An Overview of Deep Generative Models in Functional and Evolutionary Genomics

Burak Yelmen, Flora Jay

ANNUAL REVIEWS

Annual Review of Biomedical Data Science

An Overview of Deep Generative Models in Functional and Evolutionary Genomics

Burak Yelmen[1,2] and Flora Jay[1]

[1]Laboratoire Interdisciplinaire des Sciences du Numérique, CNRS UMR 9015, INRIA, Université Paris-Saclay, Orsay, France; email: flora.jay@lisn.fr

[2]Institute of Genomics, University of Tartu, Tartu, Estonia

Annu. Rev. Biomed. Data Sci. 2023. 6:173–89

First published as a Review in Advance on May 3, 2023

The *Annual Review of Biomedical Data Science* is online at biodatasci.annualreviews.org

https://doi.org/10.1146/annurev-biodatasci-020722-115651

Keywords

deep generative models, variational autoencoder, generative adversarial network, functional genomics, evolutionary genetics

Abstract

Following the widespread use of deep learning for genomics, deep generative modeling is also becoming a viable methodology for the broad field. Deep generative models (DGMs) can learn the complex structure of genomic data and allow researchers to generate novel genomic instances that retain the real characteristics of the original dataset. Aside from data generation, DGMs can also be used for dimensionality reduction by mapping the data space to a latent space, as well as for prediction tasks via exploitation of this learned mapping or supervised/semi-supervised DGM designs. In this review, we briefly introduce generative modeling and two currently prevailing architectures, we present conceptual applications along with notable examples in functional and evolutionary genomics, and we provide our perspective on potential challenges and future directions.

1. INTRODUCTION

Hidden Markov model: a Markov model (stochastic model for Markovian systems, i.e., where future states only depend on the current state) with observable and unobservable (hidden) states

Logistic regression: a model where the probability of an event happening is linked to a linear combination of independent observations via the logit function

Supervised learning: learning from labeled datasets a mapping from the data to their label(s) (e.g., regression and classification)

Unsupervised learning: learning from unlabeled datasets, the data structure, and relevant patterns (e.g., clustering and dimension reduction)

Machine learning has a broad range of applications, from research to industry and commerce. In the past few decades, rapid developments in the field have paved the way for breakthroughs in natural language processing, image recognition, robotics, biology, and many other domains (1). Generative modeling, as a subfield of machine learning, is similarly now widely researched and applied thanks to recent algorithmic and computational advances (2). In the broader statistical context, generative approaches model the statistical distribution of given data and can create new data instances following this distribution. They model the joint probability $P(\mathbf{X})$, where $\mathbf{X}$ is the observable variable or data instances, or $P(\mathbf{X}, \mathbf{Y})$, if the data has labels $\mathbf{Y}$. In some cases, generative models are only able to sample from the model distribution without providing its explicit estimation (3). On the other hand, discriminative approaches model the conditional probability $P(\mathbf{Y}|\mathbf{X})$, where $\mathbf{Y}$ is the target variable; in other words, they try to find the decision boundaries for specific labels in the data. Based on this terminology, a hidden Markov model (HMM) is generative, as it models the joint distribution of hidden states and observations for a Markovian process, and new data points can be sampled from the HMM distribution. In contrast, logistic regression is an example of a discriminative model (**Figure 1**). A second and straightforward definition of generative models would encompass any model that aims to generate partial or full data points (e.g., pixels in an image or a full image). Finally, a third definition focuses on the training scheme rather than the final task and includes any model for which the training loss function is based on the generation of the whole or parts of the data (4). Generative models falling in at least one of these three categories can address many tasks, such as data generation, density estimation, modeling, denoising and inpainting, compression, dimension reduction, and feature learning (5).

Genomics is the study of the genetic material of an organism in terms of function, structure, and evolution. Research in this field has revolutionized our understanding of cellular mechanisms and evolutionary processes, which has not only increased our collective knowledge but also fostered the discovery and development of novel drugs and treatments for diseases. Machine learning, and, in particular, deep learning, has become fundamental in genomics thanks to its ability to utilize big data and capture high-dimensional correlations and complex genomic structures (6–8). More recently, deep generative models (DGMs) have also been gaining research attraction in the broad genomics field, especially after the introduction of generative adversarial networks (GANs) (9). While the most common goal of DGMs is data synthesis, they can also be used for dimensionality reduction (and, relatedly, data characterization by visualization) or prediction. In this review, we

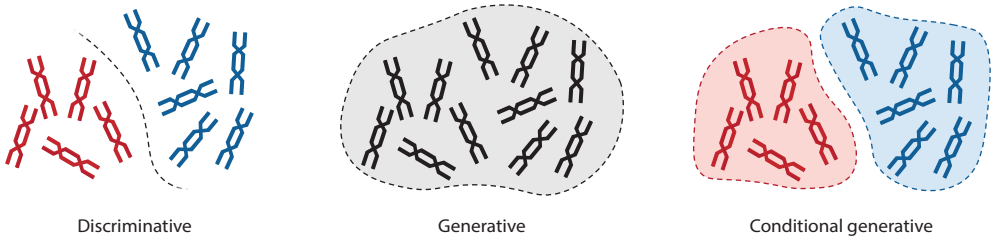

Figure 1

Discriminative and generative models: Discriminative approaches model decision boundaries for classification or regression tasks through supervised learning, whereas generative approaches model the data distribution, often through unsupervised learning. This distribution, even if not learned explicitly, can be sampled to generate new data instances. Generative models can also be conditioned on labels to generate data in a supervised manner. It is important to note that there is no strict dichotomy between these terms in practice; they are only presented here for explanatory purposes. In recent years, the term "generative" has started to include models that are generating data during training, regardless of their statistical modeling and final task (generative or discriminative).

first provide a brief technical summary of DGMs, followed by an overview of recent applications in genomics under three main utility themes: generation, dimension reduction, and prediction.

2. DEEP GENERATIVE MODELS

DGMs are a subset of generative models that use deep neural networks to approximate complex probability distributions of usually large training datasets. Since GANs and variational autoencoders (VAEs) are two of the most common DGMs for applications in genomics, we briefly introduce the fundamentals of both in this section.

2.1. Generative Adversarial Networks

GANs are part of the family of implicit density models, which do not estimate or approximate the data distribution but instead provide a direct way to sample from it (3). Although there are many variations, a GAN fundamentally consists of two neural networks: a generator (G) and a discriminator (D) (**Figure 2**). G takes a noise vector ($\mathbf{z}$) as input and generates a new sample $G(\mathbf{z})$ as output; in other words, G maps the data space to a latent space. The discriminator takes a sample ($\mathbf{x}$) as input and outputs a probability (or a score) $D(\mathbf{x})$ to assess whether $\mathbf{x}$ is sampled from the real dataset or generated by G. These two networks are trained in an adversarial manner: D is trained to maximize the probability of assigning the correct label, while G is trained to fool the discriminator by minimizing the probability of D assigning the fake label to $G(\mathbf{z})$. To put it another way, they compete in a zero-sum game until an equilibrium is reached where D cannot determine whether the output $G(\mathbf{z})$ is real or not. In a more technical definition, the basic loss function that G tries to minimize and D tries to maximize is as follows:

$$E_{\mathbf{x}}[\log D(\mathbf{x})] + E_{\mathbf{z}}[\log (1 - D(G(\mathbf{z})))],$$

where $E_{\mathbf{x}}$ is the expected value for all real data points and $E_{\mathbf{z}}$ is the expected value for generated data points. As for other deep neural networks, the loss is optimized through gradient descent. Aside from this loss function, proposed initially by Goodfellow et al. (9), many alterations and variations have been introduced. One commonly employed loss function is the Wasserstein loss used in the Wasserstein GAN (WGAN) model (10). In WGAN, instead of a discriminator, there is a critic (C), which no longer assigns the probability of real or fake to the input, but rather a score estimating the earth mover's (or Wasserstein) distance between the training and generated data. The new loss function, to be minimized by the generator G and maximized by the critic C, is as follows:

$$E_{\mathbf{x}}[C(\mathbf{x})] - E_{\mathbf{z}}[C(G(\mathbf{z})],$$

where C needs to be 1-Lipschitz continuous, which is achieved by clipping gradients (which means gradient values are clipped to a threshold before updating the weights during training) in the original WGAN study (mathematical proofs can be found in the original paper). Better approaches to achieve this constraint have since been proposed, such as using gradient penalty (GP), resulting in yet another commonly used GAN alteration called WGAN-GP (11). Overall, WGAN seems to be less prone to mode collapse (when samples are generated only for a subset of the data distribution), demonstrates less sensitivity to hyperparameter adjustments, and generally generates more realistic samples than the naive GAN model (10, 11).

