Offspring GAN Augments Biased Human Genomic Data

Supratim Das
Indian Institute of Science Education and Research - Pune
Pune, MH, India
supratim.das@students.iiserpune.ac.in

Xinghua Shi*
Department of Computer & Information Sciences
Temple University
Philadelphia, PA, USA
mindyshi@temple.edu

ABSTRACT

Genomic data have been used for trait association and disease risk prediction for a long time. In recent years, many such prediction models are built using machine learning (ML) algorithms. As of today, human genomic data and other biomedical data suffer from sampling biases in terms of people's ethnicity, as most of the data come from people of European ancestry. Smaller sample sizes for other population groups can cause suboptimal results in ML-based prediction models for those populations. Suboptimal predictions in precision medicine for some particular group can cause serious consequences limiting the model's applicability in real-world problems. As data collection for those populations is time-consuming and costly, we suggest deep learning-based models for in-silico data enhancement. Existing Generative Adversarial Network (GAN) models for genomic data like Population scale Genomic conditional-GAN (PG-cGAN) can generate realistic genomic data while trained on fairly unbiased data but fails while trained on biased data and encounters severe mode collapse. Our proposed model, Offspring GAN, can resolve the mode collapse issue even when trained in strongly biased genomic datasets. Our results demonstrate the ability of Offspring GAN to generate realistic and diverse label-aware data, which can augment limited real data to alleviate biases and disparities in genomic data. We also propose a privacy-preserving protocol using Offspring GAN to protect the privacy of genomic data.

CCS CONCEPTS

• **Applied computing** → **Computational genomics**.

KEYWORDS

machine learning; deep learning; data bias; generative adversarial networks; data augmentation; genomics; mode collapse

ACM Reference Format:
Supratim Das and Xinghua Shi*. 2022. Offspring GAN Augments Biased Human Genomic Data. In *13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '22), August 7–10, 2022, Northbrook, IL, USA*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3535508.3545537

BCB '22, August 7–10, 2022, Northbrook, IL, USA

ACM ISBN 978-1-4503-9386-7/22/08…$15.00
https://doi.org/10.1145/3535508.3545537

1 INTRODUCTION

Recent availability of large-scale genomic, epigenomic, transcriptomic, proteomic, and clinical data have opened up an opportunity for building robust predictive models toward genomic medicine utilizing those abundant datasets. Within this scope, researchers have developed various models to predict disease risks and clinical outcomes built upon analyzing and integrating such rich data. The complexity of this prediction problem arises due to the underlying complex interactome agglomerated in a hierarchical organization from the molecular level to the organ system level, which gives rise to a disease phenotype. Due to the non-linear interactions between each component, the reductionist approach for predicting disease phenotype fails in system-level applications. Machine learning (ML) algorithms such as deep learning algorithms effectively capture the effects of all linear and non-linear interactions for all components and are thus of great interest in predictive modeling.

Deep learning models are readily used in genomics and biomedical informatics to predict disease sub-type [13], disease susceptibility or treatment outcome for complex diseases such as cancer or autism spectrum disorder [50], and predicting other phenotypes using genomic[19], transcriptomic, clinical and imaging[40] data. Despite recent advances, deep learning models, similar to other ML models, are vulnerable to data biases. Training a deep learning model with a biased dataset results in sub-optimal prediction accuracy.

As of today, most biomedical data in publicly available data sets are sampled from people of European ancestry, and the sample size of other populations (e.g. Africans, Asians), is considerably low [14]. As these data sets are heavily biased towards Caucasians, deep learning models may lead to non-optimal or even wrong predictions for non-Caucasians. In precision medicine, this can cause serious undesired consequences for people of those ethnicities which have lesser representation in the training dataset when building a predictive model. For instance, researchers have used ML algorithms to identify six Single Nucleotide Polymorphisms (SNPs) as crucial contributors to Parkinson's disease[17]. As the individuals in their datasets were predominantly of European descent, they identified the limitation of their model being relevant for only one ethnic group[17]. Similarly, a lack of genomic data for particular ethnicities might affect the performance of genotype imputation, causing sub-optimal results for those specific ethnicities.However, it is a time-consuming and costly process to collect more diverse genomic data (and sometimes it may become infeasible due to various constraints). To overcome data inequality, transfer learning has been proposed [14].

In this study, we tackle this data disparity problem using a Generative Adversarial Network (GAN) to produce realistic synthetic data for minority groups, which can later be augmented with the real

data to increase sample size and mitigate biases in the dataset. GANs have become a successful method for image and video generation in the field of computer vision [23]. Augmenting GAN synthetic medical images improves Convolutional Neural Network (CNN) based classification task for predicting skin cancer [39].

With regard to augmenting or synthesizing genomic data, multiple studies have reported different approaches with distinct research objectives. In one study, researchers used both GANs and Restricted Boltzmann Machine (RBM) for synthetic haplotype data generation and experienced overfitting with the RBM model and underfitting with the GAN model [48]. Another group developed pg-gan which combines the isolation-with-migration model to recover some evolutionary parameters to produce labeled haplotypes[45]. Additionally, a Population scale Genomics conditional Generative Adversarial Networks (PG-cGAN) [9] was developed to generate realistic synthetic genotype data from real data to increase the sample size. Nonetheless, PG-cGAN was trained on unbiased Single Nucleotide Polymorphism (SNP) data of Human Leukocyte Antigen (HLA) region from the 1000 genome project. Although PG-cGAN fails to prove that it would work with heavily biased datasets and generated datasets would not reinforce existing data biases, it provided a proof of concept for translating GAN into the realm of genomic data augmentation.

An ideal GAN would have high data fidelity and variation. However, it is observed in real life that most GANs fail to achieve both at same time. Sometimes the generated data distribution fails to synthesize daUniversity of Göttingenta with variation with respect to any certain feature present in real data. Such scenarios are called mode collapse[18, 38]. The $n-$ dimensional histogram of synthetic data misses one or more modes than the real data distribution, hence the name mode collapse. In GAN literature, particularly in computer vision, we have recently witnessed the development of various new methods to mitigate the problem of mode collapse while taking care of data quality. Those methods (summarized in **Table 1**) utilize a variety of strategies including the regularization of discriminators [4, 30], changing loss functions [4, 29, 29], adding an autoencoder[5, 6, 34], replacing the discriminator with a classifier [33] or using more than two players[7, 12, 24, 47] in the GAN model.

Biased training data sets can intensify the mode collapse issue. Hence we have developed a 4-player GAN with a hinge loss to tackle this issue. Besides using one generator and one discriminator, two Mendelian hybridizers are added to the architecture. We call the input of a Mendelian hybridizer the genotype of parental generation (P_1) and output to be the genotype of offspring generation (F_1). Hence the name Offspring GAN.

