Deciphering the 3D genome organization across species from Hi-C data

Aleksei Shkolikov [1,2,3,*], *Aleksandra Galitsyna* [4,*], Mikhail Gelfand [5]

[1] *Faculty of Bioengineering and Bioinformatics, M.V. Lomonosov Moscow State University, Moscow, 119991, Russia*

[2] *Kharkevich Institute of Information Transmission Problems, Moscow, 127051, Russia*

[3] *Vavilov Institute of General Genetics, Moscow, 119991, Russia*

[4] *Independent Researcher*

[5] *Center for Molecular and Cellular Biology, Moscow, Russia*

** These authors contributed equally*

Key takeaways:

- We present Chimaera, a neural network that predicts genomic interaction maps from DNA sequences.
- Chimaera's unique architecture integrates a Hi-C autoencoder with a DNA encoder, allowing it to identify recurring chromatin structures and their associated DNA sequence patterns.
- Chimaera offers a unified metric for quantifying various 3D-genome features, such as insulation, loops, stripes, and fountains/jets.
- Chimaera enables the exploration and validation of complex hypotheses regarding the principles of 3D-genome folding, including sequence-specific folding patterns at protein binding sites and genes.
- We apply Chimaera to various tasks in chromatin biology across multiple species, demonstrating its ability to predict the consequences of genomic rearrangements, mutations, and gene insertions.

Abstract

Three-dimensional (3D) genome organization plays a critical role in gene expression regulation and function. Recent advances in Hi-C and Micro-C data across various species provide insights into the mechanisms governing 3D genome formation, such as loop extrusion. While visual patterns like topologically associating domains (TADs) and loops are conserved across species, the underlying biological mechanisms may differ. Both species-specific architectural factors and DNA sequences influence chromatin folding, complicating comparative studies on the evolution of 3D organization of the genome. This work leverages existing Hi-C data and machine learning to explore species-specific 3D genome folding mechanisms and predict chromatin structures from DNA sequences. Here, we present ***Chimaera*** (convolutional neural network for Hi-C maps prediction using autoencoder for maps representation), a neural network that not only predicts Hi-C maps from DNA sequence, but also enables the search, quantification, and interpretation of associations between DNA sequences and 3D genome patterns.

Firstly, we demonstrate that Chimaera predicts Hi-C or Micro-C contact maps from DNA sequences, enabling the interpretation and extraction of key biological mechanisms. By exploring the latent representations generated by Chimaera, we offer a tool for building an unsupervised atlas of chromatin features such as insulation, loops,

stripes, and fountains/jets. We demonstrate the capabilities of Chimaera by detecting and quantifying signatures of insulation and fountains in Hi-C data, applying it to well-characterized biological processes like the cell cycle and embryogenesis. Additionally, we perform a targeted search for DNA sequence elements associated with specific chromatin structures, advancing our understanding of genome organization.

By extending the search of DNA sequence elements to multiple species, we confirm the role of CTCF in generating insulation patterns in vertebrates and BEAF-32 in *Drosophila*, and identify motifs previously not reported in mouse and *Drosophila*. In *Dictyostelium*, Chimaera demonstrates the importance of gene arrangement on the DNA strand for the formation of loops, confirming the hypothesis about the impact of convergent gene positioning on 3D genome organization in this amoeba. A pronounced but diverse effect of genes is evident when predicting chromatin interactions in other organisms.

Finally, we train the model on data from one species and then apply it to cross-predict how the genomes of other organisms might fold within the cellular environment of the original species. We thereby test whether chromatin folding patterns are transferable between species and reveal evolutionary similarities across genomes by building a chromatin-based cluster tree of species ranging from plants to mammals.

Introduction

Three-dimensional genome (3D) organization plays an indispensable role in gene expression regulation and function. With the growing volume of Hi-C/Micro-C data for both model and non-model species, we are gaining insights into diverse mechanisms that govern the formation of 3D genome patterns.

One of the most well-known mechanisms of genome organization is loop extrusion [1], which folds DNA and facilitates promoter-enhancer communication in higher eukaryotes [2–5]. Loop extrusion generates (1) on-diagonal, block-enriched chromatin interactions on Hi-C maps, known as **topologically associating domains (TADs)** [1,6], (2) dot-like enriched interactions of locus pairs, or **loops** [1,7], (3) enriched interactions of a single genomic locus with neighboring regions, or **stripes** [1,8], and (4) enriched interactions emanating from a single on-diagonal locus that spread across a characteristic distance, also called **fountains or jets** [9,10]. Although the disruption of TAD boundaries has been linked to severe disorders [11], some of these patterns may simply result from the mechanisms that form them without having specific function [12].

