bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

DeepGANnel: Synthesis of fully annotated single molecule patch-clamp data using generative adversarial networks

Numan Celik, Sam T. M. Ball, Elaheh Sayari, Lina Abdul Kadir, Fiona O'Brien and Richard Barrett-Jolley*

Faculty of Health and Life Science, University of Liverpool, Liverpool, United Kingdom

***Corresponding Author**: rbj@liverpool.ac.uk

Abstract

Understanding and accurately quantifying ion channel molecule gating in real time is vital for knowledge of cell membrane behaviour, drug discovery and toxicity screening. Doing this with single-molecule resolution first requires the detection of individual protein pore opening and closing transitions and construction of a so-called idealised record which indicates sample-point by sample-point whether a given molecule is open or closed. Creating this can be difficult, since patch-clamp electrophysiology data can be noisy or contain multiple ion channel molecules. We have recently developed a deep learning model to achieve this called Deep-Channel, but further development is limited by the massive datasets need to train and validate models. In the past, this problem has been tackled by simulation of single molecule activity from Markov models with the addition of pseudo-random noise. In the present report we develop a new method to synthesise raw data, based on generative adversarial networks (GANs). The limitation to direct application of a GAN with this method has been that whilst there are methods to generate classified output image by image, there has been no method to generate an entire timeseries with parallel idealisation, sample-point by sample-point. In this paper, we over-come this problem with DeepGANnel, a model that splits training data raw and parallel idealised data into different rows of image windows and passes these data through a progressive-GAN. This new methodology allows generation of realistic, idealisation synchronised single molecule patch-clamp data, without the biases inherent in pseudorandom simulation methods. This method will be useful for development of single molecule analysis methods and may in the future prove useful for generation of biological models including single molecule resolution stochastic data. The model is easily extendable to other timeseries data requiring parallel labelling, such as labelled ECG.

1. Introduction

Ion channel molecules are transmembrane pore-forming proteins that allow the passage of ions from one side of the membrane to the other (Hille, 2001). They play a fundamental role in regulating key biological processes such as cell excitability; for example, voltage-activated Na^{2+} and K^{+} selective ion channel proteins are essential for the generation and propagation of action potentials (Hodgkin & Huxley, 1952).

Analysing the data from biophysical experiments is essential for understanding the principles that underlie the electrophysiological responses of excitable cells. The development of the Nobel Prize winning patch-clamp electrophysiological technique (Hamill, Marty, Neher, Sakmann & Sigworth, 1981; Neher & Sakmann, 1976) that allows the measurement of ionic currents from cells has paved the way for an improved understanding of the role of ion channel molecules in the function of excitable tissue in health and disease. Patch clamp "single-channel" molecule recording in particular has a unique ability to capture a single molecule gating in 'real time' and provides researchers with a powerful means to investigate the mechanisms by which such molecules open and close their ion conducting pore. This has made an enormous impact to the understanding of ion channel molecular function, allowing the detailed and precise information about cell function to be obtained. During

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

gating, an ion channel molecule typically transitions to many different conformational states and these transitions and states are commonly represented by continuous Markov chain models. Single molecule analysis allows functional Markovian models to be generated, which can be paired with available structural information to aid our understanding of the molecular conformational changes that occur during gating.

Computational simulation of single-molecule recordings in combination with biological experiments can describe and predict the activity of many ion channel molecules, which aids to the development of mathematical models of excitable cells (Hodgkin & Huxley, 1952). General-purpose software packages, such as MATLAB or more specialised programs such as QuB, ionChannelLab and PulseSim are typically used to simulate single molecule events based on the Markov chains for the molecule of interest. Current practice for idealising (or annotating) biological or simulated records of patch-clamp time series data involves supervised threshold-crossing or segmented-k means (SKM) algorithms using software applications such as QuB (SUNY, Buffalo (Nicolai & Sachs, 2013), MDL (Gnanasambandam, Nielsen, Nicolai, Sachs, Hofgaard & Dreyer, 2017) or SPARTAN (Juette et al., 2016). Although these approaches are generally successful, idealisation of complex and/or multi-molecule records cannot be achieved with accuracy. Moreover, it is widely accepted that the current methods of idealisation are laborious and require a high level of human input. Deep learning (LeCun, Bengio & Hinton, 2015), an artificial intelligence development, has begun to revolutionise automatic analysis of complex image modalities (Gao, Celik, Wu, Williams, Stylianides & Zheng, 2019) and time-series datasets for classification problems in single-molecule analysis (Albrecht, Slabaugh, Alonso & Al-Arif, 2017; Boza, Brejova & Vinar, 2017). In other areas, the field of physiology has benefited greatly from this advancement in artificial intelligence, since it allows for complex patterns to be found within data, given that the size of the dataset is large enough (Sun, Shrivastava, Singh & Gupta, 2017).

Physiological signals are a common target for deep learning applications since the amount of readily available data is relatively large – deep learning models have shown great promise in area s such as sleep state identification and emotional recognition using EEG data (Fraiwan & Lweesy, 2017; Jirayucharoensak, Pan-Ngum & Israsena, 2014; Supratak, Dong, Wu & Guo, 2017), and detection of arrhythmia or sleep apnea from ECG signals (Acharya, Fujita, Lih, Hagiwara, Tan & Adam, 2017; Cheng, Sori, Jiang, Khan & Liu). Recently, we developed a deep-learning method, called Deep-Channel (Celik et al., 2020) to identify single molecule events from patch-clamp data using recurrent convolutional neural networks. Typically, in model development researchers use simulated data to provide both the raw data along with the critical fiducial annotation/idealisation of the underlying molecular state (open or closed for example). In the past this method focussed on direct simulation from Markov models together with the addition of pseudo-random noise (for example,Anderson et al., 2015; Colquhoun et al., 1996; Gillespie, 1977; Nicolai & Sachs, 2013; Voldsgaard Clausen, 2020). However, in Celik *et al* (Celik et al., 2020) we adapted this methodology by adding real patch-clamp amplifier noise. Synthetic patch-clamp data was generated through fiducial records with authentic kinetic models in MATLAB, played out through a CED digital-to-analogue converter to a patch clamp amplifier that sent this signal to a model cell and recorded back on to a hard disk. This Deep-channel model provided highly accurate annotations (0.9±0.2 F-1 score) but the model relies on high computational costs to record synthetic datasets to train the model. These limitations drove us to consider other solutions to create highly realistic synthetic single molecule ion currents that can be used for developing analysis tools, and generative adversarial networks (GAN) seem an ideal solution.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

A GAN, first described by Goodfellow in 2014 is a type of deep neuronal network that compromises a generator and a discriminator that play a zero-sum game until they converge (Goodfellow et al., 2014). The discriminator is essentially a classifier which tries to distinguish real data from the data created by the generator. GANs have been shown to be an efficient method for generating high quality realistic images and videos (Ledig et al., 2017; Zhang et al., 2017).

