Enformation Theory: A Framework for Evaluating Genomic AI

eyes s. robson
Center for Computational Biology

UC Berkeley
Berkeley, CA 94720
eyes.robson@berkeley.edu

Nilah M. Ioannidis
Department of Electrical Engineering
and Computer Sciences
UC Berkeley
Berkeley, CA 94720
nilah@berkeley.edu

Abstract

The nascent field of genomic AI is rapidly expanding with new models, benchmarks, and findings. As the field diversifies, there is an increased need for a common set of measurement tools and perspectives to standardize model evaluation. Here, we present a statistically grounded framework for performance evaluation, visualization, and interpretation using the prominent genomic AI model Enformer as a case study. The Enformer model has been used for a range of applications from mechanism discovery to variant effect prediction, but what makes it better or worse than precedent models at particular tasks? Our goal is not merely to answer these questions for Enformer, but to propose how we should think about new models in general. We start by reporting Enformer's few-shot performance on the GUANinE benchmark, which emphasizes complex genome interpretation tasks, and discuss its gains and deficits compared to precedent models. We follow this analysis with visualizations of Enformer's embeddings in low-dimensional space, where, among other insights, we diagnose features of the embeddings that may limit model generalization to synthetic biology tasks. Finally, we present a novel, theory-backed probe of Enformer embeddings, where variance decomposition allows for holistic interpretation and partial 'backtracking' to explanatory causal features. Through this case study, we illustrate a new framework, Enformation Theory, for analyzing and interpreting genomic AI models.

1 Introduction

Sequence-to-function AI models are rapidly taking a leading role in genome interpretation and comprehension. Enformer [1] has become one of the most popular and powerful models, but our understanding of it and related models [23, 22, 25] is nascent and incomplete. One canonical approach for model comparison and evaluation in AI fields is the use of benchmarks, which have seen gradual but growing rates of creation and adoption in genomic AI as well [7, 35, 13, 27, 52]. These genomic AI benchmarks are not yet widely adopted nor comprehensive, but they nonetheless offer a systematic means to evaluate and describe a model's successes and failures as well as its perspectives on relevant tasks.

In this work, we examine Enformer's performance on the GUANinE v1.0 benchmark [35], a collection of large-scale human and eukaryotic genomics tasks designed for robustness and controlled evaluation. We report the linear evaluation (L2-regularized) performance of both Enformer's embeddings and its human output head, which, while largely concordant, differ considerably in their understanding of DNase hypersensitivity. We additionally perform a qualitative analysis of GUANinE's applicability to models like Enformer by visualizing the sequences from GUANinE tasks in Enformer's own principal component space. While much can be gleaned from the traditional low-dimensional data visualizations employed here, there is still a need to develop better, more informative low-dimensional visualizations for genomic AI more broadly.

For our interpretability analysis, rather than relying on feature attribution approaches that operate on a per-example basis for *decision explainability* [26, 38], we use a holistic approach to model interpretation, or characterizing *what a model has learned on the whole* [10]. This approach, similar to Shapley regression values [26], is based on the decomposition of variance explained, but it has far fewer input requirements. Namely, it does not require the use of heuristics or

Preprint. Under review.

approximations, access to model parameters, nor expensive computational resources (which are all typically necessary for decision explainability, e.g. Shrikumar et al. [38, 39]). This approach dramatically lowers the computational and technical barrier to model investigation and auditing, while enabling the use of statistical significance testing for increased methodological rigor. Finally, this interpretability approach hints at a future of reliable, sample-efficient [12] estimators of sequence function, perhaps without the need for deep learning.

The contributions via our Enformation Theory framework are thus threefold:

1. **Performance:** We evaluate Enformer's performance on GUANinE and discuss that performance relative to the precedent models,
2. **Visualization:** We visualize benchmark sequences in low-dimensional space, illustrating *why* benchmarks like GUANinE are well-suited for models like Enformer and identifying hypotheses as to *how* Enformer struggles on GUANinE's GPRA tasks, and
3. **Interpretation:** We perform a confounder-based variance decomposition, a theory-backed model interpretation technique [15, 10], where we identify *what* confounders Enformer has learned (or failed to learn) and confirm the relevance of region-level and chromosome-level features to basic sequence characteristics like DNA accessibility (DNase hypersensitivity).

Our aim is to present a straightforward, transferable approach for analyzing genomic AI models – one that lends itself to extension and integration into model development pipelines.

2 Background

The standard approach to genomic AI originates at the intersection of traditional bioinformatics with convolutional neural networks (CNNs) from computer vision, in which supervised convolutional models attempt to capture sequence motifs (and higher-order interactions between them), not unlike Position Weight Matrices (PWMs) [3, 50, 2]. Some models also employ dilated convolutions, typically strided over an extended input, for computational efficiency [23, 1, 25]. A more recent, yet underdeveloped, approach involves self-supervised autoencoders trained on tokenized sequences from (multi)organismal corpora [21, 49, 7, 35].

Regardless of the approach, scoring and validating genomic AI models has been a recurrent issue [27, 7, 35], e.g. challenges of data leakage across tracks[1] [37] or organisms [1].

2.1 Enformer

Enformer is a powerful[2] genomic AI model that attempts to attend to $131,072$ base pairs (bp) of input sequence context. It does so through bp-resolution compressive convolutional blocks whose embeddings are fed into a transformer [45]. Not including the output head for mouse, Enformer totals 246.1M parameters, of which the human output head is some 6.6%. Enformer was trained in a supervised fashion on 5313 human and 1643 mouse experimental tracks, including cap analysis of gene expression (CAGE) data, using only reference genomes for each organism.

2.2 GUANinE benchmark

GUANinE is a large-scale genomic AI benchmark designed to enable the comparison of fully supervised models and baselines while also enabling the study of alternative (self-)supervised regimes [51, 1, 22]. Compared to smaller scale benchmarks where single-task, highly specialized baselines can be difficult to fully train due to small sample sizes [24, 13, 27], GUANinE requires transfer-learning-centric or multi-task models like Enformer to achieve a performance that is competitive with deep, specialized genomic machine learning algorithms. This effectively combines the best features of large-scale benchmarks in computer vision [9] with multi-task benchmarks in natural language processing [46, 47, 42].

GUANinE v1.0's primary limitation is its lack of variant interpretation or cell-type specificity tasks, limiting its applicability for benchmarking some genetics applications [18]. However, analyses such as genome-wide DNase hypersensitivity estimation, sequence conservation annotation, or across-gene variation in gene expression are all within its purview and relate to the bulk of the training data for models like Enformer. This makes GUANinE an ideal choice for evaluating how well Enformer learned its original training data and identifying which other biological features, if any, Enformer learned along the way.

[1] Due partly to the fact that 'average' genome methylation for a cell type is largely attributable to its degree of differentiation, e.g. specialized vs progenitor cells [11].

