Interpreting *Cis*-Regulatory Interactions from Large-Scale Deep Neural Networks for Genomics

Shushan Toneyan[1] **and Peter K Koo**[1,*]

[1]Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory, NY, USA
[*]e-mail: koo@cshl.edu

ABSTRACT

The rise of large-scale, sequence-based deep neural networks (DNNs) for predicting gene expression has introduced challenges in their evaluation and interpretation. Current evaluations align DNN predictions with experimental perturbation assays, offering a limited perspective of the DNN's capabilities within the studied loci. Moreover, existing model explainability tools mainly focus on motif analysis, which becomes complex to interpret for longer sequences. Here we introduce CREME, an *in silico* perturbation toolkit that interrogates large-scale DNNs to uncover rules of gene regulation that it has learned. Using CREME, we investigate Enformer, a prominent DNN in gene expression prediction, revealing *cis*-regulatory elements (CREs) that directly enhance or silence target genes. We explore the relationship between CRE distance from transcription start sites and gene expression, as well as the intricate complexity of higher-order CRE interactions. This work advances the ability to translate the powerful predictions of large-scale DNNs to study open questions in gene regulation.

Recent advances in sequence-based genomic DNNs have shown notable success in predicting gene expression by considering significantly larger inputs[1–3]. However, the extensive sequence size presents a challenge when evaluating DNN predictions and interpreting their learned patterns. Current methods for evaluating large-scale models have relied on assessing the alignment between predictions and experimental perturbation assays[1,4,5]—such as massively parallel reporter assays[6,7] and CRISPR interference (CRISPRi)[8]—as well as statistical analyses like expression-quantitative trait loci[5,9,10]. However, these only provide a narrow evaluation of what a DNN has learned within the studied loci or the biological question being probed. Moreover, experimental measurements capture both biological and technical variability, which makes it difficult to assess generalization of the underlying biology learned by the DNN. Conversely, prevailing model interpretability methods concentrate primarily on the analysis of motifs[11–20], short DNA sequences associated with regulatory functions. As the number of motifs spanning longer sequences grows, interpreting motif analysis becomes increasingly challenging, given the difficulty in deciphering their coordination for carrying out regulatory functions.

To bridge this gap, we present CREME (*Cis*-Regulatory Element Model Explanations), an *in silico* perturbation toolkit designed to examine large-scale DNNs trained on functional genomics data. In contrast to existing model interpretability methods, CREME can provide interpretations at various scales, including a coarse-grained CRE-level view as well as a fine-grained motif-level view. CREME is based on the notion that by fitting experimental data, the DNN essentially approximates the underlying "function" of the experimental assay. Thus, the trained DNN can be treated as a surrogate for the experimental assay, enabling *in silico* "measurements" for any sequence. Drawing inspiration from CRISPRi[21,22], CREME comprises a suite of perturbation experiments to uncover how DNNs learn rules of interactions between CREs and their target genes (Fig. 1).

To demonstrate the utility of CREME, we interpret Enformer[1], a DNN that takes ∼400kb DNA sequences as input and predicts the corresponding read coverage profiles for 5,313 experiments that includes chromatin accessibility, transcription factor binding, histone marks, and gene expression across various cell types. In this study, we investigate the regulation of gene expression in K562 cells. Using a curated list of sequences centered on TSS annotations[23] (see Methods), we examine how specific sequence perturbations affect gene expression given by Enformer's predictions of TSS activity. The results are organized according to specific biological questions.

To what extent does Enformer depend on distal context for gene expression prediction? While Enformer's predictions have been shown to depend on individual nearby enhancers[5], the extent that it relies on a broader set of context is unclear. Using CREME's *TSS Context Dependence Test*, we sought to directly measure the effect of distal context on TSS activity[24,25]. Briefly, this test measures the effect of shuffling the context (i.e. the entire sequence) while keeping the proximal regions (∼5kb) centered on the TSS-under-investigation intact (see Methods). To reduce the effects of spurious patterns, we performed 10 independent dinucleotide-shuffles and averaged predictions of TSS activity, a procedure similar to global importance analysis (GIA)[14].

Interestingly, we observed that the majority of cases resulted in a drop in TSS activity, presumably due to disruption of

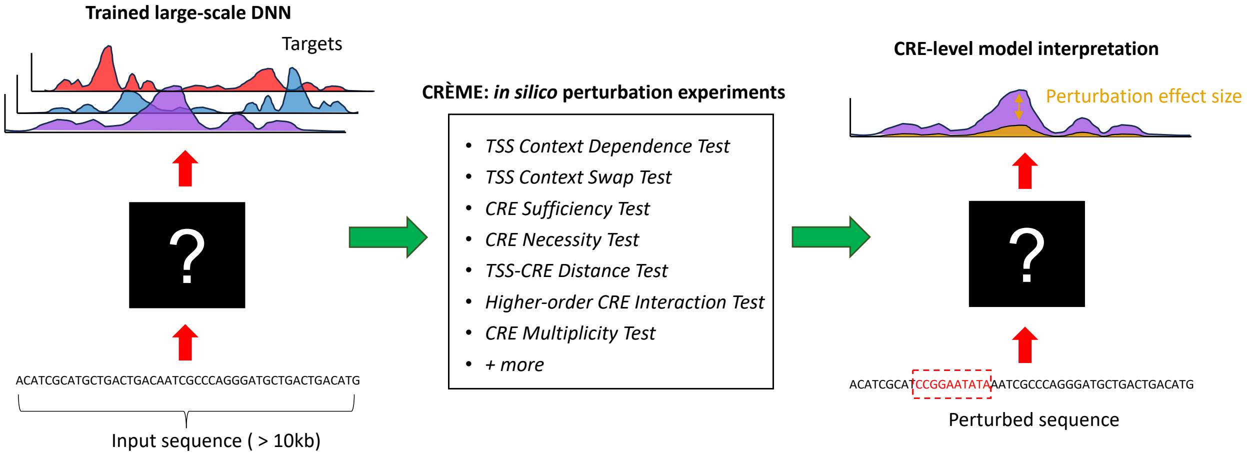

Trained large-scale DNN
Targets
?
ACATCGCATGCTGACTGACAATCGCCCAGGGATGCTGACTGACATG
Input sequence (> 10kb)
CRÈME: *in silico* perturbation experiments
TSS Context Dependence Test
TSS Context Swap Test
CRE Sufficiency Test
CRE Necessity Test
TSS-CRE Distance Test
Higher-order CRE Interaction Test
CRE Multiplicity Test
+ more
CRE-level model interpretation
Perturbation effect size
?
ACATCGCATCCGGAATATAAATCGCCCAGGGATGCTGACTGACATG
Perturbed sequence

Figure 1. CREME Overview. CREME offers a suite of *in silico* perturbation experiments that probe specific biological hypotheses. Perturbations are applied to the input sequence and the effect size is measured as the difference in the model predictions.

enhancers (Figs. 2a and 2b). On occasion, the TSS activity increased, suggesting the presence of silencers[26–29]. We also observed cases where TSS activity was robust across context, which can arise from a neutral context (i.e., depletion of CREs or net neutral effects of CREs) or the TSSs are intrinsically strong and context independent. Based on these results, we organized a subset of the sequences into 3 context categories (i.e., enhancing, silencing, and neutral) for further analysis (Fig. 2a, inset).

Do CREs yield similar effects on non-target genes? Next, we explored how gene expression changes when a TSS along with its proximal context (5kb) is inserted into different, non-native (but genomic) contexts—replacing the native TSS—using CREME's *TSS Swap Test* (see Methods). By stratifying results according to context categories, we observed that TSSs originally from neutral contexts largely maintained high activity independent of the context (Fig. 2c, top). As expected, these TSSs mostly correspond to housekeeping genes[30] (49%), whereas the TSSs in other contexts did not (6% for enhancing context and 0% for silencing context). Interestingly, swapping TSSs among the set of sequences from enhancing contexts resulted in a 50% decrease in TSS activity, on average, while a larger drop was observed when placed into other contexts (Fig. 2c, middle). This suggests that enhancers can be somewhat effective at enhancing other genes, but they are better tuned for their native target gene, perhaps through some compatibility rule[31–33]. On the other hand, we found that silencing context is more-or-less interchangeable when considering their effectiveness across genes that are actively silenced (Fig. 2c, bottom). Swapping these actively silenced genes into non-silencing context leads to a substantial increase in TSS activity. Moreover, testing TSS activity in dinucleotide-shuffled versions of the context from each category confirmed the importance of higher-order structured patterns (e.g. enhancers and silencers) beyond dinucleotide frequencies (Supplementary Fig. 1).

