SEMI-SUPERVISED SLEEP-STAGE SCORING BASED ON SINGLE CHANNEL EEG

A. M. Munk[1,*], *K. V. Olesen*[1,*], *S. W. Gangstad*[1,2,*], *L. K. Hansen*[1]

[1] Technical University of Denmark, Department of Applied Mathematics and Computer Science, DK-2800 Kongens Lyngby, Denmark
[2] UNEEG^{TM} medical, DK-3540 Lynge, Denmark

ABSTRACT

The field of automatic sleep stage classification based on EEG has enjoyed substantial attention during the last decade, which has resulted in several supervised classification algorithms with highly encouraging performance. Such supervised machine learning algorithms require large training sets that have been manually labelled, and are time- and resource-consuming to acquire. Here we present a semi-supervised approach that can learn to distinguish the sleep stages from a one-night data set where only a fraction has been manually labelled. We show that for fractions larger than 50%, our semi-supervised approach performs as good as a similar, fully-supervised model.

***Index Terms*—** EEG, semi-supervised learning, sleep stage scoring, non-negative matrix factorizaton, generalizable Gaussian mixture model

1. INTRODUCTION

Analyzing the temporal evolution of sleep stages is an important diagnostic tool in sleep medicine. Manually labeling the stages in a polysomnogram (PSG) for subsequent analysis is unfortunately highly time- and resource-consuming. The automation of PSG-labelling using machine learning techniques has therefore enjoyed substantial attention during the last decade. Excellent results have been achieved using sophisticated supervised machine learning algorithms [1] [2], and the problem of automatic sleep stage classification is by many considered solved. However, a supervised classification algorithm with a low generalization error requires a large set of labelled examples for training. In order to save time and resources, it would be beneficial if a classification algorithm could be trained on a data set where only a fraction of the training examples has been labelled. However, it is important that the performance of such a classifier still matches that of an algorithm that was trained using a fully labelled training set.

In this work, we compare the performance of a semi-supervised Gaussian Mixture Model (SS-GMM) against a fully supervised Gaussian Mixture Model (FS-GMM). The classification strategy is conceptually simple. A model is trained such that a data point is first given a set of probabilities to belong to each cluster component, much like the *responsibility* in the traditional GMM framework. Furthermore, the probability for each cluster to represent each class is learned using the information in the labelled part of the data set. A test point is classified according to the posterior probability governed by the responsibilities and the cluster-class mapping.

All models in this work were subject specific, meaning that they were trained and tested on each subject in the experimental data set separately. This is contrary to popular practice in the field [3] [4] [5] [6]. In contrast to population models, our focus on subject specific modeling allows us to account for variation in individual brain dynamics and sleep patterns.

Furthermore, the models were trained and validated on EEG that was recorded before the EEG in the test set. This mimics the use case where an initial, labelled recording is acquired from a subject, from which a model can be trained and validated. The model can then be deployed to automatically label subsequent EEG recordings from the same subject.

2. EXPERIMENTAL DATA

The experimental data in this study consists of the *SC* recordings in the *sleep-edfx*-data base that is publicly available on *PhysioNet.org* [7] [8] [9]. The data contains a pair of PSG recordings from two consecutive nights from 19 healthy subjects (10F + 9M) accompanied with expert-annotated sleep stage labels. An additional subject is present in the online repository, for whom only a single night was successfully recorded. This subject was excluded from the study. The expert annotated labels were scored according to the Rechtschaffen and Kales (R&K) classification rules [10].