[2] Though somewhat unwieldy, due to its input representation size, which is not well suited to modern deep learning hardware [34]

Model	bp	dnase-propensity	ccre-propensity	conservation 30	conservation 100	gpra-c	gpra-d
		ρ	ρ	ρ	ρ	ρ	ρ
GC-content*	≤ 512	20.5533	12.9891	38.1307	39.5828	52.9688	56.9824
T5*	≤ 512	33.0548	27.4643	52.5576	44.0542	**84.6738**	**88.5912**
Beluga*	≤ 2000	64.9577	49.6688	58.6143	48.8816	66.5479	69.9363
Basenji2*	≤ 55k	60.5149	**56.4568**	71.0417	59.1229	69.5484	72.9872
Enformer	≤ 196k	64.9164	55.3816	**74.4358**	**62.2344**	70.6259	74.1311
Enformer embeds	≤ 196k	**68.0818**†	55.0775	**74.6815**	**62.4178**	70.6976	74.2039

Table 1: Performance of Enformer's outputs and embeddings via linear evaluation on GUANinE. The T5 is a fully-supervised model while other neural models are also few-shot (L2-regularized), as Enformer. Best score per task along with close scores (if any) are **bolded**. *Non-Enformer results reported previously [35]. †Single-bin embeddings results.

GUANinE is comprised of three distinct task groups: functional elements (human), conservation (human), and gene expression (yeast), each of which contains two separate, yet related tasks. The functional elements tasks include dnase-propensity and ccre-propensity, which ask the model to estimate the cell-type-agnostic likelihood that a sequence is accessible (DNase hypersensitive) and the cell-type-agnostic likelihood that an accessible sequence exhibits H3K4Me3, H3K27ac, or CTCF epigenetic features (indicative of promoters, enhancers, and cohesin loops, respectively). The cell-type-specific raw inputs that are used to construct both of these cell-type-agnostic GUANinE tasks are present in Enformer's 5313-track training and validation set, so Enformer should perform highly on these tasks if it learned them during its training.

The conservation tasks in GUANinE are among the more challenging in the benchmark, although Enformer's bi-organismal training likely exposed it to some of the underlying signal, like Basenji2 before it [35]. Other than this training procedure, models like Enformer must solely rely on recovering information manifolds buried in training data to extract conservation signal.

While Enformer was trained on gene expression data, the gene expression tasks in GUANinE have an esoteric distribution[3] in a distinct organism, yeast. This implies different transcription factors, and patterns of their interaction, may affect performance in addition to the distribution shift induced by the dual reporter assay [8].

3 Results

We report Enformer's few-shot performance scores on GUANinE's test set in Table 1. Both the human output head (with inverse softplus transform) and embedding scores use the average features for three bins for prediction, except for the embedding's dnase-propensity scores, which exhibited unusually higher single-bin performance.

Visualizations To better characterize the relationship between Enformer and GUANinE tasks, we present a pair of superimposed plots of GUANinE sequences in the PCA space of Enformer's training data embeddings in Figures 1 and 2. The latter illustrates the distribution of the dnase-propensity task labels, colored by propensity[4]. We observe that the unusual first PC of Enformer's training dataset corresponds to a small number of inaccessible (and less conserved, see Appendix A) regions, as suggested by the predominantly class '0' sequences from the dnase-propensity task.

Figure 1 shows how GUANinE tasks are largely resemblant of Enformer's underlying training distribution (i.e. they are not out-of-distribution), except for the tight cluster of the gpra-c task sequences. These cluster near the origin, suggesting the plotted PCs do not capture much variation in their embeddings. We next sought to investigate whether this visual lack of variation was indicative of an *statistical* lack of variation in the embeddings, so we examined pairwise distances between the embeddings as illustrated in Figure 3. As shown in the pairwise distance distribution, computed in the original, 3072-dimensional embedding space of Enformer, the over-similarity of GPRA-task embeddings demonstrates a lack of statistical variation, which suggests the GPRA tasks are out-of-distribution and that they may be pushed towards an information-poor 'null space' of Enformer's embedding space.

[3]The creators of the GPRA technique went to great lengths to design diverse sequences – this increased variation shifts their distribution considerably from natural sequences [8, 44].

[4]Which represents the ubiquity of DNase hypersensitivity across experimental tracks.

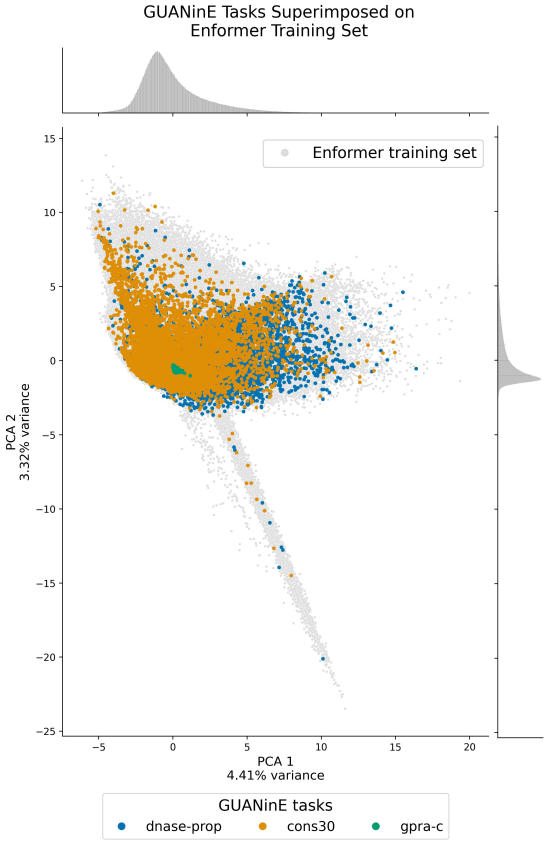

Figure 1: Enformer embedding PCA visualization of 1% of examples from select tasks of GUANinE superimposed on a grey background of 5% of Enformer's training set. GUANinE's dnase-propensity and cons30 tasks fall squarely within Enformer's training distribution, while the gpra-c task is overly clustered near the origin. This suggests at minimum these PCs, if not Enformer itself, have failed to capture the task.

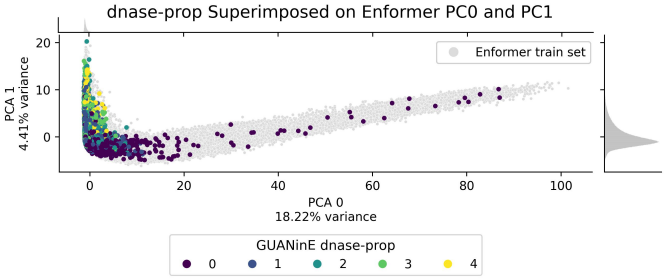

Figure 2: Enformer embedding PCA visualization of 1% of GUANinE's dnase-propensity examples (colored by propensity label) superimposed on a grey background of 5% of Enformer's training set sequences. The marginal density plots show that only a small number of sequences constitute the long tail of the 'cirrus' shape. A similar plot showing cons30 labels is presented in Appendix A.