Simulation approaches for medical imaging and physiological data have gathered significant traction due to the possibility of a large anonymised annotated dataset to be created for use in machine learning models or other systems. Until recently these models mostly used mathematical models to generate synthetic data, however the introduction of GANs has allowed for a data-driven approach to data synthesis; this approach has seen significant success in the medical imaging field, ranging from brain MRI synthesis (Chartsias, Joyce, Giuffrida & Tsaftaris, 2018; Shin et al., 2018) to retinal image synthesis (Costa et al., 2018). GANs can also be used in conjunction with other models to achieve greater results, for example combining GANs with reinforcement learning to build models to enable the drug discovery process (Putin et al., 2018), or using GAN generated data to train AI (Zhang, Wang, Lu, Won & Yoon), circumventing issues with data quantity and quality.

However, there have been only a few studies examining time series data; recently an interesting study was published showing an artificial audio generated by a GAN, called WaveGAN, synthesising one second slices of audio waveforms (Donahue, McAuley & Puckette, 2018). The generation of synthetic electrophysiological signals would have applications in many different areas such as ECG-GAN (Golany & Radinsky, 2019) and EEG-GAN (Hartmann, Schirrmeister & Ball, 2018), but to our best knowledge, the current research is the first for the generation of raw single molecule currents with parallel classification of each molecular transition event using GANs.

In this work, we develop a GAN model to generate synthetic ion channel currents using genuine patch-clamp data. These output data can then be made available for biological modelling or further single molecule software development. The generation of synthetic time series data is generally performed with a regular convolutional neural network based GANs, such as conditional GANs (CGAN) (Esteban, Hyland & Rätsch, 2017) because of the advantages of local and hierarchical structure of convolutional layers. While these GAN architectures leverage the generator output for their applications, the recent approaches in GAN have altered the discriminator and adapted it to implement multiclass classification such as in the semi-supervised GAN (Odena, 2016), deep convolutional GAN (DC-GAN) (Radford, Metz & Chintala, 2015), InfoGAN (Chen, Duan, Houthooft, Schulman, Sutskever & Abbeel, 2016), and the auxiliary classifier GAN (AC-GAN) (Odena, Olah & Shlens, 2017). To generate realistic synthetic single-molecule records in time series along with continuous idealisation (labels), we use a DC-GAN by adapting time series single molecule data into image modalities for the input of generator. Furthermore, we evaluate output data on a range of different metrics, including biological standard kinetic analysis. In the future we believe this approach will prove useful for generation of biological models including single molecule resolution stochastic data.

2. Methods

2.1 Model design

We propose a GAN based model to generate synthetic time series data that includes realistically similar features to the real ion channel molecule currents. Figure 1 describes the complete pipeline of the proposed DeepGANnel model in this work. The architecture of the GAN model introduced in this work is following the regular DC-GAN, but applies the convolutional neural networks that demonstrated efficiently to produce time series datasets in previous works (Delaney, Brophy & Ward, 2019; Zhu, Ye, Fu, Liu & Shen, 2019). In this part, utilized neural networks briefly are covered along with the processed data information and evaluation metrics criteria.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

2.2 Generative Adversarial Networks

A generative adversarial network (GAN) typically consists of two neural networks competing against each other; one called the generator and the other the discriminator. The generator (G) tries to convert random noise *(z)* of the Gaussian variable $z \sim p(z)$ $(G(x|z))$ into observations that seem as if they have been sampled from the original data *p(x)*; while the discriminator (D) aims to classify whether the sampled data comes from the original dataset or output of the generator network by predicting a class probability $D(x) \in [0,1]$. The training process is employed by an adversarial manner between *G* and *D* by updating the parameters of *G* based on updates of *D*; *D* is trained to maximize the probability of assigning correct labels to the real and generated samples. *G* is initially fixed to a random noise and then tries to maximize the discriminator's uncertainty by minimising $log\left(1 - D\big(G(z)\big)\right)$ to fool *D* into believing that the generated samples are real. This results in an objective function *V (G, D)* with this min-max competition and is described by the following equation:

$$\min_{G}\max_{D} V(G,D) = E_{z\sim p(x)}[\log D(x)] + E_{z\sim p(z)}\left[\log\left(1 - D\big(G(z)\big)\right)\right] \#(1)$$

To further develop solutions for the classification of generated time series data we utilized deep convolutional GAN (or DC-GAN). In the DC-GAN, *D* is trained to distinguish real given data $x_{s,r}$, from generated samples $G(x_{s,n})$, and *G* is trained to transform random sample data (e.g. Gaussian sample, uniform sample) into realistically similar fake data $G(x_{s,n})$ to fool *D* . Similarly, optimization objective functions can be defined as follows for both *D* and *G* architectures; *D* tries to minimize the classification loss (L_d) for the real/fake modalities:

$$L_d = E_{x(s,r)\sim P(x(s,r))}\big[\log D\big(x_{s,r}\big)\big] - E_{x(s,n)\sim P(x(s,n))}\left[\log(1 - D\left(G\big(x_{s,n}\big)\right))\right] \#(2)$$

while the *G* aims to minimize the following loss (L_g) to maximize L_d:

$$L_g = E_{x(s,n)\sim P(x(s,n))}\left[\log(1 - D\left(G\big(x_{s,n}\big)\right))\right] \#(3)$$

2.3 Convolutional Neural Networks

Convolutional neural networks (CNNs) have shown great potential in processing images and sequences through computer vision studies (Yang, Nguyen, San, Li & Krishnaswamy, 2015). In this study, we use two-dimensional convolution layers as image inputs to the generator and discriminator (Figure 2). The convolutional layers in CNNs are utilizing sliding filters over the image input in intervals introduced by the stride. The same filter is applied across all other step sizes which adapts the learning independently of its position in the series image input. The common notation of the convolution is given by (6)

$$C_t = f\left(\omega * X_{t-\frac{l}{2}:t+\frac{l}{2}} + b\right) \forall t \in [1,T] \#(4)$$

where C_t is the outcome of the convolution of an input data *X* of length *l* with ω filter, a bias parameter *b*, and with non-linear activation function *f* at time *t (Zhu, Ye, Fu, Liu & Shen, 2019)*. The neural networks can learn more abstract features by applying multiple filter layers. The weights of the neural network can be fitted through hidden neurones given by:

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

$$\frac{W - K + 2P}{S} + 1 \# (5)$$

where W is the number of input dimensions, K is the kernel filter size, S defines the stride and P shows the number of zero padding points that are applied around the data border(Zhu, Ye, Fu, Liu & Shen, 2019). After initiating the convolution layer, an activation function (Rectified Linear Unit - ReLU) is introduced to increase the non-linearity. In the last step, a max pooling layer is applied to reduce the size of the represented data as well as keeping the spatial information of the filtered image data. Multiple CNN layers are used in this study to capture more features associated with the data. To prevent overfitting, we employ some layers with the dropout regularization technique and also batch normalization to effectively optimize the training process.

