Polygraph: A Software Framework for the Systematic Assessment of Synthetic Regulatory DNA Elements

Anay Gupta*[1], Avantika Lal*[2], Laura Gunsalus*[2], Tommaso Biancalani[2], Gokcen Eraslan[2]

[1]College of Computing, Georgia Institute of Technology, Atlanta, Georgia, USA
[2]Biology Research | AI Development, gRED Computational Sciences, Genentech, USA
* These authors contributed equally.

Abstract

The design of regulatory elements is pivotal in numerous therapeutic interventions, including gene and cell therapy, wherein the typical objective is to engineer DNA sequences exhibiting specific attributes like cell-type specificity and elevated expression levels. However, the systematic assessment of these constructed DNA sequences remains challenging due to the absence of robust metrics and an integrated software framework. Here, we introduce Polygraph, a Python framework for evaluating synthetic DNA sequences. Polygraph provides a variety of features to streamline the synthesis and scrutiny of regulatory elements, incorporating features like a diversity index, motif and k-mer composition, similarity to endogenous regulatory sequences, and screening with predictive and foundational models. Consequently, Polygraph stands as the first instrument for the assessment of synthetic regulatory sequences, enabling expedited progress in therapeutic applications and improving our understanding of gene regulatory mechanisms.

Introduction

Cis-regulatory DNA elements (CREs) are DNA sequences that drive and tune gene expression, in part through the selective binding of transcription factors or other regulatory molecules necessary to initiate and maintain transcription[1]. Because their activity can be targeted and cell-type specific, careful design of these elements offers a promising direction for synthetic biology and gene therapy[2–6]. Recent deep learning models trained on high-throughput sequencing data integrate the complex grammar embedded in DNA, and serve as powerful oracles to predict the activity of native DNA sequences and screen synthetically generated regulatory elements[7–11]. However, evaluating and prioritizing designed sequences remains challenging without an understanding of their properties and potential mechanisms of regulatory activity.

Recent DNA design methods span from classic search and optimization algorithms[12–14] to generative modeling[15–18] to drive expression, transcription factor binding, and other functional profiles. These approaches invite exciting open questions: What mechanisms do they exploit? Do they converge on the same regulatory grammar observed in natural sequences or do they identify diverse and novel combinations of binding motifs? How should sequences be selected from millions of computationally generated options? The prioritization of sequences is currently limited because (1) there is no integrated software package to predict and analyze the performance of CREs in various biological contexts; (2) there are no defined metrics for the comprehensive evaluation of these elements, which makes it difficult to standardize assessments across different studies or design methods; and (3) the biological underpinnings of designed sequences is poorly understood, including how CREs may behave differently in across cell types and tissues.

Here, we present the Polygraph package to perform systematic evaluation of designed DNA sequences through sequence analysis, transcription factor motif composition, predictive modeling, and language modeling. This package compares generative approaches and prioritizes sequences for experimental testing, enabling applications in cell and gene therapy. By benchmarking design algorithms

to aid in selecting good CRE candidates, Polygraph provides a path toward robust and tunable CRE engineering.

Results

The Polygraph package

Polygraph is a python package that accepts DNA sequences of any length and analyzes their properties and predicted activity to compare design methods. Polygraph also supports significance testing across groups and predictions on custom models. In this paper, we demonstrate the uses of this package on two datasets: yeast promoter design with directed evolution[8] and first-order optimization method Ledidi[13], and human enhancer design on the existing synthetic sequences from Gosai et al.[11] where the authors used FastSeqProp[12], simulated annealing[19], and AdaLead[14]. Here, we first describe the functionalities of the package.

