YARE-GAN: Yet Another Resting Sate EEG-GAN

Yeganeh Farahzadi
Institute of Psychology, Eötvös Loránd University
Budapest, 1064, Hungary

Morteza Ansarinia
Department of Behavioural and Cognitive Sciences, University of Luxembourg
Belval, Luxembourg

Zoltán Kekecs
Institute of Psychology, Eötvös Loránd University
Budapest, 1064, Hungary

Abstract

In this study, we implement a Wasserstein GAN with Gradient Penalty (WGAN-GP) to generate multi-channel resting-state EEG data and assess the quality of the synthesized signals through both visual and feature-based evaluations. Our results indicate that the model effectively captures the statistical and spectral characteristics of real EEG data, although challenges remain in replicating high-frequency oscillations in the frontal region. Additionally, we demonstrate that the Critic's learned representations can be reused for gender classification task, achieving an out-of-sample accuracy, significantly better than a shuffled-label baseline and a model trained directly on EEG data. These findings suggest that generative models can serve not only as EEG data generators but also as unsupervised feature extractors, reducing the need for manual feature engineering. This study highlights the potential of GAN-based unsupervised learning for EEG analysis, suggesting avenues for more data-efficient deep learning applications in neuroscience.

Introduction

Generative AI—including techniques such as Generative Adversarial Networks (GANs) (Goodfellow et al. (2014)), Variational Autoencoders (VAEs) (Kingma (2013)), and diffusion models (Ho et al. (2020))—is being increasingly recognized as a powerful tool in neuroscience. It has found various applications ranging from synthesizing neural data to also learning reusable representations of neural signals in an unsupervised manner.

One of the challenges in neuroscience is the limited availability of high-quality neural data, as collecting such data is often costly and time-consuming. Generative AI can help address this issue by creating realistic synthetic datasets that capture the statistical properties of real neural recordings. Additionally, in cases where labeled data is sparse or certain conditions are underrepresented, synthetic data can help balance datasets, leading to more robust analyses and improved model performance (Murphy (2022)). The use of generative AI—particularly GANs—for data augmentation in neural datasets has been relatively well studied. Prior research has demonstrated that augmenting datasets with generated neural signals can enhance classification accuracy in tasks such as motor imagery (Hartmann et al. (2018); Fahimi et al. (2020)), emotion recognition (Pan & Zheng (2021)), and detection of epileptic seizure (Pascual et al. (2020)).

Beyond data augmentation, generative AI—particularly GANs—also provides a way to learn meaningful feature representations in an unsupervised manner (Radford (2015)). By learning a latent space that effectively captures the underlying structure of neural data, GANs can generate interpolated samples that preserve key statistical and physiological properties. These learned representations can be repurposed for supervised tasks, reducing the need for extensive and often labor-intensive feature engineering. This approach is particularly beneficial for neural signal processing, where preprocessing and manual feature extraction can significantly influence final results (Robbins et al. (2020); Botvinik-Nezer et al. (2020)). This way, an unsupervised generative model can first be trained using publicly available, large-scale resting-state datasets and later fine-tuned on task-specific data, which is often more limited. In this way, generative AI not only mitigates the issue of limited case-specific samples through data augmentation but also tackles this issue by offering transferable, reusable representation of the data that can later be fine-tuned for a downstream, case-specific supervised task.

While recent work has introduced novel architectures like criss-cross transformers for unsupervised EEG representation learning Wang et al. (2024), the use of generative adversarial networks (GANs) for this purpose remains relatively underexplored, with some studies investigating their potential Liang et al. (2021). This may be due to the fact that GANs in neuroscience have primarily been applied for data augmentation, often on small, domain-specific datasets, and mainly focused on generating extracted features rather than raw time-series data. As a result, GANs have not been widely utilized as direct neural signal processors or feature extractors.

In this study, we aim to extend the use of GANs beyond data augmentation by incorporating larger-scale EEG recordings—particularly resting-state, task-free data—to train a GAN model for neural data generation. We further explore the utility

of the learned representations from its intermediate layers of its discriminator for downstream classification tasks, demonstrating their potential for improving neural signal analysis.

Methods

Data

We used the MPI-LEMON dataset (Babayan et al. (2019)), a publicly available resource designed to study mind-body-emotion interactions. This dataset includes resting-state EEG (rs-EEG) recordings from 216 participants, utilizing 61 EEG channels during both eye-closed and eye-open experimental blocks. Data from 13 participants were excluded due to missing event information, different sampling rates, mismatching header files, or insufficient data quality (Babayan et al. (2019)). Each participant contributed a 16-minute EEG recording, resulting in a total of 54.13 hours of rs-EEG data.

Preprocessing To preserve the natural structure of the EEG data, we kept preprocessing to a minimum (Delorme (2023)). We first downsampled the EEG signals to 98 Hz. This sampling rate was deliberately chosen as a workaround to remove power line noise (50 Hz) without requiring an additional low-pass or notch filter.

Following the steps outlined in Défossez et al. (2023), we then applied Baseline correction by subtracting the average value of the first 0.5 seconds from each channel. The data was then normalized using Scikit-Learn's robust scaler, followed by standard scaling. To mitigate the impact of extreme outliers, values exceeding 20 standard deviations were clamped. Data clamping is found to be as effective as more complex artifact rejection methods such as AutoReject (Défossez et al. (2023)).

Finally, EEG signals were high-pass filtered at 0.5 Hz to remove low-frequency drift and were segmented into around 5-seconds windows (512 time points). In this study, we included eight EEG channels (F1, F2, C1, C2, P1, P2, O1, and O2). Selecting 8 out of the 61 EEG channels were primarily for practical reasons, since this subset allowed for faster experimentation and model development. However, the proposed model is designed to be scalable. Specifically, the capacity of each layer—such as the number of filters in the convolutional layers—is proportional to the number of input channels. This allows the model to be adapted to accommodate and process additional EEG channels as needed. In the supplementary materials we compared the model's performance with additional 16 and 56 channels to demonstrate the scalability of the results