* These authors contributed equally to this work and share the first-authorship. The authors would like to thank Jonas Duun-Henriksen for important comments on the manuscript. LKH is supported by Innovation Fund Denmark - the Danish Center for Big Data Analytics Driven Innovation. SWG is supported by Innovation Fund Denmark under the project Ultra-long term subcutaneous EEG monitoring of brain function and disease. Source code: https://github.com/ammunk/SemiSupSleep

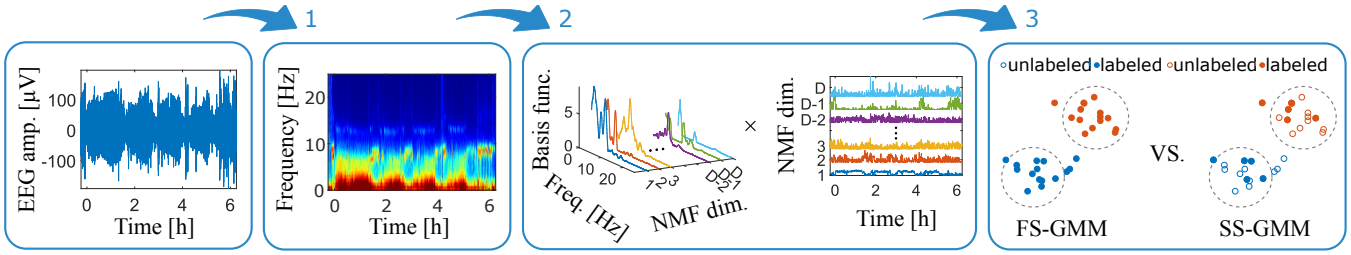

Fig. 1. Conceptual illustration of the classification method. In the first step, the sleep EEG is converted to a multi-tapered spectrogram. In the second step, the spectrogram is decomposed into two matrices, $\boldsymbol{W}$ and $\boldsymbol{H}$, using NMF. In the third step, the weight values in the matrix $\boldsymbol{H}$ are scaled and used as features for a GMM-classifier. Two versions of the classifier are compared, a fully supervised model and a semi-supervised model. This figure was made with data from the first subject's first night.

3. METHODS

The recording from the first night was used as training and validation set, and the recording from the second night was used as test set. The Pz-Oz channel, which was sampled at 100 Hz, was used as it has proven successful for sleep stage classification in previous work, [1] [11]. The period of interest for the classification problem was extracted from the PSG recording using the labels. This period was defined as 15 min prior to sleep onset (defined as first epoch of N1) in the evening until 15 minutes after the last sleeping epoch in the morning. The rest of the recording was discarded. Epochs of the sleeping EEG that were not scored or scored as *Body movement* were excluded from the data set. Furthermore, the stages N1 and N4 were in some cases too small or even non-existing, so the N1 and N2 stages were merged into a single "light sleep"-class, and the stages N3 and N4 were merged into a single "deep sleep"-class. This resulted in the four classes *Awake, REM, Light sleep* and *Deep sleep*, as also done in [12] [13] [14].

The sleep EEG in the training and test sets were decomposed into the time-frequency domain using multi-tapered spectrograms, based on the Discrete Prolate Spheroidal Sequence (DPSS) taper functions [15]. The advantage of multi-tapered spectrograms compared to single-tapered spectrograms is the reduction of both the bias and the variance of the spectral estimate. The spectrogram matrix, $\boldsymbol{Spec} \in \mathbb{R}_+^{M \times N}$, was computed using the Chronux MATLAB toolbox [16] [17] using 29 tapers, and in 30-second, non-overlapping windows. $\boldsymbol{Spec}$ was log-transformed as $\boldsymbol{S} = \ln(\boldsymbol{Spec} + 1)$. An example of $\boldsymbol{S}$ is shown in Fig. 1. One column of $\boldsymbol{S}$ now corresponds to one observation in the data set (one epoch of 30 seconds). In order to reduce the dimensionality, $\boldsymbol{S}$ was approximated using Non-Negative Matrix Factorization [18], which seeks to minimize the Frobenius norm between the original matrix and the matrix approximation

$$\begin{aligned} \min \quad & \|\boldsymbol{S} - \boldsymbol{WH}\|_F^2, \\ s.t. \quad & \boldsymbol{W} \in \mathbb{R}_+^{M \times D}, \quad \boldsymbol{H} \in \mathbb{R}_+^{D \times N}. \end{aligned}$$