Which CREs are necessary for TSS activity? Using CREME's *CRE Necessity Test*, we can identify a CRE's influence on a target gene, mapping the locations of enhancers and silencers and their effect size on TSS activity. Specifically, we binned the input sequence into 5kb tiles and monitored how TSS activity is altered upon shuffling each tile (Fig. 3a), which is effectively an occlusion perturbation[5,34]. As expected, individual tiles in enhancing backgrounds tend to yield a positive influence on TSS activity, while tiles in silencing context are enriched to yield a negative influence (Fig. 3b). Notably, all context seem to have a mix of enhancers and silencers. In general, individual tiles in neutral context have an overall weak impact on TSS activity, though some may yield large effects.

Are individual CREs sufficient for TSS activity? To test whether an individual tile is *sufficient* to activate or silence a target TSS[35], we employed CREME's *CRE Sufficiency Test*, which embeds a tile-of-interest along with the TSS tile in dinuceotide-shuffled sequences at their original positions and measures TSS activity. This GIA experiment uncovers the global importance of the combined tiles while effectively removing contributions from background context[14]. The results indicate that individual enhancer tiles often yield low effect sizes on their own (Fig. 3c). Surprisingly, the effect sizes of individual tiles on TSS activity increased when taken out of the original (neutral or silencing) context, suggesting competition among CREs in the original context leads to a lower effect . Evidently, necessity does not imply sufficiency (Supplementary Fig. 2), suggesting that multiple CREs are needed to drive most TSS activity in non-neutral contexts.

How far are CREs from target TSS? We next mapped the distance distribution of TSSs for enhancer and silencer CREs identified from CREME's Sufficiency Test (see Methods). Our findings indicate that the majority of enhancers recognized

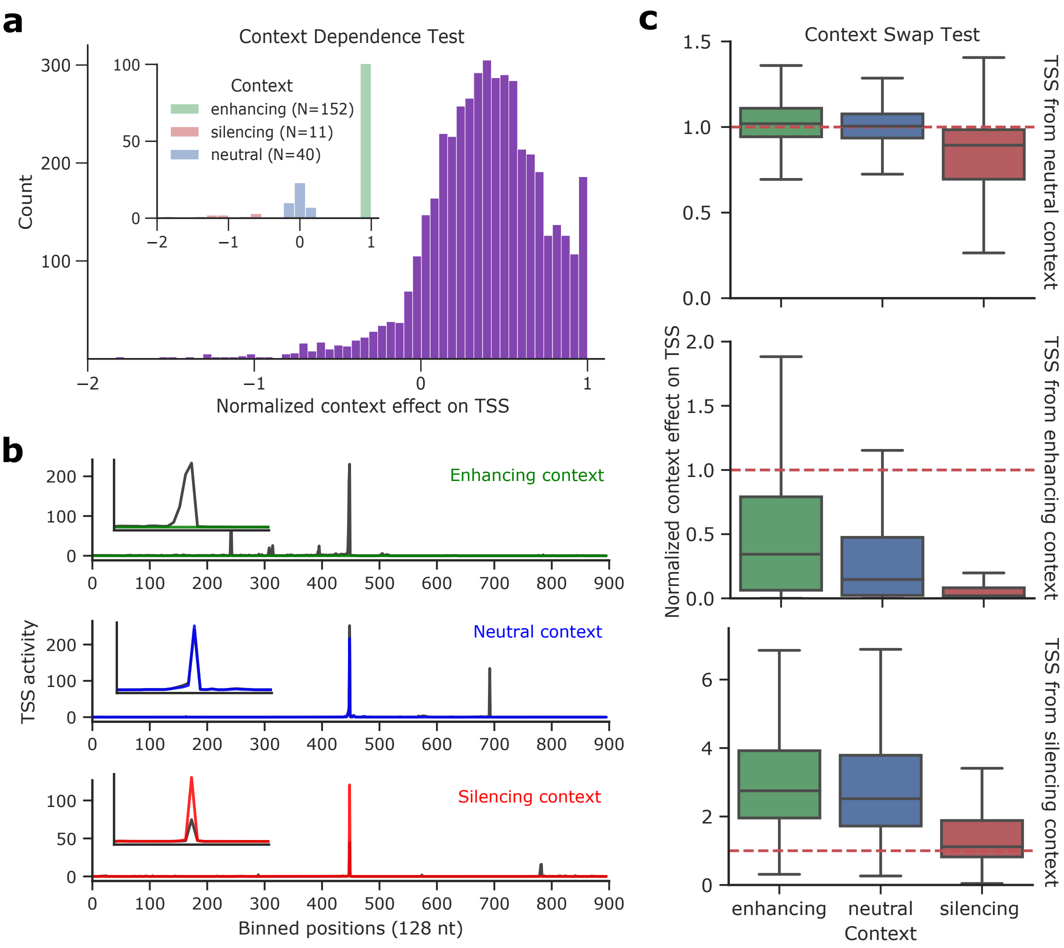


Figure 2. TSS robustness to context. **a**, Histogram of normalized context effect on TSS for 4,527 sequences that contain an active, annotated gene in K562 cells. The sequence context was perturbed via a dinucleotide-shuffle while keeping the central 5kb tile centered on a TSS-of-interest intact. Inset shows the subset of sequences for enhancing, silencing and neutral contexts. **b**, Representative sequences from the three context categories showing Enformer's predictions before and after a context perturbation, with a zoomed in version shown in the inset. **c**-**e**, Context Swap Test results. Box plots of normalized context effect on TSS for sequences with context perturbations given by insertion of the original TSS in different context categories. Results are organized according to the original TSS category: neutral (**c**, top), enhancing (**c**, middle), and silenced (**c**, bottom). The number of datapoints in each box-plot represent an all-vs-all comparison of each respective TSS in each possible context. Box plots show the first and third quartiles, the median (central line) and the range of data with outliers removed (whiskers).

by Enformer are located in close proximity to their target TSS, and the number of enhancers progressively diminishes as the distance increases (Fig. 3d), in agreement with previous observations[5]. On the other hand, silencers recognized by Enformer are more-or-less homogeneously distributed (Fig. 3d).

Does changing the distance between CRE and TSS alter its effect? Using CREME's *TSS-CRE Distance Test*, we performed a GIA experiment where the TSS activity was monitored while the distance of an enhancer from a target TSS was systematically varied in random sequences (see Methods). Surprisingly, we found that Enformer learns a similar distance dependence relationship, on average, across different enhancer-TSS pairs (Fig. 3e). This suggests that a weak enhancer can increase its effect on gene expression by moving closer to the TSS[36]. Similarly, silencer-TSS pairs did not exhibit any noticeable distance dependence.