2.2. Variational Autoencoders

Similarly to GANs, there are many variations of VAEs, but a simple VAE is a deep neural network with the same architectural basis as an autoencoder (AE), consisting of an encoder E and a

Deep learning: subset of machine learning involving neural networks with multiple layers that can learn hierarchical representations from data

Neural network: a computational architecture of connected nodes inspired by how biological neurons work

Latent space: a meaningful encoding of data in a lower dimensional space

Gradient: derivative of a multivariate function denoting the direction of greatest change at any point; also called slope

Gradient descent: for optimizing the neural network weights, an iterative algorithm for finding a local minimum of the network loss function by stepping in the gradient's opposite direction

Earth mover's (or Wasserstein) distance: a distance measure between two multidimensional distributions, which is the minimum cost of turning one distribution into another

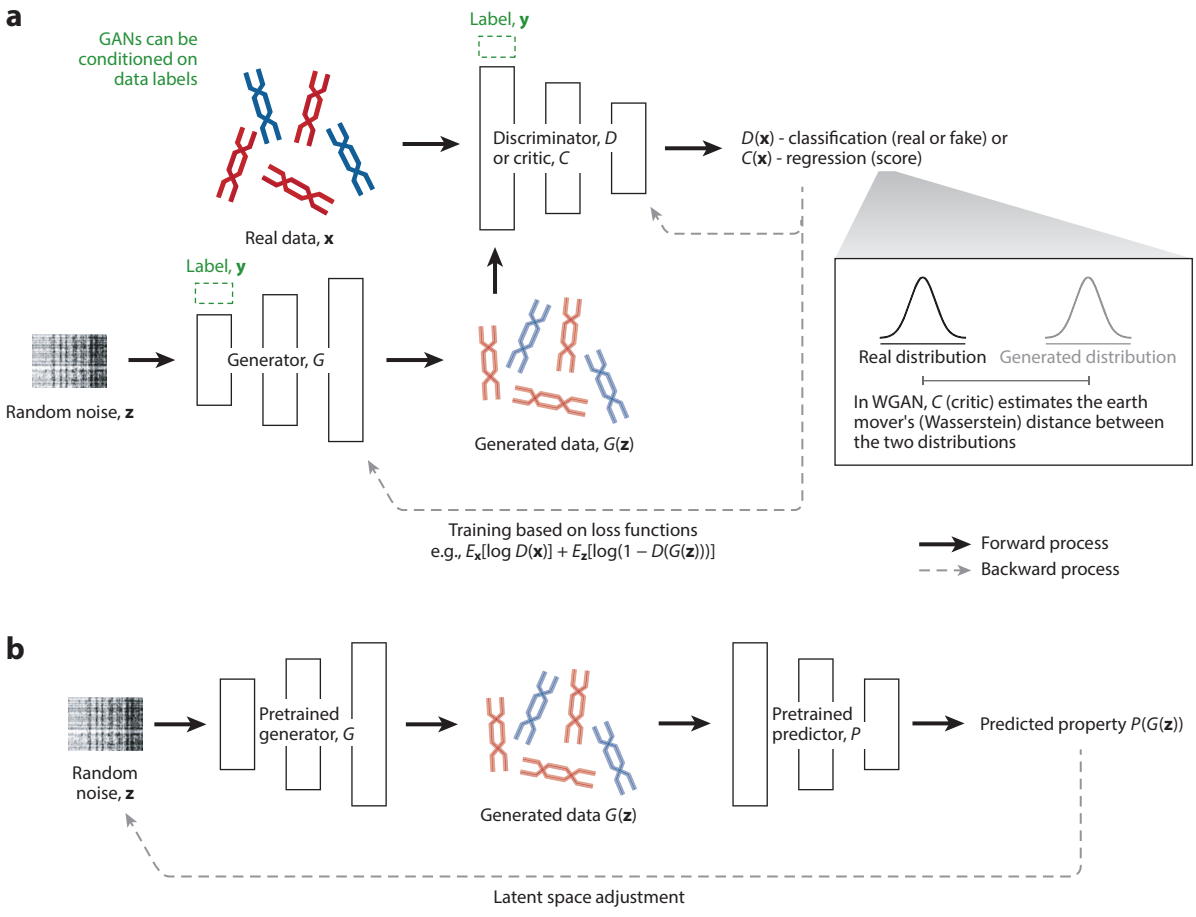

Figure 2

(*a*) GANs consist of a generator *G*, which generates new data instances, and a discriminator *D* or a critic *C*, which assesses the realness of the generated data. These two architectures are trained adversarially up to an equilibrium point where the discriminator cannot identify whether the generated data are real or fake. GANs can also be conditional, allowing for novel data to be generated with specified labels. (*b*) There are many modifications of the GAN concept to generate directed outputs similar to conditional GANs. One type of application that generates genomic sequences with desired properties, such as higher protein binding, uses the pretrained generator of a GAN model and a predictor that scores sequences for the desired property. The gradient of the score put out by the predictor $P(G(\mathbf{z}))$ with respect to the latent space $\mathbf{z}$ is calculated, and the latent space is adjusted based on the direction of this gradient, which guides the generated sequences toward the desired property with each adjustment step (13). Abbreviations: GAN, generative adversarial network; WGAN, Wasserstein GAN.

Hyperparameters: parameters that control a learning process and are not learned by training; they can be tuned through hyperparameter optimization

decoder *D* (**Figure 3**) (12). In a typical AE, *E* reduces the dimension of the input data $\mathbf{x}$ through a succession of layers leading to an embedding vector in the so-called latent space. *D* then decodes the embedding with the goal of reconstructing the input data as well as possible. Additionally, the VAE's goal is to ensure that the latent space is regular (organized in a desired way); consequently, small variations in the latent space will yield small variations in the decoded outputs. This is a valuable property for sampling meaningful embeddings directly from the latent space. For that, *E* encodes $\mathbf{x}$ as a distribution (a so-called latent distribution), generally a Gaussian characterized by its mean $\mu_{\mathbf{x}}$ and standard deviation $\sigma_{\mathbf{x}}$. Then a vector $\mathbf{z}$ is sampled from this distribution and decoded by *D*. The VAE loss function has two parts, a reconstruction loss between $\mathbf{x}$ and $D(\mathbf{z})$

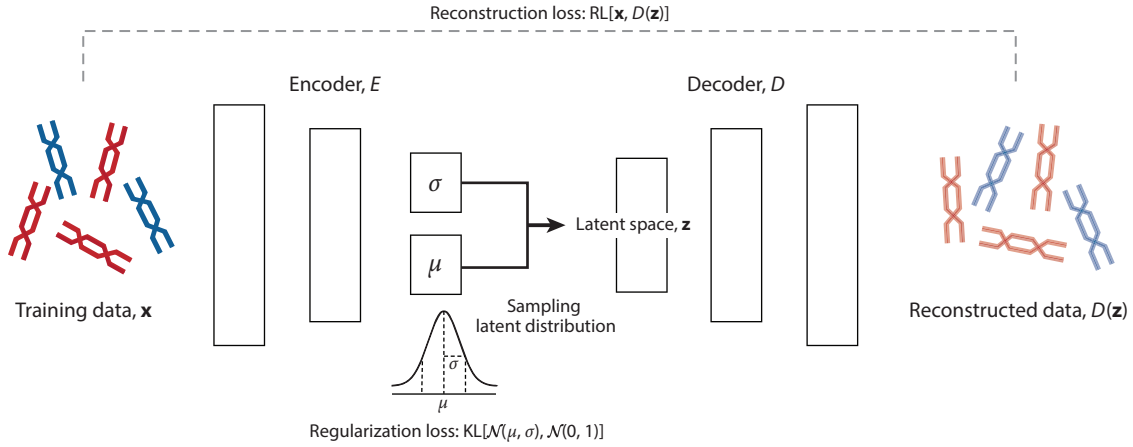

Figure 3

The variational autoencoder (VAE) architecture has two components: an encoder E, which encodes the training data into a parametric latent distribution, and a decoder D, which decodes a latent encoding $\mathbf{z}$ drawn from the latent distribution back to the original data space. Unlike conventional AEs, VAE latent space is regulated toward a known distribution. Therefore, the loss function used for training consists of a reconstruction term based on the difference between training and reconstructed data, and a regularization term based on the difference between the latent distribution and a target distribution (e.g., the standard normal distribution). After training, one can sample embeddings from the target distribution and decode them to generate novel data instances.