Minimization of the Frobenius norm is equivalent to assuming the entries of $\boldsymbol{S}$ are independently and identically distributed with Gaussian noise around the means $\boldsymbol{WH}$ [19]. This corresponds to maximizing the likelihood function, $p(\boldsymbol{S}|\boldsymbol{WH}, \sigma^2) = \mathcal{N}(\boldsymbol{S}|\boldsymbol{WH}, \sigma^2\boldsymbol{I})$, where $\mathcal{N}(\cdot|\cdot,\cdot)$ is the normal distribution

$$\mathcal{N}(\boldsymbol{S}|\boldsymbol{WH}, \sigma^2\boldsymbol{I}) = \prod_{i,j} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{\left(s_{ij} - [\boldsymbol{WH}]_{ij}\right)^2}{2\sigma^2} \right].$$

The optimal value of the common dimension D is computed by minimizing the Bayesian Information Criteria (BIC) [19]

$$\text{BIC} = -\ln p(\boldsymbol{S}|\boldsymbol{WH}, \sigma^2) + D(M+N)\ln(M \times N). \quad (1)$$

The columns of $\boldsymbol{W}$ are basis functions that can be interpreted as band pass filters. Columns in $\boldsymbol{H}$ are weights indicating how much of each basis functions in $\boldsymbol{W}$ is needed to represent the spectral contents in each epoch in the data set. One column in $\boldsymbol{H}$ corresponds to one observation. The basis functions of $\boldsymbol{W}$ were learned from the first night, and the same functions were used when estimating the weights for the second night. The matrix $\boldsymbol{X} = \boldsymbol{H}^{\mathrm{T}}$ was scaled column-wise by the 99th percentile across the entire training set, and was used as input for the classifier.

The classification model was build such that it can be trained with a variable amount of labelled data. Let the entire training set $\mathcal{D}$ be composed of a labelled and an unlabelled subset, $\mathcal{D} = \mathcal{D}^l \cup \mathcal{D}^u$. Here, $\mathcal{D}^l$ consists of pairs of data observations and their respective labels $(\boldsymbol{x}^l, \boldsymbol{y})$, whereas $\mathcal{D}^u$ consists of data observations $\boldsymbol{x}^u$ only. The relative size of the two subsets is described by the fraction $f = |\mathcal{D}^l|/|\mathcal{D}|$. If the model was trained using all the labels in the training set, such that $|\mathcal{D}^u| = 0$ and $f = 1$, the model is fully supervised. Conversely, if the model is trained using only some of the labels in the training set, such that $0 < f < 1$, the model is semi-supervised. The classification model, which is inspired by [20], is a generalizable Gaussian Mixture Model (GMM) with $k \in \{1, 2, \ldots, K\}$ mixture components that was optimized using the EM algorithm. The classifier models the posterior class distribution $P(c|\boldsymbol{x})$, where $c \in \{1, 2, \ldots, C\}$

is one of C mutually exclusive classes. The posterior distribution was learned through the complete generative distribution of data $\mathcal{D}$, $p(\mathcal{D}|\boldsymbol{\theta}) = p(\mathcal{D}^l|\boldsymbol{\theta})p(\mathcal{D}^u|\boldsymbol{\theta})$, assuming independence between unlabelled, $\mathcal{D}^u$, and labelled, $\mathcal{D}^l$, data. Using the GMM framework allowed for an optimization scheme in terms of the log-likelihood,

$$\ln p(\mathcal{D}|\boldsymbol{\theta}) = \ln \sum_{\boldsymbol{Z}^l} p(\boldsymbol{X}^l, \boldsymbol{Y}, \boldsymbol{Z}^l|\boldsymbol{\theta}) + \ln \sum_{\boldsymbol{Z}^u} p(\boldsymbol{X}^u, \boldsymbol{Z}^u|\boldsymbol{\theta}),$$

where $\boldsymbol{Z} \in \mathbb{R}^{N\times K} = \left[(\boldsymbol{Z}^l)^{\mathrm{T}}, (\boldsymbol{Z}^u)^{\mathrm{T}}\right]^{\mathrm{T}}$ denotes the unknown latent variables, with mutually exclusive row values and $\boldsymbol{\theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$. By further assuming observations to be i.i.d. as well as classes to be only conditionally dependent on mixture components, the authors in [20] use the following density for single observations:

