

COMPOSING GRAPHICAL MODELS WITH GENERATIVE ADVERSARIAL NETWORKS FOR EEG SIGNAL MODELING

Khuong Vo^x, Manoj Vishwanath^x, Ramesh Srinivasan^{y,z}, Nikil Dutt^{x,y}, and Hung Cao^z

^x Department of Computer Science, ^y Department of Electrical Engineering and Computer Science, ^y Department of Cognitive Sciences, ^z Department of Biomedical Engineering, University of California, Irvine, USA

fkhuongav,manojv,srinivar,dutt,hungcao@uci.edu

ABSTRACT

Neural oscillations in the form of electroencephalogram (EEG) can reveal underlying brain functions, such as cognition, memory, perception, and consciousness. A comprehensive EEG computational model provides not only a stochastic procedure that directly generates data but also insights to further understand the neurological mechanisms. Here, we propose a generative and inference approach that combines the complementary benefits of probabilistic graphical models and generative adversarial networks (GANs) for EEG signal modeling. We investigate the method's ability to jointly learn coherent generation and inverse inference models on the CHI-MIT epilepsy multi-channel EEG dataset. We further study the efficacy of the learned representations in epilepsy seizure detection formulated as an unsupervised learning problem. Quantitative and qualitative experimental results demonstrate the effectiveness and efficiency of our approach.

Index Terms— EEG, GAN, Bayesian networks

1. INTRODUCTION

Electroencephalogram (EEG) is a non-invasive technique that measures the spontaneous electrical activity of the brain. EEG has been a driver of studies from basic neurological research to clinical applications. EEG modeling is essential to understanding the underlying mechanisms that generate brain signals and serve to design experiments and test hypotheses in silico. There exist extensive prior works on EEG computational models [1] that derived principled neuroscience laws, empirically validated rules, or other domain expertise. Those are often in the form of general time-dependent and nonlinear partial differential equations. Nevertheless, they rely on strong assumptions which are not always generalizable. Further, those are slow to simulate and often suffer from model misspecifications.

Generative Adversarial Networks (GANs) [2] provide a powerful framework and tools for machine learning, especially for deep representation learning and generative models. Over the past few years, GANs have witnessed tremendous advancements and achieved state-of-the-art performance in a variety of prominent tasks, including photo editing, video prediction, text generation, and signal synthesis [3, 4]. As a data-driven method, GANs are flexible and do not depend on rigid assumptions. Therefore, GANs hold great potential in modeling the inherent stochasticity and extrinsic uncertainty of EEG signals.

Recent work [5, 6, 7] applying GANs in EEG synthesis tend to simply characterize the spatio-temporal characteristics of EEG data

subject to latent spaces of basic distributions, e.g., Gaussian or uniform distributions. Such assumptions impose limitations in capturing the intrinsic dependence among latent variables. Also, the GANs require deeper networks to synthesize longer sequences, which are computationally expensive and challenging to train, e.g., vanishing or exploding gradient problems. Moreover, the lack of inference capability in vanilla GANs hinder insight into structural information of EEG signals. On the other hand, probabilistic graphical models [8] enable inference through structured representations but often lack the capability to model arbitrarily complex distributions.

To address these challenges, we propose a novel GAN-based approach for EEG signal modeling that couples deep implicit likelihoods [9] with structured latent variable representations to combine their complementary strengths. Our method uses graphical models for representing underlying structures of the signals, and applies ideas from the Graphical-GAN [10] for effectively learning not only a generative model mapping from latent distributions to complex high-dimensional EEG data space but also an inverse inference model mapping from the data space to the latent space. Our study paves the way for leveraging implicit probabilistic models to comprehensively investigate the mechanisms that generate brain waves.

