Towards creating longer genetic sequences with GANs: Generation in principal component space

Antoine Szatkownik, Cyril Furtlehner, Guillaume Charpiat, Burak Yelmen, Flora Jay

Towards creating longer genetic sequences with GANs: Generation in principal component space

Antoine Szatkownik[1], **Cyril Furtlehner**[1], **Guillaume Charpiat**[1], **Burak Yelmen**[1,2,*], **Flora Jay**[1,*]
[1] Université Paris-Saclay, CNRS, INRIA, LISN, Paris, France
[2] University of Tartu, Institute of Genomics, Tartu, Estonia
[*] These authors contributed equally.
Corresponding author: Antoine Szatkownik <szatkownik@lisn.fr>

Abstract

Synthetic data generation via generative modeling has recently become a prominent research field in genomics, with applications ranging from functional sequence design to high-quality, privacy-preserving artificial in silico genomes. Following a body of work on Artificial Genomes (AGs) created via various generative models trained with raw genomic input, we propose a conceptually different approach to address the issues of scalability and complexity of genomic data generation in very high dimensions. Our method combines dimensionality reduction, achieved by Principal Component Analysis (PCA), and a Generative Adversarial Network (GAN) learning in this reduced space. We compare the quality of AGs generated by our approach with AGs generated by the established models and report improvements on capturing population structure and linkage disequilibrium.

1 Introduction

Growing advances in DNA sequencing have spurred a data surge, elevating machine learning's importance in genomics [1–3]. However, a substantial amount of these data, held by companies, institutions or governmental agencies, are tied to privacy concerns impeding their accessibility. As of today, the unsupervised training of generative models on genomes aims at enhancing visualization by revealing fine-scale population structure [4–6], or generating synthetic data [7–12]. These generated synthetic genomes, or artificial genomes (AGs), can potentially become privacy-preserving alternatives to real biobanks, which could be used as reference panels for genome-wide association studies (GWAS), local ancestry inference methods or other downstream tasks such as imputation or selection scans [7]. In this context, Yelmen et al. [9] successfully tackled the challenging problem of generating very high dimensional genetic data despite small sample sizes by developing sophisticated GANs, Restricted Boltzmann Machines (RBMs) and Variational autoencoders (VAEs). Unlike fully connected architectures, the proposed scheme of convolving along the genomic sequences combined with location-specific variables is computationally feasible [5, 9], yet it still poses challenges due to high dimensions and the intrinsic complexity of whole genome sequences. These difficulties drive the need to seek alternative methods for larger genome-wide frameworks.

Fully-connected architectures are not affordable for direct application to large-scale genetic data: for instance, a simple model consisting of just one dense linear layer with $n = 1000$ latent variables and dealing with genetic sequences of L single nucleotide polymorphisms (SNPs), with L potentially of the order of millions (e.g., $L = 1$M), would have $L \times n = 1$B parameters. Thus, in order to be able to consider dense linear layers, and to make the learning process possibly easier, one needs to reduce the dimensions, to boil the important information down to a smaller space.

18th Conference on Machine Learning in Computational Biology (MLCB 2023).

To address the high-dimensional generative task, we thus propose to project the genetic data into a lower-dimensional subspace through dimension reduction, and to train a GAN within this reduced space. Principal Component Analysis (PCA) was preferred for its easy and faithful reconstruction from Principal Component (PC) space to data space, unlike t-SNE and UMAP, which either lack invertibility or result in poor reconstruction. Furthermore, PCA is nonparametric and used widely in population genetics studies, since genomic variation based on ancestry is well represented in PC space [13–21]. While autoencoders are also of interest to us, they are heavily parameterized, leading to long training times, and require careful regularization adjustments. Regarding PCA, in the principal subspace, each new dimension being a linear combination of the input features, the notion of locality within the sequence disappears, making dense layers or attention layers ideal candidates for the generative network architecture. Although PCA is linear, the generator output is not, as the GAN can learn non-linearities.

In this study, we introduce two latent generative modeling methods, a PCA Wasserstein GAN with gradient penalty (PCA-WGAN) and a chained alternative designed to preserve local information (**FIG. 2**). Both approaches leverage dimension reduction for high dimensional generative modeling. We then showcase the usefulness of our approaches on The 1000 Genomes Project Consortium [22] genetic dataset encompassing diverse human populations spread worldwide.

2 Materials and Methods

2.1 Overview

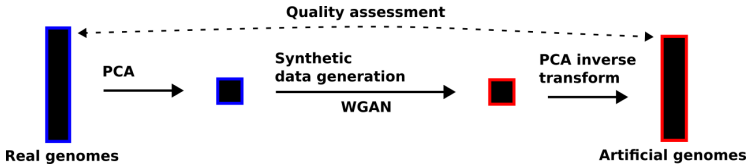

Figure 1: **Overview of the general method.**

Our general approach consists in (i) reducing the training dataset dimensions through principal component analysis; (ii) training a WGAN on these PC scores; (iii) generating synthetic PC scores by sampling the generator; (iv) inverse-transforming these scores back to synthetic genome samples; (v) assessing the quality of the artificial genomes using multiple descriptive population genetics statistics (**FIG. 1**).

2.2 Dataset description

The 1000 Genomes project is a database gathering human genome data obtained by sampling individuals from populations spread worldwide. The purpose of this consortium was to provide a diverse set of genetic data; it contains 26 populations, with a number of samples for each population ranging from one to roughly a hundred. The dataset used in this study was curated by Yelmen et al. [9]. It consists of 2504 individuals corresponding to 5008 phased haplotypes (*i.e.* for each individual there is one haplotype coming from the mother and one from the father), and contains 65,535 contiguous SNPs spanning chr1:534247-81813279 (*i.e.* $\sim$80 Mega base pairs), within the Omni 2.5 genotyping array framework. The data is formatted in the following way: rows are phased haplotypes; columns are positions of alleles represented by 0 when the corresponding nucleotide is identical with respect to the one in the reference genome (GRCh37) and 1 when it is point-wise mutated. Hence the dataset is represented as a binary matrix.