and a regularization term, which is an estimation of the distance between the latent and the prior distribution—most commonly between $\mathcal{N}(\mu_{\mathbf{x}}, \sigma_{\mathbf{x}})$ and the standard normal distribution $\mathcal{N}(0, 1)$:

$$\mathrm{RL}(\mathbf{x}, D(\mathbf{z})) + \mathrm{KL}[\mathcal{N}(\mu_{\mathbf{x}}, \sigma_{\mathbf{x}}) \,||\, \mathcal{N}(0, 1)],$$

where the first part is the reconstruction loss (such as cross-entropy for binary data or mean-squared error for Gaussian data), which measures the likelihood of the reconstructed data, and the second part is the Kullback–Leibler divergence, which measures the distance between two distributions. The regularization term is critical, as it allows the convergence of the latent space toward the standard normal distribution through training, which can then be used to sample new data instances.

2.3. Network Architectures

DGM architectures, including VAEs and GANs, consist of different types of neural networks, such as fully connected, convolutional and recurrent neural networks, or a combination of these (**Figure 4**). This architectural choice depends on the nature of the data, the task, and the available computational resources. Since in fully connected layers, all nodes in a given layer are connected to all nodes in the next layer, training will become memory intensive with larger input sizes, yet fully connected networks are adapted to the processing of data with an unknown structure. Initially designed for image data, convolutional layers are particularly suited for capturing local shift-invariant patterns that are then combined into features of higher complexity. In most cases, they are less parameter heavy than fully connected layers, as they share weights along the input. They have been widely applied in genomics (for an overview in functional genomics, see Reference 7, and in population genetics, see Reference 14). Alternatively, recurrent layers, traditionally applied in text and speech recognition, account for temporal or sequential dynamics and are thus pertinent for experimental time series data or omic sequences [e.g., bidirectional recurrent layers for a DNA sequence (15–17)]. Finally, graph neural networks are suited for non-Euclidian data with a graph

Kullback–Leibler divergence: a statistical nonsymmetric distance measuring how much one probability distribution differs from another probability distribution

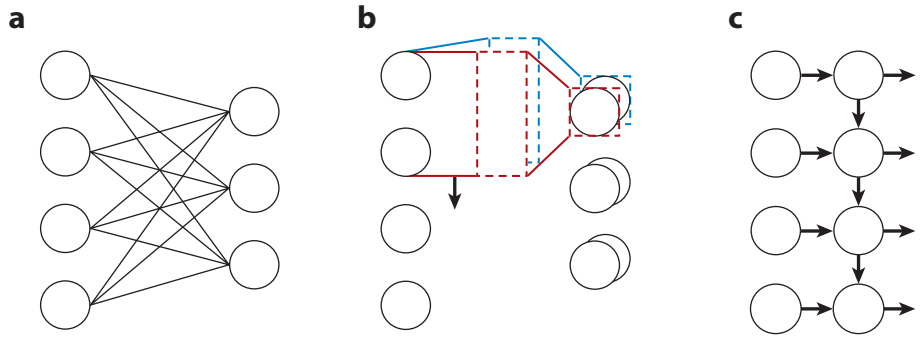

Figure 4

Types of neural networks. (*a*) In fully connected neural networks, each node in a given layer is connected to each node in the subsequent layer. In the genomics context, this full connectivity is useful for capturing any association in sequence data, whether it be short range, long range, or arbitrary correlation patterns. Yet, using this architecture for long sequences is not feasible due to the drastic increase in parameters to be learned with increase in input size. (*b*) Convolutional neural networks relay the information from one layer to the next through filters (or kernels) sliding along the input. These filters can capture spatial patterns, such as edges or shapes in image data. Deeper into the architecture (i.e., getting closer to the output), basic and local patterns are combined into more complex and global features. For genomics applications, this might be particularly advantageous for modeling local structures in genomic data, such as linkage disequilibrium patterns or sequence motifs. (*c*) Recurrent neural networks process a sequence of inputs and produce a sequence of outputs. They allow feedback connections where the information from the output of a previous position is used by subsequent inputs. This type of memory keeping is specifically utilized for temporal and sequential data types in which the inputs are not independent, such as DNA or RNA sequences.

structure. This makes them relevant for bioinformatics applications since biological networks, such as molecular structures, gene ontologies, regulatory pathways, or other biological systems, are ubiquitous in the field (18). Although discriminative neural networks have largely explored these architecture types, the vast majority of current DGM applications in genomics consist of fully connected and convolutional architectures.

3. THE GENERATION OF GENOMIC DATA

Linkage disequilibrium: nonrandom association of alleles at different loci

Genome-wide association study: the analysis of genotype–phenotype associations by looking at the allele frequencies of individuals with different phenotypes, correcting for differences in ancestry

As the cost of sequencing continues to decrease and new technologies are developed, the amount of genomic data increases immensely. With a cursory assessment, one might assume that the need for simulation of novel DNA sequences is nominal in this era, yet generative approaches are imperative for both functional and evolutionary genomics. For example, benchmarking data processing pipelines and inference methods related to next- and third-generation sequencing depend on simulated sequence data (19–22). In evolutionary biology and population genetics, coalescent and forward simulations of genetic variants among individuals have been fundamental for modeling evolutionary histories and estimating parameters related to demography and natural selection (23). Another approach for simulating genomic variants is the use of resampling methods, which mimic the characteristics of real haplotypes, such as linkage disequilibrium (LD) patterns. They are beneficial for simulating disease-associated variants and, consequently, for evaluating genome-wide association study (GWAS) methods and their statistical power (24, 25).

Although these more traditional DNA generation methods are still fundamental and relevant, they mostly require prior domain knowledge and simplified assumptions, and they fail to capture the full complexity of real sequences in most cases, which in turn limits their application to certain problems. Additionally, they either generate sequences that cannot be directly used along with real

sequences (as real and generated data exist in different spaces) or fail to generate enough diversity and overfit the real dataset (26). In this context, DGMs, as a new approach to sequence generation, can provide interesting and exciting solutions.

Overfitting: when a model models a particular dataset (such as the training data) too well and fails to generalize

3.1. Applications in Functional Genomics

One of the main objectives in synthetic biology and bioengineering is the design of functional sequences with desired structures and properties, such as binding affinity or gene expression levels—yet this typically requires extensive biological domain knowledge. The general approach for the design of novel regulatory sequences, for instance, mainly relies on invoking random mutagenesis or combinatorial approaches with known sequences prior to candidate selection through predictive modeling and eventually in vivo analysis (27–30). However, even an excellent selection model cannot counterbalance the difficulty of covering the vast sequence space via arbitrary and undirected changes or a combination of known sequences. In recent years, several DGMs have been proposed as potentially better alternatives for functional novel sequence design. Although they have architectural differences, they all rely on (*a*) GAN-like models for capturing the main structure of the target region and (*b*) a selective function for fine-tuning the desired properties. In theory, the selective function can be any type of function that either selects suitable candidates from the generated sequence pool or is integrated into the model to adjust the generated sequences toward desired properties during training. In one of the first applications of GAN models for the generation of novel DNA, Killoran et al. (13) combined the generator of a pretrained GAN, which creates realistic sequences, with a pretrained deep neural network predictor, which predicts the target characteristic for a given sequence (such as preferential binding to one specific protein). They trained this combined model by calculating the gradient of the output of the predictor with respect to the input noise of the generator. Following the direction of this gradient, the input noise was adjusted so that the outputs of the generator could converge to the desired properties (**Figure 2**). Instead of replacing the discriminator, Gupta & Zou (31) included a third component, called the analyzer, which can predict how desirable a sequence is (in terms of targeted antimicrobial properties, in this case). The original GAN and the analyzer were pretrained independently before being linked through a feedback loop: At each epoch, the generated sequences scored by the analyzer as most desirable were fed back to the discriminator as real examples, gradually replacing the training set of real genes and guiding the sequence generation toward the target. Similar generative models showed promising results for creating novel promoter regions, protein-binding motifs, protein-coding sequences, sequences with antimicrobial properties, and even whole regulatory structures (e.g., promoter, 5′ UTR, 3′ UTR, terminator) with desired expression levels (13, 30–34).