2. METHODOLOGY

2.1. EEG Signal Synthesis with GANs

A GAN is a generative model trained by a pair of neural networks in a game-theoretic approach [2]. In GANs, a discriminator neural network D is trained to distinguish real from synthetic EEG signals, while a neural generator network G is trained to generate EEG signals from a latent space to make them indistinguishable by the discriminator. With EEG signal x drawn from data generating distribution $q(x)$, z drawn from noise prior p_z , and $p(x)$ is the generator's distribution over synthetic data, G and D jointly optimize the following objective:

$$\begin{aligned} L_{GAN}(G; D) &= E_{x \sim q(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \\ &= E_{x \sim q(x)}[\log D(x)] + E_{p(x)}[\log(1 - D(x))] \quad (1) \end{aligned}$$

The discriminator is expected to output a high probability for a valid EEG signal and a low probability for a synthesized one, corresponding to the values of $\log D(x)$ and $\log(1 - D(G(z)))$, respectively. G and D are trained simultaneously until G is able to successfully fool D .

Following the proofs in [2], given a fixed generator G , the optimal discriminator is given by $D(x) = \frac{q(x)}{q(x) + p(x)}$

Under an optimal discriminator D , the generator minimizes the Jensen-Shannon (JS) divergence, which attains its minimum if and only if $p(x) = q(x)$.

2.2. Conjoining GANs with Bayesian Networks

2.2.1. Generative and Inverse Inference Process

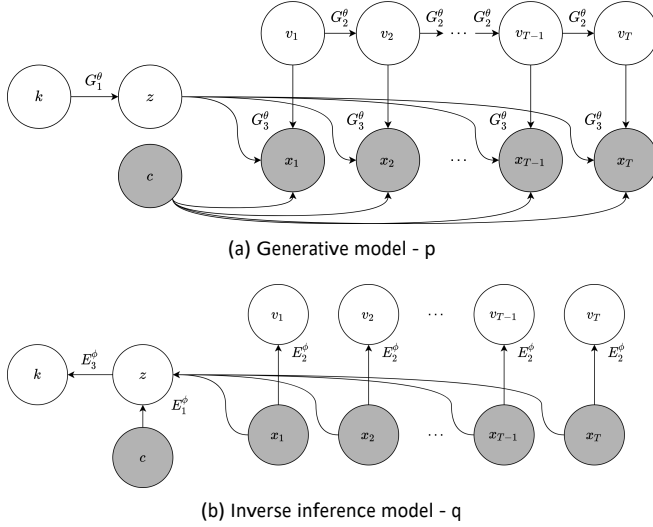


Fig. 1: Directed graphical models for EEG signal modeling. Each time step corresponds to a -second multi-channel signal. Shaded nodes represent observed variables. Clear nodes represent latent variables. Directed edges indicate statistical dependencies between variables.

As shown in Figure 1, we model the generative process and the inverse inference process by a generative model and an inverse inference model in the Bayesian network. The framework exploits a Gaussian mixture model (GMM) to characterize the static latent variable structure with its capability to approximate arbitrary distributions, and a Markov model for the dynamic latent characterization. We use the notations p and q to denote the generative and inverse inference models.

The joint distribution of the generative model p is

$$p(x_{1:T}; v_{1:T}; z; k; c) = p(k)p(z|k)p(c) \prod_{t=1}^T p(v_t|v_{t-1})p(x_t|z; v_t; c) \quad (2)$$

where $p(k)$ and $p(c)$ are simple prior distributions for Gaussian mixture indicator k and condition c , e.g., a categorical distribution and a uniform distribution, $p(z|k)$ models a component selecting procedure for sampling noise z which encodes the temporal-spatial relationships invariant across time, v_t 's form a first-order Markov chain, with $p(v_1|v_0) = N(0; I)$, to encodes the temporal relationships variant across time, $p(x_t|z; v_t; c)$ specifies the conditional probability of the data at each time step t given noise z , state v_t , and condition c , and is of interest for the final generation.

The distribution function $p(x_{1:T}; v_{1:T}; z; k; c)$ is parametrized as generator neural networks. It consists of three parts: $z^p = G_1(k^p)$, $v_{t+1}^p = G_2(v_t^p; t)$, $t \sim N(0; I)$, and $x_t^p = G_3(z; v_t^p; c)$. G_1 is responsible for a mapping from the input prior to a mixed Gaussian distribution with respect to k_p . G_2 transitions to a new

state v_t^p given the previous state. G_3 uses noise z^p , state v_t^p , and condition c to generate the synthetic -second EEG signal x_t^p .