How do CREs interact to regulate gene expression? Moving from single-tile perturbations to multi-tile perturbations, we used CREME's *Higher-order CRE Interaction Test* to identify minimal sets of CREs that maximally alters TSS activity (Fig. 4a). In anticipation that CRE interactions are complex[24,37,38], we elected to search for CRE sets via an iterative greedy search, instead of grouping CREs based on their individual effects. In each round, the greedy search identifies a new CRE (given the

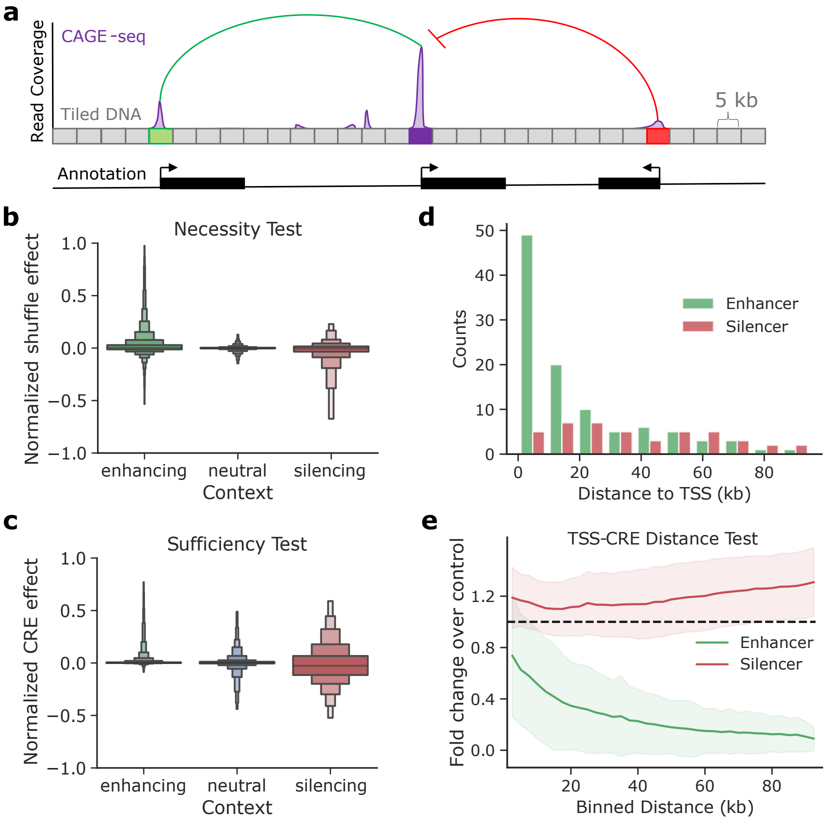


Figure 3. CRE effects on TSS activity. **a**, Schematic of tile perturbation experiments showing a TSS centered sequence with a toy gene annotation below, a CAGE track is shown above with putative up and down regulating CREs. **b**, CRE Necessity Test results. Normalized shuffle effect on TSS activity for tiles in enhancing, neutral and silencing context sequences. Normalized shuffle effect is calculated according to how the TSS activity changes upon tile shuffles, normalized by the original TSS activity. A value of 1 represents a strong enhancer and a value of 0 represents a strong silencer. **c**, CRE Sufficiency Test results. Normalized CRE effect of adding a given tile along with the TSS tile to dinucleotide-shuffled sequences in their original positions for different context categories. **b**,**c**, Boxen-plots have 5928, 1560, 429 context-derived tiles in enhancing, neutral, and silencing context, respectively. **d**, Histogram of the distance between CRE from TSS for enhancers and silencers, defined by activity thresholds from **c** (i.e., enhancers > 0.3 and silencers < -0.3). **e**, TSS-CRE Distance Test results. Average fold change over control of moving an CRE tile from a fixed TSS tile in shuffled sequences, where the control represents the TSS activity when the CRE tile is at its original position (i.e., similar to the CRE Sufficiency Test). The distance towards the 5' end and 3' end were averaged together for each CRE tile. Shaded region represents standard deviation of the mean.

set of the CREs found in previous rounds) that yields the largest effect size (see Methods) – optimizing for higher or lower TSS activity yields sets of silencers or enhancers, respectively.

When probing for enhancer sets, we observed that, on average, 5 enhancers drive more than 80% of TSS activity for sequences in enhancing contexts (Fig. 4b). In contrast, when probing for silencer sets, we found that silencing context is enriched for larger numbers of silencers, each with a smaller effect size (Fig. 4c). All contexts, including neutral context, contain enhancers and silencers, albeit with varying effect sizes. Together, this suggests that the overall net effect of enhancers and silencers drives TSS activity.

To help understand the CRE trajectories from the greedy search, we considered a hypothetical scenario where the overall influence of multi-tile perturbations on TSS activity follows an additive effects model (see Methods). We found that, according to Enformer, sets of multiple enhancers are largely additive on average (Fig. 4d), which is in contrast to previously observed multiplicative effects of pairs of CREs[5,32,38,39]. However, when stratifying the results, we observed that enhancers exhibit a range of complex behaviors, including redundancy and cooperativity. Redundancy refers to when multiple enhancers appear to each have a small effect size on TSS activity when perturbed individually (Fig. 4e), which arises due to the presence of other redundant enhancers[40–43]. Cooperativity refers to when two or more CREs depend on each other[24,37,38]. For instance, a perturbation to an individual CRE that is cooperating with another CRE will result in a large decrease in TSS activity as both

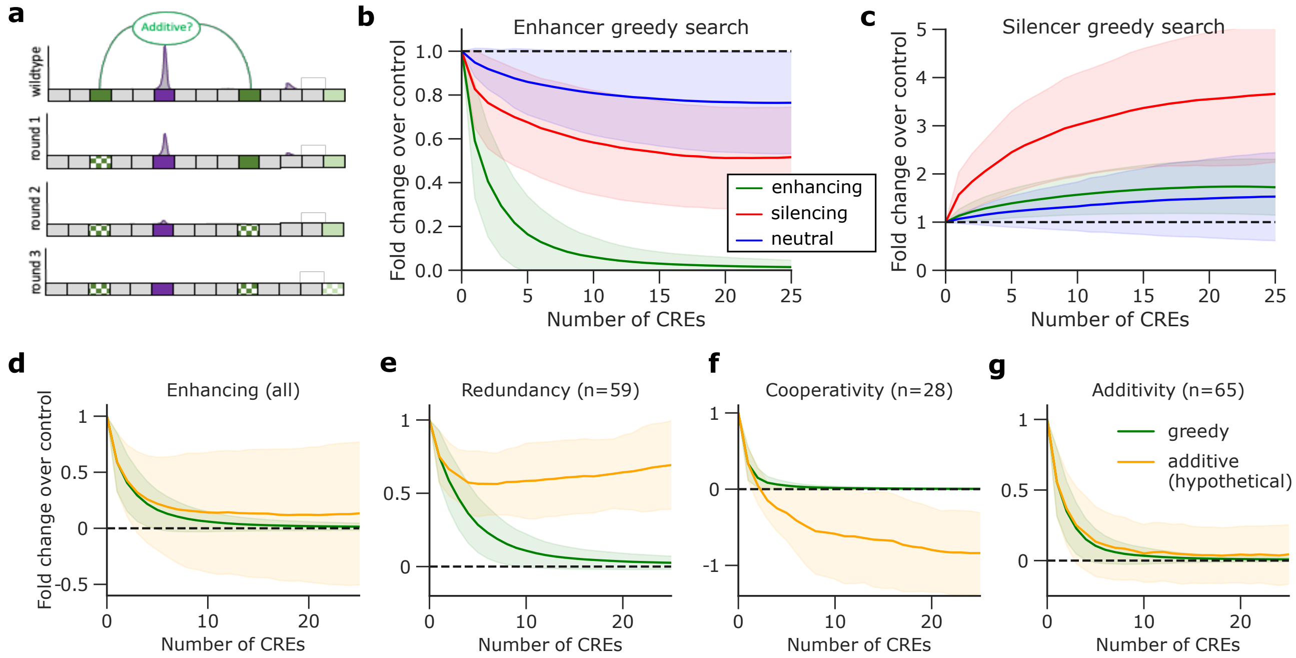


Figure 4. Optimal CRE sets reveal complex interactions. **a**, Schematic of the greedy search process for enhancer CRE sets. Checkered box represents a shuffled tile. **b**,**c** Average fold change over control of TSS activity with the shuffled CRE tiles in each round of the greedy search (indicated by the number of CREs) for sequences from different context categories optimizing for enhancer sets (**b**) and silencer sets (**c**). Control represents the TSS activity of wild-type. **d**, Comparison of the average fold change over control for enhancer sets for sequences categorized as enhancing context versus a hypothetical additive effects model. The 152 sequences from enhancing contexts are stratified according to interaction type, redundancy (**e**), cooperativity (**f**), and additivity (**g**). Shaded region represents standard deviation of the mean.

are necessary. In this case, the hypothetical additive effect will appear stronger than the observed effect when both CREs are perturbed, leading to negative TSS activities (Fig. 4f). As anticipated, the optimal CRE set identified by the greedy search do not reflect the order given by the largest individual CRE effects (Supplementary Fig. 3). This suggests that CREs exhibit strong dependencies, i.e. a CRE perturbation can influence the effect size of other CREs. Furthermore, we compared multi-tile perturbations with a hypothetical additive model for silencer sets and found that silencers largely exhibit strong redundancies (Supplementary Fig. 4).

