10.51231/2667-9507-2026-005-12-78-89

Machine Learning in Genomics, A Critical Review of Model Classes, Validation Practices, and Barriers to Translation

Ketevan Kartvelishvili[1], Giorgi Kuchava[1,2]

[1] *Ivane Beritashvili Center of Experimental Biomedicine, Tbilisi, Georgia*

[2] *Georgian Technical University, Tbilisi, Georgia*

Abstract

Machine learning (ML), and deep learning in particular, has become a standard tool in genomics, with applications ranging from prediction of regulatory activity from DNA sequence to variant-effect prediction, single-cell analysis, and polygenic risk estimation. Reported performance, however, depends strongly on evaluation design, and several independent benchmarks indicate that headline results do not always generalize to the tasks that matter biologically or clinically. This review addresses three questions: (i) which model classes are applied to which genomic tasks; (ii) how evaluation practices influence reported performance and where they risk overestimation; and (iii) which factors currently limit translation into biological and clinical use. We organize the literature by task, summarize evidence of performance limits, and illustrate the effect of sample dependence on validation using a controlled simulation. We conclude with a minimum reporting checklist and identify open problems: leakage-resistant benchmarks, evaluation on variant – and individual-level tasks, comparison with tuned simple baselines, ancestry-aware validation, and interpretable, privacy-preserving modeling.

KEYWORDS: machine learning; deep learning; genomics; validation; data leakage; variant-effect prediction; single-cell; polygenic risk

Introduction

In genomics, data generation is no longer the rate-limiting step; interpretation is [1]. Machine-learning methods, particularly deep neural networks, are now applied across the field to predict regulatory activity from sequence, prioritize genetic variants, model cell states, and estimate disease risk [2,3]. Their appeal lies in the ability to learn features directly from raw data and to capture nonlinear and long-range dependencies that hand-engineered features miss.

Genomic data, however, have properties that make standard ML practice hazardous. Dimensionality often exceeds sample size by orders of magnitude. Observations are not independent: adjacent genomic windows share sequence context, paralogous sequences and repeat families recur across the genome, and individuals share haplotypes through common ancestry. Technical variation and population structure can confound biological signal, and labels are frequently noisy, sparse, or biased. As a consequence, reported performance depends heavily on how models are evaluated, and recurrent pitfalls have been catalogued [4].

Existing reviews have mainly described architectures and applications [2,3]. Here we adopt a critical, evaluation-centered perspective and address three questions: (i) which model classes are applied to which genomic tasks; (ii) how do evaluation practices influence reported performance, and where do they risk overestimation; and (iii) which factors currently limit translation into biological and clinical use? Beyond synthesizing the literature, we provide a task-to-model map (Fig. 1, Table 1), a summary of independent evidence on performance limits (Table 2), a controlled simulation illustrating the effect of the data-splitting strategy (Fig. 2 and Fig. 3), and a minimum validation checklist (Table 3).

Scope and Literature Search

This review covers supervised and self-supervised learning on nucleic-acid-derived data in human and model-organism genomics, specifically sequence-based prediction of regulatory activity, variant calling and interpretation, transcriptomic and single-cell modeling, and polygenic prediction. Protein structure prediction, medical imaging, and models based solely on clinical records are outside the scope.

[Literature search strategy: databases searched (e.g., PubMed, Scopus, Web of Science), date of the last search, search strings, inclusion and exclusion criteria, and the number of records screened and included. To be completed by the authors.]

Model Classes and Genomic Tasks

Fig. 1. maps the principal input data types to the model classes and tasks discussed below, and Table 1 summarizes representative models and their main limitations.