The first Mendelian hybridizer is applied to increase the sample size for the populations with lower representation. The generator of a trained Offspring GAN is capable of data generation, which is passed to the second Mendelian hybridizer, and it can produce offspring generation of synthetic data with a lower computational cost than the generator.

Strategy	GANs
Regularizing discriminator	W-GAN[4, 42] (Weight constrains), SNGAN[30] (Spectral normalization)
Adding classifier	Semi Supervised GAN[33], R^3 cGAN [25], Triple GAN[24], Enhanced Triple GAN[47]
Changing loss function	W-GAN[4] (W_1loss), MHGAN[20] (Hinge loss), Bures GAN[29] (Bures metric)
Adding autoencoder	BEGAN[5], CS-BEGAN[6], BEGANv3[34]
Including >2 players	Triangle GAN[12], Triple GAN[24], Enhanced Triple GAN[47], Microbatch GAN[31], SGAN[7]

Table 1: Existing strategies used in recent GAN architectures to mitigate mode collapse while preserving high data fidelity.

2 METHODS

2.1 Data

We evaluated our proposed Offspring GAN and PPO-GAN against a previously published PG-cGAN model to generate realistic data for Human Leukocyte Antigen (HLA) genotypes within a 3 Mbp stretch at chromosome 6p21.31. extracted from the 1000 Genome Project [1]. Studies have correlated some of those polymorphisms with various disease phenotypes like non-small cell lung cancer (NSCLC)[21], multiple sclerosis[46], Parkinson's disease [35], Takayasu arteritis[46]; prevalence and susceptibility of various infectious diseases[36] like leprosy [3]. HLA region is also strongly correlated with autoimmune diseases like type 1 diabetes melitus[37]. In the HLA dataset, there are 504 East Asian (EAS),504 American (AMR), 503 European (EUR), 504 African (AFR), and 489 South Asian (SAS) samples. Since the whole data set is fairly unbiased across continental populations, we trained our model with subsampled biased data and compared generated data with the whole unbiased data to check the model's applicability in the real world where larger unbiased data is not available.

We subsampled East Asian (EAS), African (AFR), and South Asian (SAS) genotypes to 20% of each from this fairly unbiased HLA dataset while keeping all samples of American (AMR) and European (EUR) populations to create a biased HLA dataset (illustrated in **Fig. 1**) which would imitate existing data bias in the most biomedical dataset. Unlike PG-cGAN[9] we used this biased data set to train Offspring GAN, PPO-GAN, and PG-cGAN. In the final subsampled biased real data, sample from EAS, AFR, and SAS comprise only around 7.6% each. This final data is one hot encoded for being used in the training process.

2.2 Model Architecture

In this study, we develop a novel population-scale genomic augmentation method for biased genomic data, named as Offspring GAN. Offspring GAN uses four main components, two novel Mendelian

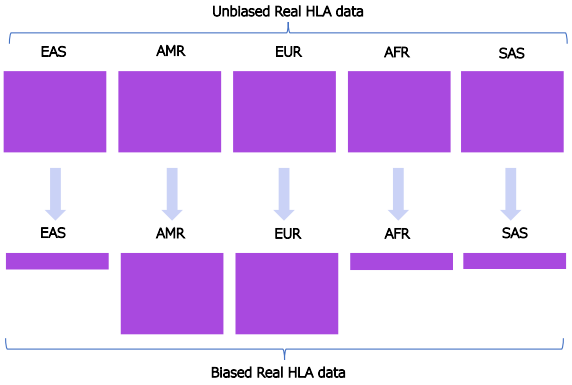


Figure 1: Data preprocessing for creating biased HLA data: Pictorial representation of subsampling scheme for creating biased real HLA dataset from the whole real dataset. For the training purpose of all GAN models, we will use the biased data only, but during evaluation, we will compare synthetic data with the whole unbiased dataset.

hybridizers, a traditional generator $G : (\mathbb{R}^z, \mathbb{R}^c) \rightarrow \mathbb{R}^x$, and a traditional discriminator $D : (\mathbb{R}^x \, \epsilon\{X_R, X_S\}) \rightarrow [0, 1]$ from cGAN. Here, z, c, x are the dimensions of noise vector, condition parameter and real data respectively. X_R is set of real data and X_S is set of synthetic data from the generator. In our case, the discriminator is modified to $D : (\mathbb{R}^x \, \epsilon\{X_{R'}, X_S\}) \rightarrow [0, 1]$, where $X_{R'}$ is the output of first Mendelian hybridizer. We used only the samples' ethnicity as the conditional parameter c. One can use necessary disease phenotypes (e.g. subtypes) along with ethnicity as the conditional parameter according to the desired downstream prediction tasks.

Given a set of real genotypes, Mendelian hybridizers produce a subset of possible genotypes for offspring generation for each population group by block-wise in-group (same ethnicity) reshuffling of the dataset. First, the Mendelian hybridizer is used for only the populations with smaller sample sizes. Generated F_1 generation genotype augmented with real data is fed into the discriminator as real data $X_{R'}$. During training the discriminator (D) tries to distinguish synthetic data X_S from real data $X_{R'}$, while the generator (G) tries to fool the discriminator by generating realistic data. The second Mendelian hybridizer generates F_1 generation genotypes of synthetic data generated by a trained generator (illustrated in **Fig. 2**). The second Mendelian hybridizer increases the robustness of our model by producing some data points which might have been missed by the generator due to mode collapse.

Along with that, we propose a privacy-preserving version of the Offspring GAN, namely Privacy-Preserving Offspring GAN (PPO-GAN). In PPO-GAN the first Mendelian hybridizer does not augment real data with the F_1 generation genotypes. Instead, for populations with lower data, only F_1 generation genotypes are used for training. Individual labs can share only the F_1 genotype with a central server where the rest of the model can be trained; and hence this strategy would protect the privacy of their real data.

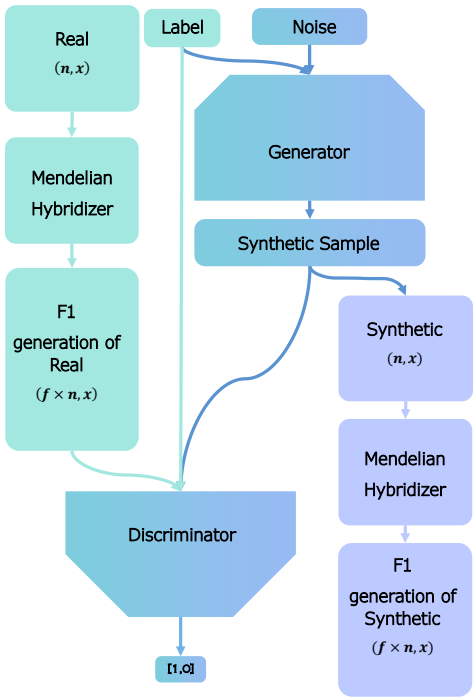


Figure 2: Architecture of Offspring GAN: Pictorial representation of the architecture of Offspring GAN. Real data is passed through the Mendelian hybridizer, and it generates genotypes offspring generation, which is passed to the discriminator as real data. On the other hand, the generator takes an input of a conditional label (which determines the ethnicity) and a noise vector and generates a synthetic genotype for the respective ethnicity. After training generator is capable of label-aware realistic data generation, which is passed to a second Mendelian hybridizer, and it produces F_1 generation genotype of synthetic data with a lower computational cost than the generator itself.

