Biosignal Oversampling Using Wasserstein Generative Adversarial Network

Munawara Saiyara Munia, Mehrdad Nourani
Predictive Analytics and Technologies Laboratory
The University of Texas at Dallas
Richardson, TX 75080, USA
MunawaraSaiyara.Munia,nourani@utdallas.edu

Sammy Houari, DDS, MD
Department of Surgery
UT Southwestern Medical Center
Dallas, TX 75390, USA
sammy.houari@utsouthwestern.edu

Abstract—**Oversampling plays a vital role in improving the minority-class classification accuracy for imbalanced biomedical datasets. In this work, we propose a single-channel biosignal data generation method by exploiting the advancements in well-established image-based Generative Adversarial Networks (GANs). We have implemented a Wasserstein GAN (WGAN) to generate synthetic electrocardiogram (ECG) signal, due to their stability in training as well as correlation of the loss function with the generated image quality. We first trained the WGAN with fixed-dimensional images of the signal and generated synthetic data with similar characteristics. Two evaluation methods were then used for evaluating the efficiency of the proposed technique in generating synthetic ECG data. We used Frechet Inception Distance score for measuring synthetic image quality. We then performed a binary classification of normal and abnormal (Anterior Myocardial Infarction) ECG using Support Vector Machine to verify the performance of the proposed method as an oversampling technique.**

Index Terms—**Generative Adversarial Network, Wasserstein GAN, ECG data augmentation, Oversampling, Myocardial Infarction**

I. Introduction

A. Motivation

The advancement of deep learning has brought about significant improvement in the healthcare domain. While applying deep learning techniques in healthcare domain, one of the major challenges is the scarcity of labelled training data. Acquisition of large amount of labelled patient data (e.g. electroencephalogram (EEG), ECG) is time-consuming, and labelling the data by a medical expert is immensely costly. Furthermore, the biophysiologic data is almost always imbalanced, such as in epileptic seizure EEG and heart attack ECG. Short time data of abnormal cases (e.g. heart attack or seizure) is not enough to train a typical supervised classifier. For example, according to [1], approximately 805,000 Americans suffer from a heart attack every year, among which 200,000 ($\approx 25\%$) occur to people who have already had a previous heart attack. Patients with heart attack (regardless of treatment) are at a higher risk of future heart attacks and will benefit from continuous monitoring. Another example is Atrial Fibrillation (A-Fib). According to [2], around 166,793 death certificates had a mention of A-Fib in 2017, out of which 26,077 deaths had A-Fib as the underlying cause. Monitoring A-Fib patients can be very beneficial to assess disease progress and/or effectiveness of medication. Hence, there is a need of generating synthetic training data that improves the performance of machine learning and deep learning classifiers in patient monitoring systems.

For generation of synthetic data, different methods have been used: random oversampling and undersampling [3], sampling by synthetic data generation [4], cost-sensitive methods [5], active learning methods [6] etc. But these techniques have some shortcomings: undersampling can discard potentially useful data, oversampling may cause of overfitting as it may make exact copies of existing examples (e.g. Random oversampling [3]). Moreover, the synthetic data generated by oversampling techniques is created based on local information, rather than the overall minority class distribution. Hence, they do not preserve the original data distribution. One of the recent advancements in this field is GANs, proposed by Goodfellow et al. [7], which have the capability of generating synthetic data by learning the original data distribution [8].

B. Prior Work

A basic GAN consists of a *generator* network and a *discriminator* network, which oppose each other. The generator network's objective is to deceive the discriminator network by generating realistic-looking synthetic data. This is achieved by mapping a latent space (random values from n-dimensional hypersphere, where every variable is drawn from a Gaussian distribution) to the original data distribution [7]. The objective of the discriminator network is to evaluate the generator by learning to discriminate the synthetically-generated data from real training data and forcing the generator to produce better synthetic data.

