - PCA



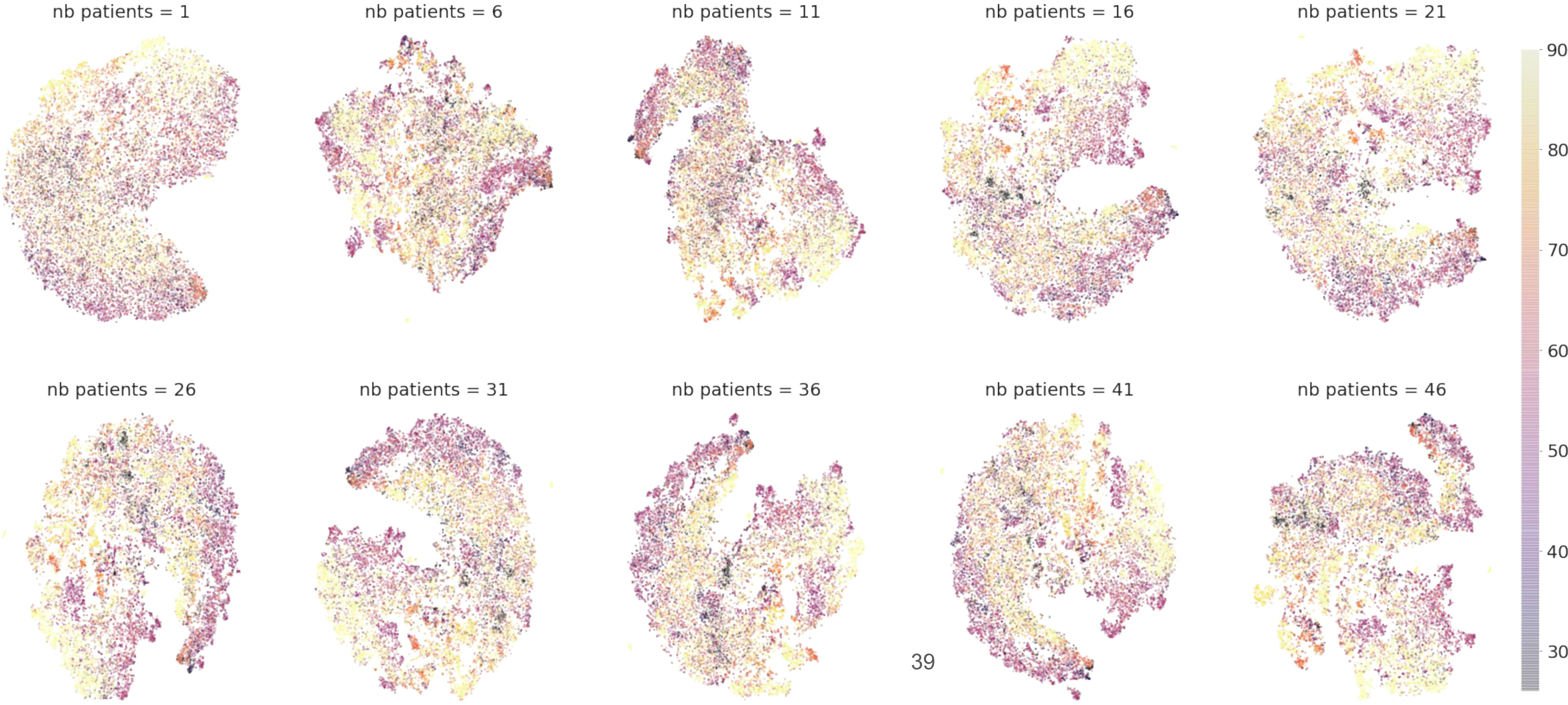
Evolution of Sleep Stages Embeddings - T-SNE



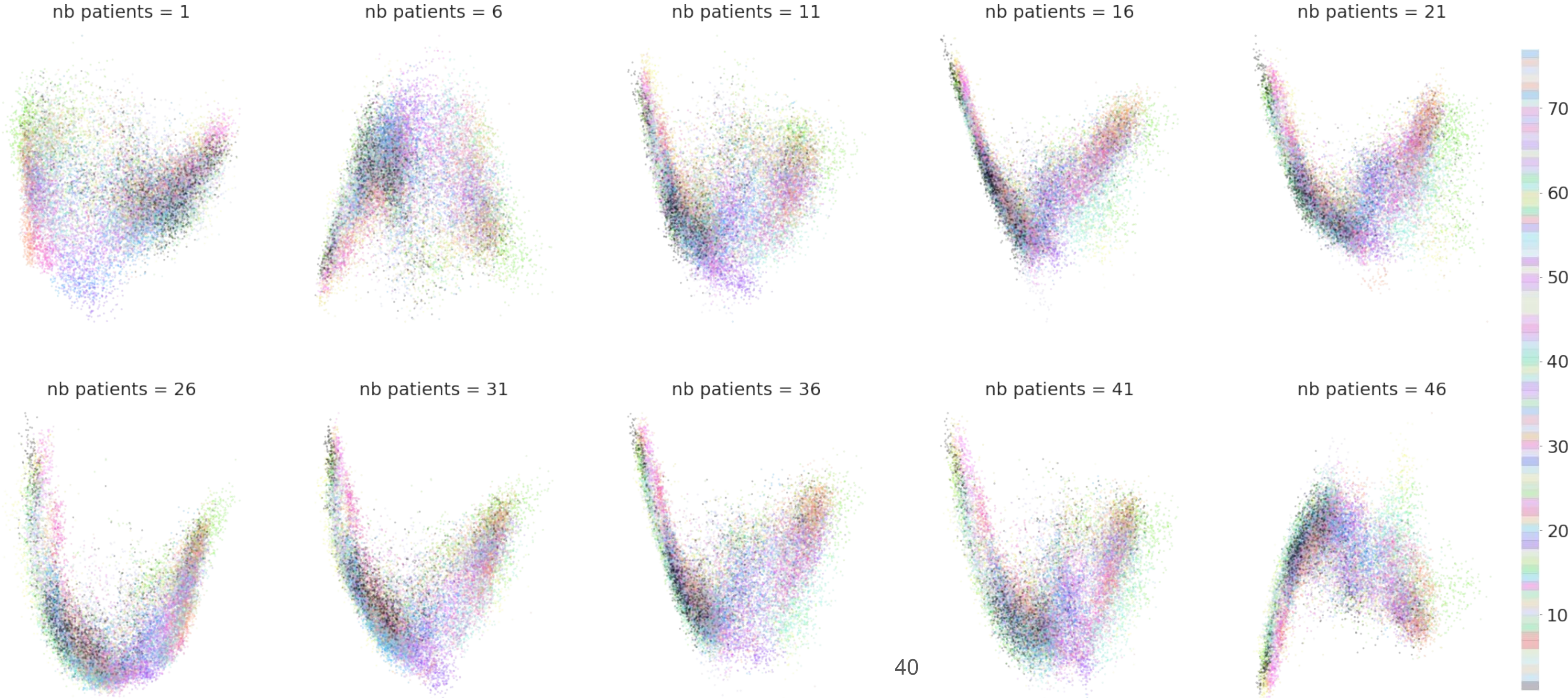
Evolution of Patient Ages Embeddings - PCA



Evolution of Patient Ages Embeddings - T-SNE



Evolution of Patient ID Embeddings - PCA



Evolution of Patient ID Embeddings - T-SNE