In evolutionary terms, Hi-C visual patterns can appear similar across species: TADs were first discovered in mammals [6], then observed in *Drosophila* [13,14] and *Arabidopsis* [15], and later found in other species [16]. Loops have been consistently detected across multiple species, including humans, *Drosophila*, *Caenorhabditis elegans*, and *Aedes aegypti* [17]. Fountains/jets have been identified in *C. elegans* [18,19], certain mouse cell types and mutants [9,10,20], and during zygotic genome activation in zebrafish, *Xenopus tropicalis*, and medaka [9].

However, visual similarities may not reflect the conservation of the biological mechanisms underlying the formation of these patterns. In mammals, TADs appear as an ensemble feature across multiple nuclei, while each individual nucleus is folded by loop extrusion with diverse domain organization [21]. Loop extrusion stalls at CTCF barriers, forming TAD boundaries, bright corner peaks (also called loops), and stripes at TAD borders in bulk Hi-C [1,7]. In *Drosophila*, however, TADs are more Suppl. Table features of individual cells [22], usually lack corner peaks [23], and may be formed mainly due to histone modifications [22,24]. The role of loop extrusion in *Drosophila* remains debated [22,25,26], suggesting it may not function in the same regulatory capacity as it does in mammals.

A complementary feature of TADs is the avoidance of interactions across specific genomic regions, also known as TAD boundaries or areas of **insulation**. In mammals, CTCF is a key extrusion-dependent insulator, a protein causing insulation at its binding site [7]. In *Drosophila*, multiple proteins can be found at insulating boundaries, including BEAF-32, Su(Hw), CTCF and others [14,27,28]. Recent studies suggest that genes themselves can serve as insulating elements [29,30], even without loop extrusion in yeast [31].

Thus, 3D genome organization is species-specific in a broad sense, meaning that the rules governing chromatin folding can differ significantly over large evolutionary distances. More narrowly, 3D genome organization is species-specific due to variations in DNA sequence and the positioning and properties of architectural factor binding sites. For example, CTCF sites are often conserved throughout mammalian evolution [32] and deuterostomes more generally [33], leading to conserved TAD organization in syntenic regions. The evolutionary

stability of TADs has been linked to conserved gene regulation, with well-studied examples including *Hox* loci [34], *Six* homeobox genes [33], and globin genes [35].

Evolutionary studies of the 3D genome usually focus on a single source of variability between species: (1) evolution of chromatin factors or (2) evolution of DNA sequence. The studies of the first type investigate species-specific chromatin factors across the nucleus and analyze general 3D genome patterns from Hi-C maps [16,36]. The studies of the second type assume conserved folding mechanisms but focus on syntenic genomic regions conserved through evolution [32,37–39], or even specific conserved genes [33,35,40].

A major breakthrough in the field came with the development of neural networks capable of predicting chromatin structure from DNA sequences [41–43]. These models learn DNA folding rules and can be tested on unseen sequences, such as rearranged cancer DNA [42]. This approach enabled a unique experiment never performed *in vivo*: visualizing the chromatin folding of one species' DNA within another species' cellular environment, as simulated by the neural network Akita [41]. Interpreting these cross-species predictions revealed significant differences in chromatin folding mechanisms between mouse and human: CTCF binding at B2 SINE elements in mice is hindered, resulting in a loss of insulation at these CTCF sites, but not in human [41].

In our study, we leverage the wealth of existing Hi-C and Micro-C datasets from diverse species, ranging from plants to mammals, and build upon the idea of interpretable machine learning to predict DNA folding patterns from DNA sequence. We introduce ***Chimaera***, a neural network-based tool for predicting species-specific 3D genome organization. Chimaera's autoencoder-based architecture [44] combined with a two-step training process that first learns Hi-C patterns and then links them to DNA sequence, enables its application to various tasks in chromatin biology. These tasks include predicting 3D structures from DNA sequences, searching for and quantifying Hi-C patterns, and interpreting associations between DNA sequences and 3D genome patterns. This work presents the first case study of species-specific 3D genome folding mechanisms analyzed through neural networks across a wide variety of organisms, shedding light on the conservation and variability of chromatin folding in evolution, and paving the way for future advancements in understanding 3D genome evolution and distal gene regulation.