2.3 PCA-WGAN

A Generative Adversarial Network (GAN) comprises two competing neural networks, the discriminator and the generator. The discriminator learns to dissociate real data from synthetic one, while the generator aims at fooling the discriminator in a way that it fails at this classification task. Vanilla GANs face several problems such as mode collapse, training instability, vanishing gradients, and sensitivity to hyperparameters [23, 24]. To account for these challenges, we used a Wasserstein GAN

[23] with gradient penalty [24] (WGAN-GP) implemented in pytorch [25]. Whereas the discriminator of a vanilla GAN produces a probability of being fake or real, the critic of a WGAN produces an unbounded "realness" score for fake and real samples. The loss of a WGAN-GP, based on the estimation of the Earth Mover's distance by the critic of two distributions on a batch of samples, is as follows :

$$\mathbb{E}_{\tilde{z}\sim\mathbb{P}_{\text{generated}}}\big[C(\tilde{z})\big] - \mathbb{E}_{z\sim\mathbb{P}_{\text{real}}}\big[C(z)\big] + \lambda\mathbb{E}_{\hat{z}\sim\mathbb{P}_{\hat{z}}}\big[(\|\nabla_{\hat{z}}C(\hat{z})\| - 1)^2\big]$$

$$\text{s.t.} \quad \hat{z} = t\tilde{z} + (1-t)z \quad \text{and} \quad t \sim \mathbb{U}[0,1],$$

where C is the critic; $\tilde{z}$ (resp. z) is a sample from the generator modeled distribution $\mathbb{P}_{\text{generated}}$ (resp. from the data distribution $\mathbb{P}_{\text{real}}$). The gradient penalty term, evaluated at a point $\hat{z}$ interpolated between a real and a fake point, is scaled by a constant factor λ and constrains the critic's gradient to be close to 1; and $\mathbb{U}[0,1]$ is the uniform distribution on $[0,1]$, yielding a distribution $\mathbb{P}_{\hat{z}}$ over $\hat{z}$.

The PCA-WGAN workflow involves an initial PCA on a binary matrix (5008 samples, 65K SNPs, see **2.2**), retaining 4507 components (90% PCs, see **section 3.1 & S1.2**), and a generative learning process in that PC space (**FIG. 2**). To generate new fake binary SNP data, we sample points in the space learned by the GAN (modeling the PC space), then perform PCA inverse transformation using eigenvectors from real data PCA, followed by a binarization step based on a threshold of 0.5 (see **S1.2** for more details).

2.4 Glocal-PCA-WGAN: Chaining multiple local PCA-WGAN

We designed an improvement of PCA-WGAN, called Glocal-PCA-WGAN. To circumvent difficulties arising from performing dimension reduction in the $\frac{N_{\text{features}}}{N_{\text{samples}}} \gg 1$ regime, our approach segments the SNP matrix into K multiple blocks. The procedure goes as follows: we split the SNP matrix into three successive blocks of equal number of SNPs and apply PCA to each block (*i.e.* each genomic region) independently of each other (colored blocks labeled 1, 2 and 3 in **FIG. 2B**) so as to preserve local information. In this architecture, there are multiple local critics and one global critic. Each local critic aims to capture the local information of its assigned block, while the global critic should recover the relations between the blocks. The WGAN-GP is trained in the space of the concatenated PCA projections. Once the training is done, we sample the generator, split the generated PC scores into three parts and inverse transform each chunk with the corresponding PCA (see **S1.3** for more details).

Constraints on the choice of block number *K*. Increasing K leads to higher chances of breaking LD. If K is too large, then the blocks have less features than samples, leading to no reduction at all (i.e., the output size of the generator & input size of critic will be 65K). If K is too small then we are compressing too many SNPs into N_{samples} dimensions, and for small K, we should not keep all PCs, otherwise the global discriminator will take as input a vector of size $K \times N_{\text{samples}}$. All in all, we want K to be small but not so much so that PCA blocks still capture local information, hence we chose $K = 3$ to keep the balance between reasonable parameter size and retaining a number of PCs that still lead to a nearly optimal reconstruction error.

3 Results

3.1 Principal components space is fertile ground for learning

We perform various pre-hoc training experiments to decide how many dimensions we should keep when performing PCA and to gauge the errors caused by reconstruction. These experiments are performed considering a separate test dataset consisting of real genomes as the best-case generation outcome.

Will learning in PC space work? Specifically, we aim to determine whether the data generated in PC space can be reliably mapped back to the original space, assuming the generation step was successful. To check this, we randomly partition the dataset into a train and a test set. Both sets follow the same distribution; in particular, the same modes appear in both, and they have similar densities. The test set acts as the perfect case where an ideal generative model completely captures the target distribution. Hence, a test set should give us hints on the expected behavior of properly generated data after reconstruction.