$$p(\boldsymbol{x}^l, \boldsymbol{y}|\boldsymbol{\theta}) = \sum_{\boldsymbol{z}^l} P(\boldsymbol{y}|\boldsymbol{z}^l) p(\boldsymbol{x}^l|\boldsymbol{z}^l, \boldsymbol{\theta}) P(\boldsymbol{z}^l, \boldsymbol{\theta}), \tag{2a}$$

$$p(\boldsymbol{x}^u|\boldsymbol{\theta}) = \sum_{\boldsymbol{z}^u} p(\boldsymbol{x}^u|\boldsymbol{z}^u, \boldsymbol{\theta}) P(\boldsymbol{z}^u, \boldsymbol{\theta}). \tag{2b}$$

The latent conditional distribution of $\boldsymbol{x}$ can be written as

$$p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\theta}) = \prod_{k=1}^{K} p(\boldsymbol{x}|k)^{z_k},$$

where $p(\boldsymbol{x}|k) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. The final class posterior is thus given by

$$P(c|\boldsymbol{x}) = \sum_k P(c, k|\boldsymbol{x}) \propto \sum_k P(c|k) p(\boldsymbol{x}|k) P(k),$$

where $P(c|k)$ is a probability table, which we seek to estimate along with $\boldsymbol{\theta}$.

In contrast to [20], we propose a different implementation of the density of the unlabelled data observations in Eq. (2b). This implementation proved to be more computationally robust on our data set. The proposed density is derived by introducing $\boldsymbol{B}$ as the latent class association of $\boldsymbol{x}^u$, such that Eq. (2b) now becomes

$$\tilde{p}(\boldsymbol{x}^u|\boldsymbol{\theta}) = \sum_{\boldsymbol{b},\boldsymbol{z}} \prod_{k=1}^{K} p(\boldsymbol{x}^u|k)^{z_k} P(k)^{z_k} \prod_{c=1}^{C} P(c|k)^{b_c z_k}.$$

Notice the density now includes the class cluster posterior $P(c|k)$, that models the probability of the link between clusters and classes. The update equation for the class cluster posterior now becomes

$$P(c|k)_{t+1} = \frac{\sum_{n\in\mathcal{D}^u} P(k|x_n) P(c|k)_t + \sum_{n\in\mathcal{D}^l} \delta_{c\,y_n} P(k|x_n, y_n)}{\sum_{n\in\mathcal{D}^u} P(k|x_n) + \sum_{n\in\mathcal{D}^l} P(k|x_n, y_n)},$$

where $\delta_{c\,y_n}$ is the kronecker delta. In the new update equation, the labelled and the unlabelled observations "vote" on what the next values in $P(c|k)_{t+1}$ should be. The labelled observations vote using the information in their labels, whereas the unlabelled observations vote for the current values in $P(c|k)_t$. By contrast, in [20], only the labelled data observations are taken into account when determining $P(c|k)_{t+1}$. This strategy fails in the rare event that a cluster i takes responsibility for unlabelled observations only. In this case, the update $P(c|k=i)_{t+1}$ cannot be computed. By allowing the unlabelled observations to contribute to the update, we avoid having to introduce heuristic workarounds. The unlabelled observations will simply assign $P(c|k=i)_{t+1} = P(c|k=i)_t$. The values in $P(c|k=i)_t$ will be influenced by the labelled observations that the cluster has previously encountered in its path. We did not observe any significant difference in average accuracy between the proposed method and the implementation in [20] combined with the rule that $P(c|k=i)_{t+1}$ should be set to the class priors when cluster i only contained unlabelled observations.

In order to find the optimal number of mixture components, a stratified 5-fold cross validation scheme was employed using the first night of each subject. For each optimal model, the test performance was measured on the second night. This approach was used for $f = \{0.2, 0.3, ..., 0.9, 1\}$. Additionally, in order to account for the randomness in performance associated with the stochastic choice of labelled observations, we ran the entire pipeline 100 times. Thus we report the expected test performance as the average test score across the 100 runs. Each run was terminated by either convergence or by running up to a maximum allowed iteration, which was set to 1000.