In a different application, a conditional GAN model was proposed to generate realistic single-cell RNA sequencing (scRNA-seq) data for different cell types (35). Since the availability of scRNA-seq data is limited due to costs and ethical reasons, it was suggested that the real data augmented with the generated data could improve downstream analyses such as distinguishing different cell populations.

3.2. Applications in Evolutionary Biology and Population Genetics

In population genetics and GWAS, biobanks with thousands of samples belonging to different populations play a vital role for both evolutionary research and discovery of genetic variant–disease associations. Although there are some publicly available databases for human genomic data, such as the 1000 Genomes Project, the Human Genome Diversity Project, and the HapMap Project (36–38), most of these data are not readily available to researchers. In addition, many populations are heavily underrepresented in such studies (39, 40). The generation of novel genomic data with

the same statistical properties as the real databases could increase data accessibility immensely and accelerate research without breaching the privacy of biobank donors. In this context, GANs, VAEs, and their derivatives have recently been suggested for generating realistic human genome segments (26, 41–45). These models have learned not only the global population stratification in real datasets but also complex underlying structures, such as LD patterns along the genome, haplotype-based selection signals, and genomic local ancestry proportions; this indicates that they might be used as reliable second-best alternatives for real genomes in the future (26, 41). Furthermore, they can be conditioned on extra variables, such as population labels, to generate targeted genomes depending on the task (41, 42). Finally, it was shown that the generated genomes could be good at preventing privacy leakage from genome donors in the training datasets, yet extensive research in this regard is still needed for further confirmation and improvements before these models can be applied in practical cases (26).

4. DIMENSIONALITY REDUCTION AND VISUALIZATION

Since omics data are often high dimensional, dimensionality reduction techniques have been important tools for initial screening and characterization of datasets in a wide range of omics studies. These techniques are commonly used for investigating the spatial genetic variation and demographic history in evolutionary studies or for characterizing the differences among cell types (46, 47). Both linear methods, such as principal component analysis (PCA) (48, 49), and nonlinear methods, such as t-distributed stochastic neighbor embedding (t-SNE) (50) or uniform manifold approximation and projection (UMAP) (51), are used for projecting the high-dimensional data space into a smaller feature space in the hope of capturing the global and local structures in a few dimensions that can be easily visualized. Dimensionality reduction methods can also be helpful for further downstream analyses, as they reduce data size and complexity. Moreover, they can be applied to many data types without prior knowledge. However, the above techniques have certain drawbacks. PCA cannot capture nonlinear relations and is sensitive to outliers, such as rare genetic variations, causing principal component axes to separate based on the rare variations rather than real clusters (52). t-SNE and UMAP can capture nonlinear relationships and the underlying local data structure with adequate cluster separation, yet the distances between clusters obtained with these methods might not be meaningful—in other words, relative distances between clusters in the projection space might not correspond to the intrinsic differences between real data clusters (53).

In more recent years, deep neural networks, such as AEs (which are not generative models) and VAEs, have gained research interest for learning the compressed embeddings of genomic data and integration of multiomics data (54–60). These dimensionality reduction techniques can be applied to various data types, such as gene expression or SNP (single-nucleotide polymorphism) data. Since the VAE loss function consists of not only the reconstruction loss but also the regularization of the latent space, the relative positions in the embeddings are expected to be more meaningful, with a better representation of global data structure.

4.1. Applications in Functional Genomics

DGM-based dimensionality reduction was applied to transcriptomic data for probabilistic modeling of gene expression, at both tissue (RNA sequencing) and single-cell (scRNA-seq) resolution (61–68). The latent space learned by VAE and GAN derivatives enables clustering and the classification of different cell types, through either 2D and 3D projections of the embeddings or further downstream analyses. One approach commonly undertaken for clustering is to perform t-SNE on the latent space. Alternatively, the architecture of DGMs is sometimes modified to enhance

interpretability, for example, by encouraging a correspondence between cell and gene embeddings (56) or by using gene annotations to guide the network connections (59). In both cases, the alterations have helped link input expression profiles and functionality.

Moving away from transcriptomics, in a noteworthy application to chromatin accessibility, Kshirsagar et al. (69) trained a Dirichlet VAE to learn latent representations of DNA *k*-mers. Because the network targeted a Dirichlet latent distribution instead of a traditional Gaussian, each open chromatin region could be represented by its membership to multiple topics (corresponding to the latent dimensions). Topics were represented as a multinomial distribution over *k*-mers and learned different binding patterns. A post hoc interpretation procedure mapped transcription factors to the VAE latent dimensions, which in turn helped to interpret the regulatory information available in chromatin accessibility peaks.

Another interesting aspect of DGM architectures is that they can be used for integrating multiple data types. In one study, Simidjievski et al. (57) investigated different VAE models trained with multiomics and clinical data and demonstrated that the latent representations learned by these integrated VAE models could be exploited to predict cancer-related parameters such as cancer subtypes and disease relapse. Similarly, VAE models have been used to integrate multiomics data for studying drug–omics associations via in silico perturbations (70).

4.2. Applications in Evolutionary Biology and Population Genetics

VAE and AE models can also capture the fine population structure present in SNP data and underline the global structure better than other dimensionality reduction methods (54, 55, 71). These studies trained convolutional and fully connected models on SNP data belonging to real samples from multiple populations or simulated samples with known demographic histories. Similar to principal components in PCA, embeddings of the latent space in these models seem to represent the genetic differentiation between genomes. This representative information is valuable for population genetics studies, as the differentiation is shaped by the species migration history (such as waves of human migration within Africa and out of Africa toward Eurasia, Oceania, and the Americas) and numerous subsequent admixture events between populations. Although not belonging to a deep architecture, the components of a restricted Boltzmann machine hidden layer have also been shown to capture fine-scale human population structure (26).

5. PREDICTION

The main utility of generative models is in learning the data distribution in an unsupervised manner; hence their use for direct predictive modeling is limited. However, in a supervised setting, they can learn conditionally on a label, $P(\mathbf{X}|\mathbf{Y})$. This differs from learning directly what in the data is informative of the label, $P(\mathbf{Y}|\mathbf{X})$, but it can still be used to perform predictions. For example, in a Naive Bayes classifier, the membership of a new point is assessed based on the learned distributions within each class. This section briefly discusses some notable predictive applications that rely on generative models in genomics-related studies.

5.1. Applications in Functional Genomics

First, it is noteworthy that predictive tasks can exploit unsupervised dimensionality reduction methods. Indeed, they yield meaningful data representations encompassing information relevant to target variables contributing to the data structure, even though the encoding has not been optimized for these targets (as illustrated in Reference 57, where multiomic encodings were used for cancer-related predictions). Any downstream predictive approach could benefit from these

compact representations, particularly those sensitive to input size. Some DGM dimensionality reduction methods have been used, through latent space vector arithmetic or alterations to classical VAE structure, to predict the cellular response (in terms of gene expression) to perturbations such as infection, treatment, or knockout of genes (72, 73). Vector arithmetic applied on latent representations can produce meaningful outputs for manipulating semantic properties underlying image data (illustrated by the famous ["man with glasses" − "man" + "woman"] operation in the latent space leading to a latent vector corresponding to an image of a "woman with glasses") (74). Similarly, vectors obtained by subtracting latent representations of gene expression profiles of different cell types have been shown to correspond to biologically meaningful differences and applied to simulate the impact of epidermal cell differentiation, interferon stimulation, *Salmonella* infection, cancer therapeutics, and other drug treatments (68, 72, 73). In a different application, vector arithmetics were used to interpolate between the latent vectors of healthy and Alzheimer's disease expression profiles generated by a GAN model (75). The interpolation was used to obtain transition curves for multiple genes demonstrating changes from healthy to disease types. This type of approach could present novel ways for inferring pathological cascades and disease progressions that would not be possible with conventional bioinformatics methodology.