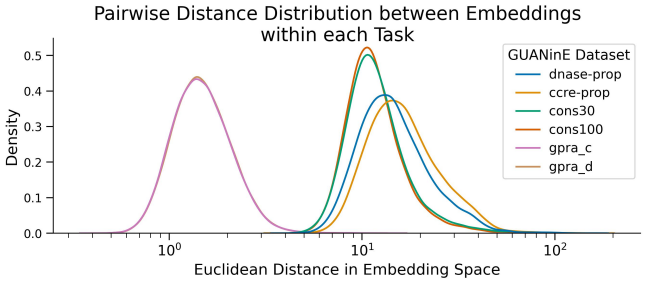

Figure 3: Pairwise distance KDE distributions, split by task, for single-bin embeddings from Enformer. For each task, we randomly sampled 100,000 pairs from 1% of the training set examples (<0.03% of all pairwise distances in the 1%). Both GPRA tasks (left, largely overlapping) have markedly lower average pairwise distances, suggesting little distinguishable difference between embeddings.

Variance decomposition The high-level findings of the confounder-based variance decomposition are presented in Figure 4, where the proportion of variance unique to Enformer (blue) is separated from the jointly explained variance (orange). Enformer surprisingly exploits nearly all of the *useful* variance from the confounders, leaving only a tiny fraction uncaptured (green, nearly invisible), suggesting it has absorbed the causal signal these features have to offer.

In precise terms, for the dnase-propensity task, these confounders explain up to 10.5% of the variance in linear contributions alone (before considering higher-order effects or interactions). Of that 10.5%, Enformer has surprisingly captured nearly 98% of it in the 10.3% jointly explained variance.

Digging into the coefficients of the combined model [10] for the dnase-propensity task, we can see which individual features contribute to this 10.5% of variance, as illustrated in Figure 5. Here, features are spatially grouped by the type of feature (e.g. GC-content) and colored by the scale of the data they examine (e.g. 'core' vs. 'distal', see Subsection 5.5). The 'core' PhyloP30way conservation signal is by far the strongest predictor, but if it were not included, the 'core' GC-content would absorb most of its causal signal via omitted variable bias [12]. A similar visualization for the multiscale features used to decompose the cons30 task is presented in Appendix A.

4 Discussion

In this section, we outline a handful of key implications about Enformer and its performance in a task-by-task overview before turning to specific discussions of each of our analyses.

Enformer incorporates conservation in predictions First and foremost, Enformer is a strong predictor of sequence conservation (cons30 and cons100), whose connection to deleteriousness suggests Enformer may succeed in medical and clinical applications, at least where region-level conservation signal is important. Additionally, Enformer maintains a good estimate of local sequence accessibility propensity (DNase hypersensitivity), which is at least in part due to its understanding of sequence conservation, as our confounder-based variance decomposition shows. The fact that Enformer misses out on only a tiny portion (0.1%, right-hand side, Figure 4) of variance explainable by our enumerated confounders is surprising – it implies Enformer has learned the implicit primary DNA sequence signatures of our chromosomal features, which are known to impact sequence conservation [48].

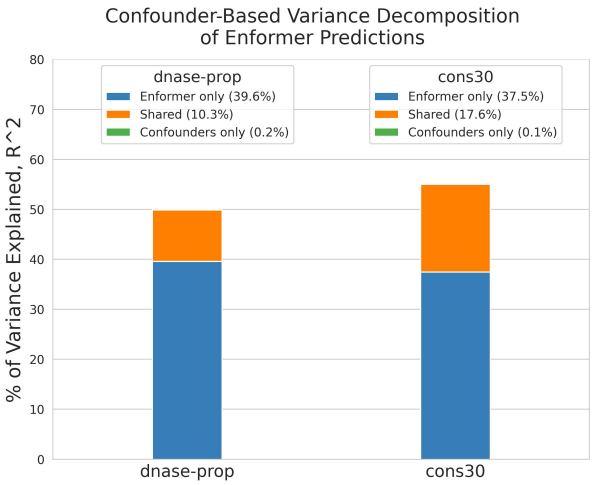

Figure 4: Percentage of variance explained by the Enformer and by enumerated sequence features ('confounders'). This tells us how informative, if at all, said sequence features are for the tasks at hand, as well as what novel information Enformer has captured beyond these features. Since the 'confounder only' portion is so small for both dnaseprop and cons30, it stands to reason that for the `GUANinE` tasks of interest, Enformer has subsumed them entirely.

It is unclear whether Enformer's grasp of *regional* conservation and accessibility is transferable to finer-grained resolutions [2] or variant-interpretation tasks, for which some evidence of non-performance exists [18]. While many researchers are interested in applying sequence models to precision genomics tasks, it is unknown how useful accessibility detection or regulatory element recognition are for precision medicine, despite these tasks being the stark majority of training data for supervised CNN models like Enformer [50, 51, 23, 2, 22, 1].

Does Enformer outperform other models? While Enformer outperforms Basenji2 and Beluga on most of the tested tasks, it lags behind Basenji2 in its ability to recognize cis-regulatory element propensity (i.e. histone marks). This is somewhat unexpected given the dependence that `GUANinE`'s ccre-propensity task has on accessibility[5]. Follow-up analyses are necessary to determine if Enformer's success in dnase-propensity is merely due to accurately detecting negatives, which would not aid in its ccre-propensity performance.

Enformer may be optimized for different objectives than accessibility, e.g. gene expression performance (CAGE). It is worth recognizing that the areas any given model "outperforms" on may be offset by decreased performance on other tasks, especially in a multi-task regime [41].

Visualization insights into generalizability A key insight enabled by our framework is Enformer's limited generalizability to the out-of-distribution GPRA tasks. We observe Enformer's tendency to map the random oligonucleotide sequences to an information-poor 'null space,' and it stands to reason that perhaps straightforward embedding-space-based solutions like adapter layers [16] or domain adaptation [4] may be able to ameliorate its performance.

An interesting potential genomics-specific form of transfer learning may be through the use of homology information, perhaps in a manner leveraging cross-organismal relationships in well-studied model organisms [5]. This could be evaluated by fine-tuning models like Enformer on such a large dataset as the GPRA tasks.

4.1 Model interpretability as a means, not an end

The confounder-based variance decomposition we undertake goes beyond motif discovery [39] or gene annotation (both of which could be supported under appropriate featurizations). For instance, we can conclude not only that Enformer has learned to leverage sequence conservation and chromosomal features (e.g. telomeric distance), but also that these features are intrinsically captured *in primary sequence alone*, as Enformer does not receive additional inputs.

This finding has profound implications in terms of how we choose to discuss DNA accessibility (i.e. relative to position), and also in terms of what confounders influence a model's test set performance and the physical assays we use to collect data [8, 44]. We leave to future work how genomics researchers can best leverage or de-confound these signals.

[5]Sequences in ccre-propensity are a subset of the dnase-propensity tasks, as a sequence must be defined as accessible in one or more cell-types before it can be declared a cis-regulatory element.

