nature methods

Brief Communication

https://doi.org/10.1038/s41592-025-02868-z

gReLU: a comprehensive framework for DNA sequence modeling and design

Received: 17 October 2024

Accepted: 17 September 2025

Published online: 15 October 2025

Check for updates

Avantika Lal[1], Laura Gunsalus[1,2], Surag Nair[1,2], Tommaso Biancalani[1] & Gokcen Eraslan[1] ✉

Deep learning models trained on DNA sequences can predict cell-type-specific regulatory activity, reveal *cis*-regulatory grammar, prioritize genetic variants and design synthetic DNA. However, building and interpreting these models correctly remains difficult, and models and software built by different groups are often not interoperable. Here we present gReLU, a comprehensive software framework that enables advanced sequence modeling pipelines, including data preprocessing, modeling, evaluation, interpretation, variant effect prediction and regulatory element design.

Deep learning models trained on DNA sequences along with functional data are capable of learning the *cis*-regulatory code across biological contexts[1,2]. These models can be queried to perform in silico experiments, such as prioritizing functional noncoding variants[3], conducting in silico genome engineering[4] and designing synthetic regulatory elements[5,6]. Furthermore, interpreting these models may reveal novel *cis*-regulatory mechanisms[7].

However, such models are difficult to train, and minor errors can result in misleading predictions[8]. In addition, the field is hampered by a lack of interoperability between tools. Instead of a common underlying framework being adapted to create new models, new models are often accompanied by custom code for data processing, training and evaluation. This makes it difficult to combine and compare models, or fine-tune them for novel tasks. Furthermore, separate tools have been built for individual tasks such as sequence design[9], or model interpretation[10], making it difficult to chain these into a workflow.

While unifying software frameworks have been proposed[11–14], these remain limited in scope, being designed largely around convolutional models that produce scalar-valued predictions for short sequences. They lack support for modern transformer architectures, long-context profile models[15,16], advanced interpretation methods and comprehensive workflows, particularly those related to sequence design[6,9]. Here, we present gReLU, an open-source Python framework that unifies diverse sequence models and downstream tasks under one umbrella, minimizing the need for custom code and switching between incompatible tools. Its main features are summarized below (Fig. 1) and detailed in Supplementary Note 1.

1. Data input: gReLU accepts DNA sequences or genomic coordinates along with functional data in standard formats. Given genomic coordinates, it can load the corresponding sequences and annotations from public databases.
2. Data processing: gReLU's preprocessing functions include filtering sequences, matching genomic regions with similar sequence content, calculating sequencing coverage and splitting datasets for training, validation and testing. gReLU provides PyTorch dataset classes to load data from each input format, which support batching, data augmentation, normalization and transformation.
3. Model design: gReLU provides customizable architectures ranging from small convolutional models to large transformer models[15–17]. Users can change parameters such as the number of layers, number and width of filters, and activation functions.
4. Model training: gReLU trains models to perform single- or multitask regression, single- or multitask binary classification, segmentation or multiclass classification. Suitable loss functions[7,16,18] are provided for each task (Supplementary Note 1). Class or example weighting allows users to emphasize subsets of examples. Training is performed using PyTorch Lightning, and logging and hyperparameter sweeps are enabled with Weights & Biases. Finally, gReLU saves model checkpoints that include comprehensive metadata, ensuring reproducibility and allowing all relevant information to be distributed in a single file.
5. Inference and evaluation: gReLU evaluates models on held-out data using appropriate metrics (Supplementary Note 1). For profile models, gReLU returns predictions along with the corresponding genomic coordinates, accounting for cropping and

[1]Biology Research | AI Development, gRED Computational Sciences, Genentech, South San Francisco, CA, USA. [2]These authors contributed equally: Laura Gunsalus, Surag Nair. ✉e-mail: eraslan.gokcen@gene.com