Does Enformer learn a sigmoidal function of gene expression? In principle, CRE redundancy can be modeled as if TSS activity follows a sigmoidal function, where saturation has been reached[44–46]. To test Enformer's extrapolation behavior, we used CREME's *CRE Multiplicity Test* to measure the effect of greedily inserting multiple enhancers (or silencers) within dinucleotide-shuffled sequences to maximize (or minimize) TSS activity (see Methods). Indeed, we observed that Enformer's predictions of gene expression saturate as more enhancers (or silencers) are incorporated, albeit different genes plateau at different TSS activity levels (Supplementary Fig. 5).

Conclusion. In summary, CREME provides a suite of *in silico* experiments for hypothesis-driven interpretations of large-scale DNNs. CREME enables moving beyond the limitations of existing model interpertability methods that are geared towards motif analysis by focusing on a CRE-level analysis. This approach reveals how DNNs consider rules of gene regulation, such as the dependence of TSS activity on distal context and the complex coordination of CREs. CREME helps to prototype experiments and generate plausible hypotheses of *cis*-regulatory mechanisms. By interrogating Enformer, we found that TSS activity of genes are affected by the complex interactions of multiple enhancers and repressors. This suggests that perturbations with CRISPRi on a single locus (and in some cases pairs of loci) would be insufficient to fully characterize dependencies between CREs.

A major limitation arises when a DNN's understanding of gene regulation is not aligned with biological reality. Insights gained through model interpretability should therefore be treated as a hypothesis and not a replacement for laboratory-based experiments. As DNNs continue to improve, CREME is a general tool that will enrich our biological knowledge as well as to understand their shortcomings. Another issue is that the perturbed sequences may introduce an out-of-distribution shift[47], for which model predictions can be less reliable – not grounded in biology. By staying close to genomic sequences, performing multiple trials, and carefully considering control experiments, we aimed to limit the negative impacts of distributional shifts.

While this study focuses on the impact of CRE-level perturbations on gene expression predictions within K562 cells given by Enformer, CREME is general and can be applied to any sequence-based DNN at any desired resolution. By restricting the tile size, the experiments performed by CREME can also be used to study motifs and their interactions. CREME provides a roadmap to probe what *cis*-regulatory mechanisms the DNN has learned, moving beyond evaluations based on alignment to experimental data. In the future, we aim to expand CREME to incorporate tests to study enhancer-promoter compatibility rules at both the CRE- and motif-level, uncover local *cis*-regulatory networks, and design synthetic CREs that are optimal for a target gene.

Methods

Enformer

Enformer is a previously established DNN that takes as input genomic sequences of length 393,216 bp and predicts 5,313 epigentic tracks for human and 1,643 epigenetic tracks for mouse through two output heads[1]. For each track, Enformer's predictions cover 896 binned positions, with each bin representing 128 bp. This represents the central 114,688 bp of the input sequence. The extended input sequence, provides context for the edge cases, i.e. the start and end of the predictions. The epigenetic tracks consist of processed coverage values of expression (CAGE), DNA accessibility (DNase-seq/ATAC-seq), transcription factor binding and histone modification (ChIP-seq). Enformer is composed of convolutional layers that initially summarizes the input sequence into representations that represent 128 bp bins. This is followed by 11 transformer blocks that use multi-head self-attention[48]. We acquired code for the Enformer model along with trained weights from `https://tfhub.dev/deepmind/enformer/1` as per instructions in the Methods section of Ref.[1]

Transcription start site selection

We acquired human annotations from GENCODE[23] (`https://www.gencodegenes.org/human/`) and filtered for 'transcript' annotations. We then extracted sequences of length 393,216 from the hg19 reference geneome centered at each filtered TSS. We converted the sequences to a one-hot representation, treating N characters as a uniform probability (i.e. 0.25). We calculated Enformer's prediction for these sequences and only considered position 448 (of the 896 binned predictions), which corresponds to the central TSS, of track 5,111 of the human output head (corresponding to K562 CAGE predictions). We refer to this scalar coverage value as the TSS activity. To focus our study on genes that yield high TSS activity, we removed sequences from this set if the predicted TSS activity was below 30, our minimum activity threshold. To simplify interpretations of perturbation experiments, we focus on sequences where the central TSS is the highest expressed gene. This was accomplished by further filtering out sequences that where the max predicted coverage value was not located at the central bin (i.e. bin 448). Further, we filtered out duplicate genes from our list, reducing the set to 4,527 total sequences.

CREME: *cis*-Regulatory Element Model Explanations

CREME is an *in silico* perturbation assay toolkit that can uncover rules of gene regulation learned by a large-scale DNN. The rationale behind CREME stems from the concept that DNNs are function approximators. Thus, by fitting experimental data, the DNN is effectively approximating the underlying "function of the experimental assay". By treating the DNN as a surrogate for the experimental assay, CREME can be queried with new sequences and provide *in silico* "measurements", albeit through the lens of the DNN. Inspired by wetlab experiments, such as CRISPRi[21,22,49], that perturb genomic loci to uncover how CREs influence gene expression, we devised a suite of *in silico* perturbation experiments that interrogate a DNN's understanding of long-standing questions of gene regulation, including the context dependence of gene expression[27,50], identification of enhancing and silencing CREs and their target genes[21,26], distance dependence of CREs to target genes on gene expression, and the complex higher-order interactions of CREs on gene expression[31,32,41–44]. Since DNN predictions may not fully capture the underlying biology when fitting experimental data, CREME is strictly a model interpretability tool. Below, we detail the different *in silico* perturbation tests explored in this paper.

CREME investigation of Enformer. For the vast majority of the experiments, we only considered TSS activity, which we define as the central 5kb tile centered on the input sequence. Enformer's receptive field for this tile covers roughly 200kb sequences, so the 200kb region centered on the sequence is what is probed in our expeirments. We split the central 200kb sequences into 40 non-overlapping 5kb tiles, with the central tile corresponding to the TSS of an annotated gene. We define the *TSS activity* as the central bin in Enformer's prediction, i.e. position 448 of track 5,111 of the human output head. Note that we only consider perturbations to the central 200kb sequences, which spans the receptive field of the central TSS.

TSS Context Dependence Test

The *TSS Context Dependence Test* aims to measure the effect size of TSS activity in random contexts (derived from dinucleotide shuffled versions of the wild type sequence). This test measures the extent to which a prediction of a given TSS activity is influenced by its context which may contain enhancers and silencers. To perform the Context Dependence Test, we executed the following steps:

1. Predict TSS activity for the wild type sequence (denoted as WT).
2. Dinucleotide shuffle the sequence (except the 5kb tile centered at the TSS).
3. Predict TSS activity for the shuffled sequence (denoted as MUTANT).
4. *Normalization:* compute context effect on TSS using WT as control: (WT - MUTANT) / WT
5. Repeat steps 2-4 10 times and average across different random dinucleotide shuffles.

Interpretation. Effect size of 0 means that the context is neutral and has no effect on the TSS predictions (i.e. wild type and mutant yield the same prediction). Positive effect size means that the central TSS prediction for the mutated sequence is lower than wild type, which indicates that we have perturbed a enhancing context. Negative effect size means that the central TSS prediction for the mutated sequence is higher than wild type, which suggests that we have perturbed a silencing context.

Analysis. We categorized the sequences into silencing, neutral, and enhancing contexts based on their context effect on TSS. We identified 3 regions: (i) enhancing context were chosen based on an effect size of more than 0.95 (N=152), (ii) neutral context was chosen if the absolute effect size was less than 0.2 (N= 40), and (iii) silencing context was chosen based on an effect size of less than -0.5 (N=11). We used these groups (combined N=203) throughout the experiments.