For the generator, we started with an input of a Gaussian noise of dimension 100 and a population label as the conditional parameter. We used one dense layer and two blocks of batch normalization, Leaky ReLU, 1– D convolution, 1–D Upsampling layers, and another block of batch normalization, Leaky ReLU, 1– D convolution. Discriminator contains two blocks of Leaky ReLU, 1– D convolution, and drop out and another block of Leaky ReLU, and drop out followed by a dense layer. Our rationale behind using dropout

layers (with a dropout rate of 0.3) is to regularize the discriminator. Better regularization of the discriminator is shown to be helpful in mitigating mode collapse[4, 30, 42]. It is also shown that weight capping for the discriminator (which is heavily exploited in W-GAN) helps regularise the discriminator. This stops the discriminator from over-performing the generator, saving it from the mode collapse[18, 22]. Till very recently, it is the W_1 loss function as an approximation of earth mover's distance (EMD) which was thought to be the reason behind the success of W-GAN in generating high-quality data without mode collapse. A latest study proved it is the weight constrain used in discriminator which helps to achieve good data quality and variation and not the w_1 loss[42]. Hence for further regularization of the discriminator, we used weight constraints in the discriminator, although we used Hinge loss, unlike PG-cGAN[9].

2.2.1 Mendelian hybridizer. Mendelian hybridizer takes an integer as a scaling factor and samples from one certain class label, here either of population EAS, AMR, EUR, AFR, or SAS as inputs. It partitions the samples along the feature axis in small blocks, then randomly reshuffled and joined back. Mendel's law of independent assortment (the Second law of inheritance) allows us to perform this task. To maximize the ability of the Mendelian hybridizer, one should use smaller blocks, but very small blocks might cause disrupt the linkage blocks and correlation between SNPs. As the biggest block of highly correlated SNP was of 895 SNPs, we decided to take boundaries of length 1000 SNP, so the highly correlated SNPs stay in one same block.For n input samples with f scaling factor Mendelian hybridizer yields total $f \times n$ samples. The Mendelian hybridizers yield the output of F_1 genotypes augmented with P_1 genotypes (illustrated in **Fig. 3**).

Although offspring genotype samples differ from parental genotype samples, for a large number of samples, F_1 samples follow the same probability distribution of P_1's distribution. Utilizing this fact, we also propose a privacy-oriented protocol for Offspring GAN namely PPO-GAN (Privacy-Preserving Offspring-GAN). As the goal of GAN is to learn the underlying probability distribution, it's enough to train the GAN on F_1 samples. Researchers can share their F_1 generation samples of real data to a central server instead of sharing the real data directly, preserving the privacy of their genomic data; on the central server, those F_1 genotype data can be augmented with other data and trained in Offspring GAN to generate synthetic data. As the Mendelian hybridizer does not require learning parameters during training, PPO-GAN provides a new privacy-preserving solution to providing GAN models on genomic data.

In federated learning, the private data never leaves the user's device; instead, intermediate results from training on that private data, such as gradients, are shared with the server. Nevertheless, this makes federated learning models vulnerable to Deep Leakage from Gradient (DLG) attacks, where one of the users can reconstruct the private data of others by using gradient matching with the shared gradients[26, 52]. Besides the gradient matching mechanism, others have suggested solving model inversion problems from shared parameter gradients, and their inverting gradients model is capable of reconstructing separate private data of the user from average gradient over bigger batch sizes in real life deep non-smooth architectures[15]. These studies prove the vulnerability of the gradient sharing method used in various distributed learning systems. As PPO-GAN does not require gradient sharing, it is safe from DLG[52], improved DLG (iDLG) [49] and inverting gradients[15] attacks. Not using real genotypes directly and using offspring genotypes of real data opens a new path towards privacy protection in genomics data for deep learning.

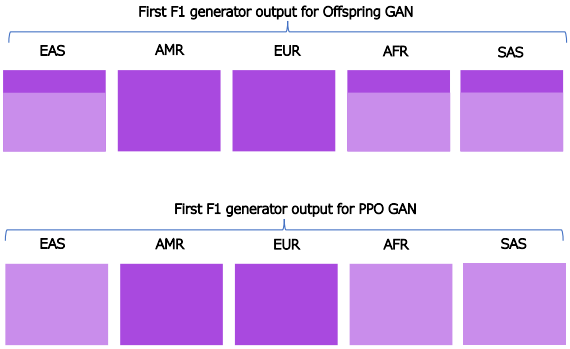


Figure 3: Difference in First Mendelian hybridizer generated output for Offspring-GAN and PPO-GAN: Pictorial representation of first Mendelian hybridizer generated output for Offspring GAN and PPO-GAN. Blocks with lighter colour represents F_1 genotype, darker colour blocks represents P_1 genotypes. For both of the models, Mendelian hybridizer is only applied for EAS,AFR, and SAS samples to balance the sample size with AMR and EUR population size. In PPO-GAN real data (i.e. P_1 generation data) is not augmented. Unlike the second Mendelian hybridizer, the first one does not affect AMR and EUR samples for our experiments.

2.2.2 Activation Function. : As discussed earlier we used Leaky relu as the activation function. Instead of setting all negative values to 0 like ReLU, Leaky relu [27] multiplies them by a coefficient α in between $[(0, 1)]$. For $\alpha = 0$, Leaky Relu turns into a normal ReLU, and for $\alpha = 1$, Leaky ReLU turns into a linear function. Equation for Leaky relu is given as:

$$h^{(i)} = \begin{cases} w^{(i)T}x & w^{(i)T}x > 0 \\ \alpha \times w^{(i)T}x & \text{else,} \quad \text{where } \alpha \text{ is the a hyper-parameter} \end{cases} \tag{1}$$

2.2.3 Loss function and optimizer. : Unlike PG-cGAN, we used a hinge loss instead of the Wasserstein loss function. For a given predicted label $\hat{y}$ (e.g. discriminator's output of real vs fake) and a real label y (e.g. real or fake data); the hinge loss is calculated as follows:

$$\mathcal{L}_{hinge}(y, \hat{y}) = max((1 - \hat{y} \times y), 0) \tag{2}$$

We used Adam optimizer for training our model.

2.3 Benchmark against PG-cGAN

We compared our models with PG-cGAN using biased genomic data. For training the PG-cGAN[9] we used exactly the same code given in their GitHub repository [8]. Their model includes one generator and discriminator with Wasserstein's loss function and RMSprop as the optimizer. For fair comparison, we trained both PG-cGAN and our models for the same number of epochs with the same biased dataset.