Architecture Details

We implemented a Wasserstein Generative Adversarial Network Arjovsky et al. (2017) with Gradient Penalty (WGAN-GP) Gulrajani et al. (2017) to generate multi-channel EEG data. The model consists of two core components: A generator, responsible for producing synthetic EEG signals, and a discriminator—referred to as a "Critic" in the WGAN framework—which evaluates whether the generated data is real or
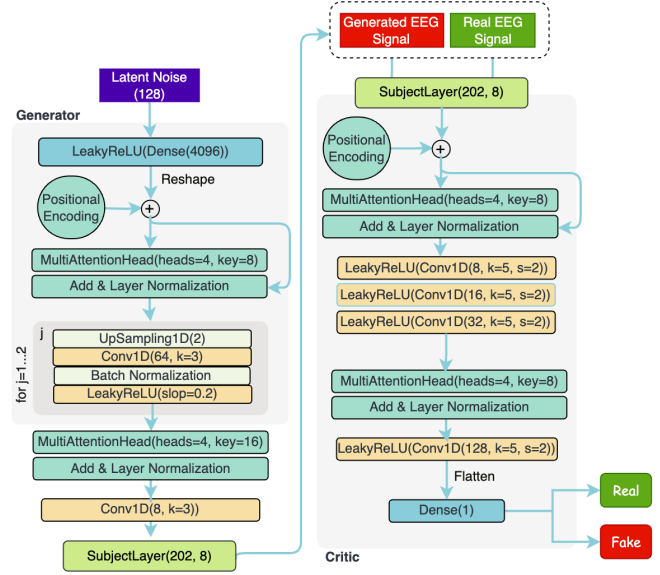
synthetic.

Figure 1: Architecture of the proposed model. This architecture combines DCGAN elements with Transformer-inspired attention modules, to more effectively capture both local and long-range temporal dependencies in EEG time-series data.

Our architecture follows the Deep Convolutional GAN (DC-GAN) (Radford (2015)) framework, incorporating a stack of convolutional layers with upsampling layers in the generator and strided convolutional layers in the Critic (Figure 1). For upsampling, we used linear interpolation, as it has been shown to produce significantly fewer high-frequency artifacts compared to nearest-neighbor upsampling (Hartmann et al. (2018)).

To improve the model's ability to capture long-range dependencies in EEG signals, we integrated a one-dimensional self-attention mechanism into both the generator and the Critic. This self-attention layers includes: (1) A multi-head attention mechanism applied along the time dimension. (2) a layer normalization module, and, (3) and a learnable positional embedding layer, similar to how they are used in Transformers. Instead of using fixed sinusoidal embeddings (as in Vaswani (2017)), we use trainable positional embeddings for each time step, which are added to the input tensor before feeding the sequence into the multi-head attention block. Our ablation experiment showed that this self-attention layer is essential in model performance in generating realistic and diverse EEG data. All weights in the convolutional and dense layers were initialized using a random normal distribution with a mean of 0 and a standard deviation of 0.02. This initialization has been shown to stabilize GAN training and prevent divergence (Radford (2015)).

To account for individual variability across participants and enhance diversity in the generated EEG signals, we incorporated a participant-specific transformation layer. Inspired by

(Défossez et al. (2023)), this layer is applied after the generator produces synthetic data learns a participant-specific transformation matrix $M \in \mathbb{R}^{D1 \times D1}$ for each participant $s \in [S]$. This modification was critical in preventing generator mode collapse — a scenario where the generator produces only a limited set of examples to deceive the critic. By leveraging this layer, the generator was able to produce more diverse outputs.

Training

As the name WGAN-GP indicates, in the training of this network we used the Wasserstein loss function.

$$L = \underbrace{\mathbb{E}_{z \sim p(z)}\big[D(G(z))\big] - \mathbb{E}_{x \sim p_r}\big[D(x)\big]}_{\text{Wasserstein loss}} + \underbrace{\lambda \, \mathbb{E}_{\hat{x} \sim p_{\hat{x}}}\Big[\big(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\big)^2\Big]}_{\text{gradient penalty}}$$

The first term, estimates the distance between the real data distribution and the generated data's distribution using the Wasserstein distance (also known as the Earth Mover's Distance). The second term, known as gradient penalty, ensures that the Critic function remains close to 1-Lipschitz by penalizing deviations of the gradient norm from 1. $P\hat{x}$ denotes the distribution of points obtained by sampling uniformly along the line segments between real and generated samples. lambda λ is the gradient penalty coefficient. We set the gradient penalty weight to 10.

The generator's loss remains:

$$L_G = \mathbb{E}_{z \sim p(z)}\big[D(G(z))\big]$$

since the generator aims to produce samples that maximize $D(G(z))$ (or equivalently, minimize $-D(G(z))$), thereby reducing the estimated Wasserstein distance between the generated and real distributions.

Both the generator and critic were optimized using the Adam optimizer ($beta_1 = 0.5, beta_2 = 0.9$), with an initial learning rate of 0.00094 and a batch size of 128. The learning rate was reduced progressively using an exponential decay schedule with a decay step of 100,000 and a decay rate of 0.90. This gradual reduction in the learning rate was crucial to preventing training instabilities, such as model blow-ups or divergence.

Latent variables z for the generator were sampled from a normal distribution with the mean and standard deviation of the real data. To ensure the Critic remains well-trained and provides meaningful feedback to the Generator, the Critic was updated twice per each Generator update. Specifically, in each training iteration, two full forward and backward passes were performed on the Critic using both real and generated samples before updating the Generator once.

Data shuffling across batch dimensions was enabled in the Keras fit function, ensuring that each training iteration included segments from different subjects. The model was trained for 900 epochs (129,600 steps) on an NVIDIA V100 GPU with mixed precision using CUDA 12.7. The entire model development and training process was implemented using Keras

(v3.5.0) with the PyTorch (v2.4.1) backend. The implementation code for this model is available at: `https://github.com/Yeganehfrh/YARE-GAN`

Evaluation Metrics

To assess the quality of the generated EEG data, we conducted both visual inspections and quantitative comparisons between real and generated signals. We first examined the generated signals in the time and frequency domains to check for realistic waveform structures and spectral properties. Next, we quantified the similarity between real and synthetic data by extracting several statistical and spectral features, including:

Hjorth parameters (activity, mobility, complexity) to capture signal variance, frequency composition, and dynamic complexity Hjorth (1970).

Higher-order statistics, such as kurtosis and skewness, to analyze the distribution of EEG amplitudes.

Relative Power spectral features across standard EEG frequency bands (delta, theta, alpha, beta, gamma) to compare oscillatory activity.