TSS Context Swap Test

A TSS Context Swap Test aims to measure the extent that TSS activity depends on a specific genomic context. To perform the Context Swap Test, we executed the following steps:

1. Cut out the central TSS (5Kb tile) from the source sequence.
2. Insert the source TSS in each of the target sequences at the central TSS position, thereby replacing the existing TSS of the target sequence.
3. Predict TSS activity for the mutant sequence (denoted as MUTANT), the wild type target sequence (UNPERTURBED), and the wild type source sequence (WT).
4. *Normalization:* compute fold change over control according to: MUTANT-UNPERTURBED / WT

As a control, we considered a control experiment, where we performed a dinucleotide shuffle of the target sequences to test the extent of influence simply from dinucleotide frequencies within each context category.

1. Cut out the central TSS (5Kb tile) from the source sequence.
2. Dinucleotide shuffle the sequence contexts of the target sequences, keeping the central TSS intact.
3. Insert the source TSS in each of the target sequences at the central TSS position, thereby replacing the existing TSS of the target sequence.
4. Predict TSS activity for the mutant sequence (denoted as MUTANT), the shuffled target sequence (DINUC), and the wild type source sequence (WT).
5. *Normalization:* compute fold change over control according to: MUTANT-DINUC / WT

Analysis. We performed the Context Swap Test on the 203 sequences filtered by the Context Dependence Test. Specifically, we placed the TSSs in each context category across all other context categories, separately keeping track of the source TSS and the context category.

CRE Necessity Test

The CRE Necessity Test measures the importance of a CRE on the central TSS activity for a given sequence context while the other CRE tiles remain intact. To perform the CRE Necesity Test, we executed the following steps:

1. Predict TSS activity for the wild type sequence (WT).
2. For each 5Kb tile not overlapping with the central TSS:
   (a) Dinucleotide shuffle the 5Kb tile under investigation.
   (b) Predict TSS activity for the shuffled sequence (SHUFFLE).
   (c) Repeat 10 times and calculate the mean TSS activity.
3. *Normalization:* compute the normalized shuffle effect as: (WT - SHUFFLE) / WT

Analysis. We performed the CRE Necessity Test on the subset of sequences from Context Dependence Test that had enhancing, silencing or neutral backgrounds (as classified by selected thresholds).

CRE Sufficiency Test

The CRE Sufficiency Test measures the effect of a given CRE on its TSS in otherwise random context, i.e. in isolation from the rest of the CRE tiles from the original wild type sequence. This essentially measures whether the CRE by itself is enough to up or downregulate the TSS. To perform the CRE Sufficiency Test, we executed the following steps:

1. Predict TSS activity for the wild type sequence (WT).
2. Dinucleotide shuffle the sequence.
3. Add the TSS 5Kb tile and predict TSS activity (TSS-CONTROL).
4. Add the CRE and the TSS tiles to the sequence and predict TSS activity (CRE-TSS-MUTANT).
5. *Normalization:* compute the normalized CRE effect as (CRE-TSS-MUTANT - TSS-CONTROL) / WT
6. Repeat each shuffle 10 times and average the normalized CRE effect per sequence.

Analysis. We performed the CRE Sufficiency Test on the same subset of sequences as CRE Necessity Test that had enhancing, silencing or neutral backgrounds (as classified by selected thresholds), 203 sequences and 7917 tiles in total. Based on CRE Sufficiency Test results, we denote CREs within 0.3 - 0.5 range as weak enhancers, 0.5 and above as strong enhancers. Similarly, we define weak silencers as tiles with addition effect size between -0.3 and -0.5 and strong - the ones with values smaller than -0.5.

TSS-CRE Distance Test

TSS-CRE Distance Test is a GIA experiment where we systematically shift the position of a tile in shuffled sequences and measure its effect on TSS activity. For each CRE tile, we performed the TSS-CRE Distance Test by executing the following steps:

1. Predict TSS activity for the wild type sequence (WT).
2. Dinucleotide shuffle the sequence except the central 5kb tile and insert the CRE at its native position and predict TSS activity (denoted as CONTROL).
3. For each test position P:
   (a) Insert the CRE tile at position P in the dinucleotide shuffled sequence (with an intact TSS) and predict TSS activity (TEST).
   (b) *Normalization:* Compute the fold change over control as TEST / CONTROL
   (c) Repeat each shuffle 10 times and average the fold change over control per sequence.

Analysis. We used the definition of enhancers and silencers based on CRE Sufficiency Test results. We performed the TSS-CRE Distance Test on CREs defined as (strong and weak) enhancers within enhancing contexts (59 in total) and (strong and weak) silencers in silencing contexts (26 in total).

Higher-order CRE Interaction Test

The aim of Higher-order CRE Interaction Test is to dissect CRE networks. Specifically, we compute the combined effect of multiple tile shuffles that have large effect through a greedy search. For enhancers, the iterative greedy search systematically identifies tiles that lead to a lower TSS activity when shuffled. We followed the same steps for silencer search but instead of choosing the minimum predicted value we chose the maximum predicted value. To perform the Higher-order CRE Interaction Test, we executed the following steps:

1. Predict TSS activity for the wild type sequence (WT).
2. For each greedy search iteration:
   (a) For each tile that is not fixed (i.e. fixed tiles are central TSS and tiles selected from previous rounds):
      i. Dinucleotide shuffle the tile
      ii. Predict TSS activity for the mutant sequence (SHUFFLE-MUTANT)
      iii. *Normalization:* Compute the fold change over control, i.e. SHUFFLE-MUTANT/WT
      iv. Repeat each shuffle 5 times and average normalized output per sequence.
   (b) Fix the tile that yields the maximum effect on TSS activity. For enhancers, maximal decrease in TSS activity; for silencers, maximal increase in TSS activity. The shuffled version that is most representative is chosen when fixing. This is selected based on the instance that yields a prediction closest to the mean across 10 shuffles.
   (c) Repeat for the desired number of rounds in the greedy search or until the entire sequence is fixed.

Analysis. We performed Higher-order CRE Interaction Test for maximally enhancing TSS activity and maximally silencing TSS activity for all sequences from different context categories.

Comparison to additive effects. To help understand the trajectories from the Higher-order CRE Interaction Test, we calculated the hypothetical effects of an additive model. In brief, the additive effects are calculated based on combining the effects on TSS activity from the individual effects of each CRE (i.e. calculated in the first round of the greedy search), following the CRE tile order found by the greedy search. This does not take into account cooperative or redundant relationships within sets of CREs, as would be captured in the greedy search.

To compute the hypothetical additive effects for sequences categorized as enhancing context, we performed the following steps:

1. Predict TSS activity for the wild type sequence (WT).
2. Get the order of tile shuffling from greedy search results iteration 1 (denote ordered vector of tiles as T).
3. Compute effect sizes (E) of tile shuffles (as done in the 1st iteration of the greedy search). For enhancer search, compute as shuffled - WT. For silencer search, compute as WT - shuffled.
4. Following the tile order of T, calculate the cumulative sum of the individual tile effects (M_additive). This assumes an additive model assumption.
5. *Normalization:* compute the hypothetical fold change over control according to (M_additive)/WT.

Interpretation. If the hypothesis of additive effect holds, we would expect the tile greedy search trace to be the same as the additive or hypothetical trace for each sequence. Let us assume two enhancers are cooperating, i.e. their combined effect is larger than individual effects (a non-additive case). We would expect their individual shuffle effects to also be larger than shuffling them simultaneously (because disabling one leads to a large effect size already). We call such cases cooperative enhancer contexts. In contrast, if two enhancers are redundant, i.e. their roles are overlapping, the effect size will be small when only a single tile is shuffled (because the other enhancer tile can compensate). Therefore, the estimated additive effect (based on single tile shuffles of iteration 1) will underestimate the effect of shuffling both of the enhancers. This will thus lead to the additive hypothetical trace being higher than the one based on the greedy search.