The discriminator network of basic GANs can be unstable during training, which results in mode collapse. Mode collapse is the situation when the discriminator only recognizes a few modes of the input distribution as real. This drives the generator to produce only a limited amount of different

978-1-7281-5382-7/20/$31.00 ©2020 IEEE

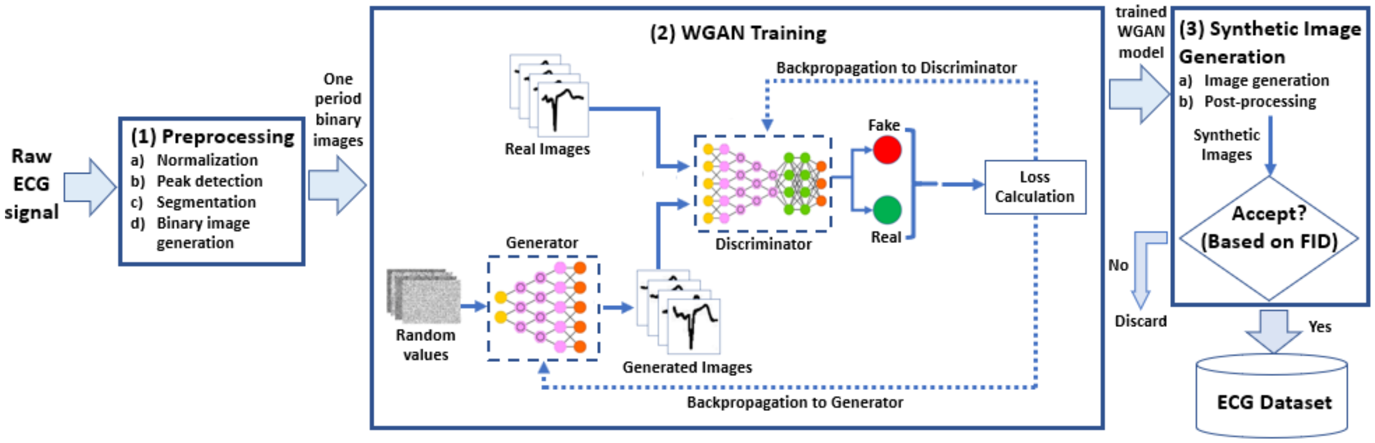


Fig. 1. Proposed oversampling methodology

outputs. Extensive research has been done to overcome the mode collapse problem in GANs, and various improvements [9]–[11] have been proposed to increase the stability of the discriminator during training. In [9], the authors have proposed an improved GAN model, called WGAN. WGAN relies on a new loss function, Wasserstein distance (also called Earth Mover's Distance) between the real and synthetic data distributions for evaluating GAN model. Earth Mover's Distance (EMD) is a method for evaluating similarity among two distributions in a specific region. It significantly improves the stability of training the discriminator, resulting in the generation of superior-quality images.

GANs have been widely used for generating high-quality synthetic images. However, very few studies [12]–[15] have reported the use of GANs in generation of biosignals. In [12], the authors proposed a novel method for generation of time series data using GANs, conditioned on class labels. They have used long short-term memory (LSTM) units in the hidden layers, which facilitates the conditional generation of the time series data. The authors in [13] used two bidirectional LSTMs in the Generator for generation of sine wave and ECG signals. The authors also showed that mini-batch discrimination layer can play a crucial role in preventing mode collapse. The authors in [14] proposed a data augmentation method based on recurrent GAN that uses LSTM cells in the hidden layers for generating ECG and EEG biosignals. The authors have shown the efficiency of the proposed method by measuring classification accuracy using an LSTM network. In [15], a bi-directional LSTM has been used as the generator network, and a convolutional neural network (CNN) as the discriminator network for generation of synthetic ECG signal.

The authors have used image-based GANs to generate time series data [16]. In this work, they have converted time series data into rasterized images and trained WGAN models with gradient penalty to generate synthetic images of sinusoidal signal, photoplethysmograph (PPG) data and electrocardiogram (ECG) data.

C. Main Contribution

The main contribution of our research is a method of generating single-channel synthetic ECG signal using WGAN. First, the raw ECG signal is preprocessed and converted into fixed-dimensional binary images such that each image contains exactly one period of the ECG signal. Afterwards, we trained a WGAN for generation of the synthetic ECG signal images. We have conducted the evaluation of the proposed method in two ways: 1) by calculating Frechet Inception distance (FID) score for quantitative assessment of the quality of the created data, 2) by performing binary classification of imbalanced normal and abnormal (Anterior Myocardial Infarction) ECG data using SVM to evaluate the effectiveness of the proposed method as an oversampling technique.