Functional connectivity between EEG channels to ensure that the generated data maintained realistic inter-channel relationships. We computed pairwise cosine similarity between EEG channels as a proxy measure of functional connectivity.

To track how well the model learned the statistical properties of real EEG, we computed the **Fréchet Distance (FD)** between the feature distributions for each EEG channel throughout training, measured every 50 epochs. This allowed us to monitor the learning process, identify potential challenges in capturing specific channels' features.

Classification task: reusing the Critic's intermediate representations for a downstream task

To assess whether the representations learned in the intermediate layers of the GAN were useful for a new task, we repurposed the Critic for gender classification. Specifically, we extracted features from its convolutional and attention layers, applied MaxPooling (with a fixed pool size of 4) to reduce dimensionality, then flattened and concatenated the resulting outputs. This process yielded an 896-dimensional activation vector, which served as the input to a downstream classifier. Our approach follows the method proposed in Radford (2015)

Our classification model consisted of a simple dense layer with 64 units, both using the GELU activation function. The final output layer was a dense layer with a Sigmoid activation function. This model received the learned embeddings from the Critic and predicted the corresponding gender class.

For this classification task, we used a publicly available EEG dataset from OpenNeuro (Kekecs et al. (2023)). Importantly, this dataset was entirely independent and had not been used during GAN training. It consists of resting-state EEG recordings from 52 participants (39 female, mean age 24.5), each with approximately 30 minutes of data. To achieve a balanced gender distribution, we resampled the data to include an equal number of male and female participants, resulting in

a final subset of 26 participants. 80% of the data was used for training and 20% for testing.

We trained the model using binary cross-entropy loss with the Adam optimizer (learning rate = 0.001). Training was conducted for 1000 epochs with a batch size of 128. To evaluate model performance, we used accuracy as the primary metric.

As baselines for the classification task, we included two comparisons. First, a randomized baseline, in which we shuffled the labels to test whether the model was learning meaningful patterns rather than overfitting to noise. Second, we used the minimally processed EEG data (as described in the methods) to predict participants' age, using the same held-out dataset. For this, we flattened each EEG segment across the time (512) and channel (8) dimensions into a single 4096-length vector. This resulted in an input shape of (batch size, 4096), which was then fed into the same classifier architecture used for the primary task.

Results

Training Statistics

Figure 2 shows the generator and Critic loss curves over 900 training epochs. Throughout this process, we also continuously monitored the gradient penalty, the norm of the critic's gradient, and the critic's prediction of the real and fake data, to ensure stable learning and prevent divergence (Figure 2).

Result 1: The proposed architecture is successful in Generating multi-channels EEG data

Figure 4 presents a visual comparison between real and generated EEG signals, highlighting their average waveforms and standard deviations in both the time and frequency domains. Despite the model not being explicitly trained on frequency-domain data, the power spectral density (PSD) analysis shows that it successfully captured the spectral characteristics of different EEG channels.

In the time domain signals remain within the same dynamic range, demonstrating that the model effectively learned key temporal properties. Also, principal component analysis (PCA) was used to assess the similarity between real and generated data in a lower-dimensional space. By reducing each EEG (512, 8) segment to 2 principal components, the PCA plot reveals real and generated EEG data exhibit similar distributions in a lower-dimensional space (Figure 3a). Furthermore, the overall distribution of real and synthetic data across all channels (Figure 3b) shows a strong resemblance, further confirming that the model successfully learned the underlying statistical properties of the EEG signals.

To further strengthen the qualitative evaluation ("eye test"), figure 5 compares the functional connectivity of real and generated signals using cosine similarity. This adds a structure-sensitive perspective beyond time- and frequency-domain comparisons. The high degree of similarity in connectivity patterns suggests that the generative model successfully captures realistic inter-channel relationships in the EEG data.

To quantify the similarity between real and generated EEG signals, we computed the Fréchet Distance (FD) on features extracted from both the time and frequency domains. Figure 6 illustrates how FD evolves during training across different electrode regions—frontal, central, parietal, and occipital.

As the graph indicates, FD gradually decreases over the course of training, suggesting that the generator becomes increasingly better at producing realistic EEG data. However, a breakdown of FD across different EEG channels reveals that the model learns lower-frequency components more effectively than higher-frequency components. It might be because those signals naturally have higher energy, making them easier for the network to learn. Additionally, the generator struggles more with frontal EEG channels compared to posterior regions, particularly in the higher gamma frequency range. This suggests that higher-order cognitive activity patterns, which are often more complex in frontal regions, are more challenging for the model to replicate.

Result 2: The proposed Critic learns useful representation, making it reusable for a downstream tasks (gender classification)

To evaluate whether the Critic's learned representations are transferable to a new task, we assessed its performance on gender classification. As shown in Figure 7, the model achieved a validation accuracy of 70%, substantially outperforming both a shuffled-label baseline and recent EEG foundation models (Figure 7c), confirming that the Critic has captured meaningful EEG features.

These results indicate that, in the process of discriminating between real and generated EEG data, the Critic has learned transferable representations of EEG structure that can be reused for an entirely new classification task using new dataset that was never seen during GAN pre-training. This underscores the potential of unsupervised representation learning with GAN-based architectures in EEG research, offering a promising path toward reducing reliance on manual feature engineering.

To further support this claim, we evaluated the Critic's intermediate representations on an additional downstream task—classifying participants' level of hypnotic suggestibility, as measured by the Harvard Group Scale of Hypnotic Susceptibility (HGSHS). Unlike gender, this label is derived from self-assessment and may be inherently noisier due to variability in subjective reporting and individual interpretation. Despite this, the model achieved a validation accuracy of 65% (chance level: 50%), indicating a meaningful signal in the learned features. More details on this experiment are provided in the supplementary materials. These findings further reinforce the robustness and generalizability of the Critic's representations across tasks and datasets, even in the face of ambiguous and less objective labeling.