2.4 Design of the Generator

Structurally, DeepGANnel is not tremendously different from the DCGAN model (Radford, Metz & Chintala, 2015) - we slightly modified the network architecture and optimized the dimension of the image input shape and noise sample and changing the input data to a time series. The architecture of generator G generally consists of strided convolutions that allow the network to undergo spatial learning with its own up-sampling; batch normalization layers that allow for stabilizing learning parameters by normalizing inputs; and Leaky ReLU activation functions for all layers, except for the Tanh function that is used in the output layer. The input data is a sliding window of time series data with a specified dimension; thus the input shape of the generator is defined as 2x1280x1 records in which the time series data is split into windows of 1280 data points. G is a series of deconvolution layers (referred to as "transposed convolution layer" in Tensorflow) to transform the noise (z) into an image with shape 2x1280x1. The output layer is a two-dimensional sample generated from the noise that can be sent to the input of the discriminator for training the model.

2.5 Design of the Discriminator

The discriminator, *D*, is a deep convolution neural network that carries Leaky ReLU activation functions at all layers for the non-linearity purpose. Similarly, this combines strided convolution layers allowing it to learn spatial down-sampling. In our D architecture, the batch normalisation layers are not used, but instead a dropout regularized technique was used at each layer. The last convolution layer in our D network is flattened and then passed into a sigmoid function for classification. Figure 2 shows the architecture of our DeepGANnel model in this work including both *G* and *D* networks. The Adam optimizer was used initially with 0.0001 learning rate for initial training process. When fine tuning some models, discriminator training was turned off temporarily (whilst the generator continued to train), if discriminator loss (see below) became too small.

2.6 Data sources

Ion channel molecule recordings came from canine articular chondrocytes isolated as described previously (Mobasheri, Lewis, Maxwell, Hill, Womack & Barrett-Jolley, 2010). Animals were previous euthanized for unassociated veterinary reasons; no animals were killed or harmed for this study. Chondrocytes were lifted from culture flasks with x1 Trypsin-EDTA and re-suspended in physiological buffer and plated onto glass bottom dishes. The cells were incubated for 30-60 minutes at 37°C in order to adhere to the dishes. Data was then recorded using cell-attached patch clamp with an Axopatch 200a amplifier (Axon Instruments, USA). Low-pass filtering was set to 1 kHz and data were digitized at 5 kHz with a Digidata 1200A interface. Recordings were made with WinEDR (John Dempster, University of Strathclyde, UK). Patch pipettes were fabricated using fire-polished 1.5 mm

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

o.d. borosilicate glass capillary tubes (Sutter Instrument, USA, supplied by INTRACEL, UK). They were pulled using a two-step electrode puller (Narishige, Tokyo, Japan) and when filled with recording solutions had a resistance of approximately 5-8 MΩ depending on the patch-clamp method used. Data idealisation/annotation was performed with QuB (Nicolai & Sachs, 2013).

2.7 Evaluation metrics

It is considered that GANs are successful when they implicitly learn the distribution of samples of the real dataset. We assess the efficiency of the proposed DeepGANnel model to simulate single molecule data by comparing real to GAN simulated data. The standard metrics for GAN are the so-called generator loss and discriminator loss. These are calculated as the logistic binary cross entropy loss – this is calculated as:

$$L = -(y\, log(p) + (1-y)\, log(1-p))\#(6)$$

Where y is the true label and p the predicted label. For the generator this is calculated once, with the true label being if the output is fake or not, and the predicted label being if the discriminator predicted if the output is fake. For the discriminator, this is calculated twice then totalled – once for the fake output as above, and again in the same manner for real outputs.

We can also use other metrics to measure model success. To do that, we will utilize a two-sample test called maximum mean discrepancy(Gretton, Borgwardt, Rasch, Schölkopf & Smola, 2007), and a classical evaluation metric for time series defined as dynamic time warping(Sakoe & Chiba, 1978).

Maximum Mean Discrepancy

The maximum mean discrepancy (MMD) measures the dissimilarity between two probability distributions P_r and P_g – one from the real data distribution and one from the GAN respectively, by comparing statistics of the samples. It is considered as the squared difference of the statistics between the two samples (the MMD^2). Given a kernel $K : X \times Y \rightarrow R$, and samples $\{x_i\}_{i=1}^{N}$, $\{y_j\}_{j=1}^{M}$, an estimation of MMD^2 is equated as follows:

$$MMD_u^2 = \frac{1}{n(n-1)}\sum_{i=1}^{n}\sum_{j\neq i}^{n} K(x_i, x_j) - \frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} K(x_i, y_j) + \frac{1}{m(m-1)}\sum_{i=1}^{m}\sum_{j\neq i}^{m} K(y_i, y_j)\#(7)$$

The smaller MMD measurement, the greater similarity between the distributions. A modified Python library's (Tensorflow) two-sample test(Gretton, Borgwardt, Rasch, Scholkopf & Smola, 2012) was used to determine MMD using the Gaussian kernel for the above calculation.

Dynamic Time Warping

Dynamic time warping (DTW) is a traditional calculation for measuring the dissimilarity between two time series data. DTW warps the series of data with temporal alignment and calculating the distance between the two different time series, such that Euclidean distance between aligned time series. The formulation of DTW is based on the following optimization problem(Serra & Arcos, 2014):

$$D_{i,j} = f(x_i, y_j) + min\{D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1}\}\#(8)$$

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

While $i = 1, \ldots, N$ and $j = 1, \ldots, M$ where N and M are the lengths of the time series data for x and y, respectively. It is also determined that $f(x_i, y_j) = (x_i - y_j)^2$. Due to the large collection of simulated ion channel records that will be studied in this work, a dedicated library (FastDTW (Salvadora & Chan, 2007)) was utilized to estimate DTW metric as it improves the computational cost.

2.8 Computing platform

A Nvidia Titan X GPU with 12 GB of RAM was used for the experiments to train and generate realistic ion channel molecule current records alongside the event classifications of generated records using the DeepGANnel model. A Python Jupyter notebook was developed to demonstrate the training process of the GAN model. Tensorflow 2.x was utilized to build the GAN model as it provides sufficient support for GPUs and provides a user-friendly interface to engage with tensors and other Keras/Tensorflow modules.

3. Results

Single (ion channel) molecule data was recorded using the patch-clamp technique in inside out mode from canine articular chondrocytes using our standard protocols, and a batch of some 30 seconds recorded under constant conditions (temperature, membrane potential etc.). This was then annotated using QuB to produce a two-dimensional signal (dimension 1 = raw signal, dimension 2 = continuous annotation). These data were copied 200 times with an invisibly small amount of gaussian noise (approximately 0.1% of signal amplitude) applied to each copy. Following min-max scaling and reshaping these data were passed to the DeepGANnel model for approximately 10,000 epochs. Figure 3A shows the characteristic evolution of discriminator and generator losses. Typical generator losses were 10x or more the discriminator loss. The first 250 epochs of the more sophisticated MMD and DTW losses are shown in Figure 3B. Figure 3 C and D show examples of raw (real) input data and a representative strip of post train DeepGANnel generated data. The two first analyses that are typically conducted in patch-clamp research are amplitude histograms and kinetic analysis. Amplitude histograms are shown in Figures 3E (real events) and Figure 3F (GAN simulated events), these are clearly similar in terms of size and shape.