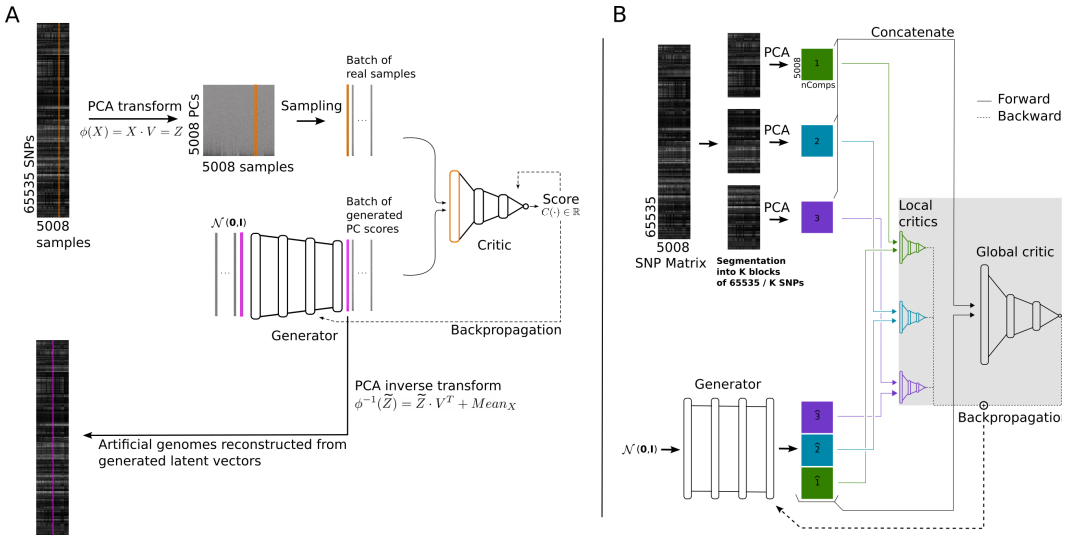

Figure 2: **PCA-WGAN architectures. A.** PCA-WGAN. The 5008 samples by 65535 SNP matrix is reduced via PCA to a 5008 samples by 4507 dimensions (90% PCs) matrix of PC scores. The WGAN learns in the PC space. Once the training is finished, the generated PC scores are mapped back to the data space via PCA inverse transform. **B.** Glocal-PCA-WGAN. The SNP matrix is split into even-sized chunks, to which PCA is independently applied. Each local critic is fed with a single PCA chunk (colored block labeled 1, 2, or 3) and a generated PC scores chunk (colored block labeled $\hat{1}$, $\hat{2}$, or $\hat{3}$), while the global critic is fed with the concatenated PCA projections and generated PC scores (i.e. all colored blocks).

We consider the following protocol that does not rely on generated samples. First, we split randomly the dataset ($N_{\text{samples}} = 5008$) into a train and a test set of equal size and perform a PCA on the training set, varying the number of kept components. Then, we project the test set onto the principal subspace derived from the training set (centered beforehand). Finally, we reconstruct the projected test set back into the original space and binarize it. Binarization is obtained through quantization with a threshold set to 0.5. We compute the mean reconstruction error of the binarized reconstructed test set with respect to the initial test set. The reconstruction error for one sample in the data space is defined as

$$\frac{1}{L}\sum_{i=1}^{L} |X^i_{\text{reconstructed test}} - X^i_{\text{test}}|,$$

where L is the number of SNPs, X^i_{test} and $X^i_{\text{reconstructed test}} \in \{0, 1\}$. The reconstruction error for the training set is zero when keeping 100% of the PCs (**FIG.3.a**), *i.e.* there is no loss of information, but is equal to 0.06 in average for the test set. Decreasing the number of retained PCs reduces the gap between the train and test errors, indicating limited overfitting when few components are kept. However, because the test reconstruction error is at its minimum when keeping all components, we apply this final procedure for all remaining analyses.

Keeping all PCs, the averaged error for the binarized reconstructed test set is around 6%. To put this value into context, we computed the distribution of genetic distance separating two individuals (measured via Hamming distance, **FIG. 3.c**). Note that the reconstruction error applied to binary data is exactly the Hamming distance normalized by the sequence length, allowing direct comparisons. We find that the test reconstruction error (6%) is way below the minimal genetic distance in the considered dataset (13200, corresponding to $\sim 20\%$ of errors along the SNP sequence), indicating that the error caused by reconstruction is smaller than the distance between the closest individuals.

We refined this analysis at the individual level by investigating, for each reconstructed point, whether its nearest neighbor is the source point (desired behavior) or another individual of the test set (**FIG. 3.d-e**). Specifically, we compared the (Hamming) distance *A* (between a source point in the test set and its reconstructed point) to the distance *B* (between the reconstructed source point and its nearest

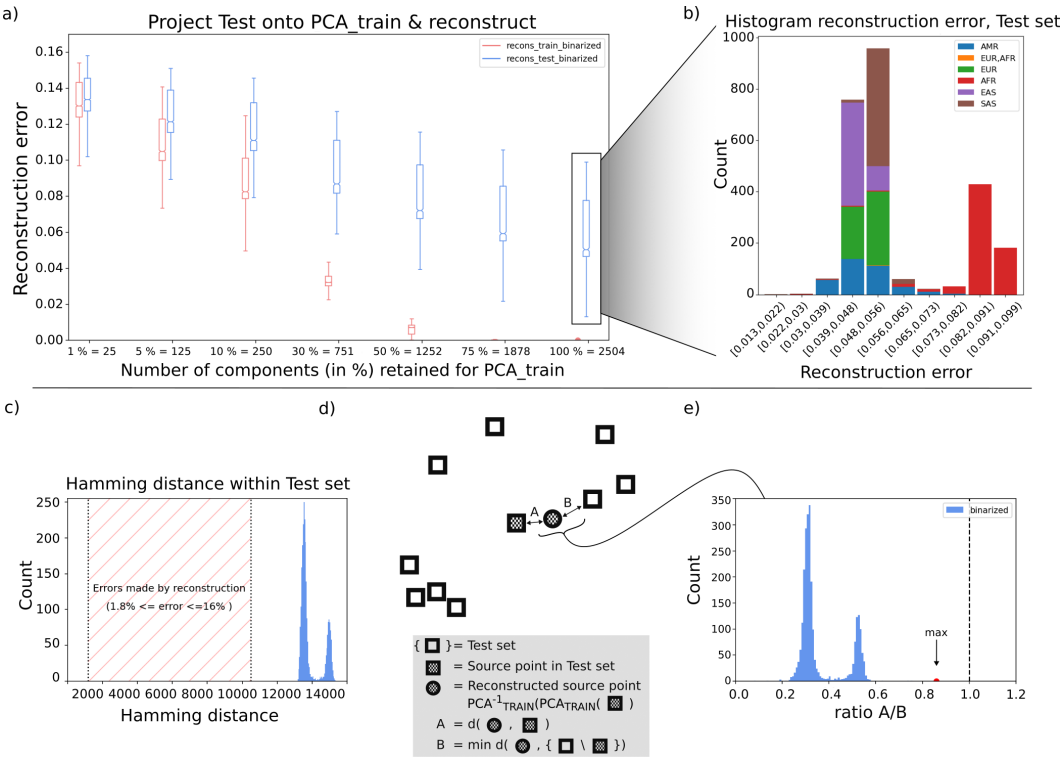

Figure 3: **PCA reconstruction analyses. a)** Reconstruction error as a function of the number of components retained when applying PCA to the training set. The reconstruction error is plotted after reconstructing the training set (red), and after projecting the test set onto the eigensubspace of the training set and reconstructing it (blue). **b)** Histogram of the reconstruction error for the reconstructed test set, keeping all components, colored by superpopulation labels. The superpopulation labels correspond to: AMR: American ancestry, EUR,AFR: European and African ancestry, EUR: European ancestry, AFR: African ancestry, EAS: East Asian ancestry, SAS: South Asian ancestry. **c)** Hamming distance within the test set. **d)** How much does reconstruction move data points? **e)** Distribution of the ratio of the distance A between a source point and its reconstruction, by the distance B between the reconstructed source point and its nearest neighbor in the test set minus the source point.