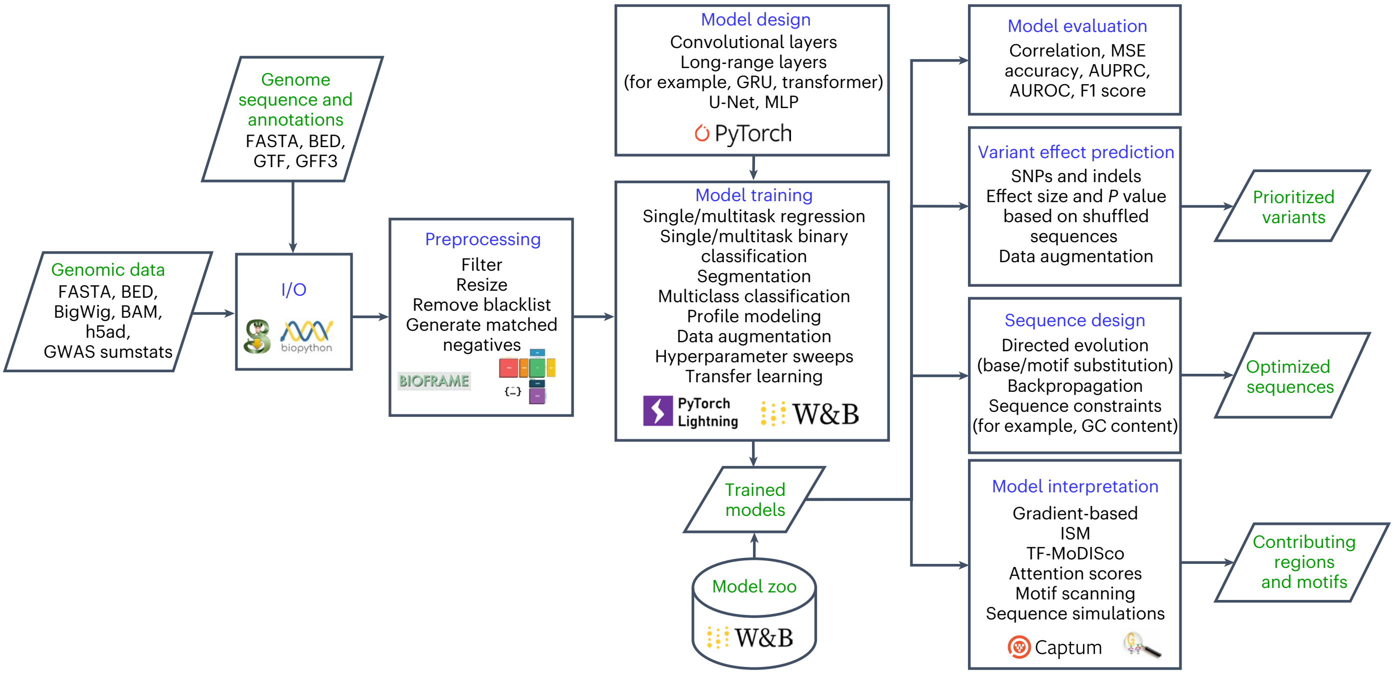


Fig. 1 | An overview of the gReLU package. A flowchart showing the main modules and features of gReLU. MSE, mean squared error; AUROC, area under the receiver operator characteristic; MLP, multilayer perceptron; I/O, input/output.

pooling performed by the model. It also enables data augmentation to improve robustness during inference.

6. Sequence interpretation: gReLU can score the importance of each base in an input sequence using in silico mutagenesis (ISM), DeepLift/SHAP or gradient-based[9] methods. Unlike previous frameworks (Supplementary Table 1), it can also annotate important regions by scanning them with position weight matrices (PWMs). Finally, it can derive learned motifs with TF-MoDISco[10] and match the results to known motifs.
7. Model interpretation: gReLU can generate synthetic DNA sequences in which motifs are shuffled, repositioned or inserted into random backgrounds at varying positions, orientations or spacing (Supplementary Note 1). Computing predictions on such sequences can reveal the regulatory grammar learned by the model[7,19,20]. For transformer models, gReLU visualizes attention matrices, which can highlight distal enhancer–gene interactions[15].
8. Variant effect prediction: Given genetic variants in tabular format, gReLU extracts sequences surrounding the reference and alternate alleles, and performs inference on them using any trained model. It computes an effect size for each variant by comparing the results for the two alleles. While this functionality already exists (Supplementary Table 1), gReLU provides additional robustness using data augmentation and statistical testing (Supplementary Note 2), along with improved interpretability via PWM scanning to identify motifs that are created or disrupted by a variant.
9. Sequence design: gReLU enables model-driven design of DNA using directed evolution or gradient-based approaches. Users can define the design objective, constrain which positions to edit and encourage or discourage specific patterns (for example, CpGs or motifs).
10. Prediction transform layers: Previous implementations of tasks such as interpretation, variant effect prediction and directed evolution were designed for short-context, single-task models that produce scalar outputs. These tasks become more complicated with multitask, long-context and/or profile models. gReLU introduces prediction transform layers, flexible layers that can be appended to such models to modify their output—for example, to compute the difference in predictions between two cell types, or the ratio of predictions over introns versus exons of a gene. This enables interpretation, variant effect prediction and design with reference to any derived function of the model's output.
11. Model zoo: gReLU includes a freely available model zoo hosted on Weights & Biases, which contains widely applicable models[6,21,22] including Enformer[15] and Borzoi[16] (Supplementary Note 1). It stores model checkpoints alongside code, datasets and logs documenting their creation. gReLU includes functions to programmatically search the zoo and download any model.

To illustrate how gReLU can nominate regulatory variants, we trained a regression model to predict DNase I hypersensitive site sequencing (DNase-seq) signal in GM12878 cells (Fig. 2a and Supplementary Table 2). Using this model, we predicted the effects of 28,274 single-nucleotide variants, of which ~2% were dsQTLs identified in lymphoblastoid cell lines[23,24]. The model classified DNase-seq quantitative trait loci (dsQTLs) with an area under the precision–recall curve (AUPRC) of 0.27, outperforming both a random predictor and a published gkmSVM model[24] (Fig. 2b).

Leveraging gReLU's model zoo and prediction transform layers, we benchmarked Enformer[7] on the same variants (Fig. 2b) and obtained a higher AUPRC of 0.60, probably due to its long input length, profile modeling and multispecies training. For both models, augmenting sequences with reverse complementation during inference increased the AUPRC (Supplementary Table 3).

To analyze the regulatory mechanisms underlying variant effects, we used gReLU to compute saliency scores surrounding both alleles of each variant and run TF-MoDISco[10] (Supplementary Fig. 1a). dsQTLs were significantly (Fisher's exact test, OR = 20, *P* value < 2.2×10^{-16}) more likely than control single-nucleotide variants to overlap a TF-MoDISco identified motif, suggesting that dsQTLs tend to alter transcription factor (TF) binding motifs (Supplementary Fig. 1b). For example, gReLU's