Discussion and Conclusion

In this study, we implemented a Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP) to gen-
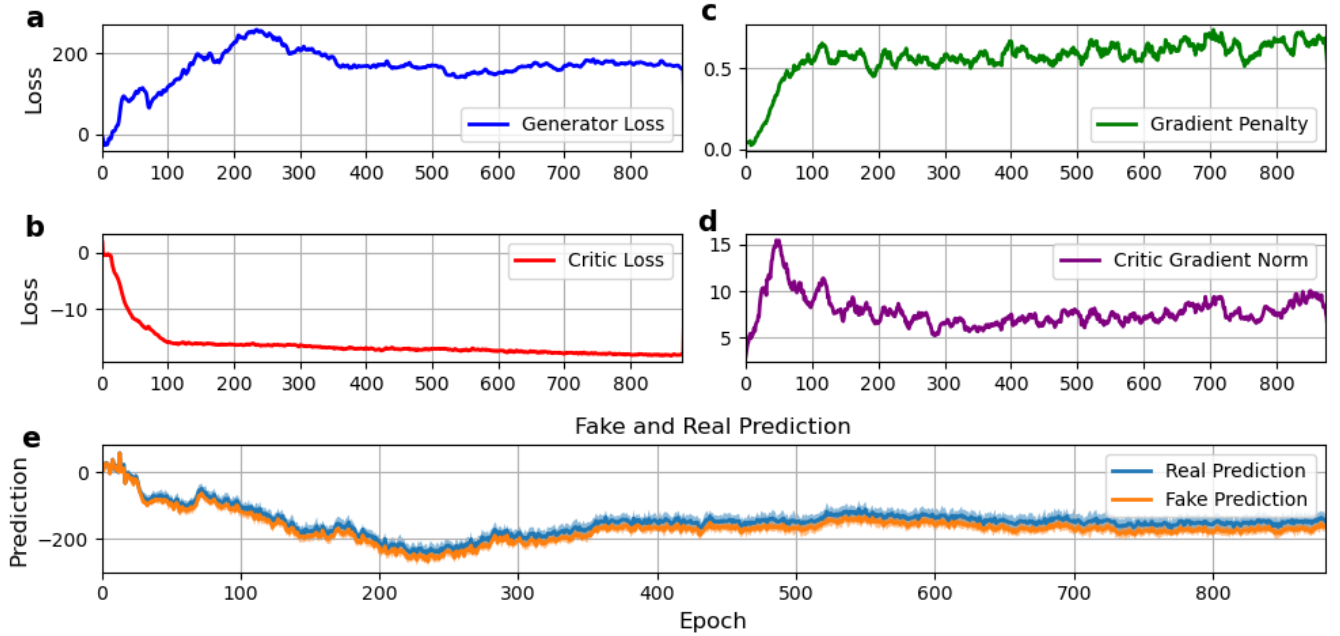
Figure 2: Training Statistics. Each graph shows key metrics tracked over 900 training epochs: (a) Loss curves for the Generator (b) and Critic. (c) Gradient penalty (the L2 norm of the gradient of the Critic with respect to its input data) and (d) Critic gradient norm (the gradient magnitude of the Critic with respect to its weights). (e) The average Critic predictions for real and fake data at each epoch. The Critic consistently assigns higher values to real data, indicating that it continues to learn and has not been fully deceived by the Generator.

Figure 3: (a) PCA visualization of real (blue) and generated (orange) EEG data in a two-dimensional space, showing the degree of overlap between the distributions. Each point represents an individual EEG segment. (b) Global distribution comparison of real and generated data, aggregated across all EEG channels and time points, illustrating their overall alignment.

erate multi-channel EEG data and evaluated the quality of the synthesized signals using both visual inspection and feature-based quantitative measures. Our results demonstrate that the model successfully learned key statistical and spectral characteristics of EEG signals, as evidenced by the decreasing Fréchet Distance (FD) over training epochs. However, we observed that the generator struggled more with replicating high-frequency components (gamma band) in the frontal region, suggesting that these aspects of neural data are more complex to capture.

Beyond data generation, we assessed the transferability of the learned features by by using the Critic's intermediate representations for a downstream gender classification task. The model achieved a validation accuracy of 70% without requiring any fine-tuning on out-of-sample data, significantly outperforming a shuffled-label baseline and the classifier trained using minimally processed EEG data. These results suggest that the GAN's Critic effectively captured meaningful EEG features that generalize across tasks. This highlights the promise of unsupervised representation learning in reducing dependence on manual feature engineering in EEG analysis, and marks a step toward the development of general-purpose foundation models for EEG.

Limitations Despite the promising results, our approach has several limitations. First, the training data was derived from a single dataset (LEMON) and limited to eight selected EEG channels, which may restrict the model's ability to generalize to datasets with different recording conditions or additional EEG channels distributed across the scalp. Although we demonstrated that the current model can scale to include more EEG channels (see Supplementary Materials), further experimentation is still needed to achieve more stable training