neighbor in the test set excluding the source point). If $B < A$ then the reconstruction step locally distorts the data, *i.e.*, the neighborhood of a point in the test set is not preserved in the reconstructed test set, which would discourage to learn the generative model in the PC space. We found that this is not the case (**FIG. 3.d**), as the distribution of the ratio $\frac{A}{B}$ is considerably lower than 1 (left hand side of dotted vertical line), with a max value equal to 0.82. This further demonstrates that the local structure is not disrupted via reconstruction.

Why is the distribution of the reconstruction error bimodal? We observe that samples with a higher reconstruction error are of African ancestry while samples from the rest of the world are mixed in the first mode (**FIG. 3.b**). This is likely due to the known higher genetic diversity present on this continent (explainable by human past demographic history) [26–28].

Are the low variance principal axes relevant when studying the test set? Are they better than random directions? To answer this question, we selected n_{Comp} principal axes and constructed the remaining $2504 - n_{\text{Comp}}$ dimensions as random directions, by sampling Gaussian vectors $\mathcal{N}_k(\mathbf{0}, \mathbf{I})$ with $k = 65535$ and orthonormalizing them to the current basis and to themselves using the Gram–Schmidt algorithm [29]. From **FIG. S1**, we observe that adding random directions to the n_{Comp} first principal axes does decrease the reconstruction error, as expected since there is more information when considering more directions, i.e. when the projection space gets larger. However, adding the next principal

axes instead of random directions decreases the reconstruction error much faster. This confirms that the low variance principal axes contain relevant information for the reconstruction step.

3.2 Quality assessment of the synthetic data

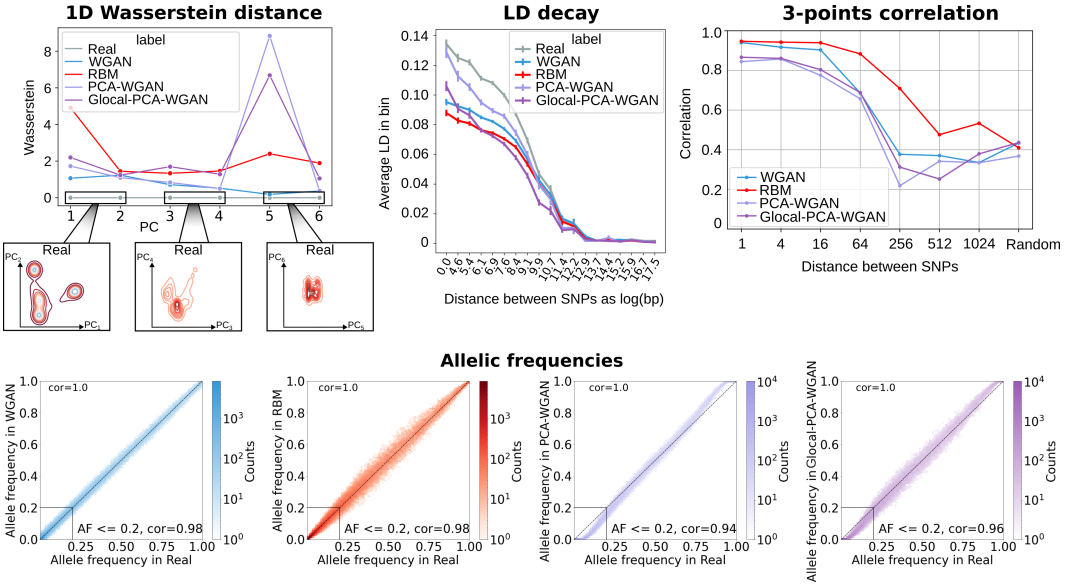

Figure 4: **Population genetics summary statistics. a)** The real and the synthetic data are concatenated into a single dataset, to which PCA is applied. The 1D Wasserstein distance is computed on each PCs. Real data is the grey curve, WGAN from [9] in blue, RBM from [9] in red, PCA-WGAN in light purple, Glocal-PCA-WGAN in dark purple. The density plot of real data for the first six PCs is detailed in first column of **FIG.S3**. **b)** LD decay approximation. **c)** Three-points correlation separated by a varying amount of SNPs. **d)** 2D Histogram of the allelic frequencies in real data (x-axis) and in AGs produced by the models.

Yelmen et al. [7] evaluated the quality of generated data, based on a set of summary statistics coming from population genetics. We present here a subset of these metrics on the synthetic 65K SNPs data generated by PCA-WGAN and Glocal-PCA-WGAN, and compare it to previous WGAN and RBM models which learn directly in the genetic space [9].

Real data exhibits a strong population structure pattern captured by PC1 to PC4 (**FIG. 4.a**), which is effectively replicated in AGs from all models, as can be seen through the 1D Wasserstein distance between the real and AG projections. Overall the error curve for PCA-WGAN is always below that of the RBM except for PC5, where the model is generating a concentrated density on that PC, causing an increased Wasserstein distance. This concentration issue could potentially be resolved by multiplying the corresponding PC scores by the inverse empirical shrinkage factor [30].

The linkage disequilibrium (LD) curve shows how the correlation between pairs of SNPs decreases as a function of their physical distance (**FIG. 4.b**). PCA-WGAN reproduces these statistics better than all the other models, *i.e.*, the LD curve is the closest to the real curve, notably for short-range interactions. The lower LD observed for Glocal-PCA-WGAN is likely due to the uniform splitting of the SNP matrix (yielding blocks with the same number of SNPs). A potential future improvement in this regard is to partition into regions that preserve LD blocks [31] (e.g., by splitting at recombination hotspots).