The GMM mixture positions $\boldsymbol{\mu}_k$ were initialized using the "k-means++"-algorithm, which provides better convergence properties than the traditional k-means algorithm [21]. The mixture covariances $\boldsymbol{\Sigma}_k$ were restricted as tied diagonals, and were all initialized by the complete data covariance matrix, and the cluster-class probability $P(c|k)$ were initialized as the class priors $P(c)$.

4. RESULTS

The average test accuracy across all subjects is presented in Fig. 2A. Naturally, the highest accuracy is achieved with $f = 1$, which is the fully supervised model, where $\mathrm{acc}_{\text{fully sup.}} \pm \sigma_{\text{fully sup.}} = 0.732 \pm 0.003$. Encouragingly, the accuracy for $f < 1$ is relatively close to the fully supervised solution for all f's. The average difference between the fully supervised model and the semi-vised model across all subjects and all runs, $\Delta_{\text{test acc}} = \mathrm{acc}_{\text{semi-sup.}} - \mathrm{acc}_{\text{fully sup.}}$ for different fractions f is further illustrated in Fig. 2B. For fractions $f \geq 0.5$, we cannot reject with 95% certainty the null-hypothesis, H_0: SS-GMM = FS-GMM. Looking at individual test performances, a few subjects showed a large deviation in test performance relative to the average across all subjects. In Fig. 2C,