5.2. Applications in Evolutionary Biology and Population Genetics

At the crossroads of functional studies and population genetics, DGMs were used to predict disease outcomes or identify risk variants in a context of insufficient data labeling. In one study, Davi & Braga-Neto (76) modified the GAN model with a discriminator that classifies not only between real and fake data but also between two phenotypes (severe or normal dengue fever). The model was trained in a semi-supervised setting on phenotype-labeled and unlabeled SNP data, and the discriminator served as a phenotype predictor after training. In another study, Frazer et al. (77) modeled the variation among amino acid sequences across multiple species using a VAE, which in return allowed them to assess sequence fitness and consequently predict possible disease variants. Although this model targeted amino acid sequences, a similar framework could be adapted to genomic data.

As a different application in population genetics, a study used a GAN-like model to infer demographic parameters from SNP data (78). Instead of a neural network, a coalescent simulator, msprime (79), was integrated as a nondifferentiable generator taking evolutionary parameters as input to generate SNP data for a pool of individuals. The discriminator indirectly assessed the plausibility of the parameters by assessing the realism of the generated data. Because of its nondifferentiability, the generator was trained using simulated annealing instead of backpropagation. Eventually, the properties of its simulations converged toward the properties of the real data, and its parameters toward the putative real evolutionary parameters.

5.3. Applications in Data Processing

DGMs have been investigated in a few studies to improve variant calling, which is the process of identifying variants from sequencing data. A recent study utilized a GAN to boost the performance of genome variant calling on low-depth data (80). In particular, generative and adversarial training was used to convert low-depth data (an image computed from the aligned reads and their quality measurements) to a high-depth equivalent. The variant calling algorithm was then applied to pairs of low-depth original and high-depth generated images. Additionally, for improving variant calling, DeepConsensus (81) implements a gap-aware, encoder-only transformer applied to multiple sequence alignment (MSA) windows in order to generate the consensus sequence. Notably, both studies used not only the nucleotide sequences but also auxiliary information such as read quality or base caller features (e.g., pulse width).

Secondly, DGMs can be used for data imputation, which is simply a partial generation of a data subset that is missing $\mathbf{X}_{\text{missing}}$ conditional on the subset that is known $\mathbf{X}_{\text{known}}$. Autoencoders, and specifically denoising autoencoders, are well suited for imputation tasks and have been applied to genomics (82–84). In a similar spirit, in one recent study, a VAE was implemented to perform transcriptome and methylation imputation (85). The authors used an iterative process that first randomly filled $\mathbf{X}_{\text{missing}}$ and then iteratively encoded and reconstructed $\mathbf{X}$. At each iteration until convergence, it updated $\mathbf{X}_{\text{missing}}$ with the reconstructed values. The variational setting allowed the latent distribution to be amended by integrating a shift correction, which is useful when a gap exists between the training dataset and the target data (e.g., due to data not missing at random).

Finally, language models (LMs) processing DNA data have very recently emerged and their training integrates a concept close to imputation. An LM models a language domain as a probability distribution over sequences of words. It can be learned with the help of machine learning, and recent LMs have leveraged deep neural networks. In particular, two frameworks named BERT (bidirectional encoder representations from transformers) (86) and GPT (generative pretrained transformer) (87) have revolutionized the natural language processing (NLP) field by providing expressive pretrained LMs. Although computationally intensive to train in the first place, they could conveniently be further fine-tuned for specific tasks (such as question answering, translation, or text classification). Shortly after its introduction, multiple DNA LMs inspired by BERT were proposed (88–93). Their common idea is the use of masked LMs, in which a portion of the input *k*-mer or nucleotide tokens is randomly masked and the model is trained to solve the pretext task of predicting those masked tokens (similar to denoising autoencoders). Thanks to this self-supervised pretraining, the model learned the underlying DNA language without requiring annotated data. In genomics, these pretrained models can then be fine-tuned for any downstream task such as predicting promoters, transcription factor binding sites, and splice sites or inferring disease mechanisms and genotype–phenotype associations.

Differential privacy: a definition of privacy where a dataset can be statistically analyzed while each of its individuals is protected

Federated learning: a machine learning approach for training algorithms over multiple servers without data exchange

6. CONCLUSIONS

With the advent of novel algorithms and increased computational capacities, deep generative modeling is now finding its way into broad genomics research. The ability to model complex data distributions without any prior knowledge required makes these models ideal for various applications with omics and medical data. An important opportunity for DGMs lies in the field of data privacy. Human genomic data are inherently very sensitive, as they encapsulate partial information on phenotypic traits, disease susceptibility, and ancestry (94, 95). Moreover, genomic data constitute a unique identifier that, if leaked, cannot be replaced. Access to human genomic data is often restricted as a result. Several privacy-preserving methods have been proposed to overcome this issue, such as encryption (96, 97), differential privacy (98), federated learning (99, 100), or a combination of those (101). In differential privacy, some amount of noise is added to the data input or the predicted output for anonymization, whereas in federated learning, algorithm training is performed without direct access to the raw data. Data synthesis via DGMs can be an alternative to these approaches that has certain advantages, such as unrestricted analysis, unlike federated learning, and potentially less distorted data compared to differential privacy (since differential privacy essentially presents a trade-off between capturing the intrinsic characteristics of the data and privacy preservation), yet extensive comparative research in this regard is still lacking. These approaches are not necessarily mutually exclusive. For instance, differential privacy can be integrated into GAN training by adding carefully adjusted noise to the gradients to reduce privacy leakage from the training data (102). It is important to mention here that donor privacy is only one aspect of the ethics of genomic and medical research. Even if privacy guarantees are provided, studies

exploiting privacy-preserving methods might still need to respect the ethical regulations designed by the original data holders/donors. In this regard, further ethical and philosophical discussion could provide useful insights especially considering the relatively novel status of DGMs.

Transfer learning: the utilization of the knowledge learned while solving a task (e.g., a pretrained network) to address a separate but related problem

Another potentially transformational aspect of DGMs is functional sequence design. As presented in Section 3.1, GAN-based models have been extensively used in recent years to generate novel sequences with desired properties and have yielded more diverse and better outcomes than more conventional sequence design methodologies. The design of highly specific biological DNA and protein sequences is one of the holy grails of synthetic biology, as it could advance drug discovery, precision medicine, and biomanufacturing significantly. In this context, DGMs are becoming critical tools by providing a substantial shift in the methodological approach to this problem.

There is also potential for advanced generative models to be employed in genomic data simulations. DGMs can both produce realistic data with minimal privacy leakage and be altered for directed generation with desired characteristics. In addition, learned characteristics from a dataset via a model could be transferred to another dataset (style transfer), as suggested by Booker et al. (45). All these factors make DGMs suitable for the generation of adjustable simulated data with known ground-truth parameters, which is essential for the development of new bionformatics methods. Furthermore, the same factors also allow DGM-generated sequence data to be used for data augmentation, especially considering that certain genomic data types are not easily accessible (due to biobank restrictions) or obtainable (due to ethical issues or costs related to sampling) (26, 35).

From a wider perspective, a major advantage of DGMs is their unsupervised or semi-supervised training, making them especially suitable for genomic data, which are abundant quantitatively but in most cases lack adequate labels (such as phenotype information or annotations). Capitalizing on extensive unlabeled or mixed datasets has been key in recent progress in computer vision and NLP research (4) and should likewise allow for modeling of complex structures and interactions present in different genomic data types. In particular, LMs can capture this underlying complexity using large unlabeled sequence databases and be fine-tuned on smaller annotated datasets for various downstream analyses, such as regulatory sequence prediction, in a manner similar to transfer learning. An additional important characteristic of most DGMs is the mapping of data space to latent space. Through directed manipulation or interpolation of the latent space vectors, sequences with novel characteristics can be obtained. This unique aspect allows DGMs to be used for sequence design and for providing innovative ways of understanding the genetic foundations of various diseases and drug responses.

DGMs are being utilized for various genomics applications, such as the characterization of population structure, cell clustering, phenotype and disease variant prediction, evolutionary parameter estimation, and imputation, as described in this review. Despite their promising results, DGMs suffer from general pitfalls hampering deep learning (103), as well as other specific issues that remain to be addressed for their broader use in genomics. One obstacle is the computational limitations associated with whole-genome generation. Even with the help of high-capacity GPUs (graphics processing units) and adjusted architectural designs, training models with large sequences of millions of base pairs is impractical with current approaches. In addition, GAN models are especially difficult to train due to the adversarial nature of the training and hard-to-reach equilibrium points. Although several improvements have been proposed (10, 11, 104), training on large data instances, in particular, is still problematic given the long training times and high dependency on hyperparameter tuning (105, 106). Another general issue with DGMs is the black-box nature of most models. Interpretability of learned features is widely researched for deep neural networks (5, 107, 108). Although a few studies tackle interpretability for DGMs, research in a biological context is still limited (56, 109).