Results

We start by designing a machine learning model (Fig. 1) that has DNA sequence as an input, predicts a Hi-C map for a given region, and encodes the information about the genome folding in a lower-dimensional vector, akin to word2vec for text [45] or DNABert for DNA sequences [46]. We adopt a two-step training process that first learns Hi-C patterns and then links them to DNA sequence.

Chimaera learns representations of Hi-C maps

The first part of the Chimaera model – the Hi-C encoder – is trained to encode the Hi-C map via low-dimensional latent representations (Fig. 1a). We first normalize input matrices of Hi-C interactions by iterative correction to avoid sequencing and mappability biases [47]. Contact frequency is known to vary over several orders of magnitude and drop rapidly with larger genomic distances [48,49], which might complicate the training and ability to automatically detect delicate patterns in Hi-C maps. We thus normalize contact frequencies by expected for each genomic separation [50]. These observed-over-expected Hi-C maps are then 45-degree rotated, truncated to remove noisy long-range interactions, snipped into fragments and subjected to a convolutional autoencoder. When Pearson correlation between true and reconstructed maps stops growing, we stop the training, check the quality of reconstruction and denoising of Hi-C matrices (Fig. 2a-d) and freeze the weights of the Hi-C decoder.

The following properties of the latent space are important: (1) the zero vector corresponds to a map with uniform interactions and no other specific structures (Suppl. Fig. 2b-c), and (2) the latent space is continuous, so that a small shift in any direction and from any point yields small changes in the decoded map (Fig. 2e). As an illustration of these properties, we implemented *in silico* generation of a given pattern in the center of Hi-C map by a stepwise movement in the latent space. For example, by moving along the vector encoding the insulation pattern, we can generate insulation in the center of the real Hi-C map in the central bin (Fig. 1d-e, 2f).

Notably, the Hi-C encoder can embed not only real Hi-C datasets but also succeeds at embedding artificial user-supplemented maps of interactions, such as stereotypical insulation, dot, or fountain (Fig. 1d-e). Of note, the insulation and fountain patterns are distant from each other in the embedded space, suggesting that they are different and rarely overlap in real Hi-C maps.

Chimaera enables quantitative calling of loops, fountains and insulation in Hi-C maps

We next asked whether the latent space representation allows us to detect typical patterns of genome organization, such as loops, insulation, and fountains/jets. Motivated by our observation of the continuity of the latent space and significant spread of representations of typical structures there, we defined a score of similarity to a given, already known, **pattern of the Hi-C map** (e.g., the insulation triangle of depleted interactions). For that, we projected the target genomic bin (the one queried for the presence of the pattern) to the vector corresponding to a map with the known pattern in the latent space. We then used this to study the development of three features (insulation, fountain, and loop) over the course of the mouse cell cycle [51] and fish embryogenesis [9] (Fig. 3).

At the early stages of both processes, we observed very small similarities to any of the studied patterns, suggesting little prominence of these structures in Hi-C maps at these stages (Fig. 3a,b). However, the similarities grew with the progression into either the cell cycle or the embryo development (Suppl. Fig. 3). We calculated the proportion of the genomic regions scoring above a given threshold and treated them as hits – for example, insulating boundaries for the map with “ideal” insulation pattern. This allowed us to quantitatively trace the changes in features of Hi-C maps in these biological processes (Fig. 3c,d).

In the cell cycle, all features gradually progress from prometaphase to late G1 (Fig. 3c). Notably, only isolated loops are present in mitosis, while both loops and TADs predominate in G1 (Fig. 3a). In fish embryogenesis, insulation remains relatively low and loops become more prominent only at the last stage, while fountains progress over time (Fig. 3b,d).

Together, these results indicate that the latent space of Chimaera preserves the information about typical patterns of the 3D genome organization (insulation, fountains, and loops), and is suitable for differentiating these patterns, akin to semantic embedding in natural language processing [52].

Chimaera predicts 3D genome architecture from DNA sequence across diverse species

The second part of the Chimaera model — the DNA encoder — is trained after the Hi-C encoder is fixed. The DNA encoder takes a DNA sequence as the input and learns to predict the representation of the Hi-C map for the respective genomic region in the latent space (Fig. 1b-c). By combining the pre-trained Hi-C decoder with the DNA encoder, we achieve the linkage between the genomic sequence and characteristic patterns of the genome organization.