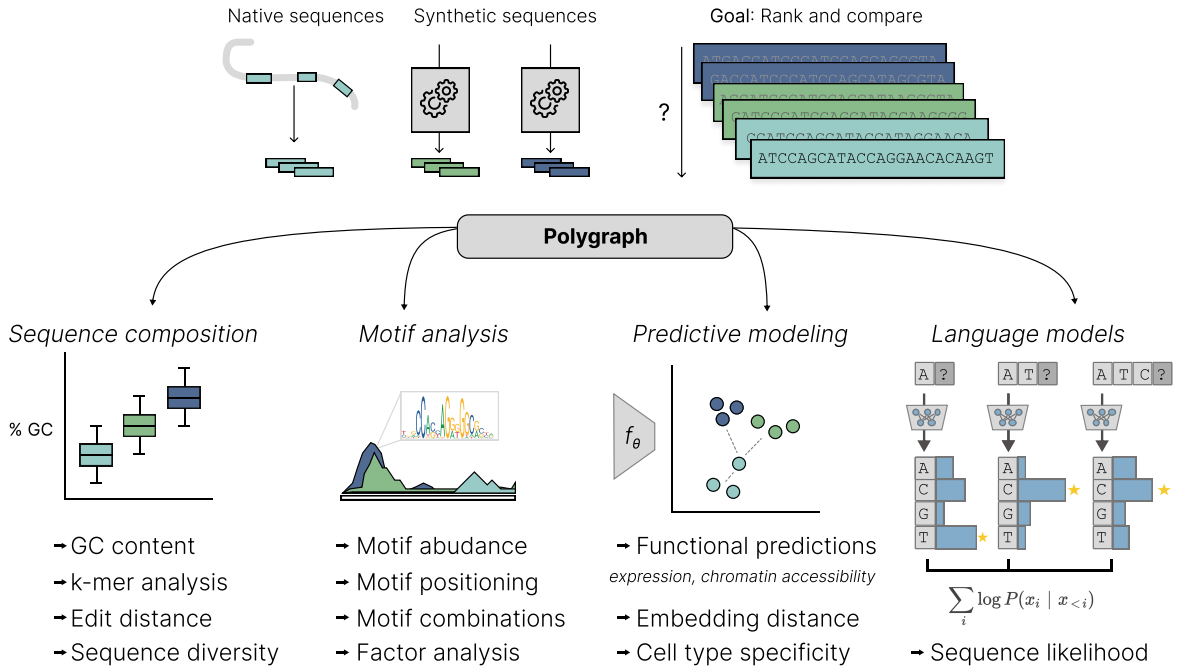

Figure 1: Schematic of Polygraph. Polygraph is a comprehensive package to evaluate and compare native and synthetic DNA sequences. Polygraph implements four classes of analysis: (A) Sequence composition, which evaluates GC content and k-mer abundance. (B) Motif analysis to quantify known transcription factor binding sites, motif combinations, motif positioning, and common transcription factor programs with NMF. (C) Predictive modeling to evaluate designed sequences on key properties such as activity, specificity, and chromatin accessibility. The latent space is used to visualize sequences and compute group diversity. (D) Application of a human DNA language model to score native sequence likelihood.

1. Sequence composition analysis

Polygraph includes sequence analysis tools to evaluate GC content (percentage of bases in a sequence that are guanine (G) and cytosine (C)), k-mer counts (abundance of nucleotide subsequences of length *k*), and edit distance to reference sequences. K-mer composition is also used to cluster sequences and visualize them in PCA space. We introduce a *sequence diversity* metric, defined as the average k-nearest neighbor (KNN) distance between a sequence and its neighbors from the same group, to quantify how similar designed sequences are to each other. Human regulatory DNA exhibits elevated GC content[20] and an abundance of particular subsequences like of GATA motifs [21], CpG islands [21], and short repeats [22]. Evaluating sequence composition with our quantitative metrics could therefore provide insight into the novelty and "humanness" of computationally designed CREs.

2. Transcription factor binding motif analysis

Motif content reveals which combinations of transcription factors are able to bind. Polygraph scans sequences against motif databases YeTFaSCo and JASPAR to describe their regulatory grammar[23,24]. We count individual binding sites as well as compute the enrichment of motif pairs. Motif scanning reports start positions along the sequence, describing if computationally placed motifs are centered and if different methods pursue different spacing rules. We additionally apply non-negative matrix factorization (NMF) to decompose the k-mer count matrix into common transcription factor programs shared across sequences [11]. Similar to k-mer analysis, we visualize and cluster sequences by motif count and compute euclidean and KNN distance between sequence groups. Altogether, motif analysis may be used to uncover higher-order multi-factor regulatory rules exploited by different design approaches.

3. Predictive modeling

Polygraph integrates trained neural network models to evaluate designed sequences on key properties including activity, specificity, and chromatin accessibility. We provide three trained and open-source models for yeast and human prediction as well as integrate any user-provided PyTorch model. Polygraph provides flexible and fast model prediction on generated sequence, cell-type specificity evaluation, and UMAP visualization of sequence embeddings. These latent vector representations created by evaluating sequences on a portion of the model capture functionally relevant properties and may therefore be more meaningful than k-mer and motif visualizations alone. Overall, predictive models describe if designed sequences will be effective in driving their intended activity and are key to engineering precise and tunable regulatory elements.

4. Language modeling

Autoregressive DNA language models are self-supervised models which can implicitly capture the rules of gene regulation as well as the rules of coding genes of the genome it is trained on. These models can be queried to infer how likely that a given sequence is sampled from the training set which in the case of human DNA language models is the human genome. Here we use HyenaDNA[25] to quantify the log likelihood of synthetic sequences to score their “humanness”.