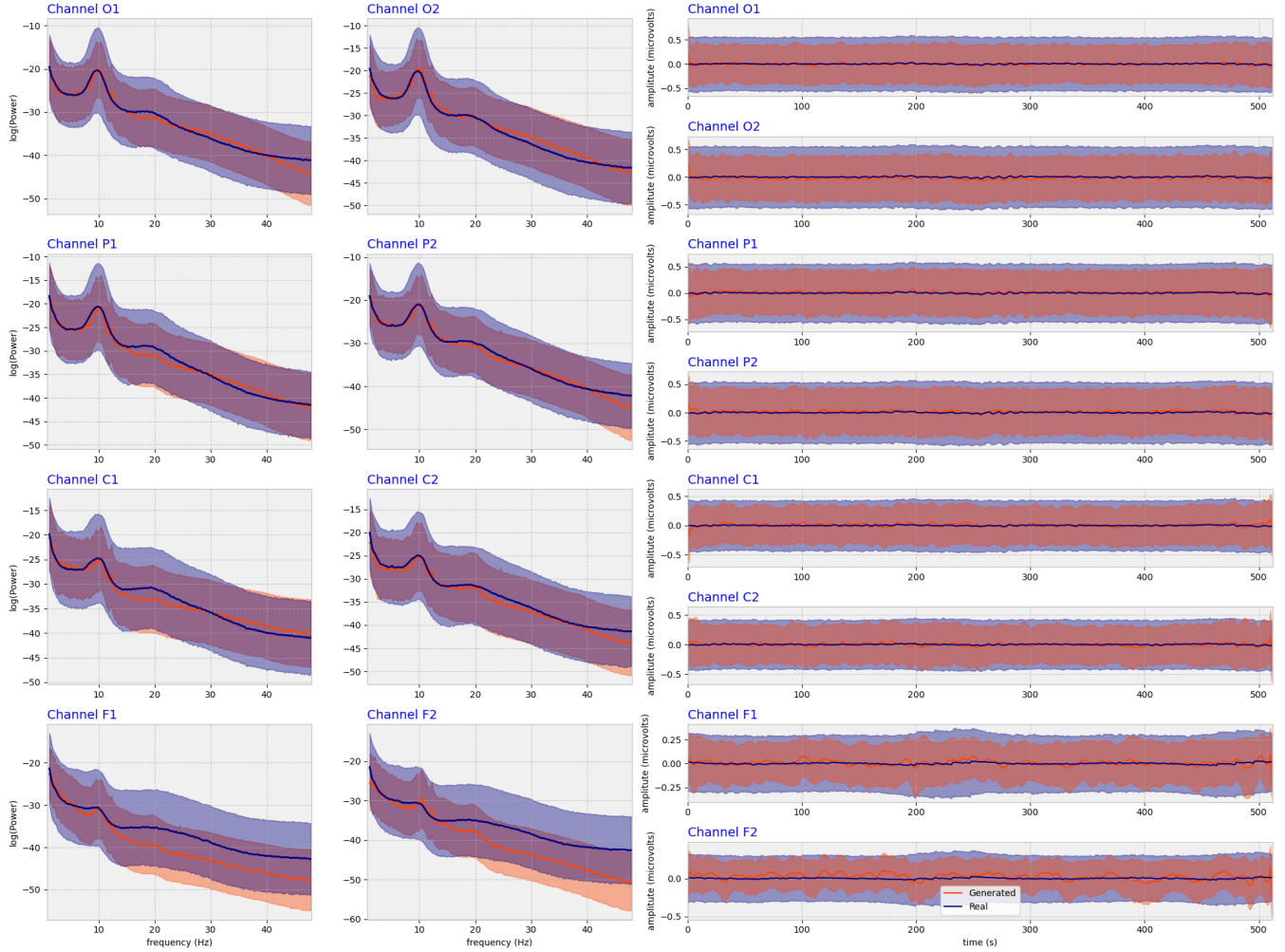Second, we have not yet benchmarked our model against

Figure 4: Visual inspection of generated and real data in the frequency and time domains across eight different EEG channels. The power spectral density (PSD) and time-series data are averaged over segments (batch dimension), with the shaded regions indicating the standard deviation. In all subplots, the blue line represents the real data, while the orange line shows the generated data.
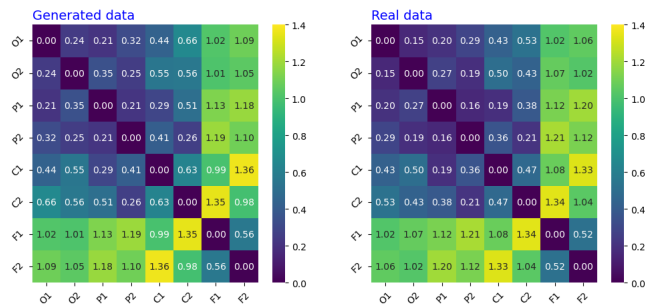

Figure 5: Cosine similarity matrices in generated and real data. The matrices show the similarity between signals from different EEG channels, demonstrating the model's ability to reproduce realistic functional connectivity patterns

other EEG representation learning models. Including such comparisons would offer deeper insights into our model's strengths and highlight areas for potential improvement.

Future Directions Moving forward, we aim to:

- Expand training data by incorporating larger, more diverse EEG datasets, including task-related and clinical recordings to improve generalizability.

- Enhance high-frequency EEG generation.

- Explore broader applications of the learned representations in tasks such as cognitive state decoding and neurological disorder detection.

- Improve model interpretability by analyzing the latent representations learned by the GAN. Specifically, we plan to investigate which EEG channels contribute most to the
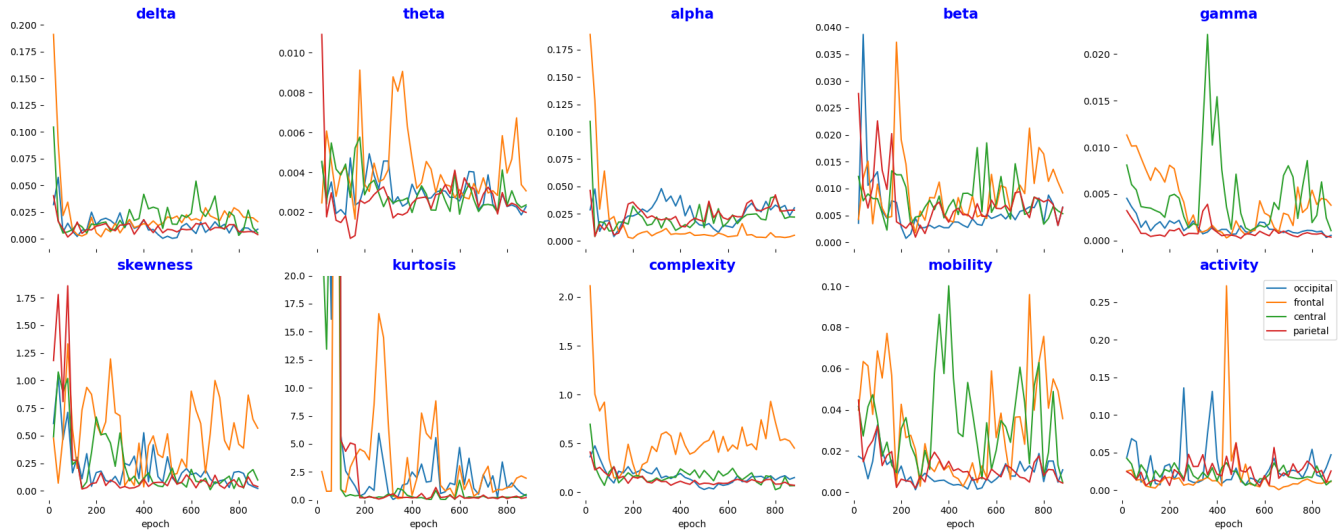
Figure 6: Fréchet Distance (FD) over training epochs. The upper row shows the FD between real and generated data for spectral features, measured as power within standard EEG frequency bands (delta, theta, alpha, beta, gamma) across different electrode regions. The lower row presents the FD evolution for manually extracted time-domain features, including Hjorth parameters (complexity, mobility, activity), skewness, and kurtosis. A decreasing FD over epochs indicates that the generated data increasingly resembles real EEG signals, though differences remain across frequency bands and electrode locations.

learned features, as well as the attention weights in downstream classification tasks. This analysis will help identify the most influential features in the model's internal representations.