The remainder of this paper is organized as follows: in Section 2, we discuss the proposed method. The biosignal generation experimentation details are described in Section 3. Section 4 presents the results of experimentation, as well as discussion. Concluding remarks are summarized in Section 5.

II. Methodology

A. Proposed Architecture

Fig. 1 shows the main steps of our proposed method which is divided into three steps: 1) Preprocessing 2) WGAN training and 3) Synthetic image generation. The details of each step are given below.

1) Preproccesing: In the preprocessing step, we first normalized the amplitudes of the ECG signal between -1 to 1 and downsampled it to 128Hz. We then segmented the signal into fixed-dimensional windows such that exactly one period of the signal can be captured in the image. For capturing exactly one ECG data period in the signal from each lead, we identified the most prominent peak during each time

TABLE I
GENERATOR AND DISCRIMINATOR MODEL ARCHITECTURE

Generator		**Discriminator**	
Layer	**Output Shape**	**Layer**	**Output Shape**
Fully Connected	32768	Conv2D	32,32,64
Leaky Relu	32768	Batch Norm.	32,32,64
Reshape	16,16,128	Leaky Relu	32,32,64
Conv2D Transpose	32,32,128	Conv2D	16,16,64
Batch Norm.	32,32,128	Batch Norm.	16,16,64
Leaky Relu	32,32,128	Leaky Relu	16,16,64
Conv2D Transpose	64,64,128	Flatten	16383
Batch Norm.	64,64,128	Fully Connected	1
Leaky Relu	64,64,128		
Conv2D	64,64,1		
Number of Generator Parameters		**Number of Discriminator Parameters**	
Total	2,229,505	Total	83,585
Trainable	2,228,993	Trainable	83,329

period and calculated the distance between each successive peak. After that the median value of these distances was determined. Then, we adjusted the distance that we want to capture before each peak and performed segmentation starting from that data point. This guarantees exactly one complete cycle per window. The data for training the WGAN is then generated by mapping the amplitude of the segmented signals to images with the dimension 64×64. For the ease of computation in our application, generated RGB images of the signal are then converted into binary images.

2) WGAN Training: The next step is to train WGAN using the binary images for generating synthetic images of the signal. The architecture of the WGAN used in this work can be found in Table I. We have used the architecture used in [17] and modified it based on our dataset. In Table I, the output shape shows the dimension of output data from each layer (e.g. output shape 32,32,128 means the upsampled image resolution is 32×32 and batch size is 128). The WGAN is composed of a generator network and a discriminator network. This composite network takes as input a point from a latent space, and then uses the generator for generating synthetic images [9]. Half batch of real images from the training dataset and half batch of synthetic images generated by the generator is then fed to the discriminator model, which then outputs the realness or fakeness of them. The training continues until the GAN model converges (when the discriminator and the generator reach a Nash equilibrium [18].

3) Synthetic Image Generation: The trained WGAN model is capable of generating new binary images by taking points from a latent space as input. These synthetic images mimic the original signal. We perform an additional post-processing step where we use a nearest-neighbor based approach to discard the noisy synthetic images, if there is any. After that, the FID score of the real and synthetic images is calculated. We have empirically chosen an FID threshold, which we use as an accept/reject criterion. Comparing the calculated FID score of the real and synthetic images with the FID threshold, we decide if the synthetic images should be accepted or discarded. If they are accepted, the remaining synthetic images are augmented to the training dataset for later application. We continue this experimentation until no more improvement is seen in FID.

B. Desired Characteristics of Signal

Let us assume using the Fourier transform, the signal has the lowest frequency of f_L and highest frequency harmonic of f_H. Then, in order to be able to capture at least one whole period of the signal in an image that captures M seconds of the signal, we need the following inequality to be true: $M \geq \frac{1}{f_L}$. Based on the Nyquist theorem, in order to be able to perfectly reconstruct the original signal from the sampled signal, the following inequality should hold: $f_s \geq 2f_H$ which if we assume that our image is of size $N \times N$ results in: $\frac{N}{M} \geq 2f_H \Rightarrow N \geq 2Mf_H$. In our case of ECG signals, f_L, f_H, and M are approximately 1.25, 34, and 0.8Hz respectively. Therefore, from the above inequality, we get $N \geq 54.4$; thus, the image size of 64×64 for one period of ECG is quite appropriate.