Yeast promoter analysis

We applied Polygraph to comprehensively analyze native and computationally designed 80 bp yeast promoters. We consider 100 native yeast promoters with high and low reported expression in media [26,27]. We additionally generated 50 synthetic promoters using directed evolution[8] and 50 using gradient-based optimization with Ledidi[13], starting from randomly generated sequences and iteratively editing to optimize expression using a model trained on native promoters (**Methods**).

Using Polygraph, we ask how designed sequences might drive expression and if they resemble native yeast sequences. Promoters created using both evolution and gradient methods have a higher GC content than their random starting points, unlike native DNA (mean: 53%, 59%, 50%, respectively, evolution p-adj = 8.74e-01, gradient p-adj = 4.46e-05, Kruskal-Wallis, **Figure 2a, Supplementary Table 1**). Reflecting this shift, k-mer analysis showed designed promoters were enriched for GC-rich patterns like 'CGCG' (gradient p-val = 1.27e-12, evolution p-val = 2.35e-08, Kruskal-Wallis) and 'GCGC' (gradient p-val = 1.04e-10, Kruskal-Wallis, **Supplementary Table 2**). Edit distance to native high promoters was higher in generated sequences than random sequences, suggesting that synthetic promoters did not grow to more closely resemble effective native sequence through iterative design (**Figure 2b**). In sum, although designed sequences drive high predicted expression, they do not resemble the yeast genome.