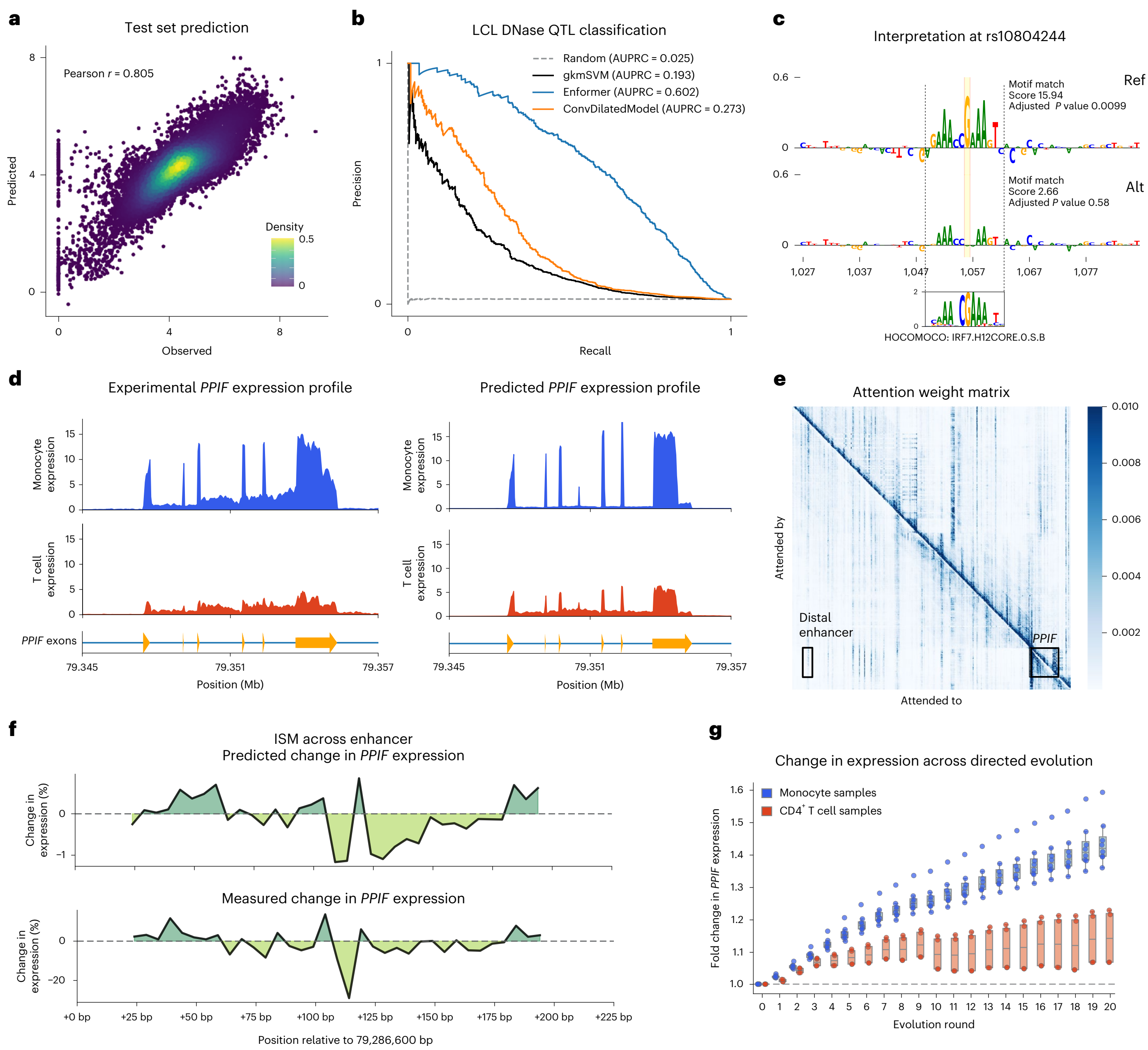


Fig. 2 | Example analyses with gReLU. a, Scatter plot showing the predictions of a convolutional regression model trained on DNase-seq in GM12878 cells using gReLU. Predictions were made on all 22,595 genomic regions in the model's test set. Each point is colored based on the number of neighboring points ('Density'). **b**, Precision–recall curve for scoring 28,274 single-nucleotide variants out of which 574 are dsQTLs in lymphoblastoid cell lines using the regression model and other baselines. **c**, Nucleotide-resolution importance scores obtained from the regression model for the reference (Ref) and alternate (Alt) alleles of rs10804244 using the saliency method highlight a disrupted IRF motif. The variant is highlighted in yellow. A HOCOMOCO PWM matching the motif is shown, along with the match score and *P* values for the Ref and Alt sequences. *P* values were calculated based on a one-sided test of the log-likelihood ratio score against a null hypothesis of a zero-order background model and adjusted for multiple testing using the false discovery rate method. **d**, Left: mean expression profile of *PPIF* across eight independent RNA-seq CD14$^+$ monocyte tracks (blue) and four independent RNA-seq CD4$^+$ T cell tracks (red). Right: Borzoi-predicted mean *PPIF* expression profile for the same 12 tracks. **e**, Averaged attention weight matrix from all heads of the final transformer layer of Borzoi, showing strong attention between the *PPIF* gene and enhancer. **f**, Functional analysis of the distal enhancer. Mean change in predicted *PPIF* expression induced by 5-bp edits (top) and experimental measurements from Variant-FlowFISH in K562 cells (bottom; Martyn et al.[4]). **g**, Predicted changes in monocyte and T cell expression over 20 rounds of directed evolution, optimizing the enhancer to maximize differential expression. Points represent the predicted RNA-seq coverage over exons of *PPIF* from eight biological replicates (independent donors) for monocytes and four biological replicates for CD4$^+$ T cells. Data are presented as box plots where the center line represents the median, box bounds represent the 25th and 75th percentiles, and whiskers extend to the minimum and maximum values (outliers excluded). Individual data points are overlaid on the box plots.

motif scanning functions revealed that the rs10804244 variant weakens an interferon regulatory factor (IRF) binding site (Fig. 2c).

This example demonstrates several unique features of gReLU. First, a recent study[20] highlighted the difficulty of comparing convolutional models with long-context models such as Enformer, because Enformer makes predictions for ~100 kb of sequence at 128 bp resolution, whereas the convolutional model returns a scalar prediction for ~1 kb. gReLU streamlined this comparison by automatically generating sequences of the correct length for each model, identifying the bins in Enformer's output corresponding to the output of the convolutional model, and subsetting and aggregating Enformer's predictions over those bins before computing variant effects. Second, gReLU's motif analysis functions enabled comparison of the model's predictions to known biology. Finally, gReLU's data augmentation functionality, the most comprehensive among current frameworks (Supplementary Table 1), improved performance during both training and inference.

Unlike previous frameworks (Supplementary Table 1), gReLU enables systematic interpretation and sequence design not only with small single-task models but also with multitask, long-context and profile models. To demonstrate, we applied the Borzoi model[16] from the model zoo to the *PPIF* gene in humans. Using gReLU, we visualized the predicted RNA sequencing (RNA-seq) coverage and found that it closely mirrored ground-truth data, showing higher *PPIF* expression in monocytes than in T cells (Fig. 2d).