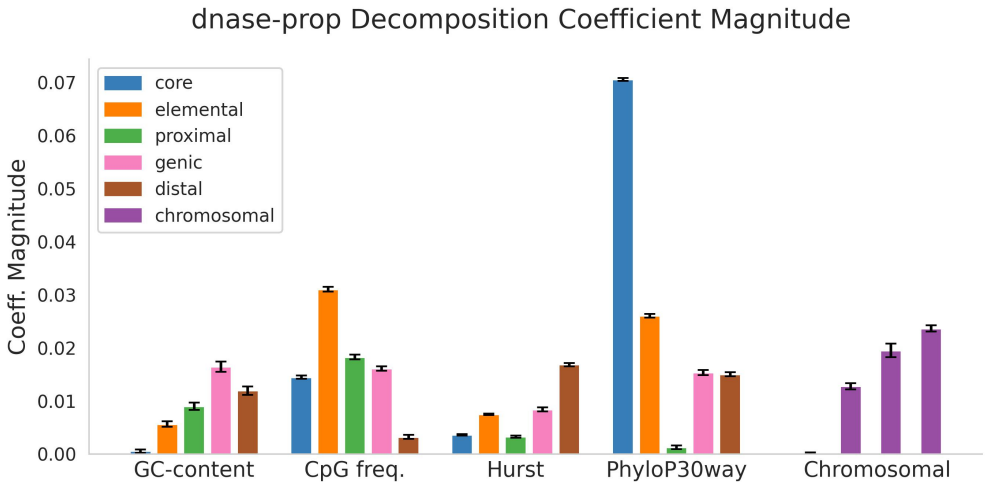

Figure 5: Coefficient magnitude (directionless) of the different features tested in the dnase-propensity confounder-based variance decomposition of Enformer's predictions. All features were standardized to mean zero and variance one before coefficient estimation. Error bars shown here are 3.259 times the test-set estimated standard error for each coefficient; this number corresponds to the Bonferroni-corrected two-sided $p < 0.01$ threshold. GUANinE's large test set sample size is the reason coefficients can be estimated accurately despite partial collinearity.

Learning more than deep learning As hinted at earlier, if genomics researchers are able to articulate the explanatory variables underlying a deep learning model's success (be it GC-content, conservation, or chromosomal features, etc.) this would reduce the need to rely upon inefficient[6] feature extraction approaches dependent on deep neural networks. At a minimum, enumerating these features can at least assist us to 'debug' and improve existing neural models by telling us whether a model has already learned a potential 'ensemble' feature [19, 17]. Our Enformation Theory framework of performance, visualization, and interpretation, indicates that augmenting Enformer with *region*-level conservation scores and simplistic chromosomal features would not improve performance on downstream tasks, as the model has already subsumed their signal.

Regional conservation as an indicator of sequence function One puzzling finding from our interpretation analysis relates to Enformer being a strong predictor of sequence conservation, although $\approx 45\%$ of PhyloP30way conservation is uncaptured by Enformer (right-hand side, Figure 4). Yet surprisingly, this untapped portion adds at most a marginal ($< 0.2\%$ of variance, left-hand side, Figure 4) amount of signal to the dnase-propensity task. We hypothesize that Enformer arrives at a "conservation-aware embedding space" that captures solely the portion of sequence conservation relevant to its downstream tasks, which follows from the original construction and utility of sequence conservation as a tool to discover functional elements [32, 40]. Plenty of sequence features unenumerated in our analyses (coding regions, binding motifs, and splice junctions, etc.) are proxies for both accessibility and conservation during training.

4.2 Reflections on GUANinE

Finally, we discuss a handful of insights regarding benchmarking that have emerged through this case study. Current models such as Enformer have strong performance on some tasks, but are far from having solved biology *in general* – or from necessarily being the proper modeling approach to do so. Given that GUANinE's propensity tasks benefit from aggregation across cell lines [35] (thus reducing the proportion of experimental noise, à la law of large numbers), we believe the performance ceiling on these tasks is far from being reached. Furthermore, models like Enformer, which have been trained on a superset of GUANinE's propensity tasks [35], should be able to achieve near perfect accuracy on dnase- and ccre-propensity, as these tasks are straightforward derivations of such data. The fact that models like Enformer explain just half of the variation in these tasks indicates that these models have not fully learned their own training data.

The primary limitation of GUANinE for evaluating a model like Enformer is its reliance on GPRA data to circumvent the small-N and collinear gene expression data of humans. Models like Enformer struggle to perform on personal gene

[6] Both in terms of computational efficiency, given the cost of larger and larger models [33], and *statistical* efficiency, which governs how well a model fits, even in small sample sizes.

expression estimation [18], but we do not believe the gpra-c or gpra-d tasks in GUANinE currently illustrate that issue. As is, they instead show the limitation of models like Enformer when attempting to predict gene expression on exogenous sequences and different organisms without additional fine-tuning – and they offer a rich opportunity for innovation in transfer learning techniques.

5 Methods

We turn now to the the different techniques used for each of our analyses before our conclusion below. Notably, we do not perform any fine-tuning of Enformer, relying instead on a linear probe fit to either the embeddings or outputs. Coefficients, formulae, and additional details on the confounder-based variance decomposition are in the Appendices.

5.1 Feature extraction with Enformer

We use the PyTorch [28], variable-length implementation of Enformer[7] for inference efficiency. For each task in GUANinE, we align inputs to the center-left bin of Enformer (the 448th bin, or 447 when zero-based indexing), and we provide the full 196kbp $+$ 256bp of sequence context. This allows us to access both the single bin embeddings and prediction for each target as well as a three-bin average. Incomplete sequences extending into telomeric regions were padded with Ns, although we do not see a strong association between Ns and Enformer embeddings, see Appendix A.

The 3072-dimensional embeddings from Enformer, three-bin-averaged and single-bin, along with the 5313-dimensional outputs from the human prediction head, three-bin-averaged and single-bin, were cached for our linear evaluation step. These embeddings and outputs were computed for the full development and test sets of each task in GUANinE, along with a 1% subsample of the training sets provided in GUANinE's Hugging Face repository[8].

All features from the output layer were passed into an inverse-softplus transformation, which slightly improved performance in every task (likely due to the softplus destroying ratio information).

5.2 Linear evaluation

We use scikit-learn's [29] ridge regression to perform L2-regularized linear evaluation on the three-bin-averaged and single-bin embeddings and outputs from Enformer. We searched over regularization coefficients of $[0.1, 0.3, 1., 3., \ldots 3.\text{e}5, 1.\text{e}6]$, evaluated on the development split, as we noticed substantial variation in optimal regularization coefficients across tasks (many of the regressions required high amounts of regularization, possibly due to the combination of high dimensionality and collinearity). The best of either the three-bin-averaged or single-bin predictions on the development set was used for test set predictions, as reported in Table 1.

5.3 Dimensionality reduction

Due to the large size of Enformer's training set, we use `incremental PCA` [36, 29] to estimate the first 30 principal components without resorting to subsetting. For all visuals, dimension reduction was performed on the single-bin embeddings. The 30 PCs explained a combined 51.6122% of Enformer's training set variance.