Despite these points that remain to be further researched, it has already been demonstrated that deep generative modeling in genomics is a robust and efficient methodology. This might be seen as the initial step toward a broad new field, artificial genomics, which can be defined as the use of artificial intelligence for in silico genomic data generation. Unlike traditional rule-based approaches or domain-specific simulators, artificial genomics can allow researchers to capture high-degree complexities in genomic data in order to design novel sequences with no or little prior knowledge.

DISCLOSURE STATEMENT

The authors are not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

We gratefully acknowledge RoDAPoG (Robust Deep Learning for Artificial Genomics and Population Genetics) for funding (grant ANR-20-CE45-0010-01); Susana Ribeiro, Gabriel Synnaeve, and Blaise Hanczar for their feedback and comments on the manuscript; and the TAU (TAckling the Underspecified) and Bioinfo groups at LISN (Laboratoire Interdisciplinaire des Sciences du Numérique), in particular, Michèle Sebag, Cyril Furtlehner, Aurélien Decelle, and Guillaume Charpiat, for fruitful discussions.

LITERATURE CITED

1. Jordan MI, Mitchell TM. 2015. Machine learning: trends, perspectives, and prospects. *Science* 349:255–60
2. Harshvardhan GM, Gourisaria MK, Pandey M, Rautaray SS. 2020. A comprehensive survey and analysis of generative models in machine learning. *Comput. Sci. Rev.* 38:100285
3. Goodfellow I. 2016. NIPS 2016 tutorial: generative adversarial networks. arXiv:1701.00160 [cs.LG]
4. Liu X, Zhang F, Hou Z, Mian L, Wang Z, et al. 2023. Self-supervised learning: generative or contrastive. *IEEE Trans. Knowl. Data Eng.* 35:857–76
5. Zhang Q, Wu YN, Zhu SC. 2018. Interpretable convolutional neural networks. arXiv:1710.00935 [cs.CV]
6. Eraslan G, Avsec Ž, Gagneur J, Theis FJ. 2019. Deep learning: new computational modelling techniques for genomics. *Nat. Rev. Genet.* 20:389–403
7. Routhier E, Mozziconacci J. 2022. Genomics enters the deep learning era. *PeerJ* 10:e13613
8. Shen X, Jiang C, Wen Y, Li C, Lu Q. 2022. A brief review on deep learning applications in genomic studies. *Front. Syst. Biol.* 2:877717
9. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, et al. 2014. Generative adversarial networks. arXiv:1406.2661 [stat.ML]
10. Arjovsky M, Chintala S, Bottou L. 2017. Wasserstein GAN. arXiv:1701.07875 [stat.ML]
11. Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville A. 2017. Improved training of Wasserstein GANs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ed. U von Luxburg, I Guyon, S Bengio, H Wallach, R Fergus, pp. 5769–79. Red Hook, NY: ACM
12. Kingma DP, Welling M. 2022. Auto-encoding variational Bayes. arXiv:1312.6114 [stat.ML]. **https://doi.org/10.48550/arXiv.1312.6114**
13. Killoran N, Lee LJ, Delong A, Duvenaud D, Frey BJ. 2017. Generating and designing DNA with deep generative models. arXiv:1712.06148 [cs.LG]
14. Korfmann K, Gaggiotti OE, Fumagalli M. 2023. Deep learning in population genetics. *Genome Biol. Evol.* 15(2):evad008
15. Quang D, Xie X. 2016. DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Res.* 44:e107

16. Whata A, Chimedza C. 2021. Deep learning for SARS COV-2 genome sequences. *IEEE Access* 9:59597–611
17. Adrion JR, Galloway JG, Kern AD. 2020. Predicting the landscape of recombination using deep learning. *Mol. Biol. Evol.* 37:1790–808
18. Zhang XM, Liang L, Liu L, Tang MJ. 2021. Graph neural networks and their current applications in bioinformatics. *Front. Genet.* 12:690049
19. Escalona M, Rocha S, Posada D. 2016. A comparison of tools for the simulation of genomic next-generation sequencing data. *Nat. Rev. Genet.* 17:459–69
20. Alosaimi S, Bandiang A, van Biljon N, Awany D, Thami PK, et al. 2020. A broad survey of DNA sequence data simulation tools. *Brief. Funct. Genom.* 19:49–59
21. Xiao T, Zhou W. 2020. The third generation sequencing: the advanced approach to genetic diseases. *Transl. Pediatr.* 9:163–73
22. Lotterhos KE, Fitzpatrick MC, Blackmon H. 2022. Simulation tests of methods in evolution, ecology, and systematics: pitfalls, progress, and principles. *Annu. Rev. Ecol. Evol. Syst.* 53:113–36
23. Yuan X, Miller DJ, Zhang J, Herrington D, Wang Y. 2012. An overview of population genetic data simulation. *J. Comput. Biol.* 19:42–54
24. Su Z, Marchini J, Donnelly P. 2011. HAPGEN2: simulation of multiple disease SNPs. *Bioinformatics* 27:2304–5
25. Wharrie S, Yang Z, Raj V, Monti R, Gupta R, et al. 2022. HAPNEST: efficient, large-scale generation and evaluation of synthetic datasets for genotypes and phenotypes. bioRxiv 2022.12.22.521552. **https://doi.org/10.1101/2022.12.22.521552**
26. Yelmen B, Decelle A, Ongaro L, Marnetto D, Tallec C, et al. 2021. Creating artificial human genomes using generative neural networks. *PLOS Genet.* 17:e1009303
27. Redden H, Alper HS. 2015. The development and characterization of synthetic minimal yeast promoters. *Nat. Commun.* 6:7810
28. Cai YM, Kallam K, Tidd H, Gendarini G, Salzman A, Patron N. 2020. Rational design of minimal synthetic promoters for plants. *Nucleic Acids Res.* 48:11845–56
29. Zrimec J, Börlin CS, Buric F, Muhammad AS, Chen R, et al. 2020. Deep learning suggests that gene expression is encoded in all parts of a co-evolving interacting gene regulatory structure. *Nat. Commun.* 11:6141
30. Zrimec J, Fu X, Muhammad AS, Skrekas C, Jauniskis V, et al. 2022. Controlling gene expression with deep generative design of regulatory DNA. *Nat. Commun.* 13:5099
31. Gupta A, Zou J. 2019. Feedback GAN for DNA optimizes protein functions. *Nat. Mach. Intell.* 1:105–11
32. Wang Y, Wang H, Wei L, Li S, Liu L, Wang X. 2020. Synthetic promoter design in *Escherichia coli* based on a deep generative network. *Nucleic Acids Res.* 48:6403–12
33. Linder J, Bogard N, Rosenberg AB, Seelig G. 2019. Deep exploration networks for rapid engineering of functional DNA sequences. bioRxiv 864363. **https://doi.org/10.1101/864363**
34. Hazra D, Kim MR, Byun YC. 2022. Generative adversarial networks for creating synthetic nucleic acid sequences of cat genome. *Int. J. Mol. Sci.* 23:3701
35. Marouf M, Machart P, Bansal V, Kilian C, Magruder DS, et al. 2020. Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. *Nat. Commun.* 11:166
36. Auton A, Abecasis GR, Altshuler DM, Durbin RM, Abecasis GR, et al. 2015. A global reference for human genetic variation. *Nature* 526:68–74
37. Cavalli-Sforza LL. 2005. The Human Genome Diversity Project: past, present and future. *Nat. Rev. Genet.* 6:333–40
38. Gibbs RA, Belmont JW, Hardenbol P, Willis TD, Yu F, et al. 2003. The International HapMap Project. *Nature* 426:789–96
39. Sirugo G, Williams SM, Tishkoff SA. 2019. The missing diversity in human genetic studies. *Cell* 177:26–31
40. Fatumo S, Chikowore T, Choudhury A, Ayub M, Martin AR, Kuchenbaecker K. 2022. A roadmap to increase diversity in genomic studies. *Nat. Med.* 28:243–50
41. Montserrat DM, Bustamante C, Ioannidis A. 2019. Class-conditional VAE-GAN for local-ancestry simulation. arXiv:1911.13220 [q-bio.GN]