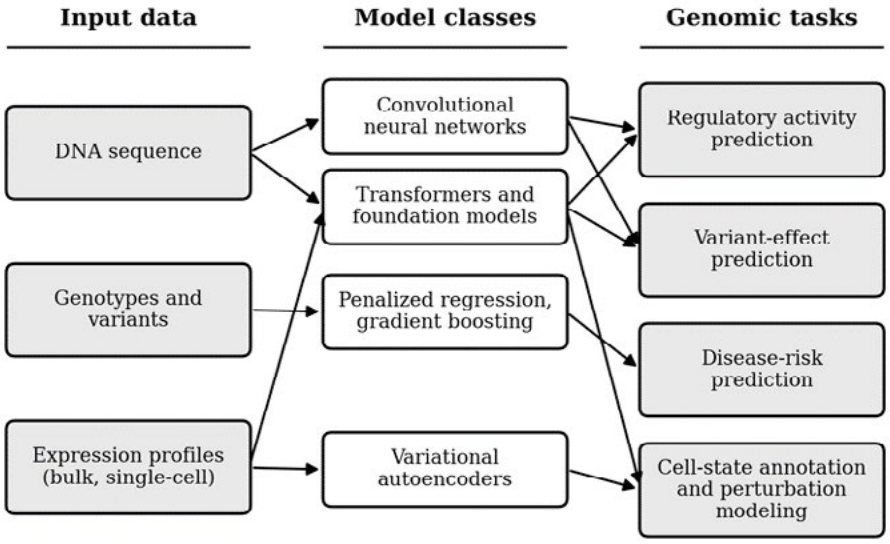


Fig. 1. Relationship between input data, model classes, and genomic tasks. Arrows indicate common pairings and are not exhaustive.

1. Regulatory Genomics from DNA Sequence

Convolutional neural networks (CNNs) were among the first deep architectures applied to genomic sequence. DeepBind predicted the sequence specificities of DNA – and RNA-binding proteins, DeepSEA predicted chromatin features and the effects of noncoding variants directly from sequence, and Basset modeled chromatin accessibility across cell types [5-7]. Subsequent work extended the receptive field: Enformer combined convolutional layers with transformer attention to integrate interactions over long genomic distances (on the order of 100 kb) when predicting expression and chromatin tracks [8]. More recently, self-supervised pretraining on genomic sequence, analogous to language modeling, has produced DNA language models such as DNABERT and the Nucleotide Transformer which are subsequently fine-tuned for downstream tasks [9,10]. A central caveat, examined in Section 4, is that accurate prediction of genome-wide regulatory profiles across loci does not imply accurate prediction of the effects of individual variants.

2. Variant Calling and Variant-Effect Prediction

DeepVariant recast variant calling as image classification on read pileups using a CNN [11]. For variant interpretation, integrative scores such as CADD combined heterogeneous annotations in a linear model [12]; later approaches learned directly from se-

quence, as in SpliceAI, which predicts splice-altering effects of variants from pre-mRNA sequence using a deep residual network [13]. For missense variants, AlphaMissense adapted a protein structure prediction architecture to estimate pathogenicity across the proteome [14]. A recurring difficulty is the nature of the labeled data: clinically classified variants are few, unevenly distributed across genes, and often classified using evidence related to the features used by predictors, which creates circularity between training and benchmarking [15].

3. Single-Cell and Transcriptomic Modeling

Single-cell RNA sequencing yields sparse, high-dimensional count matrices. Probabilistic deep generative models such as scVI use variational autoencoders with count-based likelihoods to model technical noise, batch, and latent cell state [16]. Following the transformer paradigm, foundation models pretrained on tens of millions of cells, such as Geneformer and scGPT, aim to transfer knowledge to tasks including cell-type annotation, gene-network inference, and prediction of perturbation responses [17,18]. Whether pretraining provides gains over simpler approaches on these downstream tasks remains under active evaluation.

4. Polygenic Prediction

Polygenic scores aggregate the effects of many variants, typically estimated by penalized or Bayesian regression on genome-wide association data, to estimate disease risk. Genome-wide scores for several common diseases were shown to identify individuals whose risk approaches that conferred by rare monogenic mutations [19]. Although deep learning has been explored for genotype-phenotype prediction, linear models remain strong baselines in many settings, because common-variant effects are largely additive and sample sizes are small relative to the number of variants.

TABLE 1. PRINCIPAL MACHINE-LEARNING APPROACHES IN GENOMICS, ORGANIZED BY TASK

Task	Input → output	Representative models	Principal limitations
Regulatory activity prediction	DNA sequence → chromatin accessibility, factor binding, expression	DeepBind, DeepSEA, Basset, Enformer [5-8]	Weak prediction of variant effects at the individual level; dependence on cell types present in training data
Self-supervised sequence models	Unlabeled genomes → representations for downstream tasks	DNABERT, Nucleotide Transformer [9,10]	Benefit over supervised baselines must be established task by task; high computational cost
Variant calling	Aligned reads → genotypes	DeepVariant [11]	Performance depends on sequencing platform and training data

Task	Input → output	Representative models	Principal limitations
Variant-effect prediction	Sequence and annotations → pathogenicity or splicing effect	CADD, SpliceAI, AlphaMissense [12-14]	Sparse and biased labels; circularity between training and benchmark data [15]
Single-cell modeling	Count matrices → latent states, annotations, perturbation effects	scVI, Geneformer, scGPT [16-18]	Batch effects; advantage over simple baselines not consistently shown
Polygenic prediction	Genotypes → disease risk	Penalized and Bayesian regression [19]	Reduced accuracy in ancestry groups underrepresented in training data [26]