`t-SNE` (in the Appendices) was performed on examples after first being reduced to 30 PCs for runtime efficiency.

5.4 Confounder-based variance decomposition

For both the dnase-propensity and cons30 tasks we decompose the explained variance into enumerated ("confounding") and unenumerated (unspecified) portions of our few-shot regression models on Enformer outputs. This is accomplished via the post-hoc confounder analysis methodology proposed in Dinga et al. [10], which relies on the Pythagorean properties of the RSS (residual sum of squares), in canonical-link linear models [15]. The decomposition of variance is related to the explainability application of game-theoretic Shapley values [26, 38], but it instead focuses on model interpretation rather than single-decision explainability. Our procedure is as follows:

1. Fit combined linear model $f_{C \cup E}(X_{\text{train}}) = y_{\text{train}}$ on both confounders C and Enformer predictions (the latter being the L2-regularized few-shot prediction validated on GUANinE's development set, although any predictor will do).
2. Fit reduced models f_C and f_E as before.

[7] `https://github.com/lucidrains/enformer-pytorch`

[8] `https://huggingface.co/guanine`

3. Determine the RSS and R^2 of $f_{C\cup E}$, f_C, and f_E on independent (and thus unbiased) test set $\{X_{\text{test}}, y_{\text{test}}\}$
4. Adjust for the partial correlation effects of the R^2 measures as in Dinga et al. [10].
5. Derive $R^2_{C\cup E} = R^2_C + R^2_E + R^2_{C\cap E}$

In practice, three issues can arise when attributing of variance to specific coefficients or to reduced models as we do here – we briefly address all three below.

First, as discussed in Hastie [15], there is a bias on coefficient estimates due to overfitting when performing ordinary least squares (OLS). This is easily resolved by instead estimating RSS on an independent test set, as in Dinga et al. [10].

Second, as "confounders" are almost certainly correlated with model predictions, there is a micronumerosity problem, i.e. sufficient sample size is required to reliably estimate individual coefficients in the presence of partial collinearity [14]. This is of limited concern due to our large ($N > 40000$) sample sizes in `GUANinE`, although we choose to report all p-values, coefficients, and standard errors in the Appendix B. This is not dissimilar to the application of linear models for stepwise regression in GWAS or fine-mapping, when partial collinearity occurs.

Third, the predominant – and underappreciated – limitation to our application of explained variance decomposition is *omitted variable bias* [12]. Specifically, if an identifiable, yet *un*included, covariate is correlated with a *proxy* variable in our regression model, the explanatory power of this absentee variable will be absorbed by its correlates (proxies). This will inevitably inflate our estimates of explained variance from the specific variables we are investigating. To counter this, we select multiple, redundant features for each regression, and within the dnase-propensity regression, we specifically include mean PhyloP30way scores to prevent GC-content and other variables from absorbing their signal.

A figure comparing the explained variance on the dnase-propensity task with and without conservation features is included in the Appendix C. PhyloP30way scores are not included as confounders for the cons30 task.

5.5 Multiscale features for the decomposition

Numerous endogenous and exogenous features have been associated with DNA function in multivariate analysis [30, 31, 20]. The proportion of guanine plus cytosine content (G+C or GC-content) is among the simplest yet most predictive in many organisms [43]. For our explained variance decomposition regressions described in subsection 5.4, we enumerate a number of both sequence-level and chromosome-level features measured at multiple scales.

Sequence-level features We include GC-content, CpG frequency, and the Hurst coefficient of alternating Purine-Pyrimidine content [30, 31] at each choice of scale. We also regressed DNA melting points but observed that estimates of melting point correlate perfectly with GC-content, as in Biopython's libraries for DNA melting point estimates [6]. The scales of analysis chosen for these features were 127bp, 511bp, 2047bp, 8191bp, $and 32767$bp, referred to as 'core', 'element', 'proximal', 'genic', and 'distal', respectively.

For each (scale, feature) combination, the measurement was centered to the example, then the corresponding measurement from the next-smallest region subtracted. For example, the 'genic_gc' feature measures GC-content of only the $\approx$ 6kbp surrounding the 'proximal' sequence. This nested featurization provides partially decorrelated features, reducing variance inflation. For the dnase-propensity regression, the mean PhyloP30way inputs are calculated similarly.

Chromosome-level features Unlike sequence features which operate directly on a local scale, numerous chromosome-level background features influence a chromosome's constituent sequences, e.g. smaller chromosomes having increased linkage and background conservation due to failed recombination events [48]. We choose to include a feature set that is both overdetermined yet rich: the log of the hg38 chromosome size, the distance to the arm's telomere (in bp), the relative position of a sequence in the chromosome's arm (i.e. 0.0 to 1.0, where 0.5 is midway between centromere and telomere), and the distance to the nearest centromere (in bp). The chromosomal feature coefficients included in Figure 5 are in the same order.

All missing values (e.g. unalignable regions for phyloP) were imputed to the mean, and we standardize all features to mean of zero and variance of one to aid with model fitting and coefficient interpretation.

6 Conclusion

Through our Enformation Theory framework of performance, visualization, and interpretation, we arrive at a deeper understanding of a model's performance than metrics reporting alone. We discuss the advances and limitations of Enformer on the `GUANinE` benchmark while also articulating *what* types of confounders and sequence features it has learned in order to achieve such performance. We show, with significance, its reliance on information contained in

complex features like conservation and distance to centro- and telomeric regions, and we highlight the need to better understand their role in experimental measurements and model predictions. We identify a key generalization issue of models like Enformer to synthetic GPRA sequences and affirm the role of GUANinE as a leading benchmark for genomic AI models. It is our hope that the analyses presented in this framework, such as our use of confounder-based variance decomposition to identify interpretable covariates, will become more ubiquitous throughout the field of genomic AI and supply it with a unified toolkit for model development.

7 Acknowledgements

Thanks to Ruchir Rastogi and other members of the Ioannidis lab for invaluable feedback. Special thanks to Amelia Dobis for her assistance in copyediting. This work was funded in part by grants from the UC Noyce Initiative for Computational Transformation and the Chan Zuckerberg Biohub San Francisco.

References

[1] Ž. Avsec, V. Agarwal, D. Visentin, J. R. Ledsam, A. Grabska-Barwinska, K. R. Taylor, Y. Assael, J. Jumper, P. Kohli, and D. R. Kelley. Effective gene expression prediction from sequence by integrating long-range interactions. *Nature Methods*, 18(10):1196–1203, Oct 2021. ISSN 1548-7105. doi: 10.1038/s41592-021-01252-x. URL https://doi.org/10.1038/s41592-021-01252-x.

[2] Ž. Avsec, M. Weilert, A. Shrikumar, S. Krueger, A. Alexandari, K. Dalal, R. Fropf, C. McAnany, J. Gagneur, A. Kundaje, and J. Zeitlinger. Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nature Genetics*, 53(3):354–366, Mar 2021. ISSN 1546-1718. doi: 10.1038/s41588-021-00782-6. URL https://doi.org/10.1038/s41588-021-00782-6.