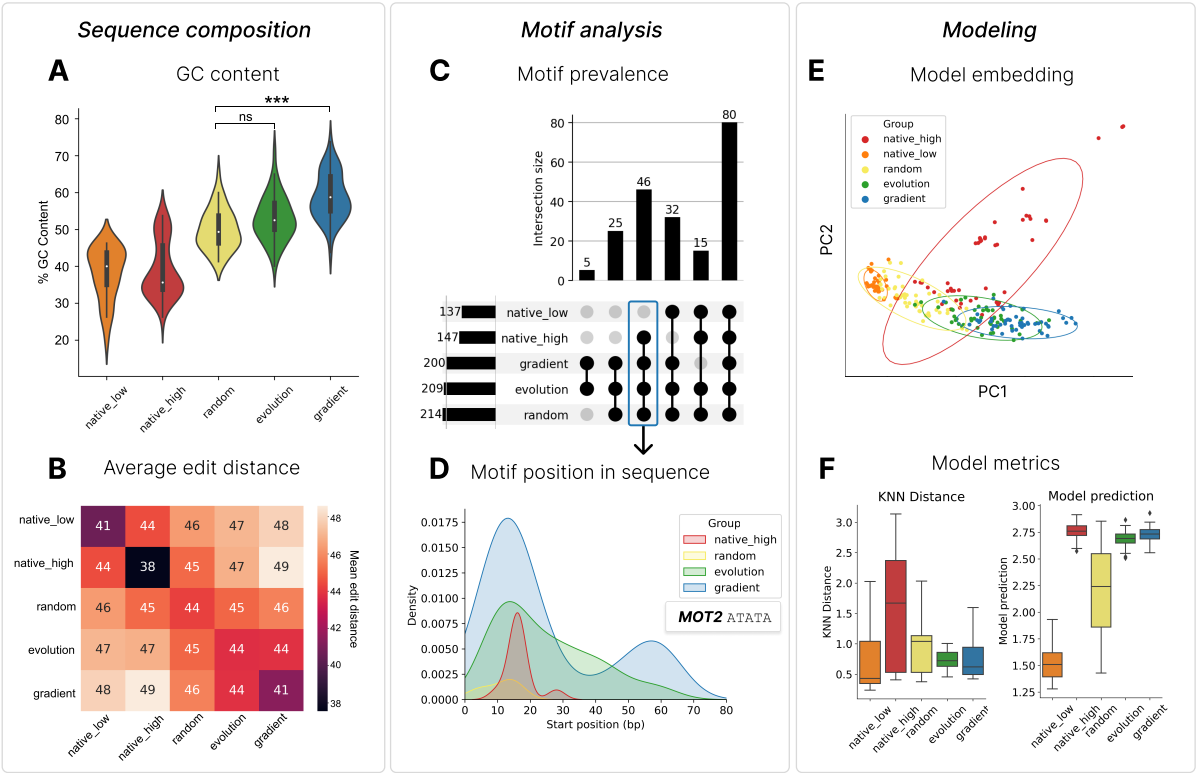

Figure 2: Sequence analysis of native, random, and designed yeast promoters. **a.** GC content distribution across native yeast promoters (labels: native_low and native_high), random sequences (random), and synthetic promoters designed either via directed evolution (evolution), or gradient-based optimization (gradient). The native_high and native_low classes are native promoters with the highest and lowest measured expression, respectively. **b.** Average edit distance between sequences in all groups. **c.** Overlap of shared transcription factor binding motifs across native and generated promoter sequences. Overlaps with fewer than 4 motifs are excluded from the plot. **d.** Distribution of MOT2 motif start locations, a motif absent in lowly expressed native promoters. **e.** PCA visualization of sequence embeddings from a sequence-to-expression predictive model. **f.** Nearest neighbor distance metrics in embedding space describe diversity within sequence groups (left). Predicted expression of native, designed, and random sequences (right).

Next, we scan the 250 sequences against the YeTFaSCo database, containing 241 yeast motifs to identify the motifs used by the natural and synthetic sequences. In total, 147 and 137 motifs were found in

the native high and native low groups, respectively (**Figure 2c**) We identified 46 motifs in the highly-expressed native promoters as well as designed promoters, but absent in lowly-expressed native promoters, while 5 motifs appeared exclusively in synthetic sequences, arising through the optimization process (HAP4, PHO2, TEA1, TEC1, a1-alpha2-dimer, **Figure 2c, Supplementary Table 3**). We conclude that design methods can insert *de novo* motifs and Polygraph can identify them. Profiling the start position of shared motifs like MOT2 revealed common spacing - the binding site frequently started between positions 10-20, hinting at functional grammar (**Figure 2d**). A higher prevalence observed in Ledidi sequences compared to directed evolution sequences suggests gradient-based methods may be more efficient in finding optimal motifs. We also find enrichment of motif combinations in designed groups compared to native sequence, including NHP6A/NHP6B and OTU1/ PDC2, (gradient p-adj = 4.29e-08, 4.29e-08, Kruskal-Wallis).

Finally, evaluating the predicted transcriptional activity across sequences using an independent model trained to predict yeast expression from sequence, we find that native sequences are the most diverse - native sequences have the most variance in embedding space, high intra-group KNN-distance, and high distance to the closest sequence of the same type (**Methods, Figure 2e, f, Supplementary Figure 1**). As expected, native high promoters showed elevated predicted activity compared to native low promoters (**Figure 2f**). However, designed promoters reached similarly high predicted expression levels despite starting from random sequences, and one exceeded all native promoters, demonstrating the ability of computational optimization to match or surpass the transcriptional activity of native regulatory DNA. In sum, these results point to a regulatory grammar underrepresented in native sequence, illustrating how integrating diverse *in silico* metrics provide insights into the sequence determinants underlying regulatory element design.

Human K562 enhancer analysis

We next applied Polygraph to evaluate 6,000 synthetic human enhancer sequences designed for K562 cells by AdaLead, FastSeqProp, and Simulated Annealing, which are generative approaches guided by an expression prediction model trained on a library of 776,474 genomic sequences tested in a massively parallel reporter assay (MPRA) for cell-line specific expression[11]. We compared these computationally designed enhancers to native human sequences from open chromatin regions specific to K562 and randomly generated DNA (**Methods**). Again, we asked if the composition and mechanisms of designed sequences resembled that of effective endogenous elements with high K562 specificity.

Performing sequence analysis with Polygraph, we find that while all sequences converge on a GC content of 50%, designed sequences demonstrate lower GC content variance (native = 49.2% ± 0.0655, simulated annealing = 50.0 ± 0.0260, **Supplementary Table 4**). K-mer distributions also differed markedly – 'GGAG', 'CCAG', 'CCTG' and 'CAGG' were common in native sequences, whereas 'TATC' and 'GATA' ranked as the top two k-mers across all design method ('GATA' p-val vs native sequences, FastSeqProp = 2.48e-295, AdaLead = 3.37e-233, simulated annealing = 1.71e-167, Kruskal-Wallis, **Figure 3a**). These designed sequences produced shared k-mer populations that differed not only from native sequence, but from random starting sequence, suggesting a common shift toward more functional sequence grammar.