C. Example of Toy Signal Generation

To show the concept, we use a simple signal with known amplitude (e.g. triangular signal) first. This way, we can evaluate the synthesized data very easily with visual inspection. This dataset was generated by varying amplitudes and time period, such that each image contains only one period of the signals. For this triangular signal, three consecutive triangles are defined as a complete period, as shown in Fig. 2. We have generated 8000 training images of size 64×64 to train the WGAN. In Fig. 2, we show an example of the real images of the triangular signal along with the synthetic samples generated by the proposed method. Note carefully how WGAN has preserved the main characteristics of the signal for which the middle peak is the largest and the third peak is the smallest.

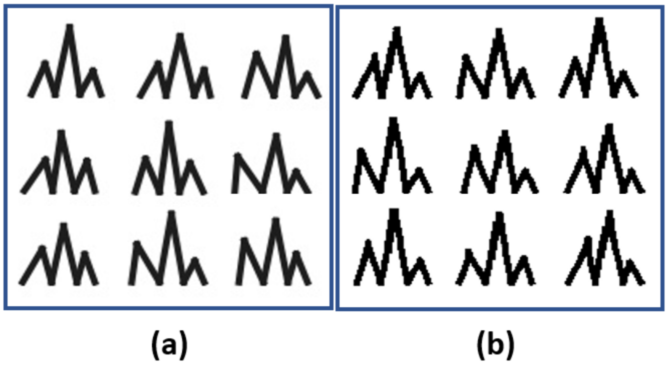


Fig. 2. Toy (triangular) signal generation example: (a) Real signals (b) Synthetic signals.

III. Biosignal Generation experimentation

A. Datasets

For demonstrating how effective the proposed method is for synthesizing realistic ECG data, we used PTB Diagnostic ECG dataset [19] from PhysioNet. From this dataset, we are specifically interested in generating ECG data for Anterior Myocardial Infarction (AMI), which results from occlusion of the left anterior descending coronary artery [20]. During AMI, the anterior leads (V3 and V4) show ST segment elevation as shown in Fig. 4. The ST elevation is often concave downward and often engulfs the T wave. It can also be seen in septal (V1 and V2) and lateral leads (V5 and V6), depending on the degree of the infarction. For simplicity, we have focused only on lead V3. Figures 3 and 4 show comparison between ECG signal of lead V3 from a healthy subject and ECG signal of lead V3 from a subject with AMI respectively.

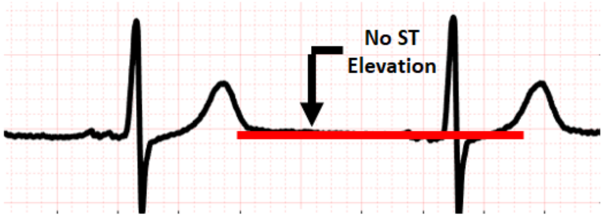


Fig. 3. Normal ECG from lead V3 (Patient 169 [19])

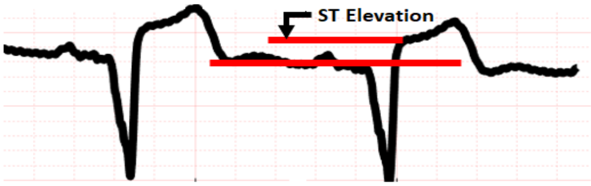


Fig. 4. Abnormal ECG from lead V3 (Patient 005 [19])

Since we were targeting ECG characteristics during AMI, we have used PTB Diagnostic ECG dataset [19]. For performing the binary classification of normal versus abnormal ECG data, we have used PTB Diagnostic ECG dataset [19] and MIT-BIH Normal Sinus Rhythm dataset [21]. The comprehensive descriptions of the datasets are given in Table II.

B. Evaluation Metrics

Evaluation of the performance of GAN is a challenging and important research area, as there is lack of consensus regarding the quantitative evaluation of GANs [22]. In this work, we evaluate our approach using two metrics: (i) FID (a metric for measuring synthetic image quality) (ii) F-measure (a metric to measure how well the signal characteristics are preserved).

