Synthetic Signals Generation

Jose Andres Reyes Cisneros
Introduction to Biomedical Signal
Lima, Perú
jose.reyes@upch.pe

Sebastian Adolfo Rios Quintanilla
Introduction to Biomedical Signal
Arequipa, Perú
sebastian.rios@upch.pe

Luis Gustavo Loja Mauricio
Introduction to Biomedical Signal
Piura, Perú
luis.loja@upch.pe

Gloria Kimberly Paurcar Centeno
Introduction to Biomedical Signal
Cusco, Perú
gloria.paucar@upch.pe

Jose Enrique Cebrián Baca
Introduction to Biomedical Signal
Piura, Perú
jose.cebrian@upch.pe

Abstract —The purpose of this work is to develop a simple, 1D-CNN based, Generative Adversarial Network model for the generation of synthetic EKG signals. Qualitatively our model has good results, but t-SNE evaluation shows that the synthetic signals' statistical distribution remains different from that of real ECG signals. We propose, in addition, ways to further improve performance for later experiments with additional Feature Extraction and a different loss function.

Keywords—electrocardiogram, signal, GAN, network, model

I. Introduction

The electrocardiogram (EKG) is a key diagnostic tool for evaluating a patient's heart condition. Automated ECG interpretation algorithms as diagnostic aids provide a great convenience for medical staff, depending only on the number of ECGs performed routinely. However, the development of such algorithms requires large training data sets and clear standard procedures.

Biomedical signals have high amounts of noise, naturally stochastic, with a great variety across individuals and within themselves. Which makes them hard to characterize.

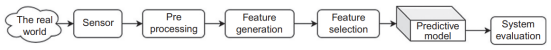


Fig 1. Typical pattern recognition system [1]

The proper characterization of a biomedical signal allows the construction of robust models (consistently accurate) that learn the most outstanding structures of the data. For example, sliding filter operations such as Fourier transform or extracting descriptive statistical measures of local regions in the data to obtain new representations for them. [1]

II. Problematic

Knowing the shapes that make up an EKG signal is essential not only for doctors, nurses, and clinical staff; but also for professionals such as Biomedical Engineers or any professional intrigued in the physiology of the heart. Therefore, it is necessary to show the EKG graph where the waves, segments, and complexes that compose them stand out, but there are not many databases that can help us in this task. Hence, the software is needed which generates simulated data and indicate the segmentation of its corresponding waves.

III. Objective

Develop a model of neural networks that allow the generation of synthetic ECG signals.

IV. Justification

Many machine learning solutions for niche domains are limited by the availability of publicly available datasets. This is the case for medicine, for instance, as access to medical data is highly monitored and this hinders the comparison and reproducibility of models. The main motivation for this project is then the need for realistic, synthetic medical data that could be used in other tasks.[2]

V. Theoretical framework

A. Definition of a neural network

Artificial neural networks are a model inspired by the functioning of the human brain. It is made up of a set of nodes known as artificial neurons that are connected and transmit signals to each other. These signals are transmitted from the input to generate an output. [3]

1. Functioning of a neural network

As mentioned, the functioning of the networks is similar to the human brain. The networks receive a series of input values and each of these inputs arrives at a node called a neuron. The neurons of the network are grouped into layers that form the neural network. Each of the neurons in the network has a weight, a numerical value, with which it modifies the input received. The new values are obtained to leave the neurons and continue on their way through the network. This operation can be seen schematically in the following image. Once the end of the network has been reached, an output is obtained that will be the prediction calculated by the network. [4]

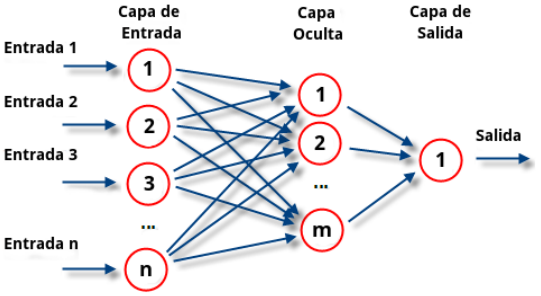


Fig 2. Structure of a functional network [3]

B. Definition of a "GAN"

Generative Adversarial Networks, or GANs for short, is a type of neural network in which two different networks are trained simultaneously.