A three-point correlation statistics is computed for SNP triplets separated by *1, 4, 16, 64, 256, 512, 1024* and a random number of SNPs respectively (**FIG. 4.c**). While PCA-WGAN has lower correlations than RBM model in Yelmen et al. [9], it remains close to the convolutional WGAN model, and demonstrates similar results to both models in the challenging scenario involving SNP

triplets separated by random distances. Multiple local critics (Glocal-PCA-WGAN) improve this statistic for all SNP distances and even marginally surpasses WGAN and RBM benchmark models for SNPs separated by random distances.

The synthetic datasets follow overall the same trends as the real one, in terms of allelic frequencies (**FIG. 4.d**). For frequencies ≤ 0.2, the correlation score between real and AGs ranges from 0.94 to 0.98. Nonetheless, PCA-WGAN fixes many alleles (20515 $\sim 31\%$ of the sites; 12487 for Glocal-PCA-WGAN; 2672 for WGAN; 47 for RBM; 8 in real), especially the ones that were rare in the original dataset, similar to previous findings [7, 9].

3.3 Performance comparison

Table 1: Training GPU & runtime comparison between models for 65K SNP dataset

	PCA-WGAN	**Glocal-PCA-WGAN**	**CRBM**	**convWGAN**
Params	74M	140M	130M	16M
GPUs	1-A100-40GB	1-A100-40GB	$\geq$1-RTX3090-24GB	1-A100-40GB
Wall clock	20 hours	2 days	13$\times$10 hours (in //)	6-8 days

We compare training compute resources between different models (**Table 1**), as well as the number of parameters as a function of sequence length (**FIG.S2**). The CRBM in [9] consists in 13 RBMs with 10M parameters each, where each RBM is trained during $\sim$ 10 hours (wall clock time). The training time of the whole set of RBMs depends on the number of available GPUs, and can benefit from parallelization (//).

4 Discussion and conclusion

This study tackles scaling generative models for high-dimensional SNP data with PCA-WGAN and a derivative model, and assesses AG quality through population genetics summary statistics. To our knowledge, while dimension reduction is routinely used as a ML preprocessing step, its application in generative modeling for genomics is unexplored, making our reduced-space genomic data generation simple and innovative. Moreover, no other studies benchmarked their models on a very high dimensional tabular dataset, except for models in [9] and the models presented in this current work. For example, Dang et al. [10] demonstrates good performances for the 805 SNPs dataset [7, 32] but weaker ones for 10K SNPs, while Zhang et al. [11] demonstrates performances comparable to [9] for 10K SNPs but did not scale up to 65K SNPs. Despite PCA-WGAN having a drawback in allelic frequencies, it achieves results comparable to recent advances [7, 9] and even captures two-point SNP correlations (LD) better. On the other hand, Glocal-PCA-WGAN improves the allele fixing issue since it applies PCA on shorter sequences, but produces worse results in terms of LD. This could potentially be improved by finding optimal splits for blocks with respect to LD [31]. Furthermore, Glocal-PCA-WGAN could scale to longer SNP sequences. As we showed that PCA-WGAN on 65K SNPs offered competitive results, we could envision a Glocal-PCA-WGAN processing 3$\times$65K SNPs. Keeping 40% of PCs for each block would lead to the same parameter size as in the current Glocal-PCA-WGAN for 65K SNPs (**S1.3**). Our newly created AGs should prove useful to a vast range of genomic analyses that do not require rare variants and filter out alleles under the 5% frequency threshold.

Overall, PCA-WGAN is a promising novel methodology which combines simple dimensionality reduction and generative modeling to produce diverse and longer artificial genomes, making it a potentially valuable tool for enhancing genome-wide analyses with synthetic data, especially when access to real datasets is limited.

5 Acknowledgements

This work benefited from Inria TAU computing resources and ANR-20-CE45-0010-01 RoDAPoG funding. We thank Michèle Sebag for insightful discussions.

References

[1] Burak Yelmen and Flora Jay. An Overview of Deep Generative Models in Functional and Evolutionary Genomics. *Annual Review of Biomedical Data Science*, 6(1):173–189, August 2023. ISSN 2574-3414, 2574-3414. doi: 10.1146/annurev-biodatasci-020722-115651.

[2] Kevin Korfmann. Deep Learning in Population Genetics. *Genome Biology and Evolution*, 15(2), February 2023. ISSN 0090-5364. doi: 10.1093/gbe/evad008.

[3] Xin Huang, Aigerim Rymbekova, Olga Dolgova, Oscar Lao, and Martin Kuhlwilm. Harnessing deep learning for population genetic inference. *Nature Reviews Genetics*, September 2023. ISSN 1471-0056, 1471-0064. doi: 10.1038/s41576-023-00636-3.

[4] C J Battey, Gabrielle C Coffing, and Andrew D Kern. Visualizing population structure with variational autoencoders. *G3 Genes|Genomes|Genetics*, 11(1):jkaa036, March 2021. ISSN 2160-1836. doi: 10.1093/g3journal/jkaa036.

[5] Kristiina Ausmees and Carl Nettelblad. A deep learning framework for characterization of genotype data. *G3 Genes|Genomes|Genetics*, 12(3):jkac020, March 2022. ISSN 2160-1836. doi: 10.1093/g3journal/jkac020.

[6] Margarita Geleta, Daniel Mas Montserrat, Xavier Giro-i-Nieto, and Alexander G Ioannidis. Deep Variational Autoencoders for Population Genetics.

[7] Burak Yelmen, Aurélien Decelle, Linda Ongaro, Davide Marnetto, Corentin Tallec, Francesco Montinaro, Cyril Furtlehner, Luca Pagani, and Flora Jay. Creating artificial human genomes using generative neural networks. *PLOS Genetics*, 17(2):e1009303, February 2021. ISSN 1553-7404. doi: 10.1371/journal.pgen.1009303.

[8] Maria Perera, Daniel Mas Montserrat, Mıriam Barrabes, Margarita Geleta, Xavier Giro-i-Nieto, and Alexander G Ioannidis. Generative Moment Matching Networks for Genotype Simulation.