2.4 Study Design

After creating strongly biased data subsampled from the HLA data, we used these biased datasets to train Offspring GAN, Privacy-Preserving Offspring GAN (PPO-GAN), and PG-cGAN. This implies that none of the models ever gets to see around 80% of East Asian (EAS), African (AFR), and South Asian (SAS) samples. After generating synthetic data from all 3 GANs, we compare those synthetic data with the whole unbiased HLA data to evaluate the capability of each model to handle data biases. For comparison, we used the evaluation metrics discussed in the next section.

2.5 Evaluation

While developing a data evaluation metric for GANs, our evaluation metric should indicate both

1) Data Fidelity : the quality of each sample generated. We want the generated data to look realistic; and

2) Data Diversity : the distribution of generated data should capture the diversity of the real data. We want the generated data set to have all possible realistic data[22].

Some of the most widely accepted metrics to evaluate the synthetic data generated in computer vision are Inception score[18] and Maurie Fréchet distance or Fréchet inception distance (FID)[18] for multiple desired features. For both of them, a pre-trained classifier is needed to classify the generated data to check if it looks like real data. In computer vision, we have lots of commonly used pre-trained image classification models such as VGG[41] and ResNET [16] trained on large datasets like ImageNet[11]; however, that is not available in the realm of genomic data. To address the incoherence among the desired features which can not be captured by FID or IS, another metric was developed termed as HYPE [51], which uses human perception to classify images instead of using pre-trained model classification. As our psychophysics can not distinguish heat-map plots of genomic data, HYPE can not be applied in our case either. Rather, we use the following metrics to compare the similarity of the real and synthetic datasets.

2.5.1 Comparison of underlying Gaussian components . The goal of any GAN is to learn the underlying data distribution of real data. To compare the underlying data distribution of real and synthetic data, we fit a Gaussian Mixture Model (GMM) with two components (illustrated in **Fig. 4**). Our rationale for fitting two components is the existence of 2 major clusters in real data, one of AFR and another of EUR, EAS, SAS, while the data points of AMR are spread across both clusters.

In a hypothetical scenario of a perfect GAN, the GMM analysis should yield two underlying components in synthetic data distribution with the same means and weights as that of real data for each SNP. Weights of a Gaussian component correspond to the probability of a randomly sampled data point being explained by that Gaussian distribution.

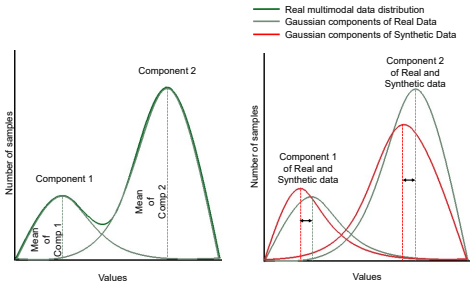


Figure 4: Pictorial representation of Comparison of Gaussian components in i^{th} dimension: At first, a Gaussian Mixture Model is used to find the underlying Gaussian components of 7160-dimensional data distribution and calculate means (for each dimension) and weights of those components or real data and synthetic data. Later, for both of the component differences between means in real and synthetic data distribution for each SNP was used to calculate total EMD.

As the Earth Mover's distance (EMD) corresponds to the total amount of shifting required for getting a data distribution from another, it is a perfect metric to calculate how far our synthetic data distribution is from that of real data. For each Gaussian components, we calculated the SNP-wise normalized EMD, the average normalized EMD and the total root mean square error (RMSE) between the real data distribution and the synthetic data distribution.

2.5.2 Minor allele frequency. In classic genomics literature, Minor Allele Frequency (MAF) has been a key metric to describe underlying population structure[2]. Allele frequency of different loci describes the underlying population structure caused by various evolutionary factors like reproductive isolation[2], selection pressure[32, 43], migration or gene flow[2, 43] and genetic drift. MAF for a population at a single SNP locus is simply a calculated frequency of the minor allele at the locus. We compare the MAFs of real and Offspring GAN generated synthetic data for all loci and plot the MAF distribution. In an ideal scenario, a GAN should be able to preserve the MAF distribution of the real data confirming a similar population structure.

2.5.3 Pearson correlation matrix. We plot the Pearson correlation matrix to compare underlying correlations between any two SNPs for both real and synthetic data. Correlation patterns of SNPs reveal underlying biological relationships between them in a large population. In an ideal scenario, the GAN-generated data distribution

should reproduce similar inter-SNP correlation yielding the same correlation patterns.

2.5.4 Principal component analysis . Principal component analysis (PCA) is a common dimension reduction tool for stratification and visualization of high-dimensional data. PCA uses linear feature combination, where the high dimensional data set is transformed into this new feature space called principal components. These principal components explain the most variation among samples. The first principal component explains the most variation and then the second, and so on. We visualize the data in the first three principal components for real and synthetic data from both offspring GAN and PG-cGAN . We also plotted PCA plots for synthetic data combined with real data in a single plot to visualize the overlap between synthetic and real data.

Although we train all our models with sub-sampled biased data, we compare our synthetic data with whole unbiased real data to check if the bias in training data is reinforced in the generated sample.

3 RESULTS AND DISCUSSIONS

3.1 Comparison of underlying Gaussian components

GMM model reveals the underlying two distinct 7160-dimensional Gaussian components. Synthetic Data generated by Offspring GAN and the privacy-preserving version of it (PPO-GAN) yields components with a very similar mean (averaged over all SNPs) and weights to that of real data. Comparison of mean and weights for real and synthetic data (summarized in **Table 2**) shows the ability of offspring GAN and PPO-GAN to capture the underlying real data distribution without being exposed to the whole unbiased dataset during training. On the other hand, PG-cGAN fails to handle biased data.

Model name	Comp 1 mean	Comp 1 weight	Comp 2 mean	Comp 2 weight
Real Data	0.999	0.29	0.897	0.71
Offspring GAN	0.988	0.29	0.878	0.71
PPO-GAN	0.980	0.28	0.875	0.73
PG-cGAN	0.741	0.19	0.628	0.80

Table 2: The Gaussian component average mean and weight comparison. Gaussian components with closer means are compared for real data and synthetic data from different GANs. Both in terms of average mean and weight of the component of Offspring GAN and PPO-GAN outperform PG-cGAN.

From the calculation of normalized Earth movers distance (EMD) (summarized in **Table 3**) and root mean square error(RMSE) (summarized in **Table 4**) in means of Gaussian components, we can see the same trend. Small normalized EMD and RMSE for offspring GAN suggest all SNP-wise means for both the component of generated data is close to that of real data. We also see that PPO-GAN reproduces results very close to Offspring GAN without sacrificing the privacy of real data for populations with lower representation in the biased dataset.