One of the networks has a focus on generation while its counterpart is on discrimination. This type of training has been gaining fame due to its usefulness in eliminating translation variations in the domain, as well as its effectiveness in generating new samples. The latter is of great importance for our research, however, it is necessary to carry out a diverse analysis of the different variations of GAN, such as the critical analysis of the advantages and disadvantages that it provides towards the solution of our problem.

1. Description of the operation of a basic gan (vanilla gan)

This generative model is designed to extract samples from the desired data distribution without the need to model the probability density function.

The generator, usually named G, at its input z is noise sampled from a prior distribution p(z) that is commonly chosen Gaussian or Uniform Distribution; the output of G is "x_g" which is expected to have a similar display with the real sample or x_r that is drawn from the real data distribution p_r(x).

On the other hand, discriminator D is provided with real or generated data, its output y_1 is a single variable that indicates the probability that the input was a real or generated sample.

The objective of D is to differentiate these groups while G is trained to confuse the discriminator D, however, how much it confuses depends on various parameters and the focus of the need to implement a GAN.

"Intuitively, G can be seen as a replicator trying to reproduce quality material, while D behaves like an officer (police) trying to detect the quality of the products." [5]

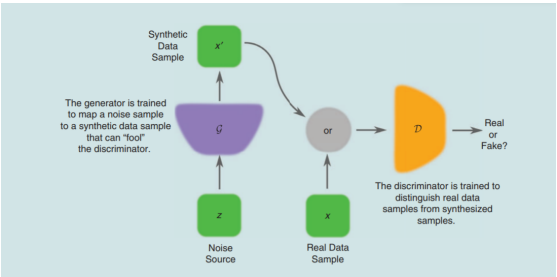


Figure 3. Example of a basic GAN and the scheme of its operation. [6]

2. Disadvantages of using the gan

Modal collapse: Natural data distributions are highly complex and multimodal. That is, the data distribution has many modes, each mode represents a concentration of similar data samples, but is different from the other modes.

During a modal collapse, the generator produces samples belonging to a limited set of modes. This happens when the generator thinks it can fool the discriminator by blocking only one mode, without evaluating the others. That is, the generator produces samples exclusively from this mode.

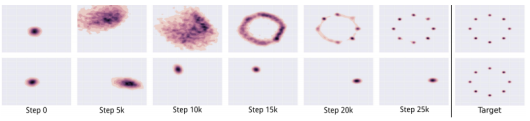


Figure 4. The image above represents the output of a GAN without mode collapse. The image below represents the output of a GAN with mode collapse. [7]

Convergence: Since the generator loss improves as the discriminator loss degrades and vice versa, we cannot judge convergence based on the value of the loss function. This is illustrated in the following image:

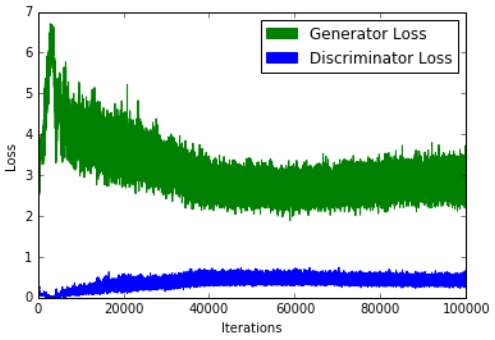


Figure 5. A plot of a typical GAN loss function [8]

VI. State of the art

A. Pseudo-realistic synthetic signal generation system

This work was based on the presentation of a synthetic signal generation system with a pseudo-realistic character for its use applied to the validation of methods and design of experiments. The generation of synthetic data has the characteristic of reducing waiting times compared to the long periods that some sensors need to obtain large volumes of samples. The data generated can be as robust as users need. The signal generation method that is used makes use of statistical models and the gradient behavior of the signal to generate new data, thus using random number generation functions to construct a signal with a given number of samples from a given limited range number of samples. The synthetic signal generation method must be able to generate digital samples that are plausible from a qualitative point of view. Furthermore, the resulting signals must satisfy the Nyquist and Shannon theorem. [9]