Transcription factor binding motif analysis identified substantial differences in regulatory grammar across groups as well. We scanned sequences for motif matches against all 871 transcription factor motifs in the JASPAR database[24] (**Supplementary Table 5)**. Performing NMF across this count by motif matrix, we find that AdaLead, FastSeqProp, and Simulated Annealing produce highly similar compositions of motif factors, reliant on a diverse combination of motifs (**Figure 3b**). Generated sequences rely on factors 0, 3, 5, and 9, using MEIS2, TBX2, TBX3, TRPS1, and GATA motifs while native and random sequences do not (e.g. factor 9 loading z-scores = 0.566 (AdaLead), 0.364 (FastSeqProp), 0.455 (simulated annealing), -0.718 (native), -2.172 (random), **Figure 3c, Supplementary Figure 2**). In contrast, PRDM9 in factor 2 is

featured prominently in native but not designed sequences. Sequence decomposition reveals how generative models converge on a shared vocabulary of motifs to produce synthetic K562 enhancers. Divergent motif strategies found in native and designed sequences can both produce cell-type specific regulatory sequences.

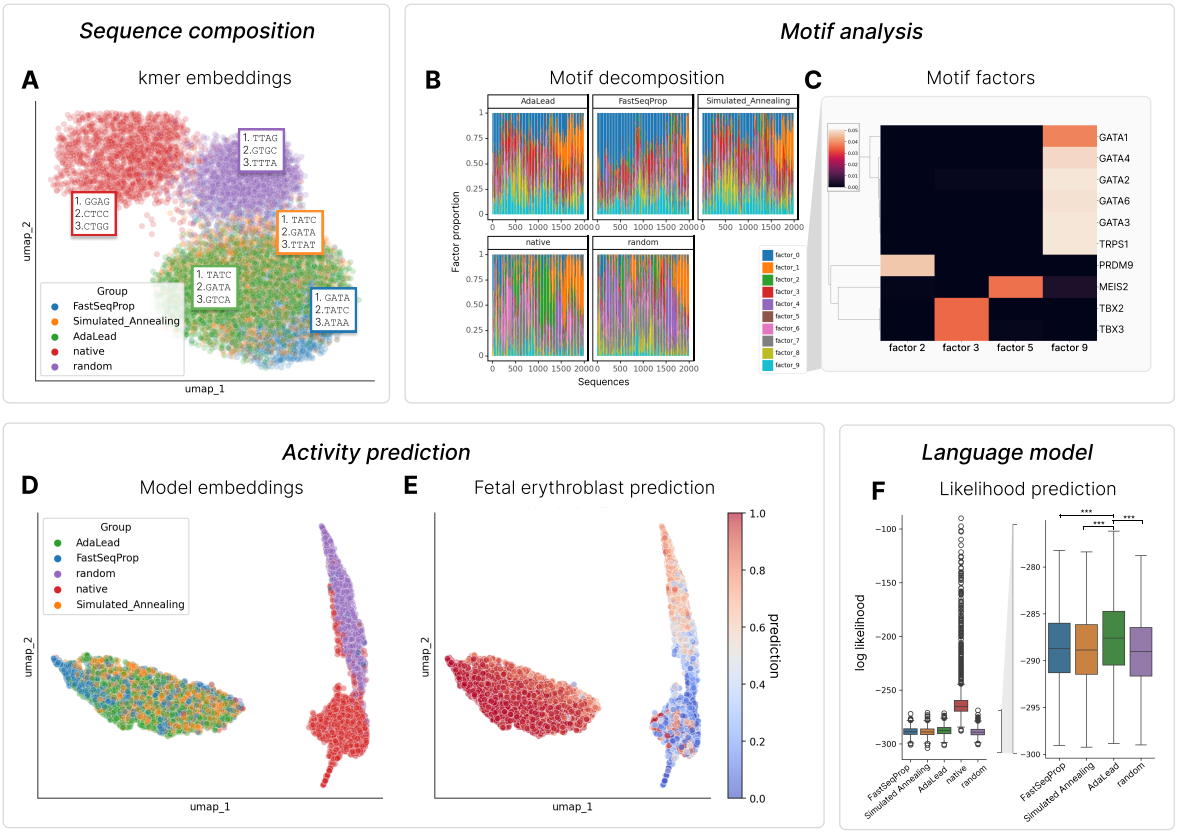

Figure 3: Sequence analysis of designed and native human enhancers in K562. **a.** UMAP visualization of 4-mer composition in uniformly random DNA, native DNA, and designed K562-specific enhancers generated with FastSeqProp, Simulated Annealing, and AdaLead. Top enriched k-mers are listed. **b.** Non-negative matrix factorization of motif counts, revealing distinct regulatory program preferences for designed and natural sequences. **c.** Motifs driving NMF components. **d.** Model embeddings from a chromatin accessibility model trained on fetal and adult human tissues[28]. **e.** UMAP of predicted chromatin accessibility in Fetal erythroblast, with designed sequences showing higher predicted erythroblast activity than native sequences. **f.** Likelihood predictions from a DNA language model, with only native sequencing scoring as probable human DNA.