Overall, this study demonstrates the promise of GAN-based unsupervised learning for extracting meaningful representations from EEG data, paving the way for future advancements in data-efficient deep learning applications in neuroscience.

References

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning* (pp. 214–223).

Babayan, A., Erbey, M., Kumral, D., Reinelt, J. D., Reiter, A. M., Röbbig, J., ... others (2019). A mind-brain-body dataset of mri, eeg, cognition, emotion, and peripheral physiology in young and old adults. *Scientific data*, *6*(1), 1–21.

Botvinik-Nezer, R., Holzmeister, F., Camerer, C. F., Dreber, A., Huber, J., Johannesson, M., ... others (2020). Variability in the analysis of a single neuroimaging dataset by many teams. *Nature*, *582*(7810), 84–88.

Défossez, A., Caucheteux, C., Rapin, J., Kabeli, O., & King, J.-R. (2023). Decoding speech perception from non-invasive brain recordings. *Nature Machine Intelligence*, *5*(10), 1097–1107.

Delorme, A. (2023). Eeg is better left alone. *Scientific reports*, *13*(1), 2372.

Fahimi, F., Dosen, S., Ang, K. K., Mrachacz-Kersting, N., & Guan, C. (2020). Generative adversarial networks-based data augmentation for brain–computer interface. *IEEE transactions on neural networks and learning systems*, *32*(9), 4039–4051.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, *27*.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of wasserstein
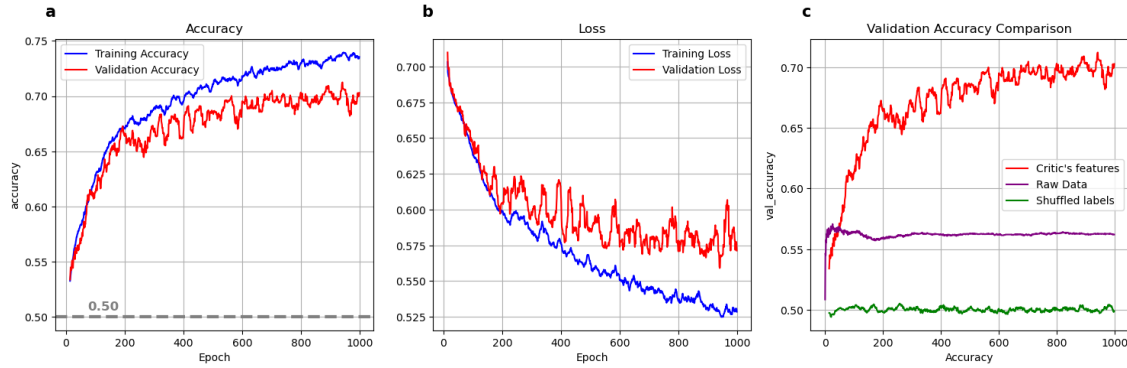
Figure 7: Classifier performance on the gender classification task using intermediate representations from the Critic network. (a) Training and validation accuracy over 1000 epochs, with horizontal dashed lines indicating the equal prevalence of each class at 50%. (b) Training and validation loss progression throughout the training process. (c) The validation accuracy trajectory compared to two baselines: a shuffled-label model and a classifier trained directly on EEG data

gans. *Advances in neural information processing systems*, *30*.

Hartmann, K. G., Schirrmeister, R. T., & Ball, T. (2018). Eeggan: Generative adversarial networks for electroencephalograhic (eeg) brain signals. *arXiv preprint arXiv:1806.01875*.

Hjorth, B. (1970). Eeg analysis based on time domain properties. *Electroencephalography and clinical neurophysiology*, *29*(3), 306–310.

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, *33*, 6840–6851.

Kekecs, Z., Girán, K., Vizkievicz, V., Lutoskin, A., & Farahzadi, Y. (2023). *The effects of sham hypnosis techniques.* https://openneuro.org/datasets/ds004504. (OpenNeuro, Dataset ds004504, Version 1.0.0) doi: 10.18112/openneuro.ds004572.v1.3.0

Kingma, D. P. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Liang, Z., Zhou, R., Zhang, L., Li, L., Huang, G., Zhang, Z., & Ishii, S. (2021). Eegfusenet: Hybrid unsupervised deep feature characterization and fusion for high-dimensional eeg with an application to emotion recognition. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *29*, 1913–1925.

Murphy, K. P. (2022). *Probabilistic machine learning: an introduction*. MIT press.

Pan, B., & Zheng, W. (2021). Emotion recognition based on eeg using generative adversarial nets and convolutional neural network. *computational and Mathematical Methods in Medicine*, *2021*(1), 2520394.

Pascual, D., Amirshahi, A., Aminifar, A., Atienza, D., Ryvlin, P., & Wattenhofer, R. (2020). Epilepsygan: Synthetic epileptic brain activities with privacy preservation. *IEEE Transactions on Biomedical Engineering*, *68*(8), 2435–2446.

Radford, A. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Robbins, K. A., Touryan, J., Mullen, T., Kothe, C., & Bigdely-Shamlo, N. (2020). How sensitive are eeg results to preprocessing methods: a benchmarking study. *IEEE transactions on neural systems and rehabilitation engineering*, *28*(5), 1081–1090.

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

Wang, J., Zhao, S., Luo, Z., Zhou, Y., Jiang, H., Li, S., . . . Pan, G. (2024). Cbramod: A criss-cross brain foundation model for eeg decoding. *arXiv preprint arXiv:2412.07236*.

Supplementary

Effects of Different Preprocessing Choices

The preprocessing of the EEG data was mainly motivated by a desire to avoid introducing any frequency-domain transformations (such as filtering) before handling raw time-domain features. This choice was made to preserve the temporal structure of the signal during early preprocessing and avoid potential artifacts introduced by edge effects during filtering. This was the reason that we first did the baseline correction and then applied a high-pass filter.

We acknowledge that standard practice often recommends the reverse order. Therefore, to evaluate the impact of the preprocessing order, we repeated the entire preprocessing pipeline with the revised ordering — applying the high-pass filter (0.5 Hz) before baseline correction — and compared the resulting signals across both time and frequency domains.