For a more in-depth analysis we conducted full kinetic analysis of both raw (real molecular data) and GAN simulated data. These are shown in Figure 4, and there are similarities and differences between the real and GAN events. In terms of closed times, it is apparent that whilst the over-all distribution is similar between real (Figure 4A) and GAN (Figure 4C) there are differences. The real data included some long closed-events that are absent from the GAN simulated equivalent. In terms of open times again the over-all distribution is similar between real (Figure 4B) and GAN simulated (Figure 4D), but there is attenuation of the very long open sojourns in the GAN simulated data.

4. Discussion

In this work, we have generated synthetic raw single-molecule timeseries data along with continuous synchronised annotation/idealisation using a generative adversarial network (GAN) based on both real ion channel single molecule data from cultured chondrocytes. We demonstrate that the GAN generated raw ion channel data was similar to those obtained by real ion channel data.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

A central problem in single molecule, including "ion channel" research is that analyses of data is laborious and frequently requires a degree of expert hand crafting to complete. The first step in such analysis is idealisation of the record, or in machine learning terms, annotating or labelling. Each time point (of which there will be many million) needs to be annotated as to how many molecule pores are open at that instant. This the becomes, effectively, a two-dimensional representation of the data. Currently we are working with simple datasets with low levels of one type of molecule in the dataset, but in the future such analysis will have to extend this to more complex datasets. Clearly new analysis methods will also be necessary to get the maximum amount of information from complex single molecule data. A number of tools have been developed to address (Colquhoun, Hatton & Hawkes, 2003; Gnanasambandam, Nielsen, Nicolai, Sachs, Hofgaard & Dreyer, 2017; Juette et al., 2016; Nicolai & Sachs, 2013) and our own Deep-Channel deep learning model (Celik et al., 2020). For further development of similar or enhanced tools there is a lack of available training data. There are two clear choices for such data; (i) real biological data that has been annotated in some way or (ii) synthetic datasets. Both of these approaches have limitations and biases. Real data cannot be perfectly labelled, and the ground truth will be only an approximate ground truth, therefore new machine learning methods will learn the errors of the existing technology. Furthermore, only simple datasets can be annotated and so this sets an upper limit on the complexity of the datasets that could be analysed by potential new tools. However, synthetic datasets also contain many biases and limitations, some of which may be entirely unanticipated or recognised. Therefore, the starting problem that our work addressed here was to create very large datasets of ion channel single molecule data that could be used to develop single molecule analysis tools. We chose to investigate if GAN technology could provide a useful alternative source of data. In principle this would have the advantages of synthetic data in that one could produce unlimited amounts, but still retain nuance and subtle authenticity missed by mere simulation. We are also hopeful that such synthetic data could be used in development of ion channel modelling software and may allow for a novel type of data inference, extracting critical features in datasets that may be overlooked by traditional analysis.

The literature includes previous examples where timeseries data can be simulated with a GAN (For example, Zhu, Ye, Fu, Liu & Shen, 2019), but these lack the synchronous labelling critical in single molecule analysis or many other physiological studies. We too found, in preliminary work (data not shown), that generating single-molecule timeseries with a basic GAN (Vanilla-GAN) model was very effective at producing authentic ion molecule signal patch-clamp signal, but this also lacked the output of the critical timepoint-by timepoint labelling necessary to meet our goals of creating valid alternative datasets. However, by switching to a 2DCNN based model in a shape of (samples x 2 x 1280 x 1) was very effective, with a small amount of carefully annotated seed data generating unlimited synthetic copies. In Figures 3 and 4 we show a direct comparison for a typical electrophysiological work up of both the original and DeepGANnel synthesised data. On the supplementary information and public repository, we include a movie of the training process. The match between synthetic kinetic analysis and the real data is not perfect, but rather close. The notable exception is that the longer states (both open and closed) are missing or have a diminished representation. We attribute this to the finite window ("image" width or "record length") size 1280 that was used. Also as stated in the methods this was cropped to remove leading and trailing artefacts. Perhaps models using far greater window lengths would be possible, but this does not appear to be a major problem for our purpose (since most single molecule events durations are within this window) and it would increase the model complexity many-fold. This model took 24-48 hours to train on our system and note that performance peaked but would deteriorate if it was left indefinitely. Our code included a call-back to reduce learning rate as epochs progressed, but we still chose to stop the modelling manually. The ever-increasing GPU power make ever larger window sizes less of an issue in the future.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

The potential for further exploitation, of GAN technology in electrophysiology beyond the current use for creation of datasets is immense. One future goal will be to simulate far more complex data signals, but we still have the limitation on how to acquire the fully annotated seed data in the first place. Potentially painstaking manual annotation of very short sections of data with known numbers of ion channel molecules by a number of human experts would be possible. Furthermore, it is possible that single-molecule GAN could be used more directly in electrophysiological modelling. Currently, single molecule behaviour within such models is derived by a set of differential equations based on a set of measured or even estimated parameters (Feetham, Nunn, Lewis, Dart & Barrett-Jolley, 2015), but it may be possible in the future to use GAN to generate more realistic stochastic behaviour directly. Additionally, future studies will investigate whether interpretability methods can be used to identify important, defining features, within each different dataset that are missed either by eye or by standard single molecule analysis techniques.

The architecture we present here, using deep learning to generate physiological timeseries data with continuous annotations, could also be adapted easily for additional usability for equivalent systems in physiology. For example, action potential simulation or ECG simulation. As proof-of-principle we show here that indeed DeepGANnel can easily synthesise telemetered ECG signal, again fully annotated (Figure 5). In this example the annotation dimension is merely beat (1) or no beat (0), but this could easily be extended to include P-wave (2), T-wave (3), or abnormal event (4) etc with trivial code adaptation.

In summary, GAN are increasingly proving a viable method to generate synthetic datasets for biological research, and here we show an implementation that allows simulation of time dependent single molecule (patch clamp ion channel protein) activity along with a continuous state annotation that is extendable for an array of physiological uses.

5. Acknowledgements

This work was funded by the BBSRC with grants: BB/R022143/1, BB/S008136/1 and a BBSRC DTP Studentship.

6. References

Acharya UR, Fujita H, Lih OS, Hagiwara Y, Tan JH, & Adam M (2017). Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network. Information Sciences 405: 81-90.

Albrecht T, Slabaugh G, Alonso E, & Al-Arif S (2017). Deep learning for single-molecule science. Nanotechnology 28: 423001.

Boza V, Brejova B, & Vinar T (2017). DeepNano: Deep recurrent neural networks for base calling in MinION nanopore reads. PLoS One 12: e0178751.