We next evaluate the activity of the K562-specific designed sequences in 30 adult and 15 fetal human tissues with 203 annotated cell types (**Supplementary Figure 3**). Using a binary sequence-to-chromatin accessibility model, we find that synthetic enhancers group together in embedding space far from real or random sequences, aside from a small minority of native sequences (**Methods**, **Figure 3d**). Designed sequences are also more diverse than both native and random elements, as measured by within-group KNN distance in embedding space (**Methods**, **Supplementary Table 4**). Expectedly, open chromatin in fetal erythroblast, a blood cell type, is predicted to be higher across synthetic enhancers optimized for erythrocyte leukemia cell line K562 [29], than native sequences (mean prediction, AdaLead=0.95, simulated annealing=0.95, FastSeqProp=0.96, native=0.37, random=0.54, **Figure 3e, Supplementary Figure 4**). Predictions were low across sequences for CD4+ T cells, an unrelated cell type. Designed sequences can

identify sequence features that represent the preserved biology between cell lines and their origins in the human body.

Finally, we computed sequence likelihoods using an autoregressive DNA language model[25] by summing the predicted log-likelihood of each nucleotide across the sequence **(Methods**). Expectedly, native sequences showed high log likelihood **(Figure 3f)**. AdaLead-generated sequences had a significantly higher log likelihood than FastSeqProp, simulated annealing, and random sequences (p-val = 1.54e-09, p-val = 4.85e-13, p-val = 7.66e-18, Kruskal-Wallis).

Conclusions

The rational design of DNA elements holds great promise for both bettering our understanding of gene regulatory mechanisms and driving cell-type specific expression in therapeutic contexts. However, systematically evaluating and comparing synthetically designed sequences remains challenging without unified frameworks and baselines. In this work, we present Polygraph - a computational toolkit for analyzing designed cis-regulatory elements through sequence composition metrics, motif analysis, predictive modeling, and language modeling.

As demonstrated in yeast and human case studies, Polygraph provides insights into the properties of computationally designed regulatory elements. Sequence analysis revealed how different generative algorithms converge on common grammar, often distinct from native DNA. Motif scanning uncovered shared transcription factor binding sites exploited for specialized activity like cell-type specific expression across methods. Predictive neural networks allowed benchmarking design methods and resolving commonalities and differences between methods in terms of high level regulatory sequence features that are implicitly parsed out by predictive models in an untargeted manner. Finally, DNA language modeling provides a “humanness” score describing similarity to genomic sequences.

While Polygraph provides an integrated toolkit for evaluating designed regulatory elements, several exciting directions remain to improve analysis and application. Integrating a wider array of predictive models, including those predicting chromatin accessibility, protein binding, histone modifications and chromatin states will add more regulatory perspectives to sequence evaluation. Condensing our metrics to create a score with which to prioritize sequences will also be critical when screening millions of candidates for experimental validation. Experimentally validating a small number of synthetic sequences and iteratively feeding the results back to the generative design processes in an active learning framework is a promising future direction. Finally, applying Polygraph across species will help to clarify how regulatory elements evolved.

Together, Polygraph's sequence evaluations empower comprehensive assessment and comparison of regulatory element design approaches. This work establishes a robust computational platform to understand the regulatory code and advance the engineering of cis-regulatory elements for therapeutic applications.

Methods

Polygraph package details

Polygraph supports several statistical analyses across groups: (1) pairwise Fisher's exact test for proportions between each non-reference group and the reference group. (2) groupwise Mann-Whitney U test for comparing the mean values between each non-reference group and the reference group, and (3) Kruskal-Wallis test followed by Dunn's post-hoc test for comparing the mean values between all groups. Several metrics may be computed across k-mer, motif, and model embedding analysis: *Sequence diversity* is defined

as within group KNN distance, or the mean distance of each sequence to its k nearest neighbors in the same group. *Distance to reference* calculates the euclidean distance of each sequence to its nearest neighbor in the reference group, in the embedding space. *1-NN statistic* calculates 1-nearest neighbor statistics based on the sequence embeddings using either group ID of nearest neighbor or distance to nearest neighbor. *Differential analysis* performs a Wilcoxon rank-sum test on sequence embeddings to find features enriched/depleted with respect to the reference group.