B. Syn siggan: generative antagonical networks for the generation of synthetic biomedical signals

This article presents a new generative model of antagonistic networks to generate synthetic biomedical signals. Synthetic data can be used to train medical students and machine learning models to advance and automate healthcare systems. The proposed model performs significantly better than existing models with a high correlation coefficient that measures the similarity

of the generated synthetic signals with the original signals.

Bidirectional long-term memory was used for the generating network and the convolutional neural network for the discriminating network of the GAN model. The model can be applied to create new synthetic biomedical signals using a small size of the original signal data set. It has been experimented with to generate synthetic signals for four types of biomedical signals (electrocardiogram (EKG), electroencephalogram (EEG), electromyography (EMG), and photoplethysmography (PPG)). [10]

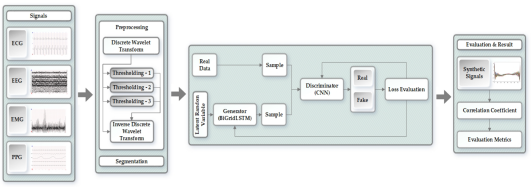


Fig 6. Overview of the proposed model. [10]

C. Synthetic EKG generation using gan to anonymize healthcare data

The reviewed article describes an approach for the generation of synthetic EKG signals based on generative adversarial networks to anonymize user information for security reasons. In this way, valuable data can be obtained for academic and research purposes, thus avoiding the leakage of confidential data. GANs are mainly exploited on images and video frames, so general processing of raw data after transformation into an image is proposed in the article so that it can be managed through a GAN and then decoded in the original data domain. [11]

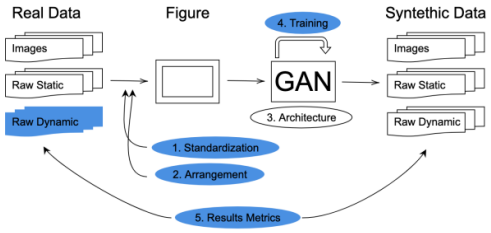


Figure 7. Overview of the proposed model. [11]

VII. Methodology

A simple GAN model using one-dimensional, single-channel convolutional layers were used. The following database was used: PTB-XL, a large publicly available electrocardiography dataset v1.0.1 (physionet.org) [12][13][14]. The database contains 21837 12-lead ECG signals, collected from 18885 patients of 10-second duration. To maintain consistency, only the first lead was used. In addition, half the signal duration was used.

All signals used for training were previously preprocessed. This included the elaboration of a bandpass filter to remove high-frequency noise and baseline drift caused by low frequencies, and the realization of a moving average to further smooth the signal. The Scipy. signal and Numpy packages were used, respectively.

The data were subsequently standardized so that they have a mean of 0 and a standard deviation of 1. This is a good practice to improve the stability of multiple Machine Learning models.

Subsequently, a discriminator and generator were defined, both using Conv1D layers, followed by a Batch Norm and Dropout layer for regularization and to prevent modal collapse. MaxPooling is used between layers to extract the most important features. Both models end with a fully connected Dense layer at the output. In the case of the discriminator, a Sigmoid activation function is used to classify signals as synthetic or real. The generator has a linear activation function since its output is intended to serve as input to the discriminator. The discriminator, and therefore the generator, uses the binary_crossentropy function as a loss since it is a binary classification problem. The optimizer in both cases is Adam. Details about the architecture are detailed in the Appendix.

The training of the model required multiple training epochs, reaching 500,000 epochs. At the end of the training, clustering was performed using t-SNE to visualize the difference in the distributions between the synthetic data and the real data samples.