Model name	Comp 1 Norm. EMD	Comp 2 Norm EMD
Offspring GAN	0.064	0.042
PPO-GAN	0.065	0.043
PG-cGAN	0.095	0.093

Table 3: Comparison of normalized EMD from Real Data distribution. Average normalized EMD was calculated from the Gaussian components of real data to that of synthetic data for all 3 GAN models. The lower EMD is the better the model we have.

Model name	Comp 1 RMSE	Comp 2 RMSE
Offspring GAN	0.00099	0.00064
PPO-GAN	0.00100	0.00065
PG-cGAN	0.00135	0.00131

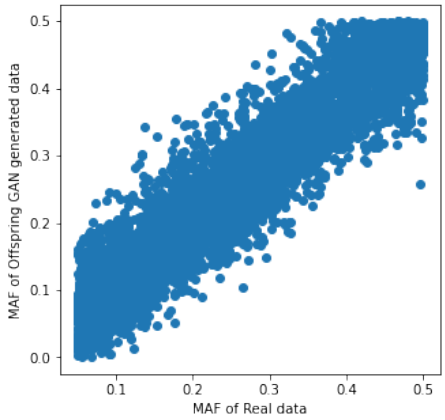
Table 4: Comparison of RMSE (root mean square error) from Real Data distribution. RMSE between Gaussian component of real and synthetic data was calculated for all 3 GAN models. The lower RMSE is the better the model we have.

3.2 Minor allele frequency

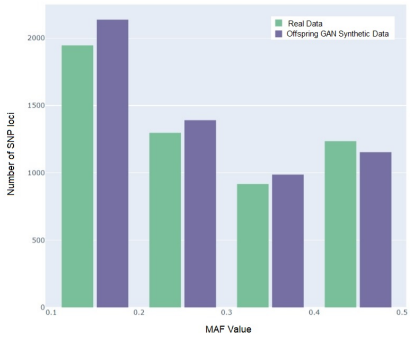
From the SNP loci-wise comparison of minor allele frequency (MAF) and MAF distribution plot, we can see synthetic data of offspring GAN reproduces the MAFs fairly well and encapture the MAF distribution nicely. Most of the data points in the scatter plot of MAF comparison of real and synthetic data fall into a tight range of $x = y$ line (see **Fig. 5**) ; this implies that for a given SNP, the MAF value of synthetic data is closer to that of the real data. Both real and synthetic dataset has a higher number of SNPs, where the alternate variant is rare (low MAF). Similar MAF in synthetic data suggests the ability of Offspring GAN to preserve the underlying population structure of real data.

3.3 Pearson correlation matrix

Heat map of Pearson correlation matrix reveals the inability of PG-cGAN to encapture underlying relationship between each SNPs, while Offspring GAN and PPO-GAN successfully encapture most of the large correlation patterns of real data(see **Fig. 6**). The large block of high correlation around the middle is seen from 3182^{th} to 4082^{th} biallelic SNP locus, offspring GAN, and PPO GAN encapture this large block while PG-cGAN does not. If we see along the secondary diagonal of the matrix, we can see even offspring GAN and PPO GAN fail to reproduce a few of the small blocks as well. The ability of our model to reproduce most of the correlation patterns suggests that synthetic data has similar inter-SNP relationships as real data, and the application of the Mendelian hybridizer did not perturb the inter-SNP correlations.



a



b

Figure 5: Minor Allele Frequency (MAF) comparison between real data and Offspring GAN generated synthetic data.(a) Scatter plot of MAFs in real and synthetic data, data points closer to $x = y$ diagonal represents a better match. (b) Histogram of MAF of real (in green) and synthetic (in violet) with a bin size of 0.1

3.4 Principal component analysis

We compared PCA plots of real and synthetic data generated by all 3 GANs. We also plotted PCA plots of real and synthetic combined data for visualization of the overlap between real and synthetic data for all 3 models. The HTML files for interactive visualization of the same plots are made available in our GitHub repository [10]. From the PCA plot of the real data, it's clear that European, East Asian, and South Asian samples form a large cluster and samples from the African population form another distinct cluster, while data points of the American population are spread across both the clusters. This same pattern is also preserved in the synthetic data of Offspring GAN as well as of PPO-GAN(see **Fig. 7**). Clustering of African Samples in a separate group can be explained by the "Out of Africa" dispersal models [28] and the spread of samples of the American population across both the cluster might be a result of its past migration of a diverse people into America. Offspring GAN preserves relative position and a good spread of data for all of the populations, but for the East Asian population, there is some heterogeneity in the spread of the population, which might be caused by a difference in mode weights in $n-$ dimensional histogram of feature space. However, Offspring GAN and PPO-GAN's PCA plots show that both have similar relative positions and spread of the cluster, while PG-cGAN fails to reproduce either of those two characteristics in its PCA plots.

4 CONCLUSION

In summary, we have developed a new model called Offspring GAN that is capable of generating realistic data while trained on biased data. Offspring GAN accomplishes this by using a hinge loss and including two Mendelian hybridizers in the model to increase its robustness to biased data in terms of producing higher data fidelity and resolving mode collapse issue. Moreover, Offspring GAN utilizes weight constraints to the discriminator that help with regularizing the discriminator to achieve robustness in the model. Our experiments on the 1000 Genomes Project genotype data shows the capability of offspring GAN for data augmentation in a heavily biased dataset. Novel techniques employed in the Offspring GAN can be applied to deal with biased genomic datasets to improve the efficiency of other deep learning based solutions.

As discussed earlier, to examine the potential of our model to produce balanced synthetic data without inheriting any archetype of bias from the training data, we created a biased training dataset from an unbiased dataset of HLA data from the 1000 genome project using random sub-sampling. When comparing synthetic data, we compared it against the unbiased dataset. Unfortunately, this HLA dataset does not have disease phenotype data associated with those genotypes. In the future, we want to apply our methods to various genomic datasets where disease phenotypes are available. Currently, our model only takes ethnicity as a conditional parameter; in the future, we want the conditional parameter to encompass disease phenotype along with ethnicity and perform disease phenotype prediction tasks.

In the meanwhile, using PPO-GAN, we demonstrated that using only offspring genotype of real data is as good as real data in terms of training the GAN model, which opens a new path in privacy preservation of genomic data without sharing the gradients from training the real data or the real data itself. Although federated learning is a popular model for preserving user-level privacy of training data, in recent studies, researchers have shown the vulnerability of gradient sharing techniques in distributed learning systems. It is now possible to reconstruct the private data of one user from its shared gradients. We suggest another strategy for protecting the privacy of models or genomic data by using the offspring generation of real data instead of using real data for training a deep learning model.