To test this capability of Chimaera, we predicted the effects of a genomic rearrangement associated with congenital F-hand syndrome (Fig. 4). While the Hi-C map for this genetic variant is not available, Chimaera predicts that, following the rearrangement, human cells form a domain around the Wnt6 gene enhancer that also includes the Wnt6 gene itself. This configuration provides a basis for the deregulation of Wnt6, which plays a crucial role in embryonic morphogenesis.

Chimaera belongs to the model category with a relatively small receptive field [53] (500 Kb), and has similar number parameters and computational resource requirements to existing alternatives (~3-5M parameters as opposed to ~700K parameters for Akita [41] and ~4M parameters for ORCA [42]). Thus, we did not aim to achieve significantly better performance of prediction against Akita [41] or Orca [42], but instead aimed to improve the interpretability of the model. Yet, when we tested Chimaera against an adapted version of Akita, we observed a better or similar prediction quality (Suppl. Table 2).

Next, we studied the limits of Chimaera's ability to learn complex DNA and chromatin folding patterns. We trained Chimaera on data from 17 different organisms, including animals, plants, fungi, and amoebozoans (see Suppl. Fig. 4, also Suppl. Table 1 for training scheme details for different species).

For species with high-quality Micro-C data, we observed substantially better prediction performance (median Pearson correlation: 0.78 in human, 0.62 in mouse, 0.75 in *Drosophila melanogaster,* 0.67 in *Saccharomyces cerevisiae*). However, some datasets showed lower prediction accuracy (e.g., 0.32 for *Culex quinquefasciatus*). We hypothesized that this is due to internal data quality and applied a transfer learning approach. By pre-training models on one species' Micro-C data and fine-tuning them for taxonomically close ones, we improved prediction performance. For *Danio rerio*, Chimaera trained on original data reached correlations of ~0.4, while pre-training on human data followed by fine-tuning on *Danio rerio* improved correlations to ~0.6 (Suppl. Fig. 6). Without fine-tuning, the human model achieved ~0.4 correlation on *Danio rerio*, indicating conserved chromatin folding patterns among vertebrates. Interestingly, transfer learning from *Drosophila* to closely related insects (mosquitos and bee) was not successful (data not shown).

For all species, we achieved significant quality of the prediction (Suppl. Table 1, Suppl. Fig. 2). Notably, the models for species trained with Micro-C data, known for its higher resolution and feature sharpness [54], exhibited superior training quality on average. This suggests that data resolution and quality are significant factors for successful training . The highest Pearson correlations were observed for *Homo sapiens* (median 0.84), *Apis cerana* (0.89), *Drosophila melanogaster* (0.83), *Saccharomyces cerevisiae* (0.81), and *Dictyostelium discoideum* (0.83).

Chimaera confirms the importance of known motifs for chromatin structure

Next, we aimed to confirm that Chimaera learns meaningful patterns from DNA sequences, performing ***in silico* mutagenesis by insertion** of known motifs of architectural proteins in different species.

As expected, insertions of several CTCF sites significantly affects predictions of models in Tetrapods (mouse, human, and frog), causing the values in the maps to change by up to 2 standard deviations (Fig. 5a,e). Insertion of this motif causes insulation at the position of insertion, while two loci of convergent sites form a loop between them (Fig. 5a). The YY1 motif produces a similar but smaller effect (Fig. 5a,e).

In *Drosophila*, we tested six motifs associated with insulation. While all of them impact the predictions to some extent, four stand out as producing the highest impact on the insulation, including BEAF-32 (the strongest effect), Cp190, M1BP, and dCTCF (*Drosophila* CTCF homolog). All these factors induce insulation, as exemplified by BEAF-32 (Fig. 5c). However, in striking contrast to tetrapods and their CTCF, dCTCF in *Drosophila* does not cause formation of loops (or corner peaks). Curiously, when we tested the importance of the *Drosophila* dCTCF motif for models in other species, we observed a significant impact in tetrapods, but not in other species (even insects closely related to *Drosophila*, Fig. 5e).

Next, we reasoned that repeated low-complexity sequences might be interlinked with genome organization and tested how *in silico* insertion of repeated nucleotides and dinucleotides would affect the predicted maps. The low-complexity sequence with the highest impact was CG-repeat across all species (Fig. 5e). While CG importance might be expected for species with a CpG-methylation mechanism (e.g., vertebrates [55]), the importance of CG repeats in yeast may result from some other chromatin folding mechanism; for example, CGs are frequently present at yeast gene bodies [56], which might be captured by the model.