Celik N, O'Brien F, Brennan S, Rainbow RD, Dart C, Zheng Y, *et al.* (2020). Deep-Channel uses deep neural networks to detect single-molecule events from patch-clamp data. Commun Biol 3: 3.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Chartsias A, Joyce T, Giuffrida MV, & Tsaftaris SA (2018). Multimodal MR Synthesis via Modality-Invariant Latent Representation. IEEE Transactions on Medical Imaging 37: 803-814.

Chen X, Duan Y, Houthooft R, Schulman J, Sutskever I, & Abbeel P Infogan: Interpretable representation learning by information maximizing generative adversarial nets.2172-2180.

Cheng M, Sori WJ, Jiang F, Khan A, & Liu S Recurrent Neural Network Based Classification of ECG Signal Features for Obstruction of Sleep Apnea Detection.

Colquhoun D, Hatton CJ, & Hawkes AG (2003). The quality of maximum likelihood estimates of ion channel rate constants. J Physiol 547: 699-728.

Costa P, Galdran A, Meyer MI, Niemeijer M, Abramoff M, Mendonca AM, *et al.* (2018). End-to-End Adversarial Retinal Image Synthesis. IEEE Transactions on Medical Imaging 37: 781-791.

Delaney AM, Brophy E, & Ward TE (2019). Synthesis of Realistic ECG using Generative Adversarial Networks. arXiv preprint arXiv:190909150.

Donahue C, McAuley J, & Puckette M (2018). Adversarial audio synthesis. arXiv preprint arXiv:180204208.

Esteban C, Hyland SL, & Rätsch G (2017). Real-valued (medical) time series generation with recurrent conditional gans. arXiv preprint arXiv:170602633.

Feetham CH, Nunn N, Lewis R, Dart C, & Barrett-Jolley R (2015). TRPV4 and KCa ion channels functionally couple as osmosensors in the paraventricular nucleus. Br J Pharmacol 172: 1753-1768.

Fraiwan L, & Lweesy K Neonatal sleep state identification using deep learning autoencoders.228-231.

Gao D, Celik N, Wu X, Williams BM, Stylianides A, & Zheng Y A Novel Deep Learning Based OCTA De-striping Method.189-197.

Gnanasambandam R, Nielsen MS, Nicolai C, Sachs F, Hofgaard JP, & Dreyer JK (2017). Unsupervised idealization of ion channel recordings by minimum description length: application to human PIEZO1-channels. Frontiers in neuroinformatics 11: 31.

Golany T, & Radinsky K PGANs: Personalized Generative Adversarial Networks for ECG Synthesis to Improve Patient-Specific Deep ECG Classification. 33:557-564.

Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, *et al.* (2014). Generative Adversarial Nets. Advances in Neural Information Processing Systems 27 (Nips 2014) 27.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Gretton A, Borgwardt K, Rasch M, Schölkopf B, & Smola AJ A kernel method for the two-sample-problem.513-520.

Gretton A, Borgwardt KM, Rasch MJ, Scholkopf B, & Smola A (2012). A Kernel Two-Sample Test. J Mach Learn Res 13: 723-773.

Hamill OP, Marty A, Neher E, Sakmann B, & Sigworth FJ (1981). Improved Patch-Clamp Techniques for High-Resolution Current Recording from Cells and Cell-Free Membrane Patches. Pflugers Archiv-European Journal of Physiology 391: 85-100.

Hartmann KG, Schirrmeister RT, & Ball T (2018). EEG-GAN: Generative adversarial networks for electroencephalograhic (EEG) brain signals. arXiv preprint arXiv:180601875.

Hille B (2001) *Ion Channels of Excitable Membranes*. 3rd edn. Sinauer Associates.

Hodgkin AL, & Huxley AF (1952). A Quantitative Description of Membrane Current and Its Application to Conduction and Excitation in Nerve. Journal of Physiology-London 117: 500-544.

Jirayucharoensak S, Pan-Ngum S, & Israsena P (2014). EEG-Based Emotion Recognition Using Deep Learning Network with Principal Component Based Covariate Shift Adaptation. The Scientific World Journal 2014: 627892.

Juette MF, Terry DS, Wasserman MR, Altman RB, Zhou Z, Zhao H*, et al.* (2016). Single-molecule imaging of non-equilibrium molecular ensembles on the millisecond timescale. Nature methods 13: 341.

LeCun Y, Bengio Y, & Hinton G (2015). Deep learning. Nature 521: 436-444.

Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A*, et al.* (2017). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. 30th Ieee Conference on Computer Vision and Pattern Recognition (Cvpr 2017): 105-114.

Mobasheri A, Lewis R, Maxwell JEJ, Hill C, Womack M, & Barrett-Jolley R (2010). Characterization of a stretch-activated potassium channel in chondrocytes. Journal of Cellular Physiology 223: 511-518.

Neher E, & Sakmann B (1976). Single-Channel Currents Recorded from Membrane of Denervated Frog Muscle-Fibers. Nature 260: 799-802.

Nicolai C, & Sachs F (2013). Solving ion channel kinetics with the QuB software. Biophysical Reviews and Letters 8: 191-211.

Nunn N, Feetham CH, Martin J, Barrett-Jolley R, & Plagge A (2013). Elevated blood pressure, heart rate and body temperature in mice lacking the XLαs protein of the Gnas locus is due to increased sympathetic tone. Exp Physiol 98: 1432–1445.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Odena A (2016). Semi-supervised learning with generative adversarial networks. arXiv preprint arXiv:160601583.

Odena A, Olah C, & Shlens J Conditional image synthesis with auxiliary classifier gans.2642-2651.

Putin E, Asadulaev A, Vanhaelen Q, Ivanenkov Y, Aladinskaya AV, Aliper A, *et al.* (2018). Adversarial Threshold Neural Computer for Molecular de Novo Design. Molecular Pharmaceutics 15: 4386-4397.

Radford A, Metz L, & Chintala S (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:151106434.

Sakoe H, & Chiba S (1978). Dynamic-Programming Algorithm Optimization for Spoken Word Recognition. Ieee T Acoust Speech 26: 43-49.

Salvadora S, & Chan P (2007). Toward accurate dynamic time warping in linear time and space. Intell Data Anal 11: 561-580.

Serra J, & Arcos JL (2014). An empirical evaluation of similarity measures for time series classification. Knowl-Based Syst 67: 305-314.

Shin H-C, Tenenholtz NA, Rogers JK, Schwarz CG, Senjem ML, Gunter JL, *et al.* (2018). Medical Image Synthesis for Data Augmentation and Anonymization Using Generative Adversarial NetworksSpringer International Publishing, pp 1-11.

Sun C, Shrivastava A, Singh S, & Gupta A Revisiting Unreasonable Effectiveness of Data in Deep Learning Era.

Supratak A, Dong H, Wu C, & Guo Y (2017). DeepSleepNet: a model for automatic sleep stage scoring based on raw single-channel EEG. IEEE Transactions on Neural Systems and Rehabilitation Engineering 25: 1998-2008.