Yeast promoter dataset and model

Yeast gigantically parallel reporter assay (GPRA) data as well as expression measurements for native promoters were obtained from Vaishnav et al.[27]. A sequence-to-expression model was trained using a convolutional architecture on 3,922 80bp sequences of native yeast promoters whose expression was measured in complex medium. The model consisted of 4 convolutional layers each with 64 channels and ReLU activation. The first convolutional layer had kernel size=15 and subsequent layers had kernel size=3. The convolutional layers were followed by average pooling across the sequence length. Finally, a 1x1 convolutional layer was used to combine the output of the 64 channels into the predicted expression value. 3,500 randomly selected promoters were used for training while 422 were held out each for validation and testing. The model was trained for 10 epochs with Poisson loss using the Adam optimizer with learning rate=1e-4 and batch size 16. This model was used as an oracle for sequence design.

To generate sequences using directed evolution, we initialized 50 random starting sequences and iterated through 10 rounds of evolution using the yeast model as an oracle to increase expression. 50 Ledidi[13] optimized sequences were generated in a similar manner with the following parameters: max_iter=500, l=100, lr=1e-2.

Yeast embedding model

We considered approximately 7.5 million 80 bp long randomly generated yeast promoters whose expression was measured in both complex and defined media from Vaishnav et al. 2022[27]. 1,000,000 randomly selected sequences were held out for training and validation. We trained a sequence-to-expression model to predict expression in both media. The model consisted of 4 convolutional layers with 512 channels each and ReLU activation. All convolutional layers after the first were followed by max pooling with width 2. The output of the final convolutional layer was flattened and passed through a dense layer with 32 nodes and ReLU activation to produce two outputs.

The output from the final convolutional layer of this model was used to embed native and synthetic sequences. The trained model can be downloaded and used as part of the Polygraph package.

Human enhancer dataset and model

Gosai et al.[11] generated a dataset of 776,474 200-nucleotide sequences paired with gene expression, as measured by a massively parallel reporter assay (MPRA), and engineered cell-type specific candidate sequences using simulated annealing, FastSeqProp[12], and AdaLead[14]. To compare the performance of these methods in one cell type, we selected the top 2,000 synthetic sequences with the highest MinGap in K562 cells, defined by Gosai et al. as sequences with the highest difference in predicted activity in K562 and the max of HepG2 and SK-N-SH. The 'native' group was selected from the top 2,000 endogenous sequences overlapping H3K27ac peaks with the highest MinGap in K562 (referred to as 'DHS-natural').

We finetuned a pretrained Enformer model[9] using the Enformer Pytorch implementation at https://github.com/lucidrains/enformer-pytorch on the single-cell atlas of chromatin accessibility in the human genome[28] to predict binarized cell type-specific pseudobulk accessibility and embed sequences (**Fig. 3d-e, Fig S3 and S4**). Sequences from chromosomes 7 and 13 were held out for validation and testing, respectively. We sliced the first residual block and left out the next six blocks, used "target_length=-1" argument to disable cropping and added a 1D average pooling to collapse the sequence dimensionality.

Input length was 200bp. Binary signal and cell type annotations were obtained from publicly available files provided by the authors (http://catlas.org/catlas_downloads/humantissues/cCRE_by_cell_type/). Cell types with less than 3% positive labels were discarded which reduced the number of cell types from 222 to 203.

Code and data availability

The codebase and tutorials for this package are available at https://github.com/Genentech/polygraph. Model weights and preprocessed datasets can be found at the following DOI: `10.5281/zenodo.8412250`.

Acknowledgments

We thank David Garfield and Oriol Fornes for helpful discussion and feedback.

References

1. Ong C-T, Corces VG. Enhancer function: new insights into the regulation of tissue-specific gene expression. Nat Rev Genet. 2011;12: 283–293.

2. Maeder ML, Gersbach CA. Genome-editing Technologies for Gene and Cell Therapy. Mol Ther. 2016;24: 430–446.

3. Naldini L. Gene therapy returns to centre stage. Nature. 2015;526: 351–360.

4. High KA, Roncarolo MG. Gene Therapy. N Engl J Med. 2019;381: 455–464.

5. Dunbar CE, High KA, Joung JK, Kohn DB, Ozawa K, Sadelain M. Gene therapy comes of age. Science. 2018;359. doi:10.1126/science.aan4672

6. Au HKE, Isalan M, Mielcarek M. Gene Therapy Advances: A Meta-Analysis of AAV Usage in Clinical Settings. Front Med. 2022;8: 809118.

7. Kelley DR, Reshef YA, Bileschi M, Belanger D, McLean CY, Snoek J. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. Genome Res. 2018;28: 739–750.