TABLE II
ECG Dataset Description

Dataset	Description
PTB Diagnostic ECG dataset [19]	• Original dataset consists of 549 records from 290 subjects. • Diagnostic classes include Myocardial infarction, Cardiomyopathy/Heart failure, Bundle branch block, Dysrhythmia, Healthy control etc. • Records per subject = 1 to 5, each approximately 2 minutes long. • 15 signals measured simultaneously in each record: the general 12 leads (i - iii, avr, avl, avf, v1-v6) along with 3 Frank lead ECGs (vx, vy, vz). • Used data of patient005 (diagnosed with AMI) for generating training dataset of ECG-WGAN. • Original data sampling frequency = 1000Hz. Downsampled to 128Hz. • 2686 training images of size 64X64, each containing only one period of the signal.
MIT-BIH Normal Sinus Rhythm Dataset [21]	• Original dataset includes 18 long-term ECG recordings from two leads for the subjects. • Sampling frequency = 128Hz • 40,000 training images of size 64X64, each containing only one period of the signal.

1) Frechet Inception Distance (FID): The FID score evaluates the quality of the synthetic images generated by GANs [23]. It examines the similarity between the synthetic images and the real images by comparing statistics of a group of synthetic images with the statistics of a group of real images. The FID score uses the Inception network [24] to calculate these statistics for each collection of images. An intermediate layer of the inception network is used to capture features of an input image (real or synthetic). The mean and covariance of the features of the real and synthetic images are then determined and the features are portrayed as two multivariate Gaussian distributions (one for real and one for synthetic images). Afterwards, the distance between these two distributions is measured using the Wasserstein-2 distance, which is also known as Frechet distance. The smaller the distance, the more similar the two distributions, and as a result, the better the quality of the synthetic images. The FID score is formally defined as:

$$FID(r,s) = ||\mu_r - \mu_s||^2 + Tr(\Sigma_r + \Sigma_s - 2(\Sigma_r \Sigma_s)^{\frac{1}{2}})$$

where μ_r and μ_s are the feature-wise means of the real and synthetic data, and Σ_r and Σ_s are the corresponding covariance matrices for the real and synthetic feature vectors. Tr sums up all the diagonal elements. Note that, FID will be a positive real number and smaller FID implies better quality of synthetic images. So, $FID(r,r) = 0$.

2) Classification Metrics: We evaluate the performance of the proposed method as an oversampling technique in terms of F-measure ($F - measure = \frac{2 \times Pre \times Rec}{Pre + Rec}$), where Pre and Rec denotes Precision ($Pre = \frac{TP}{TP+FP}$) and ($Rec = \frac{TP}{TP+FN}$) respectively. Here, TP is the number of true positives, FP is the number of false positives, TN is the number of true negatives, and FN is the number of false negatives. In our application, we consider the positive class as the abnormal class, and the negative class as the normal class.

C. Experimental Setup

An Nvidia RTX-2060 GPU with 6GB of RAM was used for the experiments in this work. Python libraries: scikit-learn, numpy, OpenCV and Keras were used to preprocess the data, build the WGAN model and perform all the experiments.

During the training, the WGAN model wad trained for 1000 epochs. We kept the training ratio of the discriminator and generator to 5:1, which helps with the training stability. The WGAN was trained using a SGD optimizer with a learning rate, $lr = 0.00005$.

To validate the performance of the proposed model, we have created two imbalanced datasets with an Imbalance Ratio (IR) of 100:1 and 200:1 between the Normal ECG class and AMI ECG class. This way, we can examine how much the classification metrics improve for various proportions to the majority class training data.

We then extracted two global features: Hu-moments [25] and color histogram [26] for each image in the new dataset. The dataset was then divided into a training set (80%) and a test set (20%) for evaluation. The ratio of Normal and AMI ECG was kept intact in both the training and test dataset. We then performed a binary classification by training an SVM classifier with RBF kernel, using the training data. We have performed stratified 10-fold cross validation [27] for validating the classifier accuracy. After the training was done, we tested the SVM classifier with the test dataset and reported the F-measure.

For evaluating the effect of augmented synthetic minority class data size on the performance of the SVM, we trained the SVM classifier by augmenting synthetic AMI ECG samples of size 0 (no oversampling), 200, 500, 1000, and 2000 to the training dataset. The test data was kept unchanged.

IV. Results and Discussion

In Figures 5 and 6, we present the images of real ECG samples and synthetic ECG samples generated by the proposed method. Fig. 6 shows that the WGAN model is able to generate realistic-looking images of the signals. Note carefully that the key characteristics of signal (ST segment elevation) has been faithfully preserved in all synthetic images.