Visualizing the model's attention matrix identified strong attention between *PPIF* and a known enhancer 61.7 kb upstream of the transcription start site[4] (Fig. 2e). Using gReLU's sequence manipulation tools, we simulated 5-bp tiled mutations across the enhancer and predicted their effect on *PPIF* expression. The results (Fig. 2f, top) were well correlated with experimental Variant-FlowFISH data[4] from analogous THP-1[25] and Jurkat[26] cell lines. Although their magnitude was lower, as previously observed[4] (Spearman's $\rho = 0.58$; Supplementary Fig. 2), they correctly identified a central region that is particularly sensitive to perturbation (Fig. 2f, bottom).

Using gReLU's directed evolution and prediction transform functions, we iteratively modified the enhancer, aiming to maximize the difference in *PPIF* exon coverage between monocytes and T cells. With 20 base edits to the enhancer, we achieved a predicted 41.76% increase in monocyte expression with only a 16.75% increase in predicted T cell expression (Fig. 2g). ISM and motif scanning over the evolved enhancer (Supplementary Fig. 3) revealed novel CEBP motifs (Extended Data Fig. 1a). This is consistent with experiments showing that inserting a CEBP motif in this enhancer increases *PPIF* expression in THP-1 relative to Jurkat cells, and with the differential expression of the CEBPA TF between these cells[4]. The specificity of the evolved enhancer was also validated using orthogonal models from the gReLU model zoo (Extended Data Fig. 1b, Supplementary Figs. 4–7 and Supplementary Methods)

In conclusion, gReLU enables scientists to easily train, apply, fine-tune and interpret state-of-the-art sequence models and to design novel regulatory elements with complex properties. It is open-source and includes instructions for users to make contributions and suggestions, along with detailed tutorials. However, there remain many useful features that have not yet been included in the package and would be valuable future extensions. These include efficient architectures to model longer genomic contexts[27,28], additional design algorithms[6], efficient training on larger, multispecies datasets, modeling biases in genomic assays[20] and modeling individual genomes[29].

Online content

Any methods, additional references, Nature Portfolio reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41592-025-02868-z.

References

1. Eraslan, G., Avsec, Ž., Gagneur, J. & Theis, F. J. Deep learning: new computational modelling techniques for genomics. *Nat. Rev. Genet.* **20**, 389–403 (2019).
2. Lal, A. Deciphering the regulatory syntax of genomic DNA with deep learning. *J. Biosci.* **47**, 47 (2022).
3. Trevino, A. E. et al. Chromatin and gene-regulatory dynamics of the developing human cerebral cortex at single-cell resolution. *Cell* **184**, 5053–5069 (2021).
4. Martyn, G. E. et al. Rewriting regulatory DNA to dissect and reprogram gene expression. *Cell* **188**, 3349–3366.e23 (2025).
5. Taskiran, I. I. et al. Cell-type-directed design of synthetic enhancers. *Nature* **626**, 212–220 (2024).
6. Gosai, S. J. et al. Machine-guided design of cell-type-targeting *cis*-regulatory elements. *Nature* **634**, 1211–1220 (2024).
7. Avsec, Ž et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nat. Genet.* **53**, 354–366 (2021).
8. Whalen, S., Schreiber, J., Noble, W. S. & Pollard, K. S. Navigating the pitfalls of applying machine learning in genomics. *Nat. Rev. Genet.* **23**, 169–181 (2022).
9. Schreiber, J. & Lu, Y. Y. Ledidi: designing genomic edits that induce functional activity. Preprint at *bioRxiv* https://doi.org/10.1101/2020.05.21.109686 (2020).
10. Shrikumar, A. et al. Technical note on Transcription Factor Motif Discovery from Importance Scores (TF-MoDISco) version 0.5.6.5. Preprint at https://arxiv.org/abs/1811.00416 (2018).
11. Chen, K. M., Cofer, E. M., Zhou, J. & Troyanskaya, O. G. Selene: a PyTorch-based deep learning library for sequence data. *Nat. Methods* **16**, 315–318 (2019).
12. Avsec, Ž et al. The Kipoi repository accelerates community exchange and reuse of predictive models for genomics. *Nat. Biotechnol.* **37**, 592–600 (2019).
13. Kopp, W., Monti, R., Tamburrini, A., Ohler, U. & Akalin, A. Deep learning for genomics using Janggu. *Nat. Commun.* **11**, 3488 (2020).
14. Klie, A. et al. Predictive analyses of regulatory sequences with EUGENe. *Nat. Comput Sci.* **3**, 946–956 (2023).
15. Avsec, Ž et al. Effective gene expression prediction from sequence by integrating long-range interactions. *Nat. Methods* **18**, 1196–1203 (2021).
16. Linder, J., Srivastava, D., Yuan, H., Agarwal, V. & Kelley, D. R. Predicting RNA-seq coverage from DNA sequence as a unifying model of gene regulation. *Nat. Genet.* **57**, 949–961 (2025).
17. Novakovsky, G., Fornes, O., Saraswat, M., Mostafavi, S. & Wasserman, W. W. ExplaiNN: interpretable and transparent neural networks for genomics. *Genome Biol.* **24**, 154 (2023).
18. Lal, A. et al. Decoding sequence determinants of gene expression in diverse cellular and disease states. Preprint at *bioRxiv* https://doi.org/10.1101/2024.10.09.617507 (2024).
19. Toneyan, S. & Koo, P. K. Interpreting *cis*-regulatory interactions from large-scale deep neural networks. *Nat. Genet.* **56**, 2517–2527 (2024).
20. Pampari, A. et al. ChromBPNet: bias factorized, base-resolution deep learning models of chromatin accessibility reveal *cis*-regulatory sequence syntax, transcription factor footprints and regulatory variants. Preprint at *bioRxiv* https://doi.org/10.1101/2024.12.25.630221 (2025).
21. Vu, H. & Ernst, J. Universal annotation of the human genome through integration of over a thousand epigenomic datasets. *Genome Biol.* **23**, 9 (2022).
22. Zhang, K. et al. A single-cell atlas of chromatin accessibility in the human genome. *Cell* **184**, 5985–6001 (2021).
23. Degner, J. F. et al. DNase I sensitivity QTLs are a major determinant of human expression variation. *Nature* **482**, 390–394 (2012).
24. Lee, D. et al. A method to predict the impact of regulatory variants from DNA sequence. *Nat. Genet.* **47**, 955–961 (2015).
25. Bosshart, H. & Heinzelmann, M. THP-1 cells as a model for human monocytes. *Ann. Transl. Med.* **4**, 438 (2016).
26. Carrasco-Padilla, C. et al. T cell activation and effector function in the human Jurkat T cell model. *Methods Cell Biol.* **178**, 25–41 (2023).