Analysis. To characterize deviations, we computed the mean squared error of the greedy and hypothetical additive outputs for each sequence. We classified the cases where the MSE value is above 0.3 (arbitrary threshold) and the greedy search results on average is greater than the average of additive. Similarly, we classified the cases where the MSE value is above 0.3 (arbitrary threshold) and the greedy search results on average is lower than the average of additive as redundant cases.

CRE Multiplicity Test

The Multiplicity Test measures how TSS activity scales upon repeated addition of an enhancing or silencing tile. With this GIA experiment, we aim to test the model's extrapolation behavior. Specifically, we probed whether TSS activity reaches saturation upon over-representation of CRE context; saturation is when the predictions reach a plateau when we enrich for enhancers or silencers. The Multiplicity Test is similar to the greedy search used in the Higher-order CRE Interaction Test, with the exception that we are systematically adding the same CRE of interest into optimal positions in each round of dinucleotide shuffled sequences. To compute the Multiplicity Test, we performed the following steps:

1. Define the CRE tile and number of times the CRE will be inserted.
2. Dinucleotide shuffle the sequence while maintaining the central TSS tile intact (S).
3. Add the CRE of interest in its original position and predict TSS activity (CONTROL).
4. Systematically scan the CRE of interest and measure predicted TSS activity at each unfixed position (MUTANT_POSITION).
5. *Normalization:* calculate the fold change over control given by MUTANT_POSITION / CONTROL
6. Fix the CRE at the position where it maximally affects TSS activity.
7. Repeat steps 3 to 6 until the number of insertions is complete.

Analysis. Using the strong and weak enhancers defined in CRE Sufficiency Test, we performed 10 iterations of the Multiplicity test for each CRE.

Data Availability

Results from this paper is deposited at zenodo: `doi.org/10.5281/zenodo.8111754`.

Code Availability

Open-source code to deploy CREME can be found at GitHub: `https://github.com/p-koo/creme-nn`. The code for reproducing the analyses in the manuscript is available at GitHub: `https://github.com/shtoneyan/CREME`.

Acknowledgements

The authors would like to thank Jack Desmarais and members of the Koo Lab for helpful comments on the manuscript. Research reported in this publication was supported in part by the National Human Genome Research Institute of the National Institutes of Health under Award Number R01HG012131, the Developmental Funds from the CSHL Cancer Center Support Grant 5P30CA045508, and the Simons Center for Quantitative Biology at Cold Spring Harbor Laboratory. This work was performed with assistance from the US National Institutes of Health Grant S10OD028632-01. We would also like to thank the support of the NVIDIA GPU Grant Program.

Author contributions

ST and PKK conceived of the method and designed the experiments. ST developed code, ran the experiments, and plotted the results. ST and PKK interpreted the results and contributed to writing the paper.

Competing interests

Nothing to declare.

References

1. Avsec, Ž. *et al.* Effective gene expression prediction from sequence by integrating long-range interactions. *Nat. Methods* **18**, 1196–1203 (2021).
2. Karbalayghareh, A., Sahin, M. & Leslie, C. S. Chromatin interaction–aware gene regulatory modeling with graph attention networks. *Genome Res.* **32**, 930–944 (2022).
3. Kelley, D. R. *et al.* Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Res.* **28**, 739–750 (2018).
4. Toneyan, S., Tang, Z. & Koo, P. K. Evaluating deep learning for predicting epigenomic profiles. *Nat. Mach. Intell.* 1–13 (2022).
5. Karollus, A., Mauermeier, T. & Gagneur, J. Current sequence-based models capture gene expression determinants in promoters but mostly ignore distal enhancers. *Genome Biol.* **24**, 1–29 (2023).
6. Kircher, M. *et al.* Saturation mutagenesis of twenty disease-associated regulatory elements at single base-pair resolution. *Nat. Commun.* **10**, 3583 (2019).
7. Arnold, C. D. *et al.* Genome-wide quantitative enhancer activity maps identified by starr-seq. *Science* **339**, 1074–1077 (2013).
8. Qi, L. S. *et al.* Repurposing crispr as an rna-guided platform for sequence-specific control of gene expression. *Cell* **152**, 1173–1183 (2013).
9. Sasse, A. *et al.* How far are we from personalized gene expression prediction using sequence-to-expression deep neural networks? *bioRxiv* 2023–03 (2023).
10. Huang, C. *et al.* Personal transcriptome variation is poorly explained by current genomic deep learning models. *bioRxiv* 2023–06 (2023).
11. Simonyan, K., Vedaldi, A. & Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv 1312.6034* (2013).
12. Lundberg, S. & Lee, S.-I. A unified approach to interpreting model predictions. *arXiv 1705.07874* (2017).
13. Shrikumar, A., Greenside, P. & Kundaje, A. Learning important features through propagating activation differences. *arXiv 1704.02685* (2017).
14. Koo, P. K., Majdandzic, A., Ploenzke, M., Anand, P. & Paul, S. B. Global importance analysis: An interpretability method to quantify importance of genomic features in deep neural networks. *PLoS Comput. Biol.* **17**, e1008925 (2021).
15. Koo, P. K. & Ploenzke, M. Improving representations of genomic sequence motifs in convolutional networks with exponential activations. *Nat. Mach. Intell.* **3**, 258–266 (2021).
16. Hammelman, J. & Gifford, D. K. Discovering differential genome sequence activity with interpretable and efficient deep learning. *PLoS Comput. Biol.* **17**, e1009282 (2021).
17. Liu, G., Zeng, H. & Gifford, D. K. Visualizing complex feature interactions and feature sharing in genomic deep neural networks. *BMC Bioinforma.* **20**, 1–14 (2019).
18. Greenside, P., Shimko, T., Fordyce, P. & Kundaje, A. Discovering epistatic feature interactions from neural network models of regulatory dna sequences. *Bioinformatics* **34**, i629–i637 (2018).
19. Jha, A., K Aicher, J., R Gazzara, M., Singh, D. & Barash, Y. Enhanced integrated gradients: improving interpretability of deep learning models using splicing codes as a case study. *Genome Biol.* **21**, 1–22 (2020).
20. Linder, J. *et al.* Interpreting neural networks for biological sequences by learning stochastic masks. *Nat. Mach. Intell.* **4**, 41–54 (2022).

21. Fulco, C. P. *et al.* Systematic mapping of functional enhancer–promoter connections with crispr interference. *Science* **354**, 769–773 (2016).

22. Gasperini, M. *et al.* A genome-wide framework for mapping gene regulation via cellular genetic screens. *Cell* **176**, 377–390 (2019).

23. Frankish, A. *et al.* Gencode 2021. *Nucleic Acids Res.* **49**, D916–D923 (2021).

24. Lin, X. *et al.* Nested epistasis enhancer networks for robust genome regulation. *Science* **377**, 1077–1085 (2022).

25. Goel, V. Y., Huseyin, M. K. & Hansen, A. S. Region capture micro-c reveals coalescence of enhancers and promoters into nested microcompartments. *Nat. Genet.* 1–9 (2023).

26. Pang, B. & Snyder, M. P. Systematic identification of silencers in human cells. *Nat. Genet.* **52**, 254–263 (2020).

27. Stampfel, G. *et al.* Transcriptional regulators form diverse groups with context-dependent regulatory functions. *Nature* **528**, 147–151 (2015).

28. Kulkarni, M. M. & Arnosti, D. N. cis-regulatory logic of short-range transcriptional repression in drosophila melanogaster. *Mol. Cell. Biol.* **25**, 3411–3420 (2005).

29. Doni Jayavelu, N., Jajodia, A., Mishra, A. & Hawkins, R. D. Candidate silencer elements for the human and mouse genomes. *Nat. Commun.* **11**, 1061 (2020).

30. Hounkpe, B. W., Chenou, F., de Lima, F. & De Paula, E. V. Hrt atlas v1. 0 database: redefining human and mouse housekeeping genes and candidate reference transcripts by mining massive rna-seq datasets. *Nucleic Acids Res.* **49**, D947–D955 (2021).

31. Martinez-Ara, M., Comoglio, F., van Arensbergen, J. & van Steensel, B. Systematic analysis of intrinsic enhancer-promoter compatibility in the mouse genome. *Mol. Cell* **82**, 2519–2531 (2022).