42. Chen J, Mowlaei ME, Shi X. 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*, Art. 26. New York: ACM
43. Das S, Shi X. 2022. Offspring GAN augments biased human genomic data. In *Proceedings of the 13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, Pap. 50. New York: ACM
44. Perera M, Montserrat DM, Barrabés M, Geleta M, Giró-I-Nieto X, Ioannidis AG. 2022. Generative moment matching networks for genotype simulation. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 1379–83. New York: IEEE
45. Booker WW, Ray DD, Schrider DR. 2023. This population does not exist: learning the distribution of evolutionary histories with generative adversarial networks. bioRxiv 2022.09.17.508145. **https://doi.org/10.1101/2022.09.17.508145**
46. Novembre J, Stephens M. 2008. Interpreting principal component analyses of spatial population genetic variation. *Nat. Genet.* 40:646–49
47. Song Y, Westerhuis JA, Aben N, Michaut M, Wessels LFA, Smilde AK. 2019. Principal component analysis of binary genomics data. *Brief. Bioinform.* 20:317–29
48. Pearson K. 1901. LIII. On lines and planes of closest fit to systems of points in space. *Lond. Edinb. Dublin Philos. Mag. J. Sci.* 2:559–72
49. Hotelling H. 1933. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.* 24:417–41
50. van der Maaten L, Hinton G. 2008. Visualizing data using t-SNE. *J. Mach. Learn. Res.* 9:2579–605
51. McInnes L, Healy J, Melville J. 2020. UMAP: uniform manifold approximation and projection for dimension reduction. arXiv:1802.03426 [stat.ML]
52. Ma S, Shi G. 2020. On rare variants in principal component analysis of population stratification. *BMC Genet.* 21:34
53. Diaz-Papkovich A, Anderson-Trocmé L, Ben-Eghan C, Gravel S. 2019. UMAP reveals cryptic population structure and phenotype heterogeneity in large genomic cohorts. *PLOS Genet.* 15:e1008432
54. Ausmees K, Nettelblad C. 2022. A deep learning framework for characterization of genotype data. *G3* 12:jkac020
55. Battey CJ, Coffing GC, Kern AD. 2021. Visualizing population structure with variational autoencoders. *G3* 11:jkaa036
56. Choi Y, Li R, Quon G. 2022. Interpretable deep generative models for genomics. bioRxiv 2021.09.15.460498. **https://doi.org/10.1101/2021.09.15.460498**
57. Simidjievski N, Bodnar C, Tariq I, Scherer P, Andres Terre H, et al. 2019. Variational autoencoders for cancer data integration: design principles and computational practice. *Front. Genet.* 10:1205
58. Dwivedi SK, Tjärnberg A, Tegnér J, Gustafsson M. 2020. Deriving disease modules from the compressed transcriptional space embedded in a deep autoencoder. *Nat. Commun.* 11:856
59. Seninge L, Anastopoulos I, Ding H, Stuart J. 2021. VEGA is an interpretable generative model for inferring biological network activity in single-cell transcriptomics. *Nat. Commun.* 12:5684
60. Svensson V, Gayoso A, Yosef N, Pachter L. 2020. Interpretable factor models of single-cell RNA-seq via variational autoencoders. *Bioinformatics* 36:3418–21
61. Lopez R, Regier J, Cole MB, Jordan MI, Yosef N. 2018. Deep generative modeling for single-cell transcriptomics. *Nat. Methods* 15:1053–58
62. Way GP, Greene CS. 2018. Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders. *Pac. Symp. Biocomput.* 23:80–91
63. Wang D, Gu J. 2018. VASC: dimension reduction and visualization of single-cell RNA-seq data by deep variational autoencoder. *Genom. Proteom. Bioinform.* 16:320–31
64. Eraslan G, Simon LM, Mircea M, Mueller NS, Theis FJ. 2019. Single-cell RNA-seq denoising using a deep count autoencoder. *Nat. Commun.* 10:390
65. Grønbech CH, Vording MF, Timshel PN, Sønderby CK, Pers TH, Winther O. 2020. scVAE: variational auto-encoders for single-cell gene expression data. *Bioinformatics* 36:4415–22
66. Liu Q, Chen S, Jiang R, Wong WH. 2021. Simultaneous deep generative modelling and clustering of single-cell genomic data. *Nat. Mach. Intell.* 3:536–44

67. Tan J, Ung M, Cheng C, Greene CS. 2015. Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders. *Pac. Symp. Biocomput.* 20:132–43
68. Ghahramani A, Watt FM, Luscombe NM. 2018. Generative adversarial networks simulate gene expression and predict perturbations in single cells. bioRxiv 262501. **https://doi.org/10.1101/262501**
69. Kshirsagar M, Yuan H, Ferres JL, Leslie C. 2022. BindVAE: Dirichlet variational autoencoders for de novo motif discovery from accessible chromatin. *Genome Biol.* 23:174
70. Allesøe RL, Lundgaard AT, Hernández Medina R, Aguayo-Orozco A, Johansen J, et al. 2023. Discovery of drug–omics associations in type 2 diabetes with generative deep-learning models. *Nat. Biotechnol.* 41:399–408
71. Meisner J, Albrechtsen A. 2022. Haplotype and population structure inference using neural networks in whole-genome sequencing data. *Genome Res.* 32(8):1542–52
72. Rampášek L, Hidru D, Smirnov P, Haibe-Kains B, Goldenberg A. 2019. Dr.VAE: improving drug response prediction via modeling of drug perturbation effects. *Bioinformatics* 35:3743–51
73. Lotfollahi M, Wolf FA, Theis FJ. 2019. scGen predicts single-cell perturbation responses. *Nat. Methods* 16:715–21
74. Radford A, Metz L, Chintala S. 2016. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434 [cs.LG]
75. Park J, Kim H, Kim J, Cheon M. 2020. A practical application of generative adversarial networks for RNA-seq analysis to predict the molecular progress of Alzheimer's disease. *PLOS Comput. Biol.* 16:e1008099
76. Davi C, Braga-Neto U. 2021. A semi-supervised generative adversarial network for prediction of genetic disease outcomes. In *2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP)*. **https://doi.org/10.1109/MLSP52302.2021.9596351**
77. Frazer J, Notin P, Dias M, Gomez A, Min JK, et al. 2021. Disease variant prediction with deep generative models of evolutionary data. *Nature* 599:91–95
78. Wang Z, Wang J, Kourakos M, Hoang N, Lee HH, et al. 2021. Automatic inference of demographic parameters using generative adversarial networks. *Mol. Ecol. Resour.* 21:2689–705
79. Kelleher J, Etheridge AM, McVean G. 2016. Efficient coalescent simulation and genealogical analysis for large sample sizes. *PLOS Comput. Biol.* 12:e1004842
80. Yang H, Gu F, Zhang L, Hua XS. 2022. Using generative adversarial networks for genome variant calling from low depth ONT sequencing data. *Sci. Rep.* 12:8725
81. Baid G, Cook DE, Shafin K, Yun T, Llinares-López F, et al. 2022. DeepConsensus improves the accuracy of sequences with a gap-aware sequence transformer. *Nat. Biotechnol.* 41:232–38
82. Badsha MB, Li R, Liu B, Li YI, Xian M, et al. 2020. Imputation of single-cell gene expression with an autoencoder neural network. *Quant. Biol.* 8:78–94
83. Talwar D, Mongia A, Sengupta D, Majumdar A. 2018. AutoImpute: autoencoder based imputation of single-cell RNA-seq data. *Sci. Rep.* 8:16329
84. Chen J, Shi X. 2019. Sparse convolutional denoising autoencoders for genotype imputation. *Genes* 10:E652
85. Qiu YL, Zheng H, Gevaert O. 2020. Genomic data imputation with variational auto-encoders. *GigaScience* 9:giaa082
86. Devlin J, Chang MW, Lee K, Toutanova K. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805 [cs.CL]
87. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, et al. 2020. Language models are few-shot learners. arXiv:2005.14165 [cs.CL]
88. Benegas G, Batra SS, Song YS. 2023. DNA language models are powerful zero-shot predictors of genome-wide variant effects. bioRxiv 2022.08.22.504706. **https://doi.org/10.1101/2022.08.22.504706**
89. Mo S, Fu X, Hong C, Chen Y, Zheng Y, et al. 2021. Multi-modal self-supervised pre-training for regulatory genome across cell types. arXiv:2110.05231 [q-bio.GN]
90. Ji Y, Zhou Z, Liu H, Davuluri RV. 2021. DNABERT: pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. *Bioinformatics* 37:2112–20