[9] Burak Yelmen, Aurélien Decelle, Leila Lea Boulos, Antoine Szatkownik, Cyril Furtlehner, Guillaume Charpiat, and Flora Jay. Deep convolutional and conditional neural networks for large-scale genomic data generation. *PLOS Computational Biology*, 19(10):e1011584, October 2023. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1011584.

[10] Meihua Dang, Anji Liu, Xinzhu Wei, and Sriram Sankararaman. Tractable and Expressive Generative Models of Genetic Variation Data.

[11] Daoan Zhang, Weitong Zhang, Bing He, Jianguo Zhang, Chenchen Qin, and Jianhua Yao. DNAGPT: A Generalized Pretrained Tool for Multiple DNA Sequence Analysis Tasks, July 2023.

[12] Sophie Wharrie, Zhiyu Yang, Vishnu Raj, Remo Monti, Rahul Gupta, Ying Wang, Alicia Martin, Luke J O'Connor, Samuel Kaski, Pekka Marttinen, Pier Francesco Palamara, Christoph Lippert, and Andrea Ganna. HAPNEST: Efficient, large-scale generation and evaluation of synthetic datasets for genotypes and phenotypes. *Bioinformatics*, 39(9):btad535, September 2023. ISSN 1367-4811. doi: 10.1093/bioinformatics/btad535.

[13] C. Tian, P. K. Gregersen, and M. F. Seldin. Accounting for ancestry: Population substructure and genome-wide association studies. *Human Molecular Genetics*, 17(R2):R143–R150, October 2008. ISSN 0964-6906, 1460-2083. doi: 10.1093/hmg/ddn268.

[14] Alkes L. Price, Noah A. Zaitlen, David Reich, and Nick Patterson. New approaches to population stratification in genome-wide association studies. *Nature Reviews Genetics*, 11(7):459–463, July 2010. ISSN 1471-0056, 1471-0064. doi: 10.1038/nrg2813.

[15] Oscar Lao, Timothy T. Lu, Michael Nothnagel, Olaf Junge, Sandra Freitag-Wolf, Amke Caliebe, Miroslava Balascakova, Jaume Bertranpetit, Laurence A. Bindoff, David Comas, Gunilla Holmlund, Anastasia Kouvatsi, Milan Macek, Isabelle Mollet, Walther Parson, Jukka Palo, Rafal Ploski, Antti Sajantila, Adriano Tagliabracci, Ulrik Gether, Thomas Werge, Fernando Rivadeneira, Albert Hofman, André G. Uitterlinden, Christian Gieger, Heinz-Erich Wichmann, Andreas Rüther, Stefan Schreiber, Christian Becker, Peter Nürnberg, Matthew R. Nelson, Michael Krawczak, and Manfred Kayser. Correlation between Genetic and Geographic Structure in Europe. *Current Biology*, 18(16):1241–1248, August 2008. ISSN 09609822. doi: 10.1016/j.cub.2008.07.049.

[16] John Novembre, Toby Johnson, Katarzyna Bryc, Zoltán Kutalik, Adam R. Boyko, Adam Auton, Amit Indap, Karen S. King, Sven Bergmann, Matthew R. Nelson, Matthew Stephens, and Carlos D. Bustamante. Genes mirror geography within Europe. *Nature*, 456(7218):98–101, November 2008. ISSN 1476-4687. doi: 10.1038/nature07331.

[17] Kai Yu, Zhaoming Wang, Qizhai Li, Sholom Wacholder, David J. Hunter, Robert N. Hoover, Stephen Chanock, and Gilles Thomas. Population Substructure and Control Selection in Genome-Wide Association Studies. *PLoS ONE*, 3(7):e2551, July 2008. ISSN 1932-6203. doi: 10.1371/journal.pone.0002551.

[18] P. Menozzi, A. Piazza, and L. Cavalli-Sforza. Synthetic Maps of Human Gene Frequencies in Europeans. *Science*, 201(4358):786–792, September 1978. doi: 10.1126/science.356262.

[19] Nick Patterson, Alkes L. Price, and David Reich. Population Structure and Eigenanalysis. *PLOS Genetics*, 2(12):e190, December 2006. ISSN 1553-7404. doi: 10.1371/journal.pgen.0020190.

[20] Olivier Hanotte, Daniel G. Bradley, Joel W. Ochieng, Yasmin Verjee, Emmeline W. Hill, and J. Edward O. Rege. African Pastoralism: Genetic Imprints of Origins and Migrations. *Science*, 296(5566):336–339, April 2002. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1069878.

[21] Alkes L. Price, Nick J. Patterson, Robert M. Plenge, Michael E. Weinblatt, Nancy A. Shadick, and David Reich. Principal components analysis corrects for stratification in genome-wide association studies. *Nature Genetics*, 38(8):904–909, August 2006. ISSN 1546-1718. doi: 10.1038/ng1847.

[22] The 1000 Genomes Project Consortium. A map of human genome variation from population-scale sequencing. *Nature*, 467(7319):1061–1073, October 2010. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature09534.

[23] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN, December 2017.

[24] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved Training of Wasserstein GANs, December 2017.

[25] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch.

[26] Sarah A. Tishkoff and Scott M. Williams. Genetic analysis of African populations: Human evolution and complex disease. *Nature Reviews Genetics*, 3(8):611–621, August 2002. ISSN 1471-0064. doi: 10.1038/nrg865.

[27] Michael C. Campbell and Sarah A. Tishkoff. African Genetic Diversity: Implications for Human Demographic History, Modern Human Origins, and Complex Disease Mapping. *Annual Review of Genomics and Human Genetics*, 9(1):403–433, September 2008. ISSN 1527-8204, 1545-293X. doi: 10.1146/annurev.genom.9.081307.164258.

[28] The 1000 Genomes Project Consortium, Corresponding authors, Adam Auton, Gonçalo R. Abecasis, Steering committee, David M. Altshuler, Richard M. Durbin, Gonçalo R. Abecasis, David R. Bentley, and Chakravarti ... A global reference for human genetic variation. *Nature*, 526(7571):68–74, October 2015. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature15393.

[29] Stephen Boyd and Lieven Vandenberghe. *Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares*. Cambridge University Press & Assessment, 1 edition, June 2018. ISBN 978-1-108-58366-4 978-1-316-51896-0. doi: 10.1017/9781108583664.