Yang JB, Nguyen MN, San PP, Li XL, & Krishnaswamy S (2015). Deep Convolutional Neural Networks On Multichannel Time Series For Human Activity Recognition. Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (Ijcai): 3995-4001.

Zhang H, Xu T, Li HS, Zhang ST, Wang XG, Huang XL, *et al.* (2017). StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. Ieee I Conf Comp Vis: 5908-5916.

Zhang Q, Wang H, Lu H, Won D, & Yoon SW Medical Image Synthesis with Generative Adversarial Networks for Tissue Recognition.

Zhu F, Ye F, Fu YC, Liu Q, & Shen BR (2019). Electrocardiogram generation with a bidirectional LSTM-CNN generative adversarial network. Sci Rep-Uk 9.

bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

FIGURE and LEGENDS

Fig. 1 | Pipelines for DeepGANNEL Development. Data are recorded using standard electrophysiological techniques. These raw data are then labelled, again using existing software and expert supervision. Pre-processing then makes copies of these data adding a small amount of gaussian noise to the raw data and reshapes the two signals into 2D "images" (2x1280).

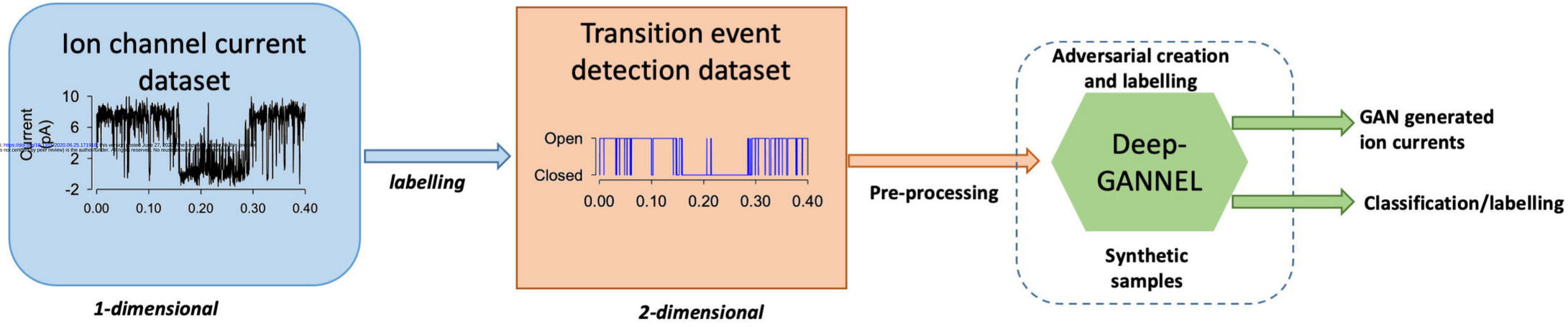
Fig. 2 | DeepGANnel network architecture. On the left is a generator network of DeepGANNEL in which the input is a random noise and output is a 2D image by 2x1280 that is converted from 1-D time series. The generator model in this architecture uses three deconvolution layers (upsampling) to produce an image from seed (random noise shape by 2x1280). Several Dense layers are used to take the noise input and transform into a desired image by up-sampling (Deconv or transposed convolutional layer) steps size of 2x1280. At each layer, Leaky Rectified Linear Unit (LeakyReLU) is used as activation function with batch normalization, except output layer which uses tanh activation function. The discriminator model comprises of several convolutional layers with the same activation function (LeakyReLU) at each layer to classify whether an image is real or fake by comparing the sampled data to real data. Dropout layers were also appended to all convolutional layers except input layer with the value of 0.3 to reduce overfitting. Finally, a Dense output layer is used after flattening the network structure with classified labels.

Fig. 3 | Realistic labelled single molecule data are synthesised by DeepGANnel (A) Discriminator and Generator losses plotted per epoch during training. (B) The maximum mean discrepancy (MMD) and dynamic time warping (DTW) evolution over the first 250 epochs as they approach equilibrium. (C and D) Examples of real (C) and GAN synthesised (D) labelled data records. (E and F) All points amplitude histograms calculated from the real (E) and GAN synthesised (F) data.

Fig. 4 | Comparative kinetic analyses of real and DeepGANnel simulated data. (A and B) Kinetic analysis from the real (input) single molecule data with closed times (A) fit with 4 exponentials and open times (B) fit with 3 exponentials. The respective time constants and weights are displayed in the inset tables. (C and D) Kinetic analysis from the GAN simulated (output) single molecule data with closed times (C) fit with 3 exponentials and open times (D) fit with 2 exponentials. The respective time constants and weights are again displayed in the inset tables.

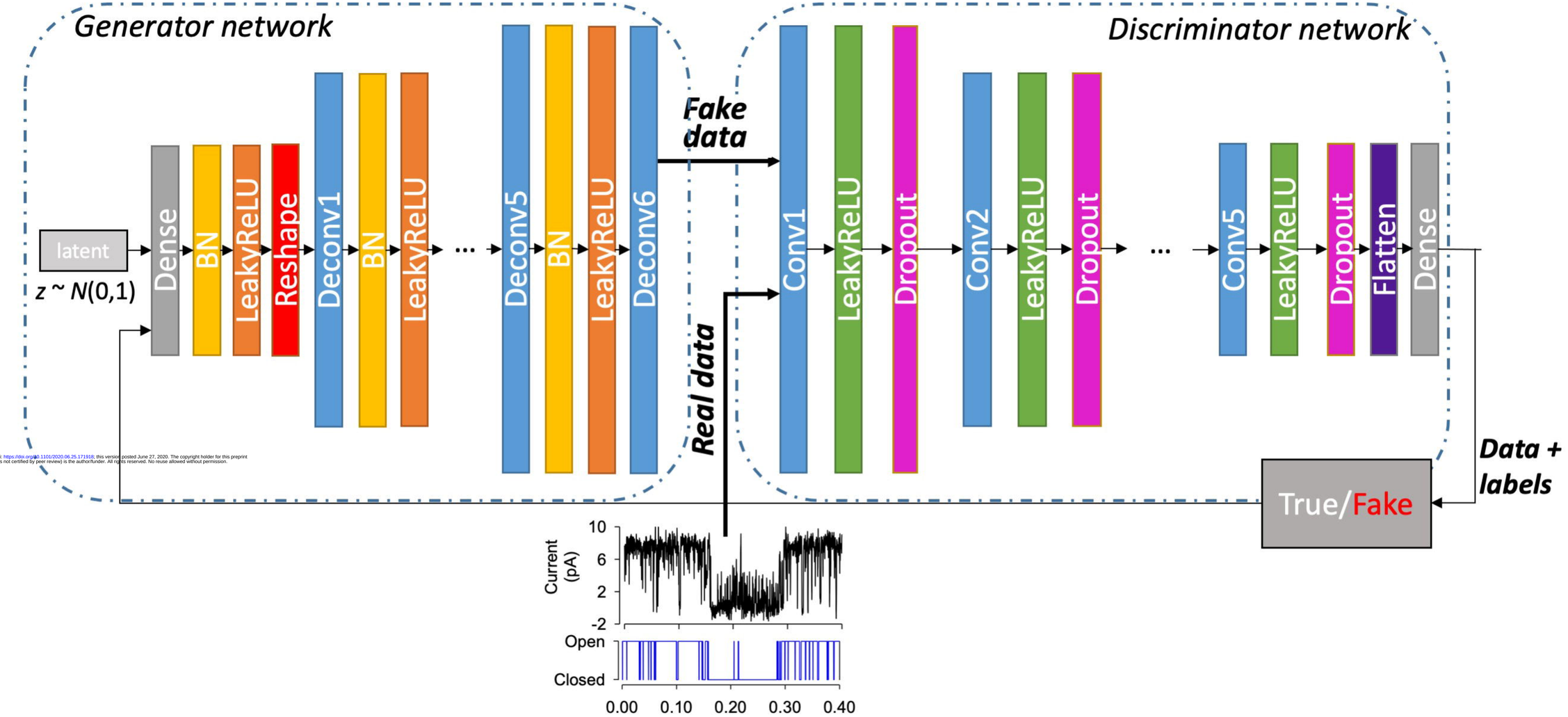
Fig. 5| General usability of DeepGANnel for 2-dimensional physiological time series data. In this example we fed rodent ECG data (dimension 1) along with a beat annotation (dimension 2, peak of r-wave) into DeepGANel and trained with 10,000 epochs until realistic rodent ECG data was generated. Data collected via electrocardiogram transmitters from male Wistar rats (ETA-F20; Data Sciences International, St Paul, MN, USA) as previously described (Nunn, Feetham, Martin, Barrett-Jolley & Plagge, 2013), briefly ECG signal was digitized to a PC with a CED Micro1401 using Spike2 at 5 kHz. In principle this could also be encoded with further annotations such as t-wave (note rats do not show a significant p-wave).