The joint distribution of the inverse inference model q is

$$q(x_{1:T}; v_{1:T}; z; k; c) = q(x_{1:T})q(z|x_{1:T}; c)q(k|z) \prod_{t=1}^T q(v_t|x_t) \quad (3)$$

where each latent variable of the Markov structure is assumed to be independent using the mean-field approximation [11]. $q(x_{1:T})$ is the empirical data distribution, $q(z|x_{1:T}; c)$, $q(v_t|x_t)$, and $q(k|z)$ are of interest for the inference. Contrary to $p(v_{t+1}|v_t)$, $q(v_t|x_t)$ models a dynamic tracing procedure for reconstructing the hidden features v_t . In contrast to $p(z|k)$; $q(k|z)$ models a component tracing procedure for reconstructing the Gaussian mixture indicator k .

The distribution function $q(x_{1:T}; v_{1:T}; z; k; c)$ is parametrized as extractor neural networks. It consists of three parts: $z^q = E_1(x_{1:T}^q; c)$, $v_t^q = E_2(x_t^q)$, and $k^q = E_3(z^q)$. E_1 and E_2 are responsible for a mapping from original signals to noise z^q and state v_t^q , respectively. E_3 infers within the latent space from z^q to k^q .

2.2.2. Learning Process

Our goal is to learn the parameters of the generative model p and the inverse inference model q by jointly minimizing the Jensen-Shannon (JS) divergence

$$JS(q(x_{1:T}; v_{1:T}; z; k; c)kp(x_{1:T}; v_{1:T}; z; k; c)) \quad (4)$$

Expectation Propagation (EP) [12], a deterministic approximation algorithm, is proposed to utilize the locally structured data following [10]. The joint distributions can be factorized in terms of a set of factors $F_G = f(k; z); (v_t; v_{t-1}); (x_t; v_t; z; c)g$. For a factor a , the divergence of interest is

$$JS(q(a) \prod_{b=a}^Y q(b)kp(a) \prod_{b=a}^Y p(b)) \quad (5)$$

EP iteratively minimize a local divergence in terms of each factor individually with the assumption that $q(a) \approx \prod_{b=a}^Y q(b)$. The divergence becomes

$$JS(q(a) \prod_{b=a}^Y q(b)kp(a) \prod_{b=a}^Y q(b)) \quad (6)$$

Using the same proof sketch as in [10], the divergence for factor a is approximated as

$$JS(q(x_{1:T}; v_{1:T}; z; k; c)kp(x_{1:T}; v_{1:T}; z; k; c)) \approx E_q \log \frac{2q(a)}{p(a) + q(a)} + E_p \log \frac{2p(a)}{p(a) + q(a)} \quad (7)$$

The divergences are further averaged over all local factors as

$$\frac{1}{JF_G} \sum_{a \in F_G} \left(E_q \log \frac{2q(a)}{p(a) + q(a)} + E_p \log \frac{2p(a)}{p(a) + q(a)} \right) \quad (8)$$

Individual parametric discriminators D_a can be employed to estimate the local divergences as follows

$$\max \frac{1}{J F_G} E_q \left(\frac{1}{a} \sum_{a=2}^X \log(D_a(a)) \right)^5 + \frac{1}{J F_G} E_p \left(\frac{1}{a} \sum_{a=2}^X \log(1 - D_a(a)) \right)^5 \quad (9)$$

where θ denotes the parameters in all discriminators. The discriminative models distinguish between the variables from the generative model p and those from the inverse inference model q as synthetic and original, respectively.