[30] Seunggeun Lee, Fei Zou, and Fred A. Wright. Convergence and prediction of principal component scores in high-dimensional settings. *The Annals of Statistics*, 38(6), December 2010. ISSN 0090-5364. doi: 10.1214/10-AOS821.

[31] Florian Privé. Optimal linkage disequilibrium splitting. *Bioinformatics*, 38(1):255–256, December 2021. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btab519.

[32] Vincenza Colonna, Qasim Ayub, Yuan Chen, Luca Pagani, Pierre Luisi, Marc Pybus, Erik Garrison, Yali Xue, Chris Tyler-Smith, and The 1000 Genomes Project Consortium. Human genomic regions with exceptionally high levels of population differentiation identified from 911 whole-genome sequences. *Genome Biology*, 15(6):R88, 2014. ISSN 1465-6906. doi: 10.1186/gb-2014-15-6-r88.

S1 Supplementary Material

S1.1 Reconstruction with PCs augmented by random axes

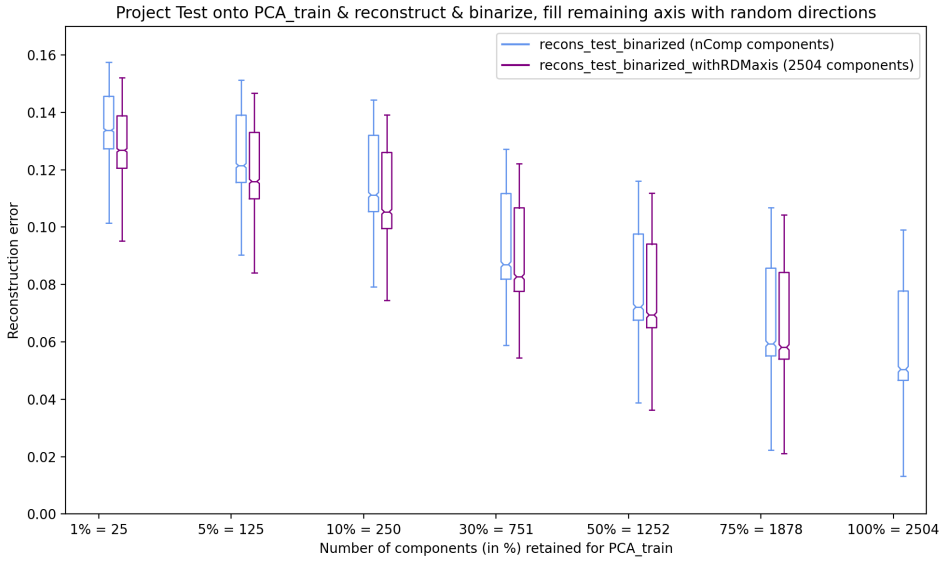

Figure S1: **PCA reconstruction analyses with random orthonormal axes.** The reconstruction error is computed for two procedures: (1) PCA is applied on the training data while varying the number of kept axes (x-axis), the test data is then projected onto the resulting eigensubspace, reconstructed and binarized (blue); (2) PCA is applied on the training data while varying the number of kept axes (x-axis), the remaining axes are filled with random gaussian vectors that are orthonormalized to the kept PCA eigenvectors and to themselves via Gram-Schmidt algorithm, the test data is then projected onto the resulting eigensubspace, reconstructed and binarized (indigo). Adding random directions decreases the reconstruction error, but not as much as considering remaining PCA modes instead.

S1.2 Implementation details of PCA-WGAN

Remarking that the reconstruction error is low enough and reasonably similar when more than 50% of the axes are kept for PCA (**FIG.3**), we decided not to keep all the PCs due to parameter size complexity. PCA is applied keeping 90% of the components, *i.e.* 4507 PCs.

The fully-connected architecture of the generator is the following :

- input layer : 5408 neurons
- 1st hidden layer: 5150 neurons
- BatchNorm1D
- LeakyReLU with negative slope of 0.01
- 2nd hidden layer: 4828 neurons
- BatchNorm1D
- LeakyReLU with negative slope of 0.01
- last layer: 4507 neurons

The fully-connected architecture of the critic is the following :

- input layer : 4507 neurons
- 1st hidden layer: 2253 neurons
- LeakyReLU with negative slope of 0.01

- 2nd hidden layer: 1502 neurons
- LeakyReLU with negative slope of 0.01
- last layer: 1 neuron

For both neural networks the optimizer is RMSprop with weight decay=1e-4. The learning rate for the generator (resp. critic) is 0.0001 (resp. 0.0008). The PCA-WGAN was trained for 1300 epochs. The batch size was set to 32. The noise fed to the generator is a multivariate normal vector with mean 0 and unit variance.

The generator takes input noise of at least the same size as the number of variables of the target distribution and outputs vectors of dimension the number of PCs. The intuition behind this is that it is rather tedious to fill a volume (hypothetical target distribution) with a curve (over-simplified input noise) (see Peano's space-filling curve). The critic takes as input a vector of 4507 dimensions, which is either generated ($\tilde{z}$), or the projection of a real sample onto the eigensubspace (z) and outputs a "realness" score.

The critic minimizes

$$\mathbb{E}_{\tilde{z}\sim\mathbb{P}_{generated}}[C(\tilde{z})] - \mathbb{E}_{z\sim\mathbb{P}_{real}}[C(z)] + \lambda\mathbb{E}_{\hat{z}\sim\mathbb{P}_{\hat{z}}}[(\|\nabla_{\hat{z}}C(\hat{z})\| - 1)^2]$$
$$\text{s.t. } \hat{z} = t\tilde{z} + (1-t)z \text{ and } t \sim \mathbb{U}[0,1],$$

while the generator maximizes

$$\mathbb{E}_{\tilde{z}\sim\mathbb{P}_{generated}}[C(\tilde{z})]$$

S1.3 Implementation details of Glocal-PCA-WGAN

The SNP matrix is evenly split (along the SNPs) into K parts. Since this approach makes the number of parameters of the model scale with the number of blocks, K was chosen to be small ($= 3$). As before, remarking that the reconstruction error is low enough and reasonably similar around 50% of kept PCA axes (**FIG. 3**), we decided not to keep all the PCs due to parameter size complexity. We thus retain 2000 principal components (i.e. 40% of PCs) per block amounting to $\sim$90% of the explained variance for each.