27. Poli, M. et al. Hyena hierarchy: towards larger convolutional language models. In Proc. *40th International Conference on Machine Learning* 28043–28078 (ICML, 2023).
28. Gu, A. & Dao, T. Mamba: linear-time sequence modeling with selective state spaces. Preprint at http://arxiv.org/abs/2312.00752 (2023).
29. Drusinsky, S., Whalen, S. & Pollard, K. S. Deep-learning prediction of gene expression from personal genomes. Preprint at *bioRxiv* https://doi.org/10.1101/2024.07.27.605449 (2024).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Methods

Processing of GM12878 DNase-seq data

From the ENCODE portal, we downloaded data corresponding to the GM12878 DNase-seq experiment ENCSR000EJD. Specifically, we downloaded the read-depth normalized bigwig (ENCFF093VXI) and the peaks narrowPeak file (ENCFF588OCA), aligned to the hg38 reference genome.

Using gReLU, we extended the peaks 250 bp on each side of the summit and merged overlapping regions. We filtered the regions to autosomes and removed regions overlapping blacklisted regions[30]. We obtained a set of GC-matched negative regions using the grelu.data.preprocess.get_gc_matched_intervals function with binwidth = 0.05. We used sequences from chromosome 10 as the validation set and sequences from chromosome 11 as the test set.

Architecture and training of the DNase-seq regression model

We used the DilatedConvModel architecture provided in gReLU, which implements a dilated convolutional architecture, similar to BPNet[7]. The following parameters were used: channels = 512, n_conv = 9. Thus, the model consists of 9 convolution layers with 512 channels each.

We created a dataset using the BigWigSeqDataset class with input length 2,114 bp and output lengths of 1,000 bp, label_aggfunc = 'sum', label_transform_func = np.log1p, augment_mode = 'random'. Thus, the labels are transformed to the log of the summed counts over the output 1,000-bp region. Therefore, the model is trained to take as input 2,114 bp of sequence and output the log of the total DNase-seq counts in the central 1,000 bp.

We performed a hyperparameter sweep using Bayesian optimization over different combinations of data augmentation parameters (rc = True or False, max_seq_shift = 0, 1 or 3, max_pair_shift = 0, 10, 50 or 100). Each model was trained using the mean squared error loss, a learning rate of 10^{-4} and batch size 512, for a total of 15 epochs. The model with the lowest validation set loss was selected as the best model, evaluated on the test set and used for variant effect prediction.

Variant effect prediction and interpretation

We downloaded the previously curated list of 574 lymphoblastoid cell line (LCL) dsQTL single-nucleotide polymorphisms (SNPs) and 27,735 control SNPs[24]. We removed regions that would be close to chromosome edges using the filter_chrom_ends function with genome = 'hg19' and pad = 98,304 (half the input length for Enformer). This resulted in 574 dsQTL SNPs and 27,700 control SNPs.

We scored the variants using grelu.variant.predict_variant_effects with compare_func = 'subtract' and genome = 'hg19', using the provided hg19 coordinates. This returns the log fold-change (LFC) of the predicted counts between the reference and the alternate alleles. We ran this step twice, setting rc = True and rc = False, respectively, to test the effect of reverse complement data augmentation. The setting rc = True causes reverse complementation to be applied, that is, for each allele-containing sequence, we predicted its activity in both orientations and averaged the results.

For gkmSVM, we used the predictions provided by the authors[24]. For Enformer, we used the model from the gReLU model zoo (project = 'enformer', model_name = 'human'). We used grelu.variant.predict_variant_effects to compute variant LFCs comparable with our regression model. To do so, we applied a prediction transform that summed predictions in the eight bins centered around the variant (total width of 128 × 8 = 1,024 bp) for the ENCFF093VXI task and log transformed the sum, along with compare_func = 'subtract'. We ran this step twice, setting rc = True and rc = False, respectively, to test the effect of reverse complement data augmentation.

We assessed performance using the sklearn.metrics.average_precision_score function with the provided labels and the absolute value of the predicted LFC. We generated the model interpretations at select loci using the grelu.interpret.score.get_attributions function with the parameters method = 'saliency', correct_grad = True, which apply the saliency method with gradient correction[31].

For a global analysis, we performed TF-MoDISco on 100-bp windows centered at the reference and alternate versions for both dsQTLs and control SNPs using grelu.interpret.modisco.run_modisco with method = 'saliency', correct_grad = True, window = 100. From the output TF-MoDISco object, we plotted the motifs. Then, we extracted the seqlets (that is, motif instances) and computed the number of SNPs overlapping any motif separately for the dsQTL and control sets. A variant was counted as overlapping if either the reference or alternate versions overlap with a motif instance.

Inference and interpretation using the Borzoi model

We used the grelu.resources.load_model function to load the Borzoi model (replicate 0) from the public gReLU model zoo. We used the following CD14$^+$ monocyte and CD4$^+$ T cell RNA-seq tracks within Borzoi:

ENCFF023YXV+ (CD14-positive monocyte female),
ENCFF946ZPT (with multiple sclerosis; CD14-positive monocyte),
ENCFF853SNW (CD14-positive monocyte),
ENCFF735XXE (CD14-positive monocyte),
ENCFF848ZVQ (with multiple sclerosis; CD14-positive monocyte),
ENCFF926QTW (CD14-positive monocyte male adult (37 years)),
ENCFF623LHV (with multiple sclerosis; CD14-positive monocyte),
ENCFF004DOF (CD14-positive monocyte),
ENCFF579IBH+ (CD4-positive, alpha-beta T cell male adult (20 years)),
ENCFF515TIF+ (CD4-positive, alpha-beta T cell male adult (20 years)),
ENCFF223MUU (CD4-positive, alpha-beta T cell male adult (37 years)),
ENCFF089BOJ (CD4-positive, alpha-beta T cell male adult (21 years)).