Results showed extremely high similarity between the two approaches: **Time domain**: Mean Pearson correlation = 0.9999 ± 0.0017, minimum = 0.9375 **Frequency domain**: Mean Pearson correlation = 0.9999 ± 0.0003, minimum = 0.9912

These correlations were computed per participant and per channel across 202 subjects and 8 EEG channels. The correlation heatmap in Figure 8 shows the correlation between two different preprocessing across channels for a subset of the participants:
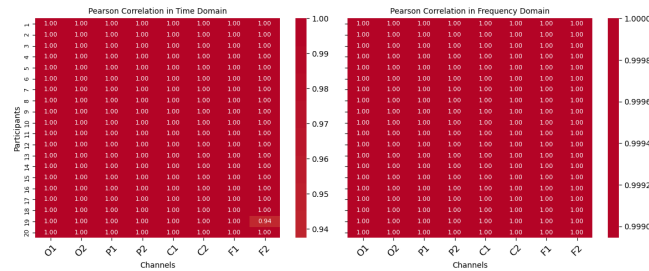

Figure 8: Comparison between signals processed with different preprocessing ordering

Time series plots (e.g., channel F2) and power spectral density comparisons using Welch's method also visually confirm the near-identical nature of the signals, however there is a vertical shifts in amplitude (Figure 9 and Figure 10):
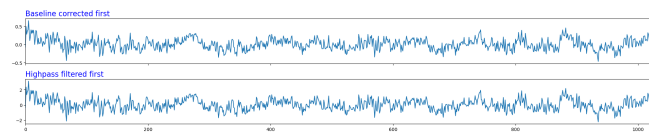

Figure 9: Time series of two signals processed using different preprocessing ordering

This shift can be interpreted as an offset transformation along the y-axis, which does not alter the temporal structure,
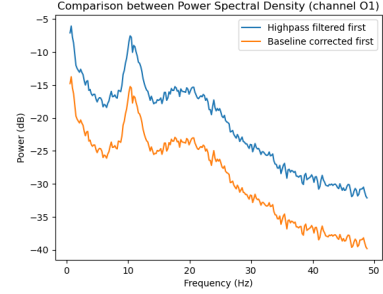

Figure 10: Effect of preprocessing order on signal Power Spectral Density
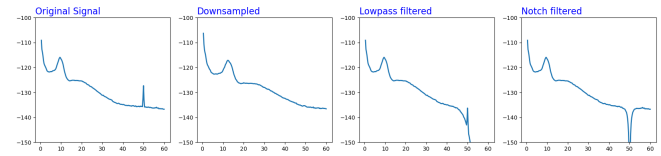

Figure 11: Comparison between different methods for eliminating the line noise at 50 Hz

spectral content, or phase dynamics of the signal. From the perspective of deep learning — particularly in architectures such as convolutional networks or GANs — this kind of transformation is unlikely to affect the learning process, as such models are largely invariant to constant additive shifts when using normalized or standardized inputs (as is the case here).

As for our choice for downsampling, we downsampled the data to 98 Hz as a practical strategy to eliminate 50 Hz line noise without introducing distortions commonly caused by filtering. Based on the Nyquist theorem, frequencies above 49 Hz are attenuated, naturally removing the 50 Hz component. While notch or low-pass filters are standard alternatives, they can significantly alter the shape of the power spectral density (PSD), especially in the gamma ranges.

As shown in the figure 11 (channel C2 as an example), both notch and low-pass filters introduce sharp discontinuities and frequency band compression, which degrades the realism of generated signals in our empirical evaluations. In contrast, downsampling preserved the natural spectral profile while effectively removing line noise. All filters were implemented using the scipy.signal module (v1.15.1)

Progression of similarity between real and generated signals across training epochs

To visualize how the distribution of generated signals evolves during training, we performed PCA on the real and generated data at various epochs. The 2D projections (figure 12 shows how the GAN progressively learns to approximate the real data distribution. Initially, the generated samples form a compact cluster, but over time, they gradually expand and overlap with the real data manifold, indicating improved realism and diversity.
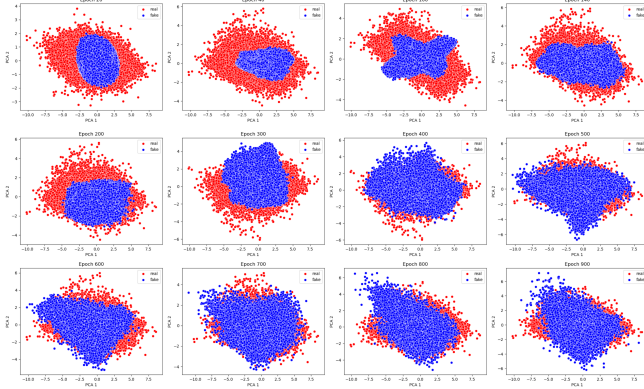
Figure 12: Progression of the GAN's learning over training epochs. PCA projections of real (red) and generated (blue) EEG samples are shown at selected epochs.
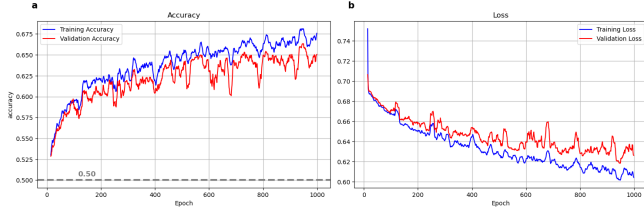

Figure 13: Classification performance on hypnotic suggestibility using the Critic's EEG embeddings. (a) Training and validation accuracy across 1000 epochs. The dashed line marks the 50% chance level. (b) Corresponding loss curves.

Hypnotic Suggestibility Classification

To further test the generalizability of the Critic's learned EEG representations, we applied the same classification setup as used in the gender prediction task (described in the main text) to a new task: predicting participants' level of hypnotic suggestibility. Labels were based on the Harvard Group Scale of Hypnotic Susceptibility (HGSHS), a subjective self-report measure.

We used the same publicly available EEG dataset from OpenNeuro as for the gender classification task, but selected a different subsample to ensure a balanced distribution of low and high hypnotizable participants, yielding a final subset of 34 participants.

Despite the increased label noise, the classifier achieved a validation accuracy of 65%, significantly above the 50% chance level (figure, 13), highlighting the robustness and transferability of the Critic's representations even in more ambiguous classification tasks.