32. Bergman, D. T. *et al.* Compatibility rules of human enhancer and promoter sequences. *Nature* **607**, 176–184 (2022).

33. Narita, T. *et al.* The logic of native enhancer-promoter compatibility and cell-type-specific gene expression variation. *bioRxiv* 2022–07 (2022).

34. Gunsalus, L. M., Keiser, M. J. & Pollard, K. S. In silico discovery of repetitive elements as key sequence determinants of 3d genome folding. *bioRxiv* 2022–08 (2022).

35. Catarino, R. R. & Stark, A. Assessing sufficiency and necessity of enhancer activities for gene expression and the mechanisms of transcription activation. *Genes & Dev.* **32**, 202–223 (2018).

36. Fulco, C. P. *et al.* Activity-by-contact model of enhancer–promoter regulation from thousands of crispr perturbations. *Nat. Genet.* **51**, 1664–1669 (2019).

37. Choi, J. *et al.* Evidence for additive and synergistic action of mammalian enhancers during cell fate determination. *Elife* **10**, e65381 (2021).

38. Zhou, J. L., Guruvayurappan, K., Chen, H. V., Chen, A. R. & McVicker, G. P. Genome-wide analysis of crispr perturbations indicates that enhancers act multiplicatively and without epistatic-like interactions. *bioRxiv* 2023–04 (2023).

39. Hong, C. K. & Cohen, B. A. Genomic environments scale the activities of diverse core promoters. *Genome Res.* **32**, 85–96 (2022).

40. Kvon, E. Z., Waymack, R., Gad, M. & Wunderlich, Z. Enhancer redundancy in development and disease. *Nat. Rev. Genet.* **22**, 324–336 (2021).

41. Frankel, N. *et al.* Phenotypic robustness conferred by apparently redundant transcriptional enhancers. *Nature* **466**, 490–493 (2010).

42. Osterwalder, M. *et al.* Enhancer redundancy provides phenotypic robustness in mammalian development. *Nature* **554**, 239–243 (2018).

43. Perry, M. W., Boettiger, A. N. & Levine, M. Multiple enhancers ensure precision of gap gene-expression patterns in the drosophila embryo. *Proc. Natl. Acad. Sci.* **108**, 13570–13575 (2011).

44. Crocker, J., Ilsley, G. R. & Stern, D. L. Quantitatively predictable control of drosophila transcriptional enhancers in vivo with engineered transcription factors. *Nat. Genet.* **48**, 292–298 (2016).

45. Melen, G. J., Levy, S., Barkai, N. & Shilo, B.-Z. Threshold responses to morphogen gradients by zero-order ultrasensitivity. *Mol. Syst. Biol.* **1**, 2005–0028 (2005).

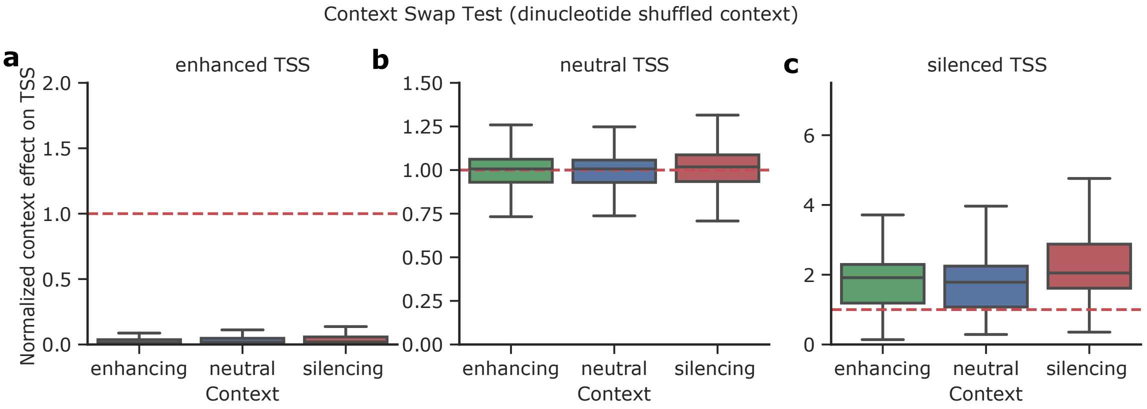
46. Burz, D. S., Rivera-Pomar, R., Jäckle, H. & Hanes, S. D. Cooperative dna-binding by bicoid provides a mechanism for threshold-dependent gene activation in the drosophila embryo. *The EMBO J.* **17**, 5998–6009 (1998).

47. Ovadia, Y. *et al.* Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Adv. Neural Inf. Process. Syst.* **32** (2019).

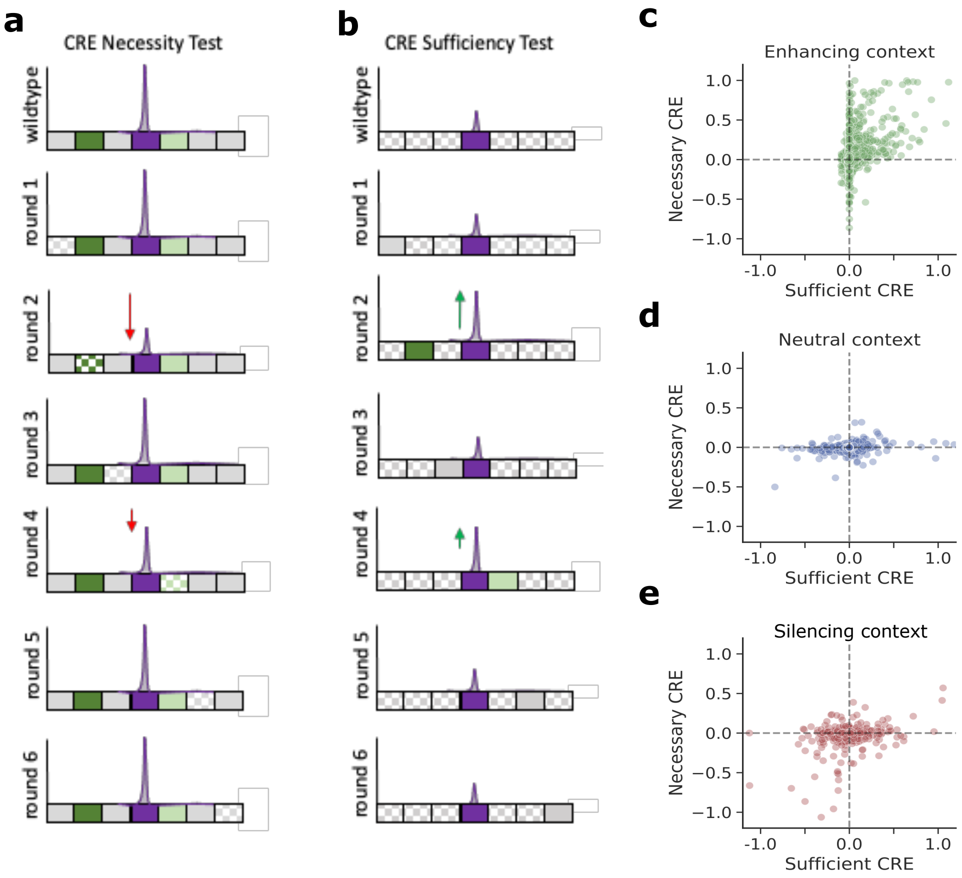
48. Vaswani, A. *et al.* Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017).

49. Chen, P. B. *et al.* Systematic discovery and functional dissection of enhancers needed for cancer cell fitness and proliferation. *Cell Reports* **41**, 111630 (2022).