Gene annotations were loaded using grelu.io.genome.read_gtf and filtered to remove overlapping transcripts using grelu.data.preprocess.filter_overlapping.

To predict and visualize RNA-seq coverage across the *PPIF* gene, we used the model.predict_on_seqs function, followed by conversion from genomic intervals to output bins with model.input_intervals_to_output_bins. The resulting predicted tracks were visualized using grelu.visualize.plot_tracks.

To quantify gene expression from model predictions, we used the 'Aggregate' prediction transform to compute the average predicted RNA-seq signal across bins overlapping the exons of the *PPIF* gene. These scores were used to compare relative expression between cell types and to track changes across sequence edits.

To produce the attention map surrounding the *PPIF* gene, we used the grelu.interpret.score.get_attention_scores function with parameter 'block_idx = −1', which extracts attention weights from the model's final attention layer. We then averaged these weights across all attention heads of this layer before visualizing the matrix.

ISM, sequence design and motif discovery

We modified the distal enhancer located at chr10:79285732–79287386 (hg38), using the grelu.sequence.mutate function to substitute 5-bp windows with an alternate sequence. Model predictions were computed for each mutated sequence, and expression scores were calculated as above. The predicted expression changes were compared with experimental results from Variant-FlowFISH assays in K562 cells, which were obtained from supplementary table 8 of Martyn et al.[4], and coordinates were converted from hg19 to hg38 using LiftOver[32].

We computed ISM scores using the grelu.interpret.score.ISM_predict function, comparing predicted expression between the reference sequence and versions with single-base substitutions. These were visualized as $\log_2$ fold-change heatmaps using grelu.visualize.plot_ISM.

To design sequences with enhanced cell-type specificity, we defined a prediction transform using the 'Specificity' class to compute the difference in predicted RNA-seq expression between monocytes and T cells, averaged across bins overlapping the *PPIF* exons. This metric served as the objective for optimization. We then used the grelu.design.

evolve function to iteratively modify the enhancer sequence. At each round, a single base was mutated to improve the specificity objective.

To investigate the mechanisms driving the evolved activity, we scanned the original and evolved enhancer for known motifs using grelu.interpret.motifs.scan_sequences. Motifs were extracted from the HOCOMOCO v12 database[33]. Comparative analysis with the original sequence was performed using grelu.interpret.motifs.compare_motifs.

Validation of the edited *PPIF* enhancer using orthogonal models

To validate the design using orthogonal models, we used a chromatin accessibility model (human-atac-catlas) from the gReLU model zoo. This is a multitask binary classification model, trained on binarized pseudobulk single-cell assay for transposase-accessible chromatin using sequencing (ATAC-seq) data from 203 human cell types[22]. Given a 200-bp input sequence, it predicts the probability of accessibility of the sequence in each cell type. We applied this model to the original and evolved *PPIF* enhancer sequences. Predictions are shown for both monocyte-like (Macrophage General, Macrophage Gen or Alv, Fetal Macrophage Placental, Fetal Macrophage Hepatic 1, Fetal Macrophage General 1, Fetal Macrophage General 2, Fetal Macrophage General 3, Fetal Macrophage Hepatic 2, Fetal Macrophage Hepatic 3 and Fetal Macrophage General 4) and T cell-like (T Lymphocyte 1 (CD8$^+$), T lymphocyte 2 (CD4$^+$), Natural Killer T, Naive T, Fetal T Lymphocyte 1 (CD4$^+$), Fetal T Lymphocyte 2 (Cytotoxic) and Fetal T Lymphocyte 3 (IL2$^+$)) tracks. We further validated that these tracks are similar to the THP-1 and Jurkat cell lines used by Martyn et al. (Supplementary Fig. 4 and Supplementary Methods).

We additionally validated the cell type specificity of the enhancer using the following DNAse tracks with the Borzoi model: ENCFF678LXL (CD14-positive monocyte female),

ENCFF659BVQ (CD14-positive monocyte female),
ENCFF724HAH (CD14-positive monocyte male adult (21 years)),
ENCFF722MXM (CD14-positive monocyte male adult (37 years)),
ENCFF759CKK (CD14-positive monocyte female adult (34 years)),
ENCFF522NPH (CD4-positive, alpha-beta T cell male adult (37 years)),
ENCFF133LMW (CD4-positive, alpha-beta T cell male adult (37 years)),
ENCFF658FXQ (CD4-positive, alpha-beta T cell male adult (21 years)),
ENCFF502EPK (CD4-positive, alpha-beta T cell female adult (33 years)),
ENCFF131RGB (CD4-positive, alpha-beta T cell male adult (21 years)).

We used the same ten DNAse tracks in the Enformer model. In these cases, grelu.transforms.prediction_transforms.Aggregate and score.compute were used to compute accessibility across the distal enhancer.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All data used in this study are publicly available at the following locations: ENCODE ENCFF093VXI (GM12878 DNase-seq read depth normalized bigwig) at https://www.encodeproject.org/files/ENCFF093VXI/; ENCODE ENCFF588OCA (GM12878 DNase-seq peaks) at https://www.encodeproject.org/files/ENCFF588OCA/; ENCODE ENCSR636HFF (hg38 blacklist) at https://www.encodeproject.org/annotations/ENCSR636HFF/; list of dsQTL and control SNPs at https://static-content.springer.com/esm/art%3A10.1038%2Fng.3331/MediaObjects/41588_2015_BFng3331_MOESM26_ESM.xlsx; JASPAR 2024 vertebrate nonredundant position frequency matrices at https://jaspar.elixir.no/download/data/2024/CORE/JASPAR2024_CORE_vertebrates_non-redundant_pfms_jaspar.txt; HOCOMOCO v12 PWMs at https://hocomoco12.autosome.org/downloads_v12; and Martyn et al. Variant-FlowFISH data (Supplementary Table 8) at https://www.biorxiv.org/content/biorxiv/early/2023/12/21/2023.12.20.572268/DC1/embed/media-1.xlsx.