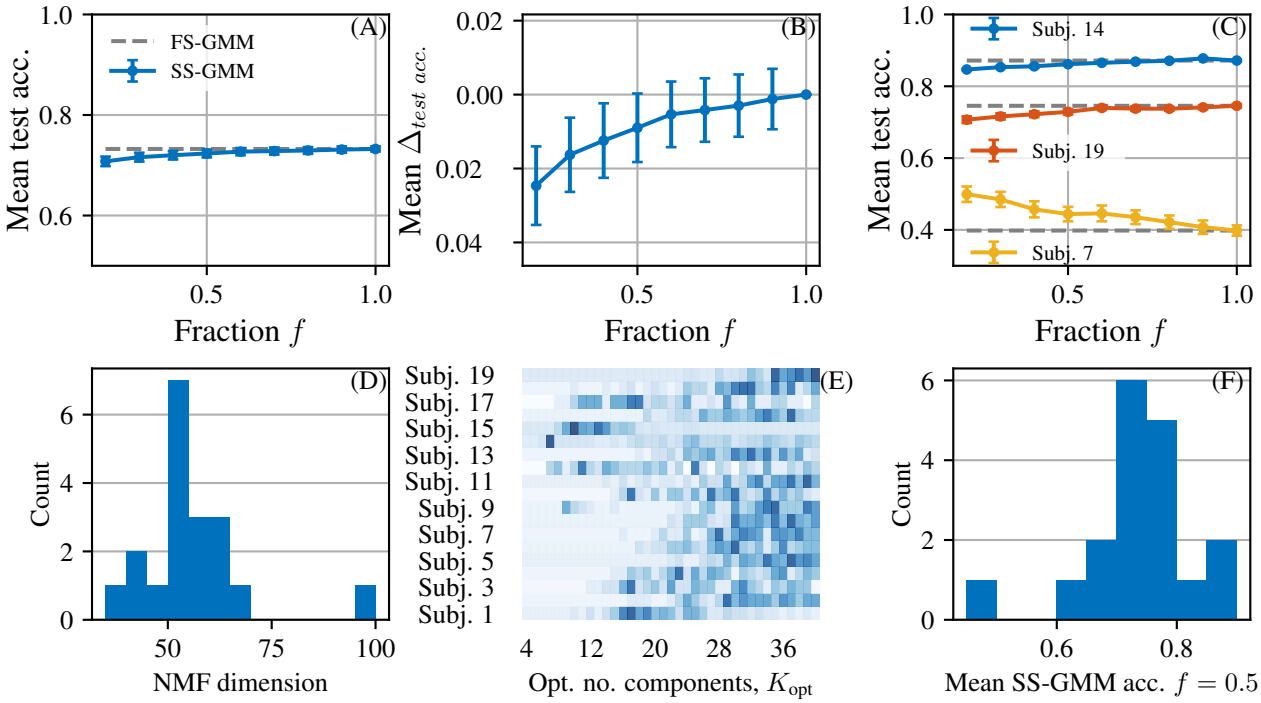

Fig. 2. (A): The mean test accuracy over all 19 subjects. (B): The mean difference in test accuracy between the FS-GMM and SS-GMM for each run across all subjects. (C): Results for the subject with the best average accuracy, one subject with a typical average accuracy, and the subject with the lowest average accuracy. All error bars represent the 95 % confidence interval. (D): Histogram of the optimal NMF dimension, D_i, minimizing the BIC criteria Eq. (1). (E): Heat map of the distribution of K_{opt} for each subject over the 100 runs. Darker values indicate more frequent chosen K_{opt}. (F): Histogram of the subject average SS-GMM accuracy. Both (E) and (F) use a labelled fraction $f = 0.5$.

we show the average test accuracy for a subject with well, poor, and typical performance. Most subjects had an average fully supervised test accuracy similar to the average across all subjects. Additionally, we found that the optimal number of dimensions D_i, $i \in \{1, \ldots, 19\}$ in the NMF decomposition for most subjects was in the interval $[50, 65]$, with a few extreme values as seen in Fig. 2D. Turning to the distribution of optimal number of GMM components K_{opt} across the 100 runs, we found relatively peaked distributions for most subjects, as illustrated in Fig. 2E. Finally, the distribution of the average semi-supervised test performance is shown in Fig. 2F with $f = 0.5$ for each subject.

5. DISCUSSION AND CONCLUSION

As seen in Fig. 2C and Fig. 2F, there is a large variability in the test performance between subjects. To address this, we made an investigation into each subjects data distributions. The prior class distribution of the training and test set was analyzed. We generally found high differences in priors for those subjects where we achieved a lower test accuracy. We suspect this might reflect a significant difference in data distributions between the two nights, thus violating the cluster-hypothesis on which our model relies. Interestingly, the SS-GMM approach outperforms the FS-GMM approach for the worst performing subject, and the test accuracy decreases with the labelled fraction f. This behaviour was only seen for this subject only. A possible explanation may be that the fractional label information acts as a regularizer to an otherwise over-fitted model.

Turning to the optimal number of dimensions D_i in the NMF decomposition, one could argue that it should be optimized as a hyper-parameter alongside K through cross-validation. Using the BIC approach, we seemed to find fairly large dimensions which may not be desirable. Regarding K_{opt}, our findings indicate an importance of optimizing the number of clusters, due to the peaked distributions.

Since we only have one test night in the data set, we have no estimate of the temporal variation of test performance over several nights. Future research should illuminate the test performance of a one-night-trained model over timescales such as weeks and months.

In conclusion, it is feasible to build an automatic sleep stage classification algorithm where only a subset of the training set has been labelled. For our approach, fractions larger than 50 % yielded an average test performance that matches that of a fully supervised model. This potentially offers a substantial reduction in work load for sleep scorers.

6. REFERENCES

[1] Ahnaf Rashik Hassan, Syed Khairul Bashar, and Mohammed Imamul Hassan Bhuiyan, "On the classification of sleep states by means of statistical and spectral features from single channel electroencephalogram," *2015 International Conference on Advances in Computing, Communications and Informatics (icacci)*, pp. 2238–2243, 2015.

[2] Khald Ali I. Aboalayon, Miad Faezipour, Wafaa S. Almuhammadi, and Saeid Moslehpour, "Sleep stage classification using eeg signal analysis: A comprehensive survey and new investigation," *Entropy*, vol. 18, no. 9, pp. 272 (31 pp.), 272 (31 pp.), 2016.

[3] Takashi Nakamura, Tricia Adjei, Yousef Alqurashi, David Looney, Mary J. Morrell, and Danilo P. Mandic, "Complexity science for sleep stage classification from eeg," *2017 International Joint Conference on Neural Networks (ijcnn)*, pp. 4387–94, 4387–4394, 2017.