The generator takes as input noise of dimension 6000 and output vectors of dimension 6000. In this model, there are as many local critics as there are blocks, plus a global critic. Each local critic aims to capture the local information of its assigned block while the global critic should capture the relations between the blocks. The global critic takes as input either the concatenated projections (1,2,3) or the generated data ($\hat{1}, \hat{2}, \hat{3}$) and outputs a "realness" score. Each local critic takes as input the projection of a block onto its principal subspace truncated to 2000 dimensions, or the corresponding block of generated samples, and also outputs a "realness" score. Each local critic i minimizes the Wasserstein loss restricted to its block. The global critic minimizes the Wasserstein loss restricted over the concatenated samples. The generator maximizes the outputs of the local critics on their assigned blocks (by splitting the generated PC scores) and maximizes the output of the global critic.

It was trained with the same set of hyperparameter values as for PCA-WGAN with a number of epochs set to 1100. For the generator, the number of neurons remains constant throughout the layers.

S1.4 Parameter size comparison

For PCA-WGAN, the model has parameter size equal to (**FIG. S2**):

- 73M for 10K SNPs
- 73M for 65K SNPs
- 73M for 1M SNPs

For CRBM [9], the model has parameter size equal to :

- 10M for 10K SNPs

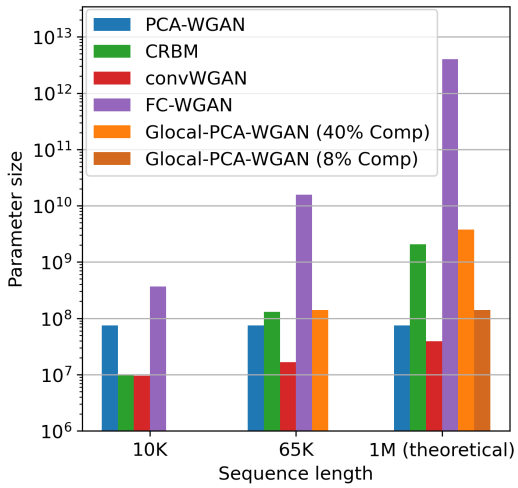

Figure S2: **Complexity & scalability.** Parameter size of the models as a function of sequence length. All models for sequence length of 1M SNPs are theoretical as they were not tested. The CRBM [9] consists in 13 RBMs having 10M parameters each. The training time of the whole set of RBMs depends on the number of available GPUs. As they can be trained in parallel or in sequence, the complexity is more a matter of time rather than parameter size.

- 130M for 65K SNPs
- 2.08B for 1M SNPs

For convWGAN [9], the model has parameter size equal to :

- 9.5M for 10K SNPs
- 16.6M for 65K SNPs
- 39.3M for 1M SNPs

For a theoretical WGAN with fully connected architecture (FC-WGAN), the model would have a parameter size equal to :

- 366.7M for 10K SNPs
- 15.7B for 65K SNPs
- 4T for 1M SNPs

For a Glocal-PCA-WGAN, the model have a parameter size equal to :

- 140M keeping 40% of PCs for 65K SNPs
- 3.8B keeping 40% of PCs and 140M keeping 8% of PCs for 1M SNPs and $K = 16$ so that the SNP blocks are of sizes 65535.

We would like to stress out that for a sequence of 1M SNPs, the implementation of the models and their training was not carried out, and is thus purely theoretical. Extrapolating the architecture to higher sequence length does not guarantee a successful convergence during training.

S1.5 Population genetics summary statistics

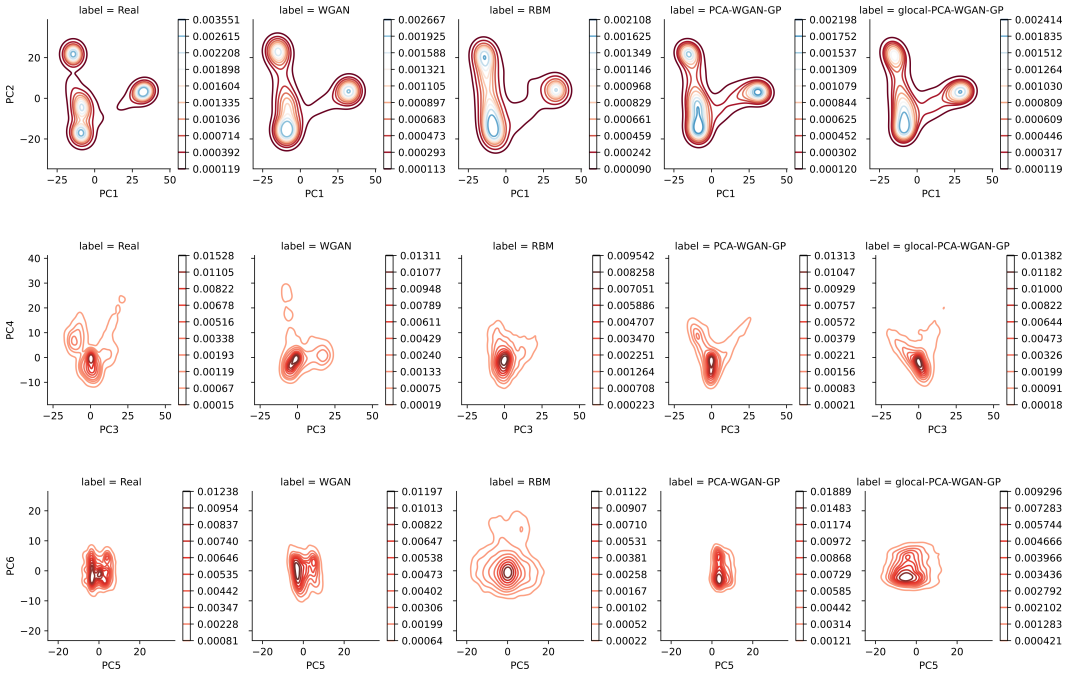

Figure S3: **Density plot of combined real and artificial genome datasets for the first six PCs.** Density increases from red to blue.