Code availability

gReLU is available under an MIT license via GitHub at https://github.com/Genentech/gReLU and via Zenodo at https://doi.org/10.5281/zenodo.15627612 (ref. 34). Documentation is available at https://genentech.github.io/gReLU/. Tutorials are available via GitHub at https://github.com/Genentech/gReLU/tree/main/docs/tutorials. The model zoo is available at https://wandb.ai/grelu/ and includes all the models described in this Brief Communication. All models are also available via Zenodo at https://doi.org/10.5281/zenodo.15603450 (ref. 35). Code used to perform the analyses in this Brief Communication is available via GitHub at https://github.com/Genentech/gReLU-applications and via Zenodo at https://doi.org/10.5281/zenodo.15603450 (ref. 35). All experiments were run on a single NVIDIA DGX-1 using a single NVIDIA A100 graphics processing unit, with Python v3.11.8, PyTorch v2.2.2, PyTorch Lightning v2.2.4, CUDA v12.1.0 and gReLU v1.0.2.

References

30. Amemiya, H. M., Kundaje, A. & Boyle, A. P. The ENCODE Blacklist: identification of problematic regions of the genome. *Sci. Rep.* **9**, 9354 (2019).
31. Majdandzic, A., Rajesh, C. & Koo, P. K. Correcting gradient-based interpretations of deep neural networks for genomics. *Genome Biol.* **24**, 109 (2023).
32. Hinrichs, A. S. et al. The UCSC genome browser database: update 2006. *Nucleic Acids Res.* **34**, D590–D598 (2006).
33. Vorontsov, I. E. et al. HOCOMOCO in 2024: a rebuild of the curated collection of binding models for human and mouse transcription factors. *Nucleic Acids Res.* **52**, D154–D163 (2024).
34. Lal, A. et al. Genentech/gReLU: v1.0.6. *Zenodo* https://doi.org/10.5281/zenodo.15627612 (2025).
35. Nair, S. Genentech/gReLU-applications: v1.0. *Zenodo* https://doi.org/10.5281/zenodo.15603450 (2025).

Acknowledgements

We thank D. Garfield, O. Fornes, K. Fletez-Brant, M. Hasan Celik, K. Gjoni, C. de Donno, J. Kancherla, J.-P. Fortin and J. Kageyama for their advice, feedback and contributions to the package.

Author contributions

A.L. and G.E. built the package. L.G. and S.N. performed analyses. T.B. provided mentorship and supervision. A.L., G.E., L.G. and S.N. wrote the paper. All authors read and approved the paper.

Competing interests

All authors are employees of Genentech, Inc.

Additional information

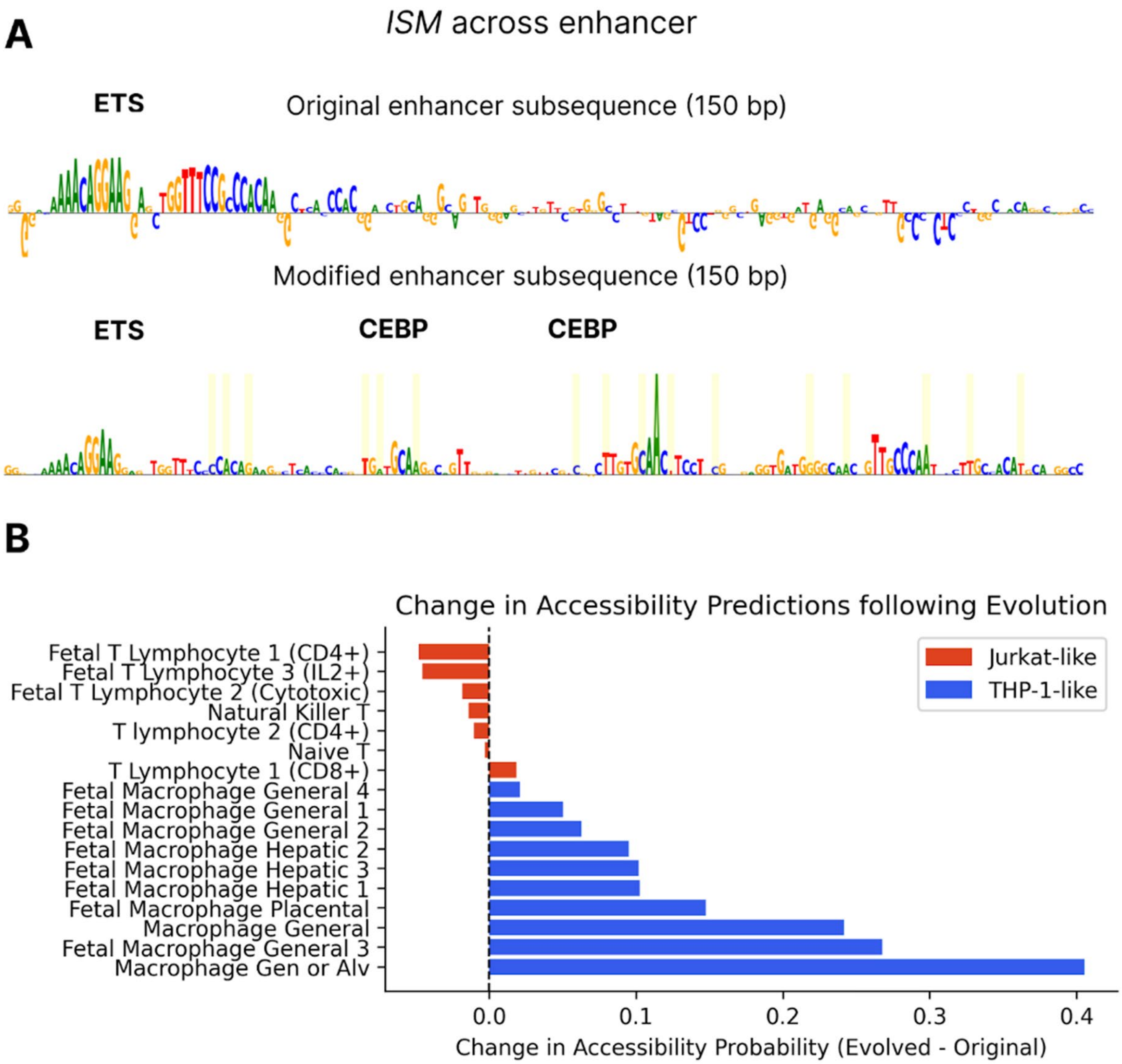
Extended data is available for this paper at https://doi.org/10.1038/s41592-025-02868-z.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41592-025-02868-z.