2.2.3. Optimization Objective

Three discriminators D_3 , D_2 and D_1 receive local variable pairs, i.e., $(k; z)$, $(v_t; v_{t-1})$, $(x_t; v_t; z; c)$, from either the generative model p or the inverse inference model q , separately. The adversarial loss is as follows

$$\begin{aligned} L_{GAN}(G; E; D) = & E_q \log D_3(k^q; z^q) + \log D_2(v_t^q; v_{t-1}^q) + \log D_1(x_t^q; v_t^q; z^q; c) \\ & + E_p \log(1 - D_3(k^p; z^p)) + \log(1 - D_2(v_t^p; v_{t-1}^p)) \\ & + \log(1 - D_1(x_t^p; v_t^p; z^p; c)) \end{aligned} \quad (10)$$

All components are trained simultaneously in an adversarial process. Let θ and ϕ denote the parameters of G and E , respectively. Iteratively, D learn to maximize Equation 10 by updating θ , while G and E learn to minimize Equation 10 by updating corresponding parameters ϕ and θ , respectively.

In order to ensure the global consistency of an entire signal across time steps, a frequency domain loss is added as

$$L_f(G) = k(x_{i;1:T}^q) - (x_{i;1:T}^p)k_1 + k(x_{i;1:T}^p) - (x_{i;1:T}^q)k_1 \quad (11)$$

where rand ' refer to the average magnitude and phase across signals i in a batch, respectively. They are computed by a fast Fourier transform (FFT). Hence, the total objective is

$$\min_{G, E} \max_D L_{GAN} + L_f \quad (12)$$

2.3. Network Architectures and Hyperparameters

Table 1 presents the architectures of the deep neural networks. Each time step corresponds to a 1-second EEG signal ($\tau = 1$). All the feature maps have 96 channels. Leaky ReLU activation functions are applied to all layers, with the slope 0.1 to stimulate easier gradient flow. Batch normalizations (BN) [13] are used at each convolutional layer of the generators and extractors. Spectral normalizations (SN) [14] are applied to the discriminators to constrain their Lipschitz constants. c are subject embeddings as one-hot vectors. The sizes of z , k , and v_t , and t are set at 128, 6, 32, and 16 respectively.

G_1 and E_2 are single-layer neural networks. We use the reparameterization trick [15] to estimate the gradients with the continuous variable z , and the Gumbel-Softmax trick [16] (the temperature of 0.1) to estimate the gradients with the discrete variable k .

α is set at 0.1 to have the training process driven mainly by the adversarial loss. In order to mitigate the issue of slow learning in regularized discriminators, a higher learning rate is provided to the discriminators than the generators and extractors by the Two Time-scale Update Rule (TTUR) [17]. The models are trained with the Adam optimizer with the initial learning rate of 0.0004 for D , the

Table 1: Network architectures. Models having similar architectures are grouped together.

| $G_2; D_3; D_2$ | G_3 | |
|--|--------------------|-----|
| Linear 512, (SN), IReLU | Linear 1536, IReLU | |
| Linear 512, (SN), IReLU | Reshape 96x16 | |
| Linear 256, (SN), IReLU | Upsample | |
| G_2 Linear 32 | Conv 6, BN, IReLU | X 4 |
| D_3 Linear 1, SN, Sigmoid | Conv 6, BN, IReLU | |
| D_2 Linear 1, SN, Sigmoid | Conv 1, Tanh | |
| D_1 | $E_2; E_1$ | |
| Get x_t or $x_{[1:T]}$ (concatenated along channels) | | |
| Conv 1, IReLU - 96x256 | | |
| Conv 6, BN/SN, IReLU | | X 4 |
| Conv 6, Stride 2, BN/SN, IReLU | | |
| Reshape 1536 | | |
| Get v_t, z, c | E_2 Linear 32 | |
| Linear 256, SN, IReLU | E_1 Linear 128 | |
| Join features of x_t, v_t, z, c | | |
| Linear 512, SN, IReLU | | |
| Linear 1, SN, Sigmoid | | |

learning rate of 0.0001 for G and E , and the exponential decay rates $\alpha_1 = 0.5$ and $\alpha_2 = 0.999$. All weights are initialized using a zero-centered Gaussian distribution with a standard deviation of 0.02. We make the implementation publicly available¹.