[3] M. Beckstette, R. Homann, R. Giegerich, and S. Kurtz. Fast index based algorithms and software for matching position specific scoring matrices. *BMC Bioinformatics*, 7(1):389, Aug 2006. ISSN 1471-2105. doi: 10.1186/1471-2105-7-389. URL https://doi.org/10.1186/1471-2105-7-389.

[4] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper_files/paper/2006/file/b1b0432ceafb0ce714426e9114852ac7-Paper.pdf.

[5] A. Breschi, T. R. Gingeras, and R. ó. Comparative transcriptomics in human and mouse. *Nat Rev Genet*, 18(7):425–440, Jul 2017.

[6] P. J. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, et al. Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11): 1422–1423, 2009.

[7] H. Dalla-Torre, L. Gonzalez, J. Mendoza-Revilla, N. L. Carranza, A. H. Grzywaczewski, F. Oteri, C. Dallago, E. Trop, B. P. de Almeida, H. Sirelkhatim, G. Richard, M. Skwark, K. Beguir, M. Lopez, and T. Pierrot. The nucleotide transformer: Building and evaluating robust foundation models for human genomics. *bioRxiv*, 2023. doi: 10.1101/2023.01.11.523679. URL https://www.biorxiv.org/content/early/2023/09/19/2023.01.11.523679.

[8] C. G. de Boer, E. D. Vaishnav, R. Sadeh, E. L. Abeyta, N. Friedman, and A. Regev. Deciphering eukaryotic gene-regulatory logic with 100 million random promoters. *Nat Biotechnol*, 38(1):56–65, Jan 2020.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[10] R. Dinga, L. Schmaal, B. W. Penninx, D. J. Veltman, and A. F. Marquand. Controlling for effects of confounding variables on machine learning predictions. *bioRxiv*, 2020. doi: 10.1101/2020.08.17.255034. URL https://www.biorxiv.org/content/early/2020/08/18/2020.08.17.255034.

[11] M. Ehrlich and M. Lacey. DNA methylation and differentiation: silencing, upregulation and modulation of gene expression. *Epigenomics*, 5(5):553–568, 2013.

[12] B. S. Everitt and A. Skrondal. The cambridge dictionary of statistics. *Cambridge University Press*, pages 149,310, 2010.

[13] K. Grešová, V. Martinek, D. Čechák, P. Šimeček, and P. Alexiou. Genomic benchmarks: a collection of datasets for genomic sequence classification. *BMC Genomic Data*, 24(1):25, May 2023. ISSN 2730-6844. doi: 10.1186/s12863-023-01123-8. URL https://doi.org/10.1186/s12863-023-01123-8.

[14] D. N. Gujarati and D. C. Porter. *Basic econometrics*. McGraw-Hill, 2009.

[15] T. Hastie. A closer look at the deviance. *The American Statistician*, 41(1):16–20, 1987. ISSN 00031305. URL http://www.jstor.org/stable/2684312.

[16] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly. Parameter-efficient transfer learning for NLP. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

[17] C. Hsu, H. Nisonoff, C. Fannjiang, and J. Listgarten. Learning protein fitness models from evolutionary and assay-labeled data. *Nature Biotechnology*, 40(7):1114–1122, Jul 2022. ISSN 1546-1696. doi: 10.1038/s41587-021-01146-5. URL https://doi.org/10.1038/s41587-021-01146-5.

[18] C. Huang, R. W. Shuai, P. Baokar, R. Chung, R. Rastogi, P. Kathail, and N. M. Ioannidis. Personal transcriptome variation is poorly explained by current genomic deep learning models. *Nature Genetics*, 55(12):2056–2059, Dec 2023. ISSN 1546-1718. doi: 10.1038/s41588-023-01574-w. URL https://doi.org/10.1038/s41588-023-01574-w.

[19] N. M. Ioannidis, J. H. Rothstein, V. Pejaver, S. Middha, S. K. McDonnell, S. Baheti, A. Musolf, Q. Li, E. Holzinger, D. Karyadi, L. A. Cannon-Albright, C. C. Teerlink, J. L. Stanford, W. B. Isaacs, J. Xu, K. A. Cooney, E. M. Lange, J. Schleutker, J. D. Carpten, I. J. Powell, O. Cussenot, G. Cancel-Tassin, G. G. Giles, R. J. MacInnis, C. Maier, C. L. Hsieh, F. Wiklund, W. J. Catalona, W. D. Foulkes, D. Mandal, R. A. Eeles, Z. Kote-Jarai, C. D. Bustamante, D. J. Schaid, T. Hastie, E. A. Ostrander, J. E. Bailey-Wilson, P. Radivojac, S. N. Thibodeau, A. S. Whittemore, and W. Sieh. REVEL: An Ensemble Method for Predicting the Pathogenicity of Rare Missense Variants. *Am J Hum Genet*, 99(4):877–885, Oct 2016.

[20] K. Jeppsson, B. Pradhan, T. Sutani, T. Sakata, M. Umeda Igarashi, D. G. Berta, T. Kanno, R. Nakato, K. Shirahige, E. Kim, and C. Björkegren. Loop-extruding smc5/6 organizes transcription-induced positive dna supercoils. *Molecular Cell*, 84(5): 867–882.e5, 2024. ISSN 1097-2765. doi: https://doi.org/10.1016/j.molcel.2024.01.005. URL https://www.sciencedirect.com/science/article/pii/S1097276524000066.

[21] Y. Ji, Z. Zhou, H. Liu, and R. V. Davuluri. DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. *Bioinformatics*, 37(15):2112–2120, 02 2021. ISSN 1367-4803. doi: 10.1093/bioinformatics/btab083. URL https://doi.org/10.1093/bioinformatics/btab083.

[22] D. R. Kelley. Cross-species regulatory sequence activity prediction. *PLOS Computational Biology*, 16(7):1–27, 07 2020. doi: 10.1371/journal.pcbi.1008050. URL https://doi.org/10.1371/journal.pcbi.1008050.

[23] D. R. Kelley, Y. A. Reshef, M. Bileschi, D. Belanger, C. Y. McLean, and J. Snoek. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Res*, 28(5):739–750, May 2018.

[24] M. Kircher, C. Xiong, B. Martin, M. Schubach, F. Inoue, R. J. A. Bell, J. F. Costello, J. Shendure, and N. Ahituv. Saturation mutagenesis of twenty disease-associated regulatory elements at single base-pair resolution. *Nature Communications*, 10(1):3583, Aug 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-11526-w. URL https://doi.org/10.1038/s41467-019-11526-w.

[25] J. Linder, D. Srivastava, H. Yuan, V. Agarwal, and D. R. Kelley. Predicting rna-seq coverage from dna sequence as a unifying model of gene regulation. *bioRxiv*, 2023. doi: 10.1101/2023.08.30.555582. URL https://www.biorxiv.org/content/early/2023/09/01/2023.08.30.555582.