VIII. Results and discussion

Two models were obtained; one trained for 500000 epochs and one trained for 569999 epochs.

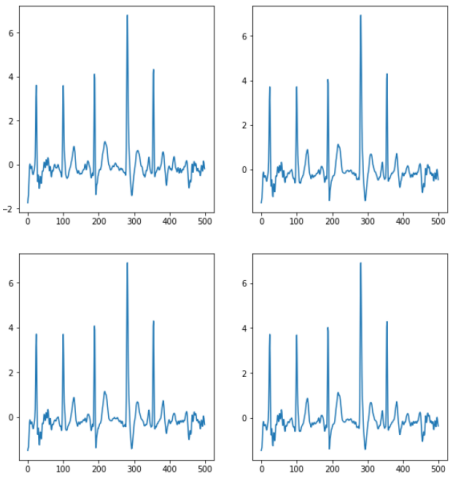


Fig 8. Synthetic signal samples with 500000 epochs.

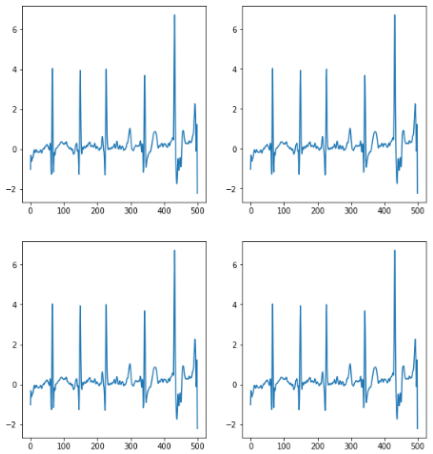


Fig 9. Synthetic signal samples with 569999 epochs

At a qualitative level and for the simplicity of GAN, the result seems to be satisfactory. In both models, a certain periodicity and the presence of what would be QRS complexes and T waves are identified. In both cases, some instability is noticed at the beginning and the end of the signals. In addition, although it seems that the same signal is generated, each signal is using a completely different latent vector space. This indicates that there was a clear problem of modal collapse.

The t-SNE results for both models are shown in the following image:

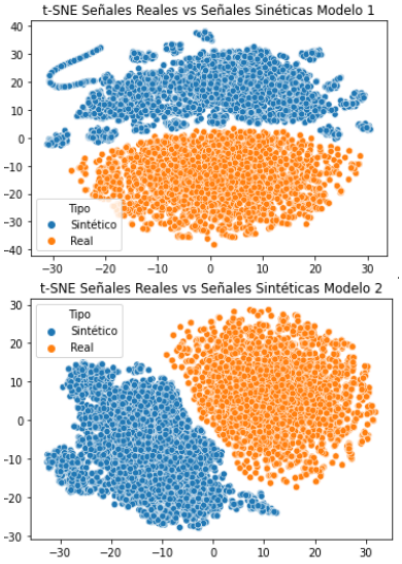


Fig 10. Results of t-SNE

In both cases, for 1000 synthetic and real signals, there is a clear distinction between the two clusters. This means that at the distribution level the synthetic data are different from the real data. The reason for this distinction has not yet been identified. It is possible that it is the instability at the beginning and end of the signals but it is not certain.

During training, the loss behavior of the discriminator was as expected. This was calculated at the moment of trying to deceive the discriminator, labeling a synthetic signal as a real signal. Low loss values represent times when the discriminator was successfully fooled and high values when it was not.:

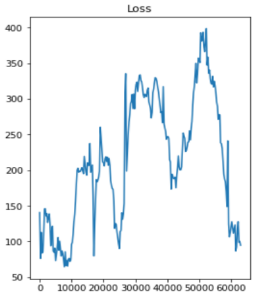


Fig 11. Loss of discriminator

On other training occasions there were convergence problems, where the discriminator cataloged all synthetic signals as real:

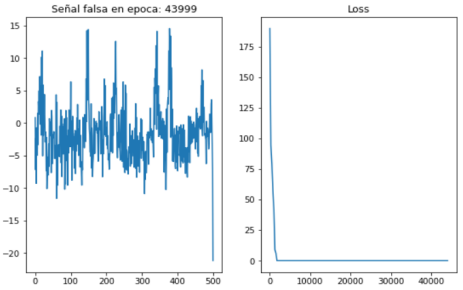


Fig 12. Convergence

Although the result was good qualitatively, there is a lot of room for improvement that can increase the training speed and the final result. First of all, it is possible to experiment with other types of architectures used in the study of time series, such as LSTM and RNN. Also, this year we have started experimenting with the use of Transformers and a GAN architecture [15] with quite promising results for motion data.