3. EXPERIMENTS

3.1. Dataset

The 23-channel interictal EEG recordings from the CHB-MIT epilepsy dataset [18] are used for the experiments. The dataset consists of scalp EEG from pediatric subjects with intractable seizures. We select a subset of 6 patients (chb01-03, chb05-06, chb10) having the same measurement setup, including males and females, 1.5-14 years old. Interictal periods are extracted at least 4-hour away before a seizure onset and after the seizure ends. The signals are low-pass filtered with a cut-off frequency at 50 Hz and scaled to the range $[-1; 1]$. Overall, the dataset contains 43593 signals, from which 70% are used for training and validation, and the other 30% are used as the test set. Each signal is 10-second long ($T=10$), at a sampling rate of 256 Hz. Additionally, 339 ictal EEG signals are extracted for evaluating epilepsy seizure detection performance.

3.2. Evaluation Metrics

Sliced 2-Wasserstein distance (SWD) [19, 20] quantifies the cost of transforming one distribution to another. It is an approximation to the 2-Wasserstein distance using 1D projections for a closed-form solution and is defined as

$$SWD_2(\mu; \nu) = \sqrt{\int_0^1 \left(F_\mu^{-1}(u) - F_\nu^{-1}(u) \right)^2 du} \quad (13)$$

where μ and ν are two probability measures, $\#$ stands for the pushforwards of the projection $R^d \rightarrow R^1$; $h; X_i$, and $U^{S^{d-1}}$ is the uniform distribution on the hypersphere of d dimensions.

Spectral entropy (SEN) measures the uniformity of the signal energy distribution in the frequency-domain. It is given by

$$H(x) = - \sum_{f=0}^{f_x=2} P(f) \log_2[P(f)] \quad (14)$$

¹<https://github.com/khuongav/Graphical-Adversarial-Modeling-of-EEG>

where P is the normalised power spectral density, and f_s is the sampling frequency of signal x .

Reconstruction error (REC) measures the differences between the values of an original signal and its reconstruction \hat{x} as

$$REC = \frac{1}{T} \sum_{t=1}^T \|x_t - \hat{x}_t\|_2^2 \quad (15)$$

3.3. Results and Discussion

Table 2: Performances of different GAN models in interictal EEG signal synthesis and reconstruction tasks.

| | SWD | REC | SEN |
|------------------------|---------|--------|--------------------|
| Original data | | | 0.620 0.070 |
| GMMarkov-GAN | 1.16e-2 | 0.0474 | 0.0392 0.608 0.063 |
| Markov-GAN | 1.34e-2 | 0.0494 | 0.0413 0.636 0.070 |
| GMMarkov-GAN (w/o FFT) | 1.70e-2 | 0.0519 | 0.0438 0.585 0.074 |
| Markov-GAN (w/o FFT) | 1.78e-2 | 0.0530 | 0.0391 0.583 0.069 |
| C-BiGAN/ALI | 2.13e-2 | 0.0562 | 0.0415 0.539 0.066 |

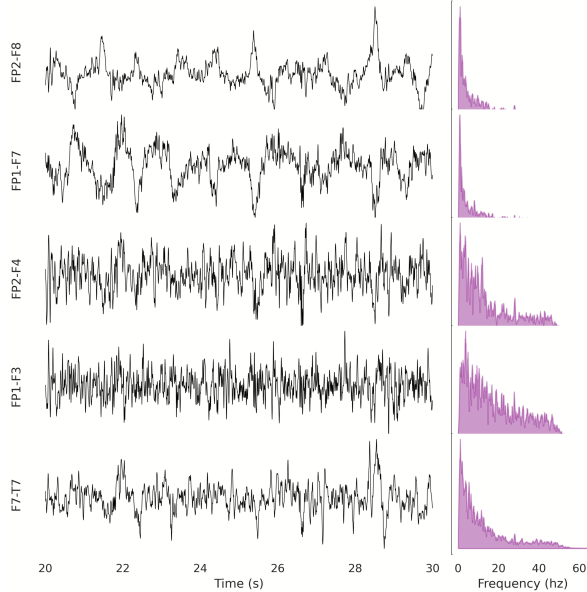


Fig. 2: Last 10-second of a 30-second synthetic 23-channel EEG signal by the GMMarkov-GAN model, conditioned on patient 3. 5 channels with the highest standard deviations are shown.