[26] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[27] F. I. Marin, F. Teufel, M. Horlacher, D. Madsen, D. Pultz, O. Winther, and W. Boomsma. Bend: Benchmarking dna language models on biologically meaningful tasks, 2023.

[28] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[30] C.-K. Peng, S. Buldyrev, A. Goldberger, S. Havlin, F. Sciortino, M. Simons, and H. Stanley. Fractal landscape analysis of dna walks. *Physica A: Statistical Mechanics and its Applications*, 191(1):25–29, 1992. ISSN 0378-4371. doi: https://doi.org/10.1016/0378-4371(92)90500-P. URL https://www.sciencedirect.com/science/article/pii/037843719290500P.

[31] C.-K. Peng, S. V. Buldyrev, A. L. Goldberger, S. Havlin, F. Sciortino, M. Simons, and H. E. Stanley. Long-range correlations in nucleotide sequences. *Nature*, 356(6365):168–170, Mar 1992. ISSN 1476-4687. doi: 10.1038/356168a0. URL https://doi.org/10.1038/356168a0.

[32] K. S. Pollard, M. J. Hubisz, K. R. Rosenbloom, and A. Siepel. Detection of nonneutral substitution rates on mammalian phylogenies. *Genome Res*, 20(1):110–121, Jan 2010.

[33] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), jan 2020. ISSN 1532-4435.

[34] E. Robson, C. Xu, and L. W. Wills. Prose: the architecture and design of a protein discovery engine. In *Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '22, page 655–668, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392051. doi: 10.1145/3503222.3507722. URL https://doi.org/10.1145/3503222.3507722.

[35] e. s. robson and N. Ioannidis. Guanine v1.0: Benchmark datasets for genomic ai sequence-to-function models. In D. A. Knowles and S. Mostafavi, editors, *Proceedings of the 18th Machine Learning in Computational Biology meeting*, volume 240 of *Proceedings of Machine Learning Research*, pages 250–266. PMLR, 30 Nov–01 Dec 2023. URL https://proceedings.mlr.press/v240/robson24a.html.

[36] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 77(1):125–141, May 2008. ISSN 1573-1405. doi: 10.1007/s11263-007-0075-7. URL https://doi.org/10.1007/s11263-007-0075-7.

[37] J. Schreiber, R. Singh, J. Bilmes, and W. S. Noble. A pitfall for machine learning methods aiming to predict across cell types. *Genome Biology*, 21(1):282, Nov 2020. ISSN 1474-760X. doi: 10.1186/s13059-020-02177-y. URL https://doi.org/10.1186/s13059-020-02177-y.

[38] A. Shrikumar, P. Greenside, and A. Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3145–3153. JMLR.org, 2017.

[39] A. Shrikumar, K. Tian, A. Shcherbina, Z. Avsec, A. Banerjee, M. Sharmin, S. Nair, and A. Kundaje. Tf-modisco v0.4.4.2-alpha: Technical note. *CoRR*, abs/1811.00416, 2018. URL http://arxiv.org/abs/1811.00416.

[40] A. Siepel, G. Bejerano, J. S. Pedersen, A. S. Hinrichs, M. Hou, K. Rosenbloom, H. Clawson, J. Spieth, L. W. Hillier, S. Richards, et al. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome research*, 15(8):1034–1050, 2005.

[41] T. Standley, A. Zamir, D. Chen, L. Guibas, J. Malik, and S. Savarese. Which tasks should be learned together in multi-task learning? In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

[42] Y. Tay, M. Dehghani, S. Abnar, Y. Shen, D. Bahri, P. Pham, J. Rao, L. Yang, S. Ruder, and D. Metzler. Long range arena : A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=qVyeW-grC2k.

[43] The G10K Consortium et al. Towards complete and error-free genome assemblies of all vertebrate species. *Nature*, 592 (7856):737–746, Apr 2021. ISSN 1476-4687. doi: 10.1038/s41586-021-03451-0. URL https://doi.org/10.1038/s41586-021-03451-0.

[44] E. D. Vaishnav, C. G. de Boer, J. Molinet, M. Yassour, L. Fan, X. Adiconis, D. A. Thompson, J. Z. Levin, F. A. Cubillos, and A. Regev. The evolution, evolvability and engineering of gene regulatory dna. *Nature*, 603(7901):455–463, Mar 2022. ISSN 1476-4687. doi: 10.1038/s41586-022-04506-6. URL https://doi.org/10.1038/s41586-022-04506-6.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[46] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446.

[47] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/4496bf24afe7fab6f046bf4923da8de6-Paper.pdf.

[48] P. D. Waters, H. R. Patel, A. Ruiz-Herrera, L. Álvarez González, N. C. Lister, O. Simakov, T. Ezaz, P. Kaur, C. Frere, F. Grützner, A. Georges, and J. A. M. Graves. Microchromosomes are building blocks of bird, reptile, and mammal chromosomes. *Proceedings of the National Academy of Sciences*, 118(45):e2112494118, 2021. doi: 10.1073/pnas.2112494118. URL https://www.pnas.org/doi/abs/10.1073/pnas.2112494118.

[49] M. Zaheer, G. Guruganesh, K. A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang, and A. Ahmed. Big bird: Transformers for longer sequences. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 17283–17297. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/c8512d142a2d849725f31a9a7a361ab9-Paper.pdf.

[50] J. Zhou and O. G. Troyanskaya. Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods*, 12(10):931–934, Oct 2015. ISSN 1548-7105. doi: 10.1038/nmeth.3547. URL https://doi.org/10.1038/nmeth.3547.

[51] J. Zhou, C. L. Theesfeld, K. Yao, K. M. Chen, A. K. Wong, and O. G. Troyanskaya. Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk. *Nat Genet*, 50(8):1171–1179, Aug 2018.

[52] Z. Zhou, Y. Ji, W. Li, P. Dutta, R. Davuluri, and H. Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome, 2023.