Although we observe the privacy-preserving protocol of the Offspring GAN to yield fairly similar results without ever seeing the real data of the populations with lower representation in biased HLA dataset, we have not tested it against existing privacy-oriented GAN models. In future, we plan to further explore privacy preserving GAN models to build a privacy-preserving distributed learning system that would be resilient to gradient leakage attacks. Currently, both of the Mendelian hybridizers are not involved during the training phase. We will use the second Mendelian hybridizer during training for a better regularization of the discriminator.

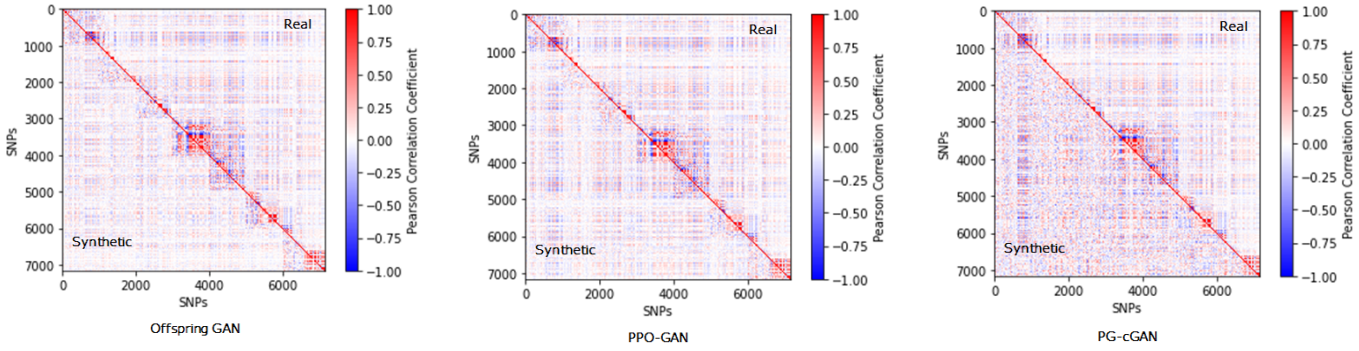


Figure 6: Heatmap of Pearson correlation coefficient: Upper half (upper triangular matrix) of each heatmap represents a correlation matrix of Real Data, and the lower half (lower triangular matrix) of that represents the correlation matrix of the synthetic data of the respective GAN model. In an ideal situation, the lower half should be a mirror image of the upper half. Dark red and blue regions depict a high positive and high negative correlation, respectively, while white areas represent no correlation between 2 biallelic SNP loci. (a) Comparison of Pearson correlation coefficient between Real and Offspring GAN generated data. (b) Comparison of Pearson correlation coefficient between Real and PPO-GAN generated data. (c) Comparison of Pearson correlation coefficient between Real and PG-cGAN generated data.

Although our model was developed using SNP genotype data at a large population level with 5 classes (East Asian, American, African, European, and South Asian), the same can be used in sub-population levels classified in counties. As we have demonstrated the capability of Offspring GAN to handle biased datasets using sub-sampled biased dataset and comparing it with the whole unbiased dataset, this can now be used in various gneomic datasets where the genomic data is already biased. For instance, we can apply Offspring GAN to cancer genomics data in the widely-used dataset such as The Cancer Genome Atlas (TCGA) [44] that are heavily biased towards populations with European ancestry, to generate unbiased datasets for model development and evaluation. In addition to genomic data, we plan to adapt Offspring GAN to augment other omics data including epigenomic and transcriptomic data.

While developing Offspring GAN, we introduced the novel idea of using offspring genotypes for training GANs. Mendelian hybridizer uses the principle of independent assortment to increase the sample size at a low computational cost as no training process is involved in this step. Hence this can be coupled with other deep learning architectures, such as CNN and transformer, to increase the sample size of training dataset.

Our offspring GAN model is capable of generating realistic data with good variation. These synthetic datasets have a much bigger sample size and are not biased to particular ancestry. Therefore, it can be used for training existing genotype imputation models to reduce the effect of bias in the training data. In the future, we also intend to examine the difference in training the same genotype imputation model with real and augmented genomic data, and investigate including synthetically generated data to enhance genotype imputation on unbalanced and biased data.

5 DATA AND CODE AVAILABILITY

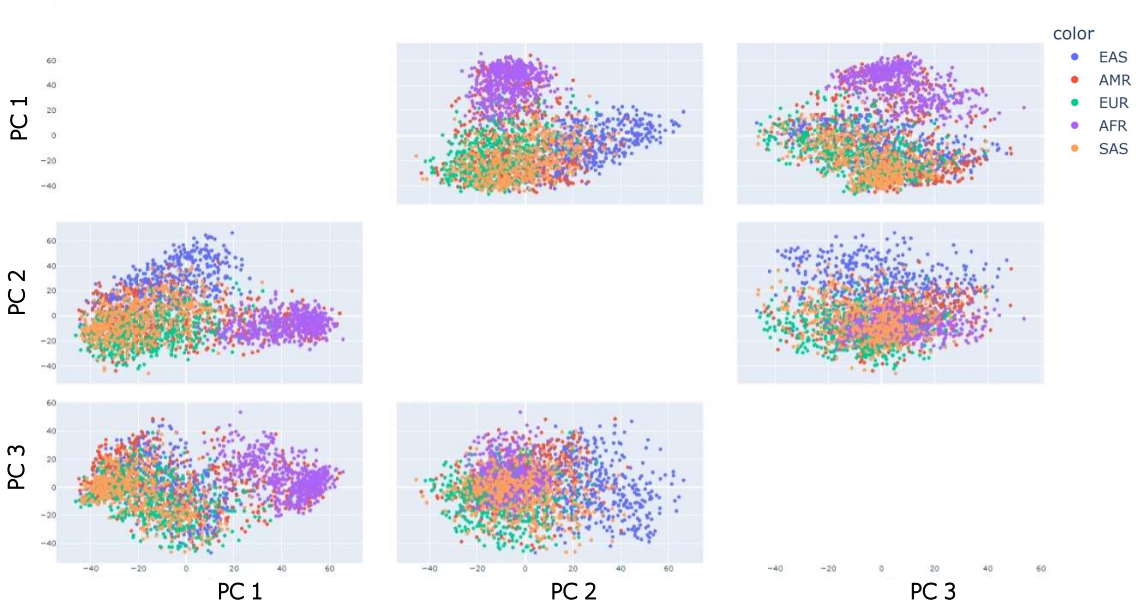
All interactive PCA plots, code for our models and training data can be accessed via our git hub repository [10].