Figure 1



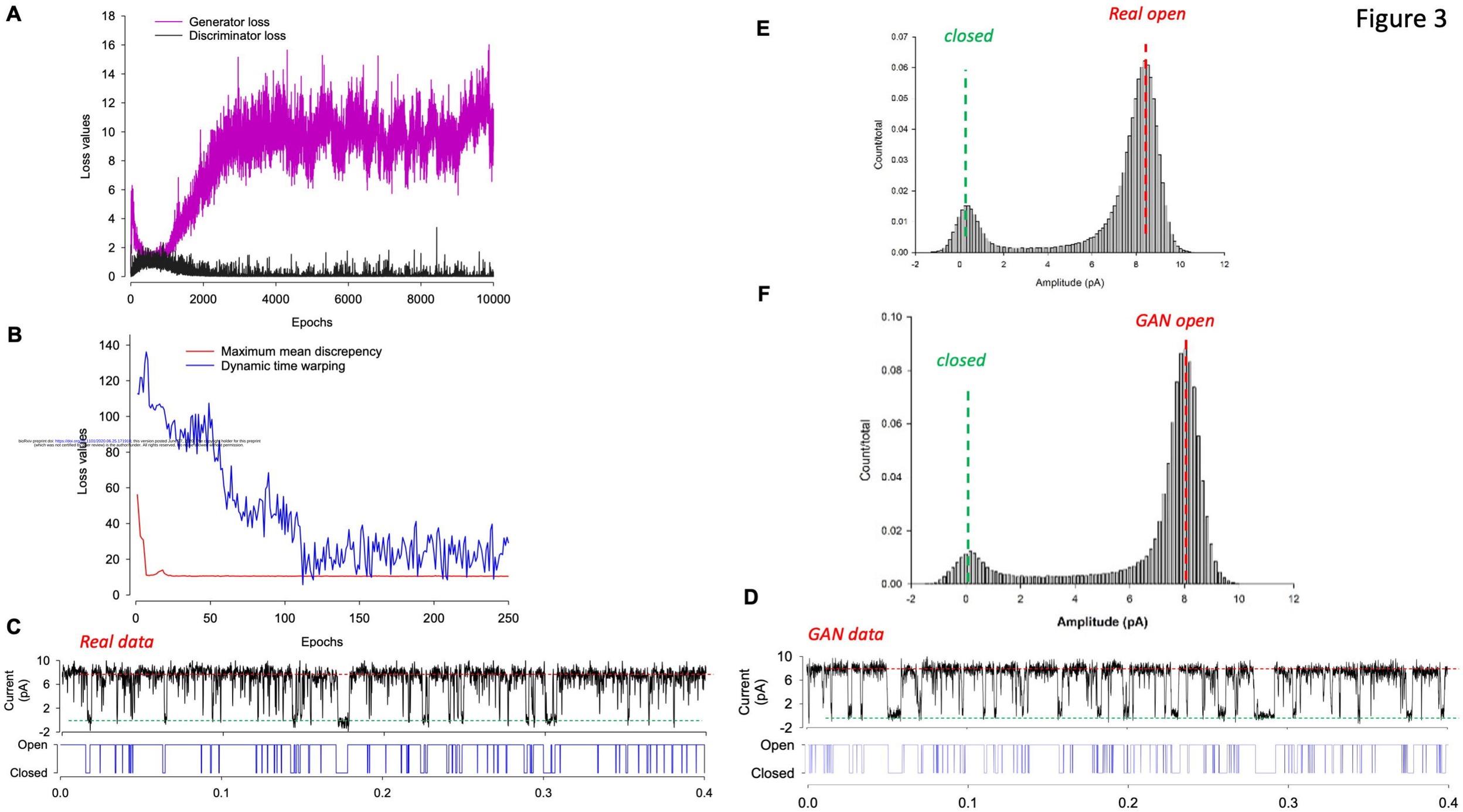
bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020 The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 2