A Additional figures

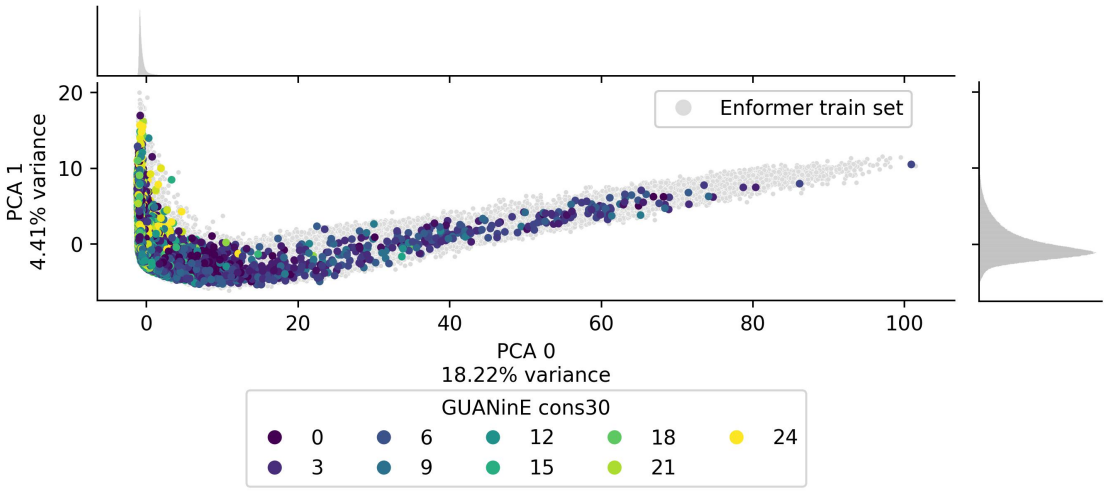

Figure 6: Enformer embedding PCA visualization of 1% of `GUANinE`'s cons30 examples (colored by binned conservation label) superimposed on a grey background of 5% of Enformer's training set sequences. The marginal density plots show that only a small number of sequences constitute the long tail of the 'cirrus' shape, which is enriched for unconserved (closer to zero) values.

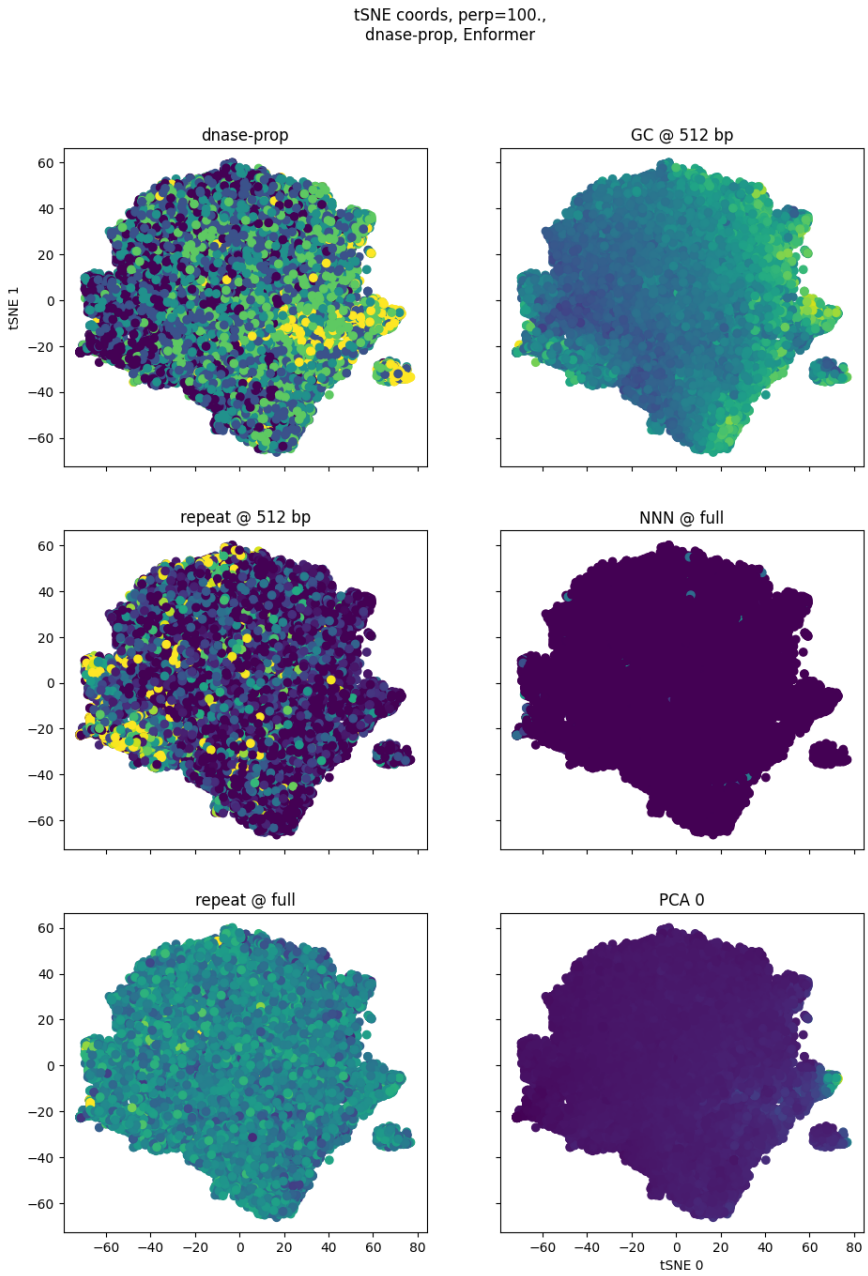

Figure 7: `t-SNE` of Enformer embeddings of examples from the dnase-propensity task. Clockwise from the top left, we visualize dnase-propensity scores (yellow is '4', purple is '0'); GC-content of the 'element' region at 512 bp; repeat frequencies of the 'element' region, as determined by RepeatMaster (yellow is greater); the number of Ns at the full context size (yellow is greater); the repeat proportion at the full context size; and the first PCA component.

B Coefficients and test statistics for the combined model regression

	Feature	Coeff. Estimate	Std. Error (test set)	Test statistic
0	core_gc	0.000598	0.000092	6.532174
1	core_cpg	0.014509	0.000087	166.952854
2	core_hurst	-0.003670	0.000037	-98.149661
3	core_cons	0.070605	0.000075	944.343870
4	element_gc	-0.005687	0.000144	-39.622319
5	element_cpg	-0.031083	0.000129	-241.677170
6	element_hurst	0.007526	0.000047	158.620686
7	element_cons	-0.026099	0.000104	-251.578393
8	proximal_gc	-0.009036	0.000199	-45.294165
9	proximal_cpg	0.018313	0.000130	140.975122
10	proximal_hurst	0.003290	0.000056	58.658573
11	proximal_cons	0.001310	0.000092	14.270689
12	genic_gc	0.016503	0.000277	59.555348
13	genic_cpg	-0.016180	0.000119	-135.915108
14	genic_hurst	-0.008436	0.000106	-79.300149
15	genic_cons	-0.015397	0.000136	-113.102824
16	distal_gc	-0.011995	0.000226	-52.960047
17	distal_cpg	0.003266	0.000110	29.747621
18	distal_hurst	0.016897	0.000085	198.094002
19	distal_cons	-0.015072	0.000110	-136.794978
20	lg_chr_size	0.000179	0.000050	3.608735
21	tel	-0.012809	0.000172	-74.578977
22	arm	-0.019565	0.000362	-54.069611
23	cen	0.023705	0.000160	147.696933

Table 2: Combined model regression for the confounder-based variance decomposition on the dnase-propensity task estimates of Enformer. The degrees of freedom for involving the standard error, which we complete on the test set, is $44801 - 24 - 3072 = 41705$, which renders the use of z-scores appropriate.

C Comparative dnase-propensity figures without conservation