91. Yang M, Huang L, Huang H, Tang H, Zhang N, et al. 2022. Integrating convolution and self-attention improves language model of human genome for interpreting non-coding regions at base-resolution. *Nucleic Acids Res.* 50:e81
92. Gwak HJ, Rho M. 2022. ViBE: a hierarchical BERT model to identify eukaryotic viruses using metagenome sequencing data. *Brief. Bioinform.* 23:bbac204
93. Zaheer M, Guruganesh G, Dubey A, Ainslie J, Alberti C, et al. 2021. Big Bird: transformers for longer sequences. arXiv:2007.14062 [cs.LG]
94. Shi X, Wu X. 2017. An overview of human genetic privacy. *Ann. N.Y. Acad. Sci.* 1387:61–72
95. Ca A. 2018. Machine learning and genomics: precision medicine versus patient privacy. *Philos. Trans. A* 376(2128):20170350
96. Kim M, Lauter K. 2015. Private genome analysis through homomorphic encryption. *BMC Med. Inform. Decis. Making* 15:S3
97. Sim JJ, Chan FM, Chen S, Meng Tan BH, Mi Aung KM. 2020. Achieving GWAS with homomorphic encryption. *BMC Med. Genom.* 13:90
98. Almadhoun N, Ayday E, Ulusoy Ö. 2020. Differential privacy under dependent tuples—the case of genomic privacy. *Bioinformatics* 36:1696–703
99. Rieke N, Hancox J, Li W, Milletarì F, Roth HR, et al. 2020. The future of digital health with federated learning. *npj Digit. Med.* 3:119
100. Aziz MMA, Anjum MM, Mohammed N, Jiang X. 2022. Generalized genomic data sharing for differentially private federated learning. *J. Biomed. Inform.* 132:104113
101. Grishin D, Raisaro JL, Troncoso-Pastoriza JR, Obbad K, Quinn K, et al. 2021. Citizen-centered, auditable and privacy-preserving population genomics. *Nat. Comput. Sci.* 1:192–98
102. Xie L, Lin K, Wang S, Wang F, Zhou J. 2018. Differentially private generative adversarial network. arXiv:1802.06739 [cs.LG]
103. Sapoval N, Aghazadeh A, Nute MG, Antunes DA, Balaji A, et al. 2022. Current progress and open challenges for applying deep learning across the biosciences. *Nat. Commun.* 13:1728
104. Nie W, Patel A. 2019. Towards a better understanding and regularization of GAN training dynamics. arXiv:1806.09235 [cs.ML]
105. Kurach K, Lučić M, Zhai X, Michalski M, Gelly S. 2019. A large-scale study on regularization and normalization in GANs. *Proc. Mach. Learn. Res.* 97:3581–90
106. Dumont V, Ju X, Mueller J. 2022. Hyperparameter optimization of generative adversarial network models for high-energy physics simulations. arXiv:2208.07715 [hep-ex]
107. Zhang Y, Tiňo P, Leonardis A, Tang K. 2021. A survey on neural network interpretability. *IEEE Trans. Emerg. Top. Comput. Intell.* 5:726–42
108. Novakovsky G, Dexter N, Libbrecht MW, Wasserman WW, Mostafavi S. 2022. Obtaining genetics insights from deep learning via explainable artificial intelligence. *Nat. Rev. Genet.* 24:125–37
109. Li C, Yao K, Wang J, Diao B, Xu Y, Zhang Q. 2022. Interpretable generative adversarial networks. *Proc. AAAI Conf. Artif. Intell.* 36:1280–88

Annual Review of Biomedical Data Science

Volume 6, 2023

Contents

Single-Cell RNA Sequencing for Studying Human Cancers
Dvir Aran 1

Challenges and Opportunities for Data Science in Women's Health
Todd L. Edwards, Catherine A. Greene, Jacqueline A. Piekos, Jacklyn N. Hellwege, Gabrielle Hampton, Elizabeth A. Jasper, and Digna R. Velez Edwards 23

Computational Methods for Single-Cell Proteomics
Sophia M. Guldberg, Trine Line Hauge Okholm, Elizabeth E. McCarthy, and Matthew H. Spitzer 47

Statistical Learning Methods for Neuroimaging Data Analysis with Applications
Hongtu Zhu, Tengfei Li, and Bingxin Zhao 73

Strategies for the Genomic Analysis of Admixed Populations
Taotao Tan and Elizabeth G. Atkinson 105

Decoding Aging Hallmarks at the Single-Cell Level
Shuai Ma, Xu Chi, Yusheng Cai, Zhejun Ji, Si Wang, Jie Ren, and Guang-Hui Liu 129

Addressing the Challenge of Biomedical Data Inequality: An Artificial Intelligence Perspective
Yan Gao, Teena Sharma, and Yan Cui 153

An Overview of Deep Generative Models in Functional and Evolutionary Genomics
Burak Yelmen and Flora Jay 173

Toward Identification of Functional Sequences and Variants in Noncoding DNA
Remo Monti and Uwe Ohler 191

A Review of and Roadmap for Data Science and Machine Learning for the Neuropsychiatric Phenotype of Autism
Peter Washington and Dennis P. Wall 211

Recent Developments in Ultralarge and Structure-Based Virtual Screening Approaches
Christoph Gorgulla 229

Human Microbiomes and Disease for the Biomedical Data Scientist
Jonathan L. Golob 259

Virus-Derived Small RNAs and microRNAs in Health and Disease
Vasileios Gouzouasis, Spyros Tastsoglou, Antonis Giannakakis, and Artemis G. Hatzigeorgiou 275

Combining Molecular and Radiomic Features for Risk Assessment in Breast Cancer
Alex A. Nguyen, Anne Marie McCarthy, and Despina Kontos 299

Single-Cell Multiomics
Emily Flynn, Ana Almonte-Loya, and Gabriela K. Fragiadakis 313

Importance of Diversity in Precision Medicine: Generalizability of Genetic Associations Across Ancestry Groups Toward Better Identification of Disease Susceptibility Variants
Lauren A. Cruz, Jessica N. Cooke Bailey, and Dana C. Crawford 339

Identification of Splice Variants and Isoforms in Transcriptomics and Proteomics
Taojunfeng Su, Michael A.R. Hollas, Ryan T. Fellers, and Neil L. Kelleher 357

Gene Interactions in Human Disease Studies—Evidence Is Mounting
Pankhuri Singhal, Shefali Setia Verma, and Marylyn D. Ritchie 377

Noninvasive Prenatal Testing Using Circulating DNA and RNA: Advances, Challenges, and Possibilities
Mira N. Moufarrej, Diana W. Bianchi, Gary M. Shaw, David K. Stevenson, and Stephen R. Quake 397

Challenges and Progress in Designing Broad-Spectrum Vaccines Against Rapidly Mutating Viruses
Rishi Bedi, Nicholas L. Bayless, and Jacob Glanville 419

The *All of Us* Data and Research Center: Creating a Secure, Scalable, and Sustainable Ecosystem for Biomedical Research
Kelsey R. Mayo, Melissa A. Basford, Robert J. Carroll, Moira Dillon, Heather Fullen, Jesse Leung, Hiral Master, Shimon Rura, Lina Sulieman, Nan Kennedy, Eric Banks, David Bernick, Asmita Gauchan, Lee Lichtenstein, Brandy M. Mapes, Kayla Marginean, Steve L. Nyemba, Andrea Ramirez, Charissa Rotundo, Keri Wolfe, Weiyi Xia, Romuladus E. Azuine, Robert M. Cronin, Joshua C. Denny, Abel Kho, Christopher Lunt, Bradley Malin, Karthik Natarajan, Consuelo H. Wilkins, Hua Xu, George Hripcsak, Dan M. Roden, Anthony A. Philippakis, David Glazer, and Paul A. Harris 443

Human Genomics of COVID-19 Pneumonia: Contributions of Rare and Common Variants
Aurélie Cobat, Qian Zhang, COVID Human Genetic Effort, Laurent Abel, Jean-Laurent Casanova, and Jacques Fellay 465

Errata

An online log of corrections to *Annual Review of Biomedical Data Science* articles may be found at http://www.annualreviews.org/errata/biodatasci