Correspondence and requests for materials should be addressed to Gokcen Eraslan.

Peer review information *Nature Methods* thanks Alexander Sasse and the other, anonymous reviewer(s) for their contribution to the peer review of this work. Primary Handling Editor: Lin Tang, in collaboration with the *Nature Methods* team. Peer reviewer reports are available.

Reprints and permissions information is available at www.nature.com/reprints.



Extended Data Fig. 1 | Interpretation and validation of the edited *PPIF* enhancer. a) ISM sequence logos of a subsequence in the edited *PPIF* enhancer, showing the emergence of new motifs. **b**) Predicted change in the probability of chromatin accessibility across the *PPIF* enhancer in T-cell/Jurkat-like and Monocyte/THP-1 like tracks using an orthogonal binary classification model trained to predict ATAC-seq peaks from DNA sequence.

nature portfolio

Corresponding author(s): Gokcen Eraslan

Last updated by author(s): Jun 15, 2025

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
□	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
□	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
□	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	□	A description of all covariates tested
□	☒	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☒	□	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
□	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	□	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	□	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
□	☒	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	No software was used for data collection.
Data analysis	gReLU is available under an MIT license at https://github.com/Genentech/gReLU and on Zenodo 34 . Documentation is available at https://genentech.github.io/gReLU/. Tutorials are available at https://github.com/Genentech/gReLU/tree/main/docs/tutorials. The model zoo is available at https://wandb.ai/grelu/ and includes the models described in this paper. All models are also available at https://zenodo.org/records/15021989. Code used to perform the analyses in this paper is available at https://github.com/Genentech/gReLU-applications and on Zenodo 35. All experiments were run on a single NVIDIA DGX-1 using a single NVIDIA A100 GPU, with Python v3.11.8, PyTorch v2.2.2, PyTorch Lightning v2.2.4, CUDA v12.1.0, and gReLU v1.0.2. 34. Lal, A. et al. Genentech/gReLU: v1.0.6. doi:10.5281/zenodo.15627612. 35. Nair, S. Genentech/gReLU-applications: v1.0. doi:10.5281/zenodo.15603450. The following additional tools were used for data analysis: LiftOver (https://genome.ucsc.edu/cgi-bin/hgLiftOver; version updated on June 15, 2021)

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

All data used in this study is publicly available at the following locations:
ENCODE ENCFF093VXI (GM12878 DNase-seq read depth normalized bigwig): https://www.encodeproject.org/files/ENCFF093VXI/
ENCODE ENCFF588OCA (GM12878 DNase-seq peaks): https://www.encodeproject.org/files/ENCFF588OCA/
ENCODE ENCSR636HFF (hg38 blacklist): https://www.encodeproject.org/annotations/ENCSR636HFF/
List of dsQTL and control SNPs: https://static-content.springer.com/esm/art%3A10.1038%2Fng.3331/MediaObjects/41588_2015_BFng3331_MOESM26_ESM.xlsx
JASPAR 2024 vertebrate non-redundant PFMs: https://jaspar.elixir.no/download/data/2024/CORE/JASPAR2024_CORE_vertebrates_non-redundant_pfms_jaspar.txt
HOCOMOCO v12 PWMs: https://hocomoco12.autosome.org/downloads_v12
Martyn et al. VariantFlowFish data (Supplemental Table 8): https://www.biorxiv.org/content/biorxiv/early/2023/12/21/2023.12.20.572268/DC1/embed/media-1.xlsx

Research involving human participants, their data, or biological material

Policy information about studies with human participants or human data. See also policy information about sex, gender (identity/presentation), and sexual orientation and race, ethnicity and racism.

Reporting on sex and gender	N/A
Reporting on race, ethnicity, or other socially relevant groupings	N/A
Population characteristics	N/A
Recruitment	N/A
Ethics oversight	N/A

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	No sample size calculation was performed for this study as this is a purely computational study and we did not collect any samples. All our analyses were perfomed on published datasets collected in other studies and we used all available data points (dsQTLs from Degner et al. 2012, training datasets for Borzoi and Enformer curated by the authors of those models).
Data exclusions	No data were excluded from analyses
Replication	We reproduced our result on the cell type specificity of the evolved PPIF enhancer , which was originally evaluated using RNA-seq tracks in the Borzoi model, using DNase-seq tracks in the Borzoi model as well as two independent models (Enformer and CATlas binary accessibility)
Randomization	We obtained a set of 28,309 single-nucleotide variants whose effect on DNase-seq signal was measured by Degner et al. (2012), who also divided them into dsQTLs and non-dsQTLs. No randomization was performed as we directly used the groups defined by Degner et al. (2012) based on their experimental measurements.
Blinding	Blinding was not relevant as our study was purely computational. No samples were collected and no experiments were performed. All statistical analyses were performed computationally and so the results are unaffected by experimenter bias.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	□ Antibodies
☒	□ Eukaryotic cell lines
☒	□ Palaeontology and archaeology
☒	□ Animals and other organisms
☒	□ Clinical data
☒	□ Dual use research of concern
☒	□ Plants

Methods

n/a	Involved in the study
☒	□ ChIP-seq
☒	□ Flow cytometry
☒	□ MRI-based neuroimaging

Plants

Seed stocks	N/A
Novel plant genotypes	N/A
Authentication	N/A