Evaluation and Its Pitfalls

1. Sample Dependence and Data Leakage

Standard ML validation assumes that training and test examples are independent draws from the same distribution. In genomics this assumption is routinely violated. Adjacent genomic windows share sequence context and regulatory elements, paralogous genes share sequence, and related individuals share haplotypes. If such related observations are divided at random between training and test sets, models can exploit local or family-specific signatures instead of learning generalizable rules, and test performance is inflated [4]. Chromosome-level hold-out has therefore become common practice in sequence-to-function modeling (Fig. 2), although it does not remove homology between chromosomes [20].

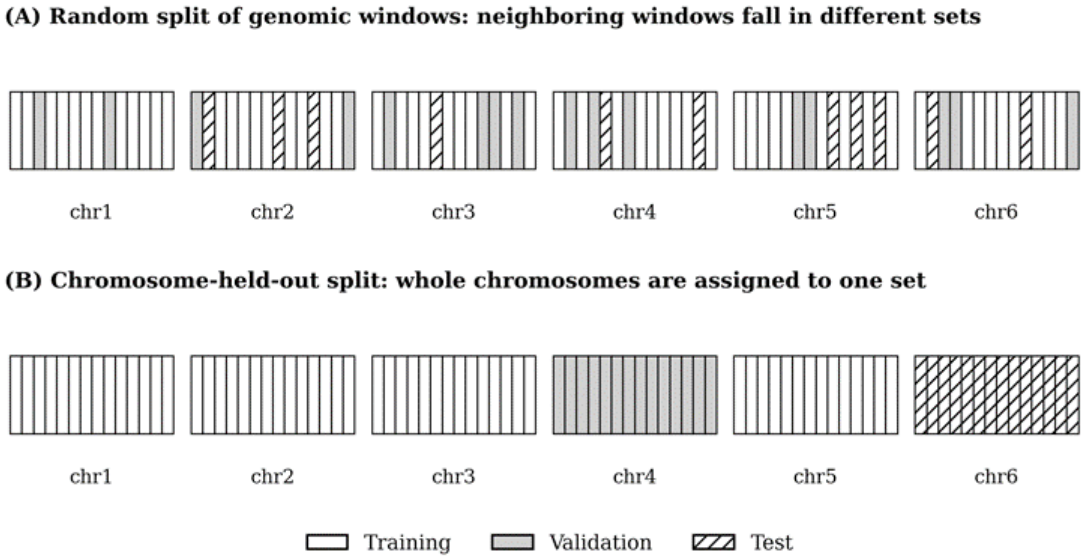


Fig. 2. Data-splitting strategies for genomic windows. (A) In a random split, neighboring windows are distributed across training, validation, and test sets, allowing information to leak between them. (B) In a chromosome-held-out split, entire chromosomes are assigned to a single set.

To illustrate the magnitude of the effect in a controlled setting, we simulated 10 “chromosomes” of 300 windows each, with 20 features. Each feature vector was the sum of a chromosome-specific signature (representing locus, batch, or local sequence context), a weak label-related signal in three features, and random noise; label prevalence differed across chromosomes. A random forest classifier evaluated with random five-fold cross-validation reached a mean area under the receiver operating characteristic curve (AUROC) of 0.70 (SD 0.02; 15 replicate simulations), because it could learn chromosome-specific signatures that predict label prevalence. When entire chromosomes were held out, mean AUROC fell to 0.54 (SD 0.07), close to chance, reflecting the weak signal that actually generalizes (Fig. 3). The simulation is deliberately simple and does not represent any specific published model; its purpose is to show that the gap can be large even when the algorithm and data are identical and only the splitting rule differs.