The FID score between the real and synthetic signal images for the WGAN models are reported in Table III. We have calculated the FID score at 100, 300, 600 and 1000 epochs for the WGAN model. From the FID values, we can say that the distance between the original and synthetic data distribution is still large, but as the training progresses, the FID score reduces and the real data distribution and synthetic data distribution get closer. Table III also shows how the quality of the synthetic images improves as the training progresses. Due to limited computational resources, we could not exhaustively search for optimal hyperparameter at this time. We have empirically chosen an FID threshold, which we use as an accept/reject criterion (See step 3 in Fig. 1).

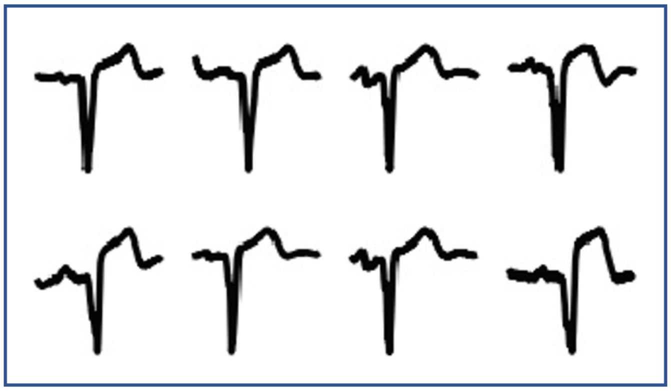


Fig. 5. Real AMI ECG samples with elevated ST-segment

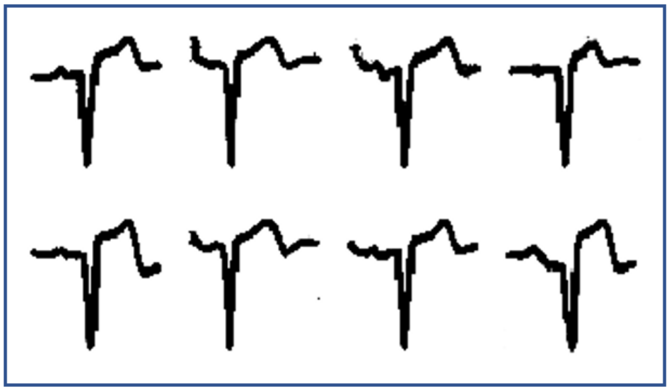


Fig. 6. Synthetic AMI ECG samples generated by WGAN where elevated ST-segment is preserved

The result of the evaluation metric for binary classification is summarized in Table IV, where we present the F-measure values for test set with different augmentation data size. Without any data augmentation, for an IR of 100:1, the F-measure of the classification was very poor (36.00%). After adding 1000 synthetic training samples, the value improved to 81.08%. Hence, increase in the number of synthetic minority class training data significantly improves the F-measure gradually. However, adding more synthetic data to the training dataset did not show any significant improvement in F-measure value after this. For an imbalance ratio of 200:1, we observe an improvement from 84.37% to 85.12% in F-measure, but again no further improvements were found for augmentation data size beyond 1000.

TABLE III
FID SCORES AT DIFFERENT EPOCHS DURING WGAN TRAINING

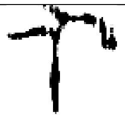
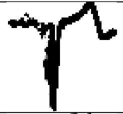
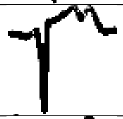
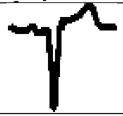
Epoch	**FID Score**	**Synthetic Image Quality**
100	137.48	
300	97.33	
600	92.19	
1000	88.27	

TABLE IV
PERFORMANCE OF SVM BEFORE AND AFTER OVERSAMPLING OF ECG DATASET

IR	Oversampling	# of synth. images	F-measure (%)
200:1	Before	0	36.00
	After	200	69.69
		500	75.36
		1000	81.08
		2000	81.08
500:1	Before	0	84.37
	After	200	83.93
		500	84.37
		1000	84.53
		2000	85.12

V. CONCLUSION

Imbalanced datasets have always obstructed the performance of machine learning classifiers in biomedical and bioinformatics applications. In this paper, we have proposed a method for synthetic ECG data generation by using WGAN for minority class oversampling. We have chosen Anterior Myocardial Infarction (AMI) ECG data from PTB Diagnostic ECG dataset [19] as the Abnormal ECG data and generated synthetic AMI ECG data using WGAN. The performance of the WGAN was evaluated by measuring the FID score and F-measure to validate the models capability as an oversampling technique. The results show that the WGAN can successfully generate synthetic ECG data that can balance the original training dataset and improve classifier performance. The future research extension of this work includes addressing more challenging signals with more ambiguities in their characteristics, such as EEG.