One improvement that can be implemented in the short term is to change the loss function. The training of a GAN is complicated because it requires the simultaneous training of a discriminator and a generator. The proper training of these requires some balance between the two models. The use of a loss function for binary classification has the problem that the discriminator quickly learns to distinguish between synthetic and real data, which brings an instability problem.

A Wasserstein-type loss function allows balancing the discriminator, preventing convergence and modal collapse problems. The discriminator goes from deciding whether a sample is synthetic or not to work as a critic, who scores whether a sample is real or not. [16]

In the long run, this also allows for some interpretability of the loss function, as the loss of the discriminator is directly related to the quality of the generator samples.

Other possible enhancements may include other features in the multiple channels of the CNN, such as an FFT, or a Wavelet Transform, both of which are common in feature extraction from ECG signals.

XIX. Conclusions

At the qualitative level, representative samples of ECG signals have been generated, created by a simple GAN based on 1D convolutional layers with some instability problems. Evaluation with t-SNE shows that there is still room for improvement for the model at the level of statistical representativeness and ways in which the work can be improved in the future have been proposed.

X. References

[1] B. Rajoub, “Characterization of biomedical signals: Feature engineering and extraction,” Biomedical Signal Processing and Artificial Intelligence in Healthcare, pp. 29–50, 2020, doi: 10.1016/b978-0-12-818946-7.00002-0.

[2] C. Esteban, S. L. Hyland, and G. Rätsch, “Real-valued (Medical) Time Series Generation with Recurrent Conditional GANs,” arXiv.org, 2017, doi: 10.48550/arXiv.1706.02633.

[3] Industria 4.0, I. A. (22 de Octubre de 2019). Qué son las redes neuronales y sus funciones. Obtenido de https://www.atriainnovation.com/que-son-las-redes-neuronales-y-sus-funciones/

[4] Matich, D. J. (s.f.). Redes Neuronales: Conceptos Básicos y Aplicaciones. Informática Aplicada a la Ingeniería de Procesos – Orientación I.

[5] Xin Yi, Ekta Walia, Paul Babyn, Generative adversarial network in medical imaging: A review, Medical Image Analysis, Volume 58, 2019, 101552, ISSN 1361-8415,https://doi.org/10.1016/j.media.2019.101552. (https://www.sciencedirect.com/science/article/pii/S1361841518308430)

[6] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta and A. Bharath, "Generative Adversarial Networks: An Overview", Ieeexplore.ieee.org, 2018. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8253599. [Accessed: 22- Apr- 2022].

[7] BUHIGAS, J. (2019). Todo lo que necesitas saber sobre las GAN: Redes Generativas Antagónicas. Puentes digitales.

[8] Ionos. (2020). Generative Adversarial Networks: el lado creativo del machine learning.

[9] F.León. (2019). Sistema de Generación de Señales Sintéticas pseudo-realistas. España: Revista Iberoamericana de automática e informática industrial.

[10] D. Hazra and Y.-C. Byun, “SynSigGAN: Generative Adversarial Networks for Synthetic Biomedical Signal Generation,” Biology, vol. 9, no. 12, p. 441, Dec. 2020, doi: 10.3390/biology9120441.

[11] E. Piacentino, A. Guarner, and C. Angulo, “Generating Synthetic ECGs Using GANs for Anonymizing Healthcare Data,” Electronics, vol. 10, no. 4, p. 389, Feb. 2021, doi: 10.3390/electronics10040389.