Table 2 presents the performance of our proposed approaches and the comparison with the BiGAN/ALI model [21, 22]. We denote its conditional version as C-BiGAN/ALI. GMMarkov-GAN is our model characterized by Gaussian mixture and Markov latent structures, while Markov-GAN is only with the Markov structure. C-BiGAN/ALI is the GAN with an inference capability but without a latent variable structure, in which the latent space is a simple Gaussian, and data at each timestep are generated independently.

Both the graphical GANs achieve significantly lower SWD, REC, and SEN differences than C-BiGAN/ALI, indicating that they are better at capturing the characteristics of EEG in both time and frequency domains. Besides, by encoding the invariant spatial-temporal features of EEG signals subject to the flexibility of a Gaussian mixture, GMMarkov-GAN enjoys better performance (SWD of 0.0173, REC of 0.0519, and SEN difference of 0.035) than the Markov-GAN. We attribute this to GMMarkov-GAN being able to learn a structured clustering of the latent space as shown in

Figure 4. These results prove the effectiveness of our proposed data structures and confirm our inverse inference strategy.

By training with the additional FFT loss, GMMarkov-GAN enjoys the highest performance (SWD of 0.0116, REC of 0.0474, and SEN difference of 0.012). It should be noted that the frequency-domain loss added little time for training, yet it noticeably improved the results.

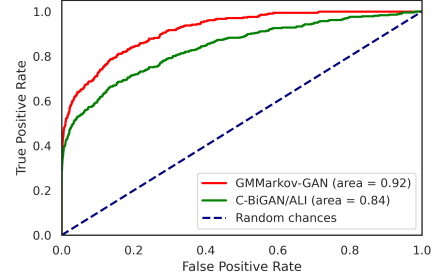


Fig. 3: ROC curve for epilepsy seizure detection.

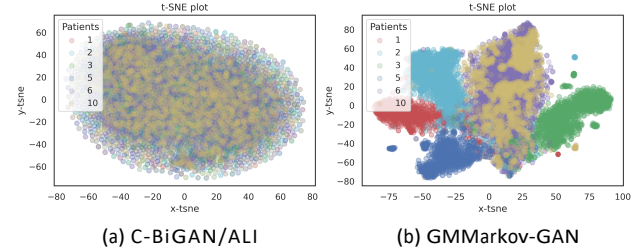


Fig. 4: t-SNE visualization of the static latent spaces.

In Figure 2, synthetic multi-channel EEG signals are plotted. The signals are naturally realistic across channels and show good fits in different frequency bands. Although our model is trained on 10 second-long signals, it can generate much longer sequences of 30 seconds, thanks to the Markov structure.

To demonstrate the efficacy of our generative and inverse mapping approach for auxiliary tasks, we further evaluated our approach in epilepsy seizure detection. As the model is trained on the interictal EEG signals, seizure segments are detected with reconstruction error thresholds in an anomaly detection framework. Figure 3 shows a high detection performance from our model by the ROC curve with the area under the curve of 0.92, competitive with contemporary approaches in supervised learning [23]. We plan to build on these results in our future work for interpreting more encoded features in the low-dimensional manifolds and further investigate the partial mode collapse issue of GANs.

4. CONCLUSION

In this work, we proposed an EEG modeling scheme that combines the strengths of probabilistic graphical models and generative adversarial networks. Our experimental results demonstrate that our method effectively characterized EEG latent variable structure via a Gaussian mixture and a Markov model. The structured representations can provide interpretability and encode inductive biases to reduce the data complexity of neural oscillations. Our approach holds promise to new generative applications in neuroscience and neurology. Future directions include generalizing learning and inference algorithms with more complicated structures to truly model the underlying relationships at different scales spanning from the single cell spike train up to macroscopic oscillations.