[4] Orestis Tsinalis, Paul M. Matthews, and Yike Guo, "Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders," *Annals of Biomedical Engineering*, vol. 44, no. 5, pp. 1587–1597, 2016.

[5] Akara Supratak, Hao Dong, Chao Wu, and Yike Guo, "Deepsleepnet: a model for automatic sleep stage scoring based on raw single-channel eeg," *Ieee Transactions on Neural Systems and Rehabilitation Engineering*, 2017.

[6] Albert Vilamala, Kristoffer H Madsen, and Lars K Hansen, "Deep convolutional neural networks for interpretable analysis of eeg sleep stage scoring," *arXiv preprint arXiv:1710.00633*, 2017.

[7] B Kemp, AH Zwinderman, B Tuk, HAC Kamphuisen, and JJL Oberye, "Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the eeg," *Ieee Transactions on Biomedical Engineering*, vol. 47, no. 9, pp. 1185–1194, 2000.

[8] 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," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[9] I. Silva and G. Moody, "An open-source toolbox for analysing and processing physionet databases in matlab and octave.," 2014.

[10] Allan Rechtschaffen and Anthony . Kales, *A manual of standardized terminology, techniques and scoring systems for sleep stages of human subjects*, 1968.

[11] Marina Ronzhina, Oto Janousek, Jana Kolarova, Marie Novakova, Petr Honzik, and Ivo Provaznik, "Sleep scoring using artificial neural networks," *Sleep Medicine Reviews*, vol. 16, no. 3, pp. 251–263, 2012.

[12] Luis Javier Herrera, Antonio Miguel Mora, C Fernandes, Daria Migotina, Alberto Guillén, and Agostinho C Rosa, "Symbolic representation of the eeg for sleep stage classification," in *Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on*. IEEE, 2011, pp. 253–258.

[13] Chih-Sheng Huang, Chun-Ling Lin, Li-Wei Ko, Sheng-Yi Liu, Tung-Ping Sua, and Chin-Teng Lin, "A hierarchical classification system for sleep stage scoring via forehead eeg signals," in *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2013 IEEE Symposium on*. IEEE, 2013, pp. 1–5.

[14] Christian Berthomier, Xavier Drouot, Maria Herman-Stoïca, Pierre Berthomier, Jacques Prado, Djibril Bokar-Thire, Odile Benoit, Jérémie Mattout, and Marie-Pia d'Ortho, "Automatic analysis of single-channel sleep eeg: validation in healthy individuals," *Sleep*, vol. 30, no. 11, pp. 1587–1595, 2007.

[15] Michael J. Prerau, Ritchie E. Brown, Matt T. Bianchi, Jeffrey M. Ellenbogen, and Patrick L. Purdon, "Sleep neurophysiological dynamics through the lens of multitaper spectral analysis," *Physiology*, vol. 32, no. 1, pp. 60–92, 2017.

[16] Partha Mitra, Hemant Bokil, Hiren Maniar, Catherine Loader, Samar Mehta, Dan Hill, Siddhartha Mitra, Peter Andrews, Rafael Baptista, S. Gopinath, Hariharan Nalatore, and Sumanjit Kaur, "chronux analysis software," 2016.

[17] Partha Mitra and Hemant Bokil, "Observed brain dynamics," *Observed Brain Dynamics*, pp. 1–404, 2009.

[18] Daniel D. Lee and H. Sebastian Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[19] Morten Mørup and Lars Kai Hansen, "Automatic relevance determination for multi-way models," *Journal of Chemometrics*, vol. 23, no. 7-8, pp. 352–363, 2009.

[20] Jan Larsen, Anna Szynkowiak Have, and Lars Kai Hansen, "Probabilistic hierarchical clustering with labeled and unlabeled data," *International Journal of Knowledge-Based and Intelligent Engineering Systems*, vol. 6, no. 1, pp. 56–62, 2002.

[21] David Arthur and Sergei Vassilvitskii, "k-means++," *Proceedings of the Eighteenth Annual Acm-siam Symposium*, pp. 1027–1035, 2007.