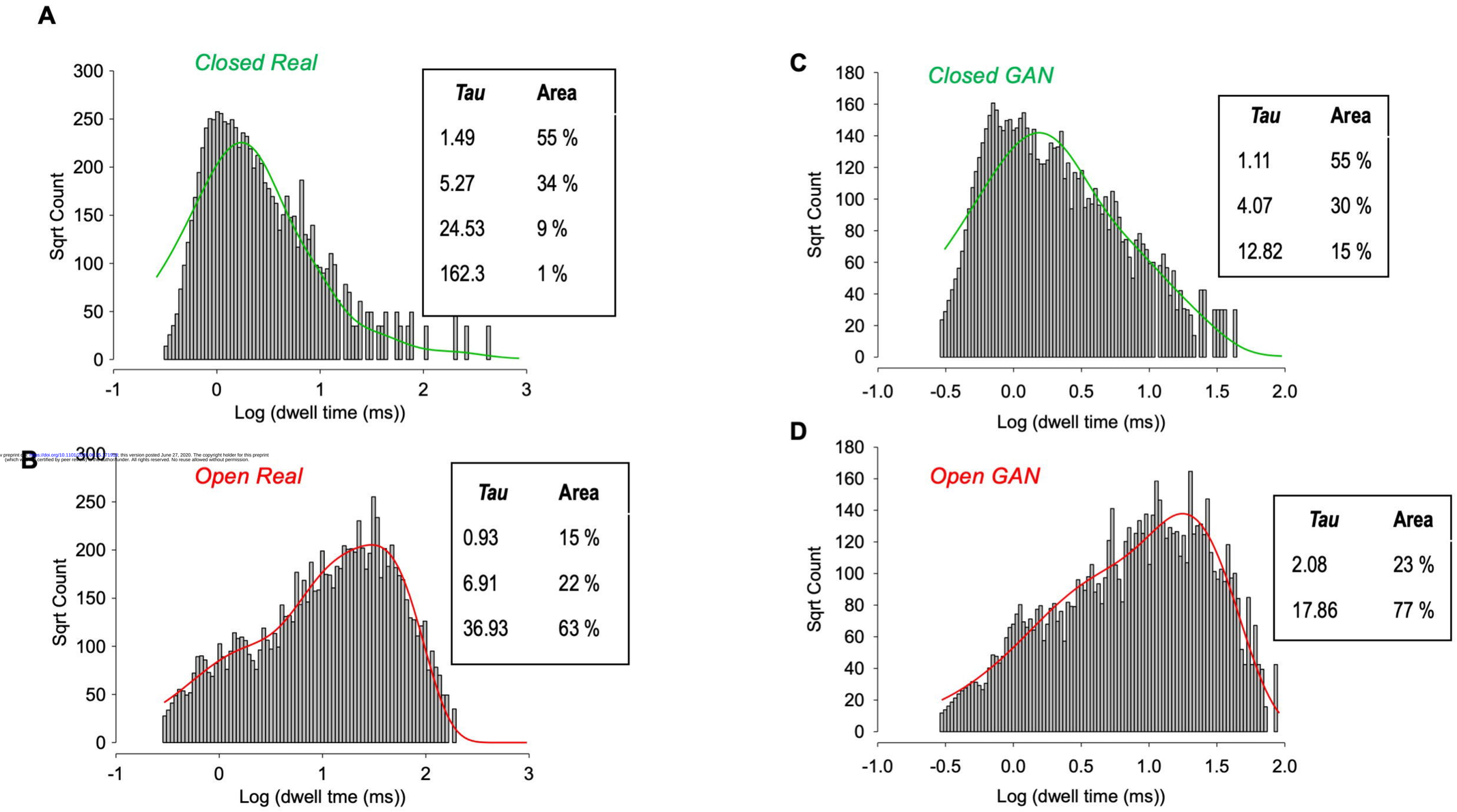
bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 3



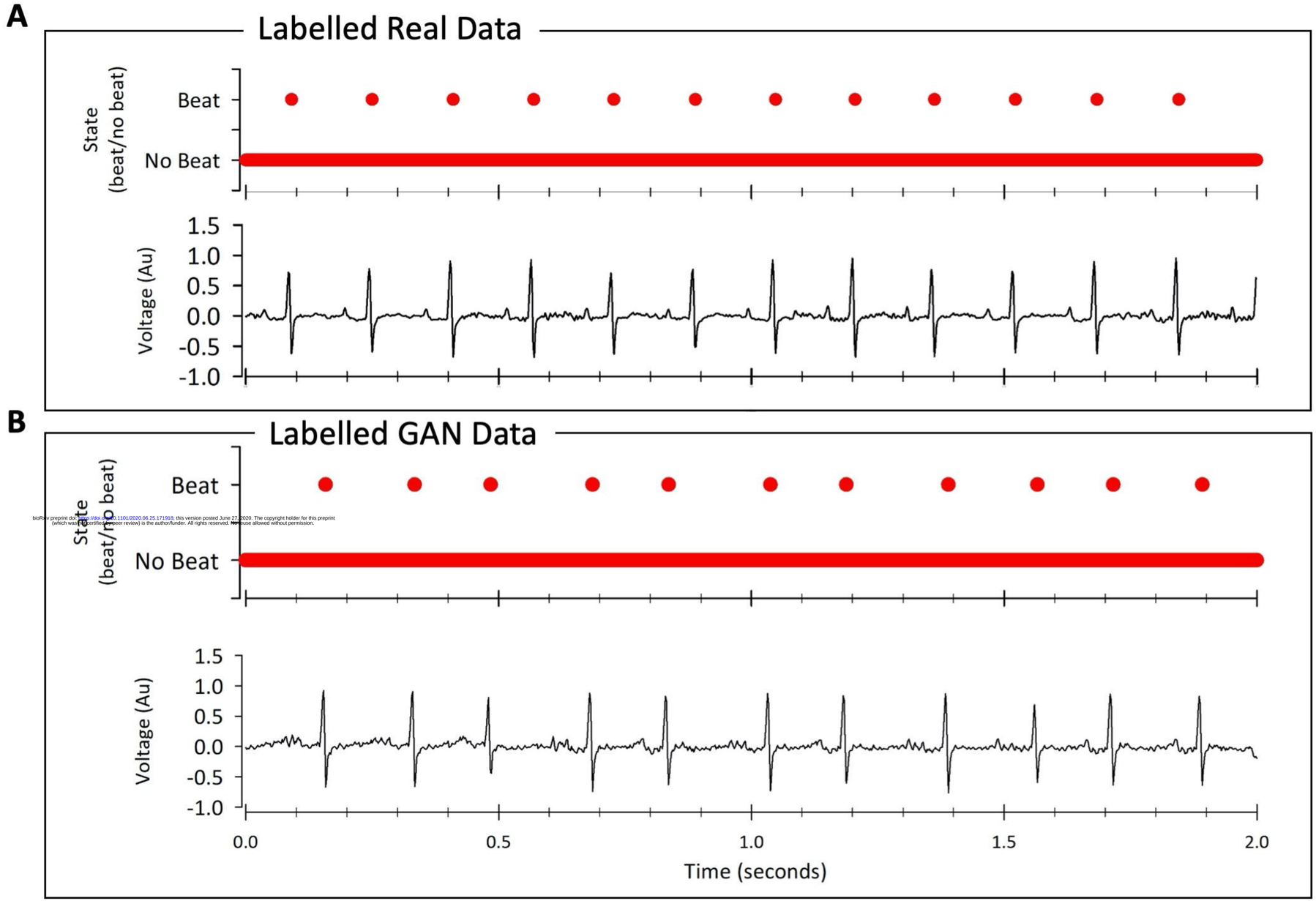
bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 4



bioRxiv preprint doi: https://doi.org/10.1101/2020.06.27.175175; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 5



bioRxiv preprint doi: https://doi.org/10.1101/2020.06.25.171918; this version posted June 27, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.