5. ACKNOWLEDGEMENTS

The authors would like to acknowledge the financial support from the NSF grant #2051186, the NSF CAREER Award #1917105 (H.C.), and the NIH SBIR grant #R44OD024874 (H.C.).

6. REFERENCES

- [1] Katharina Glomb, Joana Cabral, Anna Cattani, Alberto Mazzone, Ashish Raj, and Benedetta Franceschiello, "Computational models in electroencephalography," arXiv preprint arXiv:2009.08385, 2020.
- [2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [3] Abdul Jabbar, Xi Li, and Bourahla Omar, "A survey on generative adversarial networks: Variants, applications, and training," arXiv preprint arXiv:2006.05132, 2020.
- [4] Khuong Vo, Emad Kasaeyan Naeini, Amir Naderi, Daniel Jilani, Amir M Rahmani, Nikil Dutt, and Hung Cao, "P2e-wgan: Ecg waveform synthesis from ppg with conditional wasserstein generative adversarial networks," in *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, 2021, pp. 1030–1036.
- [5] Kay Gregor Hartmann, Robin Tibor Schirrmester, and Tonio Ball, "Eeg-gan: Generative adversarial networks for electroencephalographic (eeg) brain signals," arXiv preprint arXiv:1806.01875, 2018.
- [6] Nik Khadijah Nik Aznan, Amir Atapour-Abarghouei, Stephen Bonner, Jason D Connolly, Noura Al Moubayed, and Toby P Breckon, "Simulating brain signals: Creating synthetic eeg data via neural-based generative models for improved ssvep classification," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.
- [7] Damián Pascual, Amir Aminifar, David Atienza, Philippe Ryvlin, and Roger Wattenhofer, "Synthetic epileptic brain activities using gans," *Machine Learning for Health (ML4H) at NeurIPS*, 2019.
- [8] Daphne Koller and Nir Friedman, *Probabilistic graphical models: principles and techniques*, MIT press, 2009.
- [9] Shakir Mohamed and Balaji Lakshminarayanan, "Learning in implicit generative models," arXiv preprint arXiv:1610.03483, 2016.
- [10] Chongxuan Li, Max Welling, Jun Zhu, and Bo Zhang, "Graphical generative adversarial networks," arXiv preprint arXiv:1804.03429, 2018.
- [11] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul, "An introduction to variational methods for graphical models," *Machine learning*, vol. 37, no. 2, pp. 183–233, 1999.
- [12] Thomas P Minka, "Expectation propagation for approximate bayesian inference," arXiv preprint arXiv:1301.2294, 2013.
- [13] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. PMLR, 2015, pp. 448–456.
- [14] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida, "Spectral normalization for generative adversarial networks," arXiv preprint arXiv:1802.05957, 2018.
- [15] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114, 2013.
- [16] Eric Jang, Shixiang Gu, and Ben Poole, "Categorical reparameterization with gumbel-softmax," arXiv preprint arXiv:1611.01144, 2016.
- [17] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.
- [18] Ali Hossam Shoeb, *Application of machine learning to epileptic seizure onset detection and treatment*, Ph.D. thesis, Massachusetts Institute of Technology, 2009.
- [19] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister, "Sliced and radon wasserstein barycenters of measures," *Journal of Mathematical Imaging and Vision*, vol. 51, no. 1, pp. 22–45, 2015.
- [20] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Zahdi Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, et al., "Pot: Python optimal transport," *Journal of Machine Learning Research*, vol. 22, no. 78, pp. 1–8, 2021.
- [21] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville, "Adversarially learned inference," arXiv preprint arXiv:1606.00704, 2016.
- [22] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell, "Adversarial feature learning," arXiv preprint arXiv:1605.09782, 2016.
- [23] Mohammad Khubeb Siddiqui, Ruben Morales-Menendez, Xiaodi Huang, and Nasir Hussain, "A review of epileptic seizure detection using machine learning classifiers," *Brain informatics*, vol. 7, pp. 1–18, 2020.