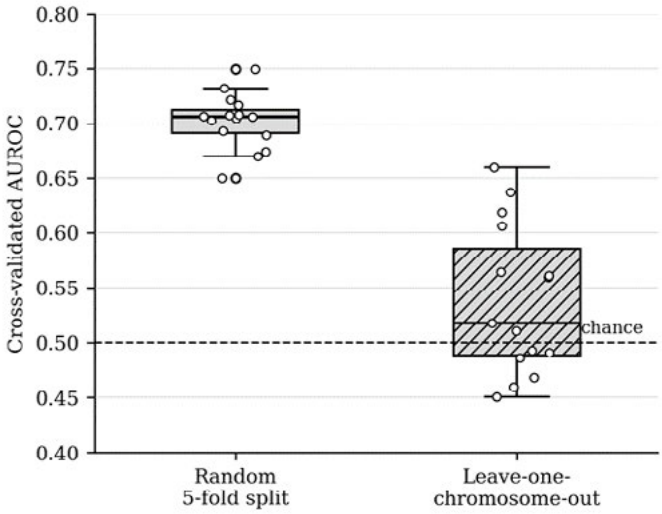


Fig. 3. Effect of the splitting strategy on apparent performance in a simulation (15 replicates; points are individual replicates, boxes show the interquartile range). The dashed line indicates chance performance (AUROC=0.5).

2. Confounding, Bath Effects, and Preprocessing Leakage

High-throughput data are affected by technical variation arising from platform, laboratory, reagent lot, and processing date, which can exceed the biological effects of interest [21]. A model trained on data in which biology is confounded with batch may learn batch. Population structure plays an analogous role in genotype-based prediction. Leakage also arises from preprocessing: normalization, feature selection, or hyperparameter tuning performed on the full dataset before splitting allows information from test data to influence training [4]. Mitigation includes designing splits that hold out entire batches or cohorts, fitting all preprocessing within training folds, and reporting performance stratified by cohort.

3. Performance Limits Reported by Independent Benchmarks

Independent benchmarks have documented gaps between headline performance and performance on biologically or clinically relevant tasks (Table 2). Two studies evaluated sequence-to-expression models on paired personal genomes and transcriptomes and found that current models, including Enformer, explain little of the variation in gene expression between individuals and often predict the wrong direction of effect of variants [22,23]. A comparison of deep-learning models for predicting the effects of gene perturbation, including single-cell foundation models, found that none outperformed simple linear baselines [24]. These findings do not imply that the models are without value: they perform well on the tasks for which they were trained, such as predicting genome-wide regulatory profiles across loci. They indicate that transfer to a different task, such as predicting the effect of a single variant in an individual, cannot be assumed and must be demonstrated.

TABLE 2. SELECTED STUDIES REPORTING LIMITS OF CURRENT MACHINE-LEARNING MODELS IN GENOMICS

Study	Task evaluated	Principal finding
Sasse et al. 2023 [22]; Huang et al. 2023 [23]	Prediction of between-individual gene expression from personal genomes	Current sequence-to-expression models, including Enformer, explained little of the variation in expression between individuals, and the predicted direction of variant effects was often incorrect
Ahlmann-Eltze et al. 2025 [24]	Prediction of transcriptional responses to genetic perturbation	Deep-learning models, including single-cell foundation models, did not outperform simple linear baselines
Martin et al. 2019 [26]	Transferability of polygenic risk scores across ancestries	Prediction accuracy was lower in individuals of non-European ancestry, reflecting the composition of the underlying association studies
Whalen et al. 2022 [4]	Survey of evaluation practice in genomic machine learning	Recurrent pitfalls include distributional differences between training and deployment data, dependence among examples, confounding, and leaky preprocessing

4. Baselines and Metrics

Comparisons are informative only if baselines are competitive. Simple models, such as k-mer-based linear models for sequence tasks, additive or linear models for perturbation prediction, and penalized regression for genotype-based prediction, should be tuned with effort comparable to that invested in the proposed model [4,24]. Metric choice also matters. The AUROC can appear high in imbalanced problems,

such as rare regulatory elements or pathogenic variants, for which precision–recall measures and calibration are more informative. Metrics computed across loci, such as the correlation between predicted and observed expression across genes, can mask failure to explain variation across individuals within a locus [22,23].

Barriers to Translation

1. Interpretability

Attribution methods such as DeepLIFT and integrated gradients assign importance scores to input nucleotides and are widely used to derive sequence motifs and to nominate candidate causal variants [27,28]. Their outputs are, however, explanations of model behavior, not of biology: they may reflect artifacts of the training data and require experimental validation, for example by reporter assays or genome editing. For high-stakes decisions, it has been argued that inherently interpretable models should be preferred over post hoc explanations of black-box models when their performance is comparable [29].

2. Ancestry Bias and Equity

Genomic datasets used to train and evaluate models are dominated by individuals of European ancestry [25]. Polygenic scores derived from such data are less accurate in other ancestry groups, and their clinical use may therefore widen health disparities [26]. Comparable concerns apply to variant-effect predictors trained on reference data with unequal representation. Ancestry-stratified evaluation should accordingly be a reporting requirement rather than an option.