REFERENCES

[1] "Heart disease facts," https://www.cdc.gov/heartdisease/facts.htm, 2019, [Online; accessed 2-Dec-2019].

[2] "Atrial fibrillation," https://www.cdc.gov/heartdisease/atrial_fibrillation.htm, 2019, [Online; accessed 9-Dec-2019].

[3] R. C. Holte, L. Acker, B. W. Porter *et al.*, "Concept learning and the problem of small disjuncts." in *Proc. Int'l J. Conf. Artificial Intelligence*. Citeseer, 1989, pp. 813–818.

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[5] G. M. Weiss, "Mining with rarity: a unifying framework," *ACM Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 7–19, 2004.

[6] S. Ertekin, J. Huang, L. Bottou, and L. Giles, "Learning on the border: active learning in imbalanced data classification," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 2007, pp. 127–136.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] Y. Lu, Y.-W. Tai, and C.-K. Tang, "Attribute-guided face generation using conditional cyclegan," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 282–297.

[9] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[10] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in neural information processing systems*, 2017, pp. 5767–5777.

[11] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[12] S. Harada, H. Hayashi, and S. Uchida, "Biosignal generation and latent variable analysis with recurrent generative adversarial networks," *CoRR*, vol. abs/1905.07136, 2019. [Online]. Available: http://arxiv.org/abs/1905.07136

[13] A. M. Delaney, E. Brophy, and T. E. Ward, "Synthesis of realistic ecg using generative adversarial networks," *arXiv preprint arXiv:1909.09150*, 2019.

[14] S. Haradal, H. Hayashi, and S. Uchida, "Biosignal data augmentation based on generative adversarial networks," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 368–371.

[15] F. Zhu, F. Ye, Y. Fu, Q. Liu, and B. Shen, "Electrocardiogram generation with a bidirectional lstm-cnn generative adversarial network," *Scientific reports*, vol. 9, no. 1, pp. 1–11, 2019.

[16] E. Brophy, Z. Wang, and T. E. Ward, "Quick and easy time series generation with established image-based gans," *arXiv preprint arXiv:1902.05624*, 2019.

[17] J. Brownlee, "Develop a wasserstein generative adversarial network (wgan) from scratch," https://machinelearningmastery.com/how-to-code-a-wasserstein-generative-adversarial-network-wgan-from-scratch/, 2019, [Online; accessed 17-July-2019].

[18] F. A. Oliehoek, R. Savani, J. Gallego, E. van der Pol, and R. Groß, "Beyond local nash equilibria for adversarial networks," in *Benelux Conference on Artificial Intelligence*. Springer, 2018, pp. 73–89.

[19] R. Bousseljot, D. Kreiseler, and A. Schnabel, "Nutzung der ekg-signaldatenbank cardiodat der ptb über das internet," *Biomedizinische Technik/Biomedical Engineering*, vol. 40, no. s1, pp. 317–318, 1995.

[20] "Anterior wall st segment elevation mi ecg review," https://www.healio.com/cardiology/learn-the-heart/ecg-review/ecg-topic-reviews-and-criteria/anterior-wall-st-elevation-mi-review, 2017, [Online; accessed 19-Jun-2017].

[21] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[22] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, 2016.

[23] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in neural information processing systems*, 2017, pp. 6626–6637.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[25] "Shape matching using hu moments," https://www.learnopencv.com/tag/hu-moments/, 2018, [Online; accessed 10-Dec-2018].

[26] “Color histogram,” https://gogul.dev/software/image-classification-python, 2020, [Online; accessed 01-jan-2020].

[27] “Stratified k-fold,” https://datascience.stackexchange.com/questions/13126/sklearn-stratifiedkfold-code-explanation, 2019, [Online; accessed 01-jan-2019].