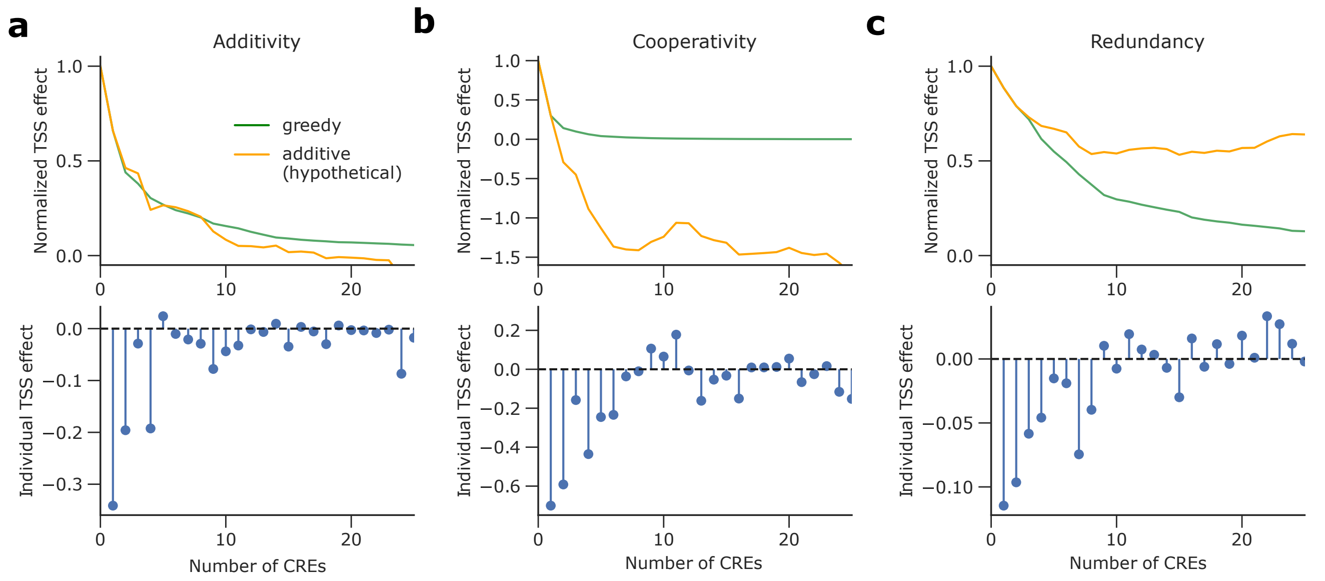
50. Crocker, J. *et al.* Low affinity binding site clusters confer hox specificity and regulatory robustness. *Cell* **160**, 191–203 (2015).



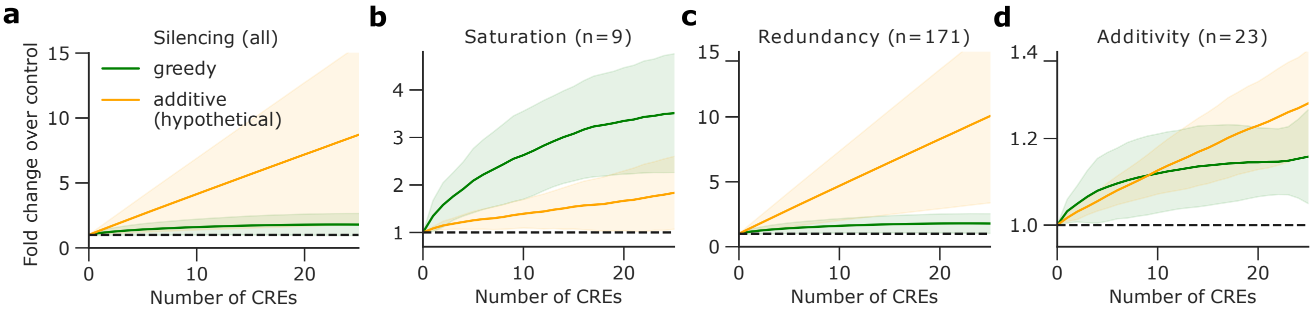
Supplementary Figure 1. Context Swap Test results for dinucleotide shuffled sequences. **a**-**c**, Box plots of normalized context effect on TSS for sequences with context perturbations given by a dinucleotide-shuffle of the sequences from different context categories. Results are organized according to the original TSS category: neutral (**a**), enhancing (**b**), and silenced (**c**). The number of data points in each box-plot represent an all-vs-all comparison of each respective TSS in each possible context. Box plots show the first and third quartiles, the median (central line) and the range of data with outliers removed (whiskers).



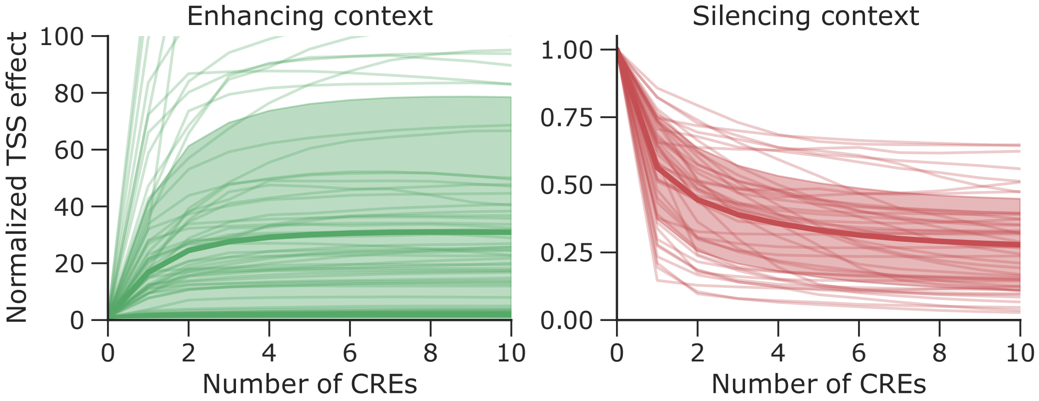
Supplementary Figure 2. Comparison of CRE necessity and sufficiency. **a**,**b**, Schematic of the tile perturbation process for CRE Necessity Test (**a**) and the CRE Sufficiency Test (**b**) in a toy sequence categorized as enhancing context. Checkered tile represents shuffled sequence. **c**-**e**, Scatter plot of the results for matched CRE tiles from the Necessity Test versus the Sufficiency Test for sequences categorized as enhancing context (**c**), neutral context (**d**), and silencing context (**e**). Each dot represents a different sequence, which contains a different shuffled tile. The value for Necessary CREs is given by the normalized shuffle effect size, which is calculated by the average change in TSS activity upon 10 tile shuffles, normalized by the original TSS activity. The Sufficiency CREs represent the normalized CRE effect of adding a given tile along with the TSS tile to dinucleotide-shuffled sequences in their original positions for different context categories. Values represent the average across 10 dinucleotide-shuffled sequences.



Supplementary Figure 3. Comparison of order of CREs from greedy search versus individual CRE effect size. Top row, Example comparison of the normalized TSS effect for a representative sequence categorized as enhancing context versus a hypothetical additive effects model for different interaction types, additivity (**a**), cooperativity (**b**), and redundancy (**c**). Bottom row, the corresponding effect size f individual CREs on TSS activity following the same order as the greedy search. Normalization is given according to the fold change over wild type TSS activity. Non-monotonicity demonstrates that greedy search leads to more effective interaction sets than the naive approach of grouping CREs according to their individual effects.



Supplementary Figure 4. Comparison of silencer sets from greedy search versus a hypothetical additive model. **a**, Comparison of the average fold change over control for silencer sets for sequences across all context categories versus a hypothetical additive effects model. The 203 sequences from all context categories are stratified according to interaction type, redundancy (**b**), cooperativity (**c**), and additivity (**d**). Shaded region represents standard deviation of the mean.



Supplementary Figure 5. Saturation of gene expression predictions. The results from a CRE Multiplicity Test applied to sequences from enhancing context (left) and silencing context (right). Each line represents a CRE identified as an enhancer (or silencer) and placed in dinuclotide shuffled sequence along with the central TSS tile. Each greedy search round adds the same CRE along the sequence from the previous round in a location that aims to maximize (or minimize) TSS activity. 10 dinucleotide shuffled sequences were explored for each CRE. The normalized TSS effect represents the TSS activity of the mutated sequence divided by the TSS activity of the control, which is the shuffled sequence with the TSS tile and the CRE in their original positions (i.e. the same as the CRE Sufficiency Test). The average across all CREs is shown with a thicker line and the shaded region represents the standard deviation of the mean.