3. Privacy and Governance

Genomic data are inherently identifying: even de-identified genomic data can be linked to individuals through auxiliary information, as demonstrated by the inference of surnames from Y-chromosome markers [30]. Training ML models on human genomic data therefore raises consent and governance issues. Federated learning, in which models are trained across institutions without pooling raw data, is one approach studied for health data, but it does not by itself provide formal privacy guarantees and must be combined with additional safeguards such as differential privacy or secure aggregation [31].

Recommendations and Open Problems

The evidence reviewed above suggests a set of minimum practices for studies applying ML to genomic data (Table 3).

TABLE 3. MINIMUM VALIDATION AND REPORTING CHECKLIST FOR MACHINE-LEARNING STUDIES IN GENOMICS

Issue	Mechanism of bias	Recommended safeguard
Sample dependence	Neighboring windows, paralogs, and related individuals share information across training and test sets	Split by chromosome, locus, homology cluster, or kinship; report the splitting rule
Confounding	Models learn batch, cohort, or ancestry signals rather than biology	Hold out entire cohorts or batches; stratify results by cohort and ancestry
Preprocessing leakage	Normalization, feature selection, or tuning uses test-set information	Fit all preprocessing and tuning within training folds only
Weak baselines	Gains are overstated when comparators are untuned or absent	Include simple linear or k-mer baselines tuned with equal effort
Metric mismatch	Aggregate or threshold-free metrics hide failure on the intended task	Report precision–recall, calibration, and task-specific metrics (e.g., within-locus correlation across individuals)
No external validation	Performance reflects idiosyncrasies of a single dataset	Validate on independent cohorts and, where feasible, with experimental data

Beyond these practices, we identify the following open problems:

1. Community benchmarks with standardized, leakage-resistant splits and held-out cohorts, maintained over time.
2. Evaluation on the tasks that matter, such as variant-level and individual-level effects, rather than on proxy tasks alone.
3. Systematic comparison with strong, tuned, simple baselines.
4. Ancestry-diverse training and evaluation data, with stratified reporting.
5. Uncertainty quantification and calibration that support decision-making.
6. Integration of experimental perturbation data, such as reporter assays and genome-editing screens, for validation.
7. Privacy-preserving training methods with formal guarantees.

Conclusion

Machine learning has expanded what can be predicted from genomic data, but the reliability of reported performance depends on evaluation design at least as much as on model architecture. Sample dependence, confounding, weak baselines, and mismatched metrics can inflate results, and independent benchmarks show that success on proxy tasks does not guarantee success on individual-level or clinically relevant tasks. Progress will require leakage-resistant benchmarks, transparent reporting, ancestry-aware validation, and closer integration of computational predictions with experimental evidence.

Code availability

The code used for the simulation in Figures 2 and 3 is available at [repository URL].

```
import numpy as np, json
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import KFold, LeaveOneGroupOut
from sklearn.metrics import roc_auc_score
def run(seed, n_chr=10, n_per=300, p=20, effect=0.35):
 rng = np.random.default_rng(seed)
 chrom = np.repeat(np.arange(n_chr), n_per)
 prev = rng.uniform(0.2, 0.8, n_chr)
 y = (rng.random(n_chr * n_per) < prev[chrom]).astype(int)
 offsets = rng.normal(0, 1.0, (n_chr, p)) # chromosome-specific (locus/batch) signature
 signal = np.zeros(p); signal[:3] = effect # weak, generalizable signal
 X = offsets[chrom] + np.outer(y, signal) + rng.normal(0, 1.0, (n_chr * n_per, p))
 def cv_auc(splitter, groups=None):
 preds = np.zeros(len(y));
 for tr, te in splitter.split(X, y, groups):
 m = RandomForestClassifier(n_estimators=40, min_samples_leaf=10, n_jobs=1, random_
state=seed)
 m.fit(X[tr], y[tr]); preds[te] = m.predict_proba(X[te])[:, 1]
 return roc_auc_score(y, preds)
 return (cv_auc(KFold(5, shuffle=True, random_state=seed)),
 cv_auc(LeaveOneGroupOut(), chrom))
res = np.array([run(s) for s in range(15)])
json.dump({"random": res[:,0].tolist(), "chrom": res[:,1].tolist()}, open("sim.json","w"))
print("random mean %.3f sd %.3f" % (res[:,0].mean(), res[:,0].std()))
print("chrom mean %.3f sd %.3f" % (res[:,1].mean(), res[:,1].std()))
```