ACKNOWLEDGMENTS

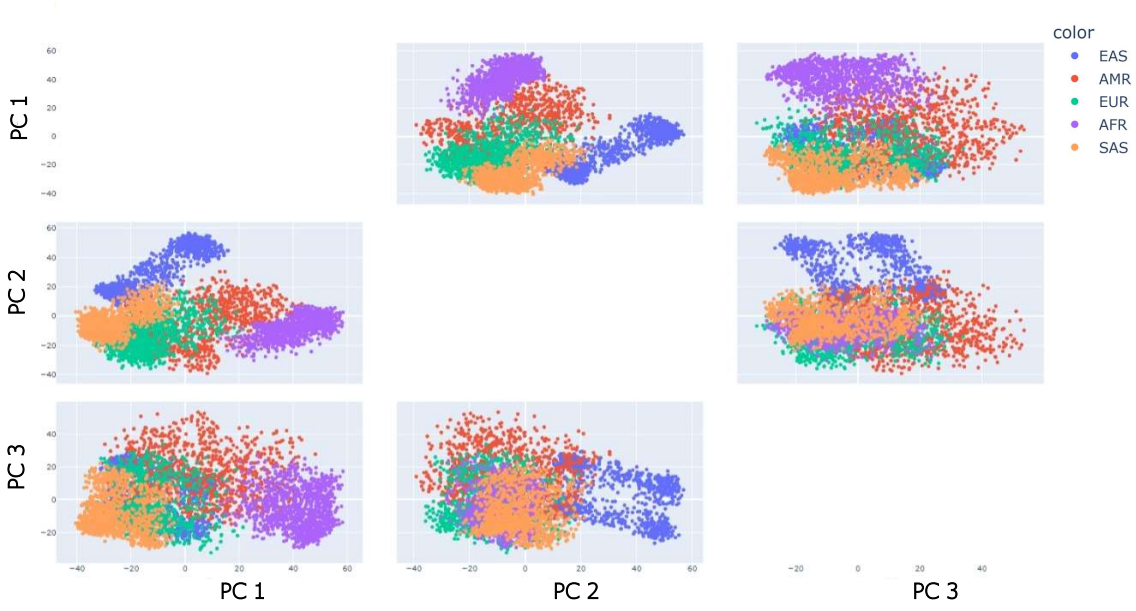
We would like to thank Erfan Mowlaei for his help with the manuscript. This work is partially supported by the National Science Foundation of the United States (Award Number: 1750632). Part of the work was performed on the high performance computation facility of IISER-Pune provided under the National Supercomputing Mission, Government of India.

REFERENCES

[1] 1000 Genomes Project Consortium. 2015. A global reference for human genetic variation. *Nature* 526, 7571 (2015), 68.
[2] Fred W. Allendorf and Stevan R. Phelps. 1981. Use of Allelic Frequencies to Describe Population Structure. *Canadian Journal of Fisheries and Aquatic Sciences* 38, 12 (1981), 1507–1514. https://doi.org/10.1139/f81-203 arXiv:https://doi.org/10.1139/f81-203
[3] Andrea Alter, Nguyen Thu Huong, Meenakshi Singh, Marianna Orlova, Nguyen Van Thuc, Kiran Katoch, Xiaojiang Gao, Vu Hong Thai, Nguyen Ngoc Ba, Mary Carrington, et al. 2011. Human leukocyte antigen class I region single-nucleotide polymorphisms are associated with leprosy susceptibility in Vietnam and India. *Journal of Infectious Diseases* 203, 9 (2011), 1274–1281.
[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International conference on machine learning*. PMLR, 214–223.
[5] David Berthelot, Thomas Schumm, and Luke Metz. 2017. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717* (2017).
[6] Chia-Che Chang, Chieh Hubert Lin, Che-Rung Lee, Da-Cheng Juan, Wei Wei, and Hwann-Tzong Chen. 2018. Escaping from collapsing modes in a constrained space. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 204–219.
[7] Tatjana Chavdarova and François Fleuret. 2018. Sgan: An alternative training of generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9407–9415.
[8] Junjie Chen, Mohammad Erfan Mowlaei, and Xinghua Shi. 2020. PG-cGAN. https://github.com/shilab/PG-cGAN.git.



a



b

Figure 7: PCA plots of real and synthetic data: These 6 sub-plots for each plot are plotted for all 6 combinations of first three principal components. All 5 populations East Asian (EAS), American (AMR), European (EUR), African (AFR), South Asian (SAS) are labeled in different colors. (a)PCA plots of real HLA genotypes (b)PCA plots of Offspring GAN generated genotypes.

[9] Junjie Chen, Mohammad Erfan Mowlaei, and Xinghua Shi. 2020. Population-scale Genomic Data Augmentation Based on Conditional Generative Adversarial Networks. In *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*. 1–6.

[10] Supratim Das and Xinghua Shi. 2022. Offspring GAN. https://github.com/shilab/Offspring-GAN.git.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

[12] Zhe Gan, Liqun Chen, Weiyao Wang, Yuchen Pu, Yizhe Zhang, Hao Liu, Chunyuan Li, and Lawrence Carin. 2017. Triangle Generative Adversarial Networks. In *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/bbeb0c1b1fd44e392c7ce2fdbd137e87-Paper.pdf

[13] Feng Gao, Wei Wang, Miaomiao Tan, Lina Zhu, Yuchen Zhang, Evelyn Fessler, Louis Vermeulen, and Xin Wang. 2019. DeepCC: a novel deep learning-based framework for cancer molecular subtype classification. *Oncogenesis* 8, 9 (2019), 1–12.

[14] Yan Gao and Yan Cui. 2020. Deep transfer learning for reducing health care disparities arising from biomedical data inequality. *Nature communications* 11, 1 (2020), 1–8.

[15] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. 2020. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in Neural Information Processing Systems* 33 (2020), 16937–16947.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[17] Daniel Ho, William Schierding, Sophie L Farrow, Antony A Cooper, Andreas W Kempa-Liehr, and Justin M O'Sullivan. 2021. Machine Learning Identifies Six Genetic Variants and Alterations in the Heart Atrial Appendage as Key Contributors to PD Risk Predictivity. *Frontiers in genetics* 12 (2021), 785436–785436.

[18] Yongjun Hong, Uiwon Hwang, Jaeyoon Yoo, and Sungroh Yoon. 2019. How Generative Adversarial Networks and Their Variants Work: An Overview. *ACM Comput. Surv.* 52, 1, Article 10 (feb 2019), 43 pages. https://doi.org/10.1145/3301282

[19] Taeho Jo, Kwangsik Nho, Paula Bice, and Andrew J Saykin. 2021. Deep learning-based identification of genetic variants: Application to Alzheimer's disease classification. *medRxiv* (2021).

[20] Ilya Kavalerov, Wojciech Czaja, and Rama Chellappa. 2021. A multi-class hinge loss for conditional gans. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 1290–1299.

[21] Aneta Kowal, Andrzej Wiśniewski, Piotr Kuśnierczyk, and Renata Jankowska. 2015. Human leukocyte antigen (HLA)-G gene polymorphism in patients with non-small cell lung cancer. *Thoracic cancer* 6, 5 (2015), 613–619.

[22] Jakub Langr and Vladimir Bok. 2019. *Gans in action: Deep learning with generative adversarial networks*. Manning.