8. Taskiran II, Spanier KI, Christiaens V, Mauduit D, Aerts S. Cell type directed design of synthetic enhancers. bioRxiv. 2022. p. 2022.07.26.501466. doi:10.1101/2022.07.26.501466

9. Avsec Ž, Agarwal V, Visentin D, Ledsam JR, Grabska-Barwinska A, Taylor KR, et al. Effective gene expression prediction from sequence by integrating long-range interactions. Nat Methods. 2021;18: 1196–1203.

10. Avsec Ž, Weilert M, Shrikumar A, Krueger S, Alexandari A, Dalal K, et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. Nat Genet. 2021;53: 354–366.

11. Gosai SJ, Castro RI, Fuentes N, Butts JC, Kales S, Noche RR, et al. Machine-guided design of synthetic cell type-specific cis -regulatory elements. bioRxiv. 2023. doi:10.1101/2023.08.08.552077

12. Linder J, Seelig G. Fast activation maximization for molecular sequence design. BMC Bioinformatics. 2021;22: 510.

13. Schreiber J, Lu YY. Ledidi: Designing genomic edits that induce functional activity. bioRxiv. 2020. p. 2020.05.21.109686. doi:10.1101/2020.05.21.109686

14. Sinai S, Wang R, Whatley A, Slocum S, Locane E, Kelsic ED. AdaLead: A simple and robust adaptive greedy search algorithm for sequence design. arXiv [cs.LG]. 2020. Available: http://arxiv.org/abs/2010.02141

15. Zrimec J, Fu X, Muhammad AS, Skrekas C, Jauniskis V, Speicher NK, et al. Controlling gene expression with deep generative design of regulatory DNA. Nat Commun. 2022;13: 5099.

16. Jain M, Bengio E, Hernandez-Garcia A, Rector-Brooks J, Dossou BFP, Ekbote CA, et al. Biological Sequence Design with GFlowNets. In: Chaudhuri K, Jegelka S, Song L, Szepesvari C, Niu G, Sabato S, editors. Proceedings of the 39th International Conference on Machine Learning. PMLR; 17--23 Jul 2022. pp. 9786–9801.

17. Gupta A, Kundaje A. Targeted optimization of regulatory DNA sequences with neural editing architectures. bioRxiv. 2019. p. 714402. doi:10.1101/714402

18. Avdeyev P, Shi C, Tan Y, Dudnyk K, Zhou J. Dirichlet Diffusion Score Model for Biological Sequence Generation. ArXiv. 2023. Available: https://www.ncbi.nlm.nih.gov/pubmed/37292476

19. van Laarhoven PJ, Aarts EH. Simulated Annealing: Theory and Applications. Springer Science & Business Media; 2013.

20. Colbran LL, Chen L, Capra JA. Short DNA sequence patterns accurately identify broadly active human enhancers. BMC Genomics. 2017;18: 536.

21. Colbran LL, Chen L, Capra JA. Sequence Characteristics Distinguish Transcribed Enhancers from Promoters and Predict Their Breadth of Activity. Genetics. 2019;211: 1205–1217.

22. Horton CA, Alexandari AM, Hayes MGB, Marklund E, Schaepe JM, Aditham AK, et al. Short tandem repeats bind transcription factors to tune eukaryotic gene expression. Science. 2023;381: eadd1250.

23. de Boer CG, Hughes TR. YeTFaSCo: a database of evaluated yeast transcription factor sequence specificities. Nucleic Acids Res. 2012;40: D169–79.

24. Khan A, Fornes O, Stigliani A, Gheorghe M, Castro-Mondragon JA, van der Lee R, et al. JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. Nucleic Acids Res. 2018;46: D1284.

25. Nguyen E, Poli M, Faizi M, Thomas A, Birch-Sykes C, Wornow M, et al. HyenaDNA: Long-Range Genomic Sequence Modeling at Single Nucleotide Resolution. ArXiv. 2023. Available: https://www.ncbi.nlm.nih.gov/pubmed/37426456

26. de Boer CG, Vaishnav ED, Sadeh R, Abeyta EL, Friedman N, Regev A. Author Correction: Deciphering eukaryotic gene-regulatory logic with 100 million random promoters. Nat Biotechnol. 2020;38: 1211.

27. Vaishnav ED, de Boer CG, Molinet J, Yassour M, Fan L, Adiconis X, et al. The evolution, evolvability and engineering of gene regulatory DNA. Nature. 2022;603: 455–463.

28. Zhang K, Hocker JD, Miller M, Hou X, Chiou J, Poirion OB, et al. A single-cell atlas of chromatin accessibility in the human genome. Cell. 2021;184: 5985–6001.e19.

29. Lozzio CB, Lozzio BB. Human chronic myelogenous leukemia cell-line with positive Philadelphia chromosome. Blood. 1975;45: 321–334.