References

1. Stephens ZD, et al. Big Data: Astronomical or Genomical? PLoS Biol. 2015;13(7):e1002195.
2. Eraslan G, Avsec Ž, Gagneur J, Theis FJ. Deep learning: new computational modelling techniques for genomics. Nat Rev Genet. 2019;20:389-403.
3. Zou J, Huss M, Abid A, Mohammadi P, Torkamani A, Telenti A. A primer on deep learning in genomics. Nat Genet. 2019;51:12-18.
4. Whalen S, Schreiber J, Noble WS, Pollard KS. Navigating the pitfalls of applying machine learning in genomics. Nat Rev Genet. 2022;23:169-181.
5. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA – and RNA-binding proteins by deep learning. Nat Biotechnol. 2015;33:831-838.
6. Zhou J, Troyanskaya OG. Predicting effects of noncoding variants with deep learning–based sequence model. Nat Methods. 2015;12:931-934.
7. Kelley DR, Snoek J, Rinn JL. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. Genome Res. 2016;26:990-999.
8. Avsec Ž, et al. Effective gene expression prediction from sequence by integrating long-range interactions. Nat Methods. 2021;18:1196-1203.
9. Ji Y, Zhou Z, Liu H, Davuluri RV. DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. Bioinformatics. 2021;37:2112-2120.
10. Dalla-Torre H, et al. Nucleotide Transformer: building and evaluating robust foundation models for human genomics. Nat Methods. 2025;22:287-297.
11. Poplin R, et al. A universal SNP and small-indel variant caller using deep neural networks. Nat Biotechnol. 2018;36:983-987.
12. Kircher M, et al. A general framework for estimating the relative pathogenicity of human genetic variants. Nat Genet. 2014;46:310-315.
13. Jaganathan K, et al. Predicting splicing from primary sequence with deep learning. Cell. 2019;176:535-548.
14. Cheng J, et al. Accurate proteome-wide missense variant effect prediction with AlphaMissense. Science. 2023;381:eadg7492.
15. Grimm DG, et al. The evaluation of tools used to predict the impact of missense variants is hindered by two types of circularity. Hum Mutat. 2015;36:513-523.
16. Lopez R, Regier J, Cole MB, Jordan MI, Yosef N. Deep gener-

ative modeling for single-cell transcriptomics. Nat Methods. 2018;15:1053-1058.
17. Theodoris CV, et al. Transfer learning enables predictions in network biology. Nature. 2023;618:616-624.
18. Cui H, et al. scGPT: toward building a foundation model for single-cell multi-omics using generative AI. Nat Methods. 2024;21:1470-1480.
19. Khera AV, et al. Genome-wide polygenic scores for common diseases identify individuals with risk equivalent to monogenic mutations. Nat Genet. 2018;50:1219-1224.
20. Kelley DR, et al. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. Genome Res. 2018;28:739-750.
21. Leek JT, et al. Tackling the widespread and critical impact of batch effects in high-throughput data. Nat Rev Genet. 2010;11:733-739.
22. Sasse A, et al. Benchmarking of deep neural networks for predicting personal gene expression from DNA sequence highlights shortcomings. Nat Genet. 2023;55:2060-2064.
23. Huang C, et al. Personal transcriptome variation is poorly explained by current genomic deep learning models. Nat Genet. 2023;55:2056-2059.
24. Ahlmann-Eltze C, Huber W, Anders S. Deep-learning-based gene perturbation effect prediction does not yet outperform simple linear baselines. Nat Methods. 2025;22:1657-1661.
25. Popejoy AB, Fullerton SM. Genomics is failing on diversity. Nature. 2016;538:161-164.
26. Martin AR, et al. Clinical use of current polygenic risk scores may exacerbate health disparities. Nat Genet. 2019;51:584-591.
27. Shrikumar A, Greenside P, Kundaje A. Learning important features through propagating activation differences. Proc Mach Learn Res (ICML). 2017;70:3145-3153.
28. M, Taly A, Yan Q. Axiomatic attribution for deep networks. Proc Mach Learn Res (ICML). 2017;70:3319-3328.
29. C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat Mach Intell. 2019;1:206-215.
30. M, McGuire AL, Golan D, Halperin E, Erlich Y. Identifying personal genomes by surname inference. Science. 2013;339:321-324.
31. N, et al. The future of digital health with federated learning. npj Digit Med. 2020;3:119.