[23] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 105–114. https://doi.org/10.1109/CVPR.2017.19

[24] Chongxuan LI, Taufik Xu, Jun Zhu, and Bo Zhang. 2017. Triple Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/86e78499eeb33fb9cac16b7555b50767-Paper.pdf

[25] Yi Liu, Guangchang Deng, Xiangping Zeng, Si Wu, Zhiwen Yu, and Hau-San Wong. 2020. Regularizing discriminative capability of CGANs for semi-supervised generative learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5720–5729.

[26] Lingjuan Lyu, Han Yu, and Qiang Yang. 2020. Threats to federated learning: A survey. *arXiv preprint arXiv:2003.02133* (2020).

[27] Andrew L. Maas. 2013. Rectifier Nonlinearities Improve Neural Network Acoustic Models.

[28] Brian P McEvoy, Joseph E Powell, Michael E Goddard, and Peter M Visscher. 2011. Human population dispersal "Out of Africa" estimated from linkage disequilibrium and allele frequencies of SNPs. *Genome research* 21, 6 (2011), 821–829.

[29] Hannes De Meulemeester, Joachim Schreurs, Michaël Fanuel, Bart De Moor, and Johan AK Suykens. 2021. The Bures Metric for Generative Adversarial Networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 52–66.

[30] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. 2018. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957* (2018).

[31] Gonçalo Mordido, Haojin Yang, and Christoph Meinel. 2020. microbatchgan: Stimulating diversity with multi-adversarial discrimination. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 3061–3070.

[32] John Novembre and Anna Di Rienzo. 2009. Spatial patterns of variation due to natural selection in humans. *Nature Reviews Genetics* 10, 11 (2009), 745–755.

[33] Augustus Odena. 2016. Semi-supervised learning with generative adversarial networks. *arXiv preprint arXiv:1606.01583* (2016).

[34] Sung-Wook Park, Jun-Ho Huh, and Jong-Chan Kim. 2020. BEGAN v3: avoiding mode collapse in GANs using variational inference. *Electronics* 9, 4 (2020), 688.

[35] Andreas Puschmann, Christophe Verbeeck, Michael G Heckman, Alexandra I Soto-Ortolaza, Timothy Lynch, Barbara Jasinska-Myga, Grzegorz Opala, Anna Krygowska-Wajs, Maria Barcikowska, Ryan J Uitti, et al. 2011. Human leukocyte antigen variation and Parkinson's disease. *Parkinsonism & related disorders* 17, 5 (2011), 376–378.

[36] Alicia Sanchez-Mazas. 2020. A review of HLA allele and SNP associations with highly prevalent infectious diseases in human populations. *Swiss medical weekly* 150, 1516 (2020).

[37] Diana Clobeth Sarrazola, Alejandra Marcela Rodríguez, Martín Toro, Alejandra Vélez, Jorge García-Ramírez, María Victoria Lopera, Cristian M Alvarez, Vital Balthazar González, Juan Manuel Alfaro, and Nicolás Pineda-Trujillo. 2018. Classical HLA alleles tag SNP in families from Antioquia with type 1 diabetes mellitus. *Biomedica* 38, 3 (2018), 329–337.

[38] Divya Saxena and Jiannong Cao. 2020. Generative Adversarial Networks (GANs): Challenges, Solutions, and Future Directions. *CoRR* abs/2005.00065 (2020). arXiv:2005.00065 https://arxiv.org/abs/2005.00065

[39] Pooyan Sedigh, Rasoul Sadeghian, and Mehdi Tale Masouleh. 2019. Generating synthetic medical images by using GAN to improve CNN performance in skin cancer classification. In *2019 7th International Conference on Robotics and Mechatronics (ICRoM)*. IEEE, 497–502.

[40] Deepa Sheth and Maryellen L Giger. 2020. Artificial intelligence in the interpretation of breast cancer on MRI. *Journal of Magnetic Resonance Imaging* 51, 5 (2020), 1310–1324.

[41] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[42] Jan Stanczuk, Christian Etmann, Lisa Maria Kreusser, and Carola-Bibiane Schönlieb. 2021. Wasserstein GANs work because they fail (to approximate the Wasserstein distance). *arXiv preprint arXiv:2103.01678* (2021).

[43] Hao Sun, Zhaoqing Yang, Keqin Lin, Shuyuan Liu, Kai Huang, Xiuyun Wang, Jiayou Chu, and Xiaoqin Huang. 2015. The adaptive change of HLA-DRB1 allele frequencies caused by natural selection in a Mongolian population that migrated to the south of China. *PloS one* 10, 7 (2015), e0134334.

[44] The Cancer Genome Atlas. 2014. Sample Counts for TCGA Data. https://tcga-data.nci.nih.gov/datareports/sampleSummaryReport.htm.

[45] Zhanpeng Wang, Jiaping Wang, Michael Kourakos, Nhung Hoang, Hyong Hark Lee, Iain Mathieson, and Sara Mathieson. 2021. Automatic inference of demographic parameters using generative adversarial networks. *Molecular ecology resources* 21, 8 (2021), 2689–2705.

[46] Xiaoting Wen, Si Chen, Jing Li, Yuan Li, Liubing Li, Ziyan Wu, Hui Yuan, Xinping Tian, Fengchun Zhang, and Yongzhe Li. 2018. Association between genetic variants in the human leukocyte antigen-B/MICA and Takayasu arteritis in Chinese Han population. *International Journal of Rheumatic Diseases* 21, 1 (2018), 271–277.

[47] Si Wu, Guangchang Deng, Jichang Li, Rui Li, Zhiwen Yu, and Hau-San Wong. 2019. Enhancing TripleGAN for Semi-Supervised Conditional Instance Synthesis and Classification. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), 10083–10092.

[48] Burak Yelmen, Aurélien Decelle, Linda Ongaro, Davide Marnetto, Corentin Tallec, Francesco Montinaro, Cyril Furtlehner, Luca Pagani, and Flora Jay. 2021. Creating artificial human genomes using generative neural networks. *PLoS genetics* 17, 2 (2021), e1009303.

[49] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. 2020. idlg: Improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610* (2020).

[50] Jian Zhou, Christopher Y Park, Chandra L Theesfeld, Aaron K Wong, Yuan Yuan, Claudia Scheckel, John J Fak, Julien Funk, Kevin Yao, Yoko Tajima, et al. 2019. Whole-genome deep-learning analysis identifies contribution of noncoding mutations to autism risk. *Nature genetics* 51, 6 (2019), 973–980.

[51] Sharon Zhou, Mitchell L. Gordon, Ranjay Krishna, Austin Narcomey, Li Fei-Fei, and Michael S. Bernstein. 2019. *HYPE: A Benchmark for Human Eye Perceptual Evaluation of Generative Models*. Curran Associates Inc., Red Hook, NY, USA.

[52] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. *Advances in Neural Information Processing Systems* 32 (2019).