Proposed Architecture's Scalability

To assess the scalability of the results, we compared the model's performance with additional 16 and 56 channels. In this setup, the architectural components such as the embedding dimension in positional encoding, the size of each self-attention head (query/key), and the number of feature maps in
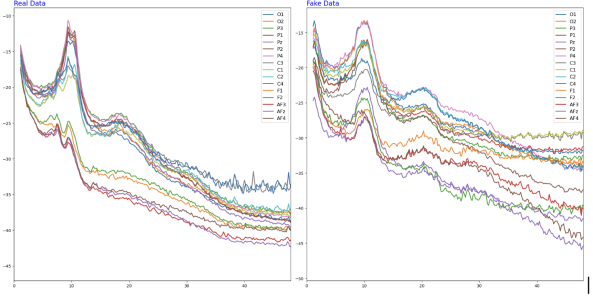

Figure 14: Comparison of the power spectral density between real and generated signals when using 16 EEG channels. This graph shows the results after 150 epochs of training.
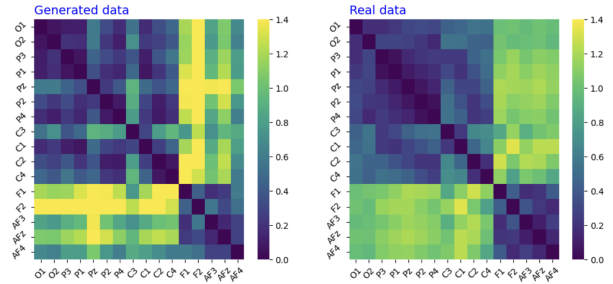

Figure 15: Comparison of cosine similarity between real and generated signals when using 16 EEG channels

convolutional layers were scaled proportionally to the number of input channels. We trained these models for 150 epochs and compared their performance at the same training stage. Table 1 summarizes this comparison:

The results indicate that training time does not scale linearly with the number of channels, an encouraging sign for scalability.

The following graphs 14, 15, 16, 17 compare the real and generated signals in terms of their power spectral density and connectivity matrix when 16 and 56 channels used, after 150 epochs of training.
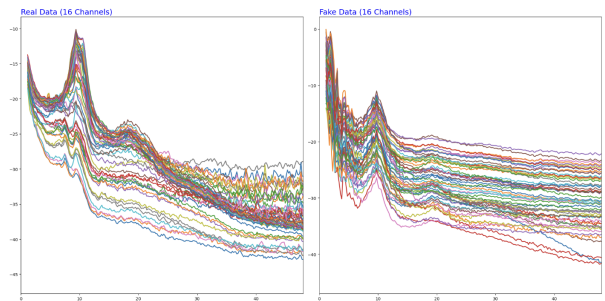

Figure 16: the power spectral density between real and generated signals when using 56 EEG channels

Channels	8	16	56
Total Trainable Parameters	636,721	872,009	8,434,385
Training speed (Milliseconds per step)	197	200	350
FD (Spectral Features; normalized by number of channels)	0.163	0.173	0.965
FD (Hjorth parameters; normalized by number of channels)	0.182	0.179	4.168

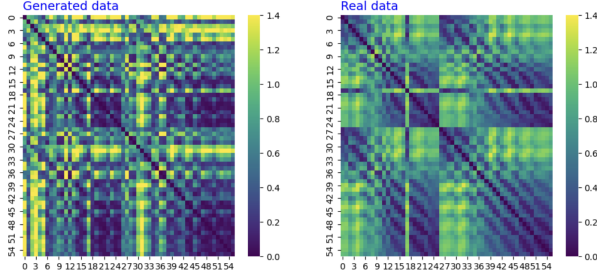Table 1: Comparison of model performance for different channel configurations.

Figure 17: cosine similarity between real and generated signals when using 56 EEG channels

More details on design choices during model development

Hyperparameter choices: Training parameters (e.g., GP weight, batch size, learning rates) were tuned empirically to ensure stable GAN training and avoid mode collapse. We chose a relatively large batch size (128), as our hardware allowed for it. This improves gradient estimates but requires lowering the learning rate to maintain stability. Gradient penalty weight was set based on the original WGAN-GP paper, but we also tested alternative values and retained the one that yielded the most stable training.

Architectural choices (e.g., number of neurons) are scaled according to the number of EEG channels as explained above in the model's Scalability section. The general structure of our model combines a DCGAN-inspired architecture with self-attention layers and positional encoding to better model temporal dependencies in EEG time-series data.

Low-Dimensional Visualization of Real and Generated EEG Features Using UMAP

In addition to PCA, we applied UMAP (Uniform Manifold Approximation and Projection) as a dimensionality reduction technique to assess the similarity between real and generated data in a lower-dimensional space. Each EEG segment, originally shaped (512, 8), was reduced to two dimensions. We then applied the HDBSCAN clustering algorithm to these 2D UMAP components. The results indicate that the regions where the real and fake data do not overlap are primarily labeled as noise by HDBSCAN (i.e., assigned a label of -1). In contrast, the dense core region—representing high-density areas—shows substantial overlap between real and fake samples, although not entirely complete. Figure 18 presents a 2D UMAP visualization of the real and generated signals.
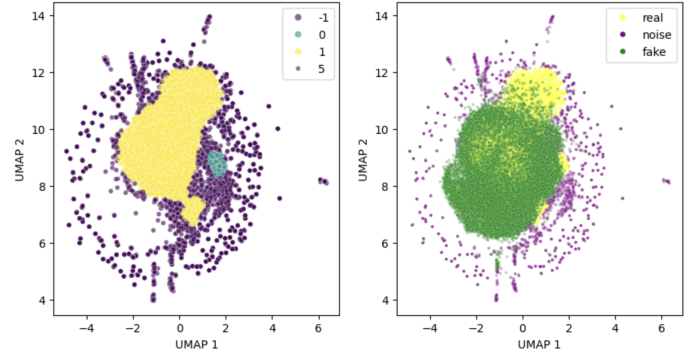

Figure 18: UMAP visualization of real (blue) and generated (orange) EEG data. the left subplot displays the HDBSCAN clustering labels of the 2D UMAP representation of the real data. The right subplot shows both real and fake samples in the same 2D space, color-coded accordingly, with the noise-labeled points from the real data highlighted