Finally, we tested Chimaera's effectiveness for ***in silico* mutagenesis by deletion**. We took a mouse-based Chimaera model and removed all instances of CTCF motifs identified by a position weight matrix (see Methods) from the genome (ΔCTCF). Next, we checked whether this *in silico* perturbation resulted in serious changes in the prediction. Predicted maps for the ΔCTCF genome showed lower similarity to untreated mouse Micro-C data [57] (correlation of 0.446) compared to maps predicted for the intact reference genome (correlation of 0.661, see Fig. 5d). Conversely, maps from the ΔCTCF genome were more similar to the CTCF-AID Micro-C data (correlation of 0.508) than those predicted for the reference genome (correlation of 0.448). This demonstrates that Chimaera effectively captures the critical role of CTCF and predicts the impact of its removal from the genome.

***De novo* search for motifs associated with chromatin structure**

Search for DNA determinants of genome architecture is challenging because it either requires *a priori* knowledge of potential DNA motifs or performs an exhaustive search by *in silico* mutagenesis. The full search space is very large: for example, exhaustive search for a significant 10-bp motif would require testing insertions of ~4^{10} sequences. While optimizations of computational time for exhaustive *in silico* mutagenesis exist [58,59], they do not help with shrinking the search space. Thus, we designed a solution that (1) would be more targeted than the existing solutions but still (2) would not require prior knowledge about the DNA motifs.

To that end, we first adapted the integrated gradient (IG) approach, which allowed us to calculate the importance of each input DNA position by estimating the change in the gradient in response to the gradual change of the given input (see Methods for a detailed explanation). This approach results in a genomic track of the importance of each genomic position (IG track) without requiring multiple model launches. Next, we detect peaks in the IG track and search for motifs with MEME [60] in the DNA sequences around them. The IG approach allowed us to detect motifs important for genome organization. However, its findings were limited to the CTCF motif causing the insulation pattern in the species where CTCF role was already demonstrated (mouse, human, frog, fish), and no other motifs were picked.

Integrated Gradients approach does not distinguish between different patterns formed in Hi-C maps. Thus, the importance of a genomic position in DNA is a cumulative measure of its contribution to insulation, TADs, loops, fountains, and other Hi-C patterns around it. To overcome this limitation, we aimed to design an approach that (in addition to the requirements listed above) will be **targeted at different patterns of 3D genome organization**.

To that end, we utilized the model's latent space and the encoded “semantics” of different chromatin patterns. As in the Hi-C pattern quantification approach above, we input a desired pattern into the Hi-C encoder (e.g., the insulation triangle of depleted interactions), and obtain its representation in the latent space. This representation serves as the **target direction** in which we can gradually push the input sequence for DNA encoder by modifying the nucleotides.

We implemented two ways of converging towards the target direction. In the approach that we called **Gradient Descent in the DNA input space** (Suppl. Fig. 1b), we define the loss between the desired target Hi-C embedding and the current DNA embedding, then backpropagate the gradients to the input DNA sequence, and iteratively repeat this procedure (see Methods).

Finally, inspired by the model of random mutagenesis and selection happening in natural biological systems, we designed **Evolutionary Search** inspired by genetic algorithms in optimization (Suppl. Fig. 1c). At each generation of Evolutionary Search, random DNA sequences are generated and then selected for goodness of fit to the desired latent representation. At the onset of each generation, DNA sequences are amplified proportionally to their fitness and then slightly randomly modified (see Methods).

Both Gradient Descent and Evolutionary Search allowed us to explore all possible DNA variants, while focusing on target chromatin patterns. This effectively helps to reduce the number of model launches (relative to exhaustive search of motifs) by shrinking the search space for the motif search, which is now focused around DNA motifs that can potentially generate specific Hi-C patterns.

We applied all three methods to the motif search across different species, with Gradient Descent and Evolutionary Search targeting insulation and fountain patterns. Methods often produced different results and demonstrated varying efficiency for different organisms (Fig. 6b, see the discussion for details), suggesting that the methods may complement each other and can be combined to produce the best results.

For all vertebrates, at least one of the methods identified motifs similar to the CTCF motif (see Methods). CTCF motif, when found by Gradient Descent or Evolutionary Search, was associated insulation and never associated with fountains. For *Drosophila*, the Evolutionary Search found motifs corresponding to known insulators (BEAF-32, M1BP). Motifs rich in alternating C and G have been found for many organisms (associated with either insulation or fountains). Also, a motif that could not be matched with any previously known has been found for *Drosophila* (Fig. 6b).