[12] Wagner P, Strodthoff N, Bousseljot R, Samek W, Schaeffter T. PTB-XL, a large publicly available electrocardiography dataset (version 1.0.1). PhysioNet. 2020. Available from: https://doi.org/10.13026/x4td-x982.

[13] Wagner, P., Strodthoff, N., Bousseljot, R.-D., Kreiseler, D., Lunze, F.I., Samek, W., Schaeffter, T. (2020), PTB-XL: A Large Publicly Available ECG Dataset. Scientific Data. https://doi.org/10.1038/s41597-020-0495-6

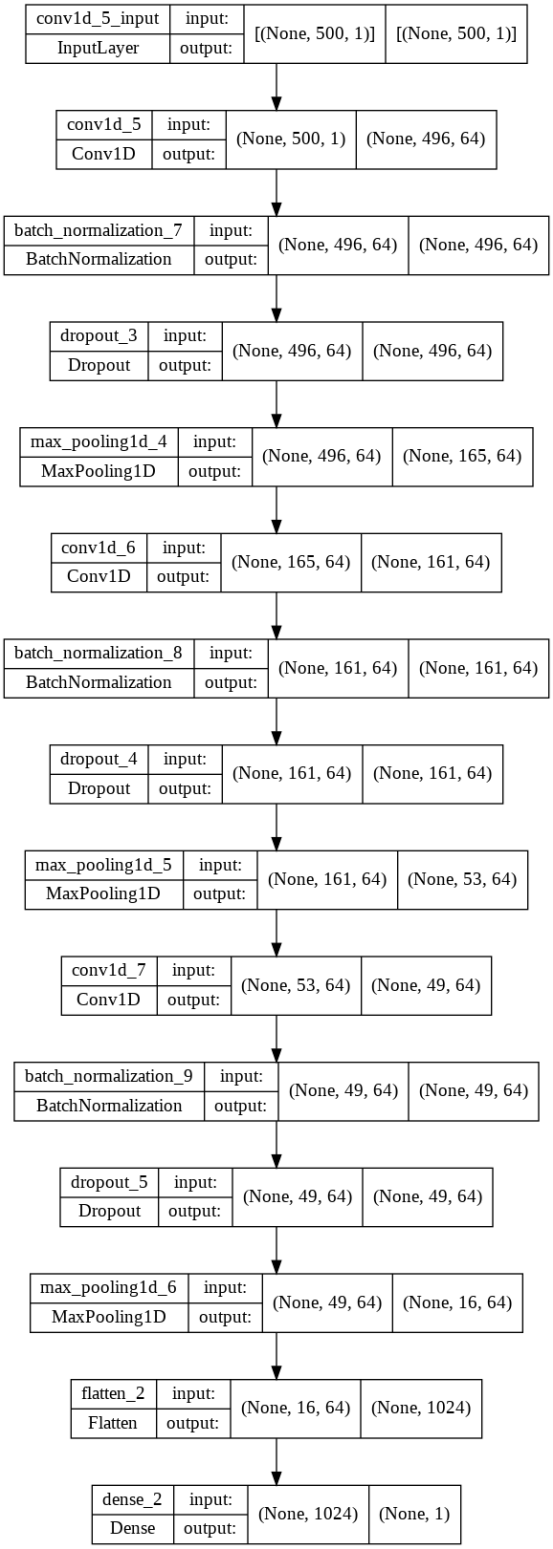
[14] Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation [Online]. 101 (23), pp. e215–e220.

[15] X. Li, V. Metsis, H. Wang, and A. Hee, “TTS-GAN: A Transformer-based Time-Series Generative Adversarial Network,” arXiv.org, 2022, doi: 10.48550/arXiv.2202.02691.

[16] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” arXiv.org, 2017, doi: 10.48550/arXiv.1701.07875.

XI. Appendix

Appendix A: Discriminator model. Plotted using the plot_model function from Keras.



Appendix B: Generator model. Plotted using the plot_model function from Keras.