Notably, as a control, we tested all the found motifs by *in silico* mutagenesis and confirmed that they indeed significantly affect the predictions (see Methods). By randomly inserting the resulting motifs into the genome and assessing the maximum change between true and predicted Hi-C maps, we estimate the significance of the motif impact by comparing to expected control (Fig. 6b). We design two expected controls: inserting random sequences of the same length and inserting of shuffled motifs (preserving nucleotide content). The first control shows that almost all found motifs are significant. The second control shows that the order of the letters in GC-rich motifs usually does not matter (except mouse and *D. rerio* embryos), confirming that these motifs represent the oligonucleotide sequence biases but not factor binding sites.

Chimaera demonstrates the importance of gene positions for chromatin structure in most studied organisms

Given the indispensable role of chromatin 3D organization in the regulation of gene expression, we next questioned whether the opposite effect holds true, that is, whether positioning of genes would influence the chromatin organization. Indeed, *in silico* gene insertions have a drastic effect on the Hi-C patterns predicted by Chimaera (Fig. 7a-f). In human and mouse, Chimaera reveals insulation at gene starts and loops between the genes (Fig. 7a,b). Interestingly, gene starts in these organisms are also important when examined by the integrated gradients (Fig. 7g-i).

In *Drosophila* and *Saccharomyces*, gene starts are also important (Fig. 7j,k) and display insulation but no loops, resulting in domains without corner peaks (Fig. 7c,d). Moreover, in *Saccharomyces,* gene bodies are more likely to form TADs than intergenic regions (Fig. 7d). Interestingly, the orientation of genes does not play a role in these species.

In *Dictyostelium*, Chimaera predicted loops between two convergently oriented gene pairs but not two divergently oriented gene pairs, suggesting that the orientation of genes plays a key role in formation of this Hi-C pattern (Fig. 7f). At that, highly expressed genes in *Dictyostelium* development [30] have a larger impact on the 3D genome (Suppl. Fig. 7a), which may be correlated with oligonucleotide frequencies in genes with various expression levels (Suppl. Fig. 7b).

Finally, in *C. elegans*, no specific patterns were associated with genes except an increased number of interactions around the genes regardless of their orientation (Fig. 7d).

Cross-Species Chimaera modeling reveals evolutionary conservation and divergence in chromatin folding mechanisms

Through the application of Chimaera—including chromatin feature quantification, *in silico* mutagenesis, and *de novo* motif discovery—we confirmed the hypothesis of species-specific chromatin folding rules. A well-trained Chimaera model tailored to a specific organism can act as a black box, effectively representing the nuclear environment and *in silico* folding distinct 3D genome organizational structures.

We thus decided to use multiple Chimaera models as a proxy to assess differences in the chromatin folding rules between species. To that end, we folded DNA sequences of one species by all possible models trained for other species, and assessed the quality of the predictions. When repeated for all species in our analysis, this allowed us to build the full matrix of cross-predictions (Suppl. Fig. 8).

In most cases, species that are taxonomically close exhibited high cross-prediction accuracy; for example, models trained on mouse data accurately predicted Hi-C structures from human DNA. This observation prompted us to generate a cluster tree based on cross-prediction matrices (Fig. 6), which confirmed that chromatin folding mechanisms are largely conserved over short evolutionary distances. Tetrapods and two *Acari* species formed distinct clades, further suggesting an evolutionary basis for the chromatin organization.

While most organisms clustered according to their taxonomic relationships, we observed notable exceptions. For instance, Hi-C data from *Apis cerana* (bee) was accurately predicted by models trained on *Pomacea canaliculata* (snail) and *Danio rerio* (zebrafish), but not other insects. Importantly, these exceptions were not attributable to model training quality (see Suppl. Fig. 8a). In case of using the model trained with transfer learning for *D. rerio*,

this organism was correctly joined with vertebrates (Suppl. Fig. 9). Nevertheless, the overall large-scale patterns of chromatin folding-based trees aligned closely with established taxonomic relationships.

These findings underscore the evolutionary conservation of chromatin folding rules across closely related species while also highlighting intriguing exceptions that may reflect unique genomic organizational features.

Discussion

Recent studies on chromatin organization have predominantly focused on mammals, from which we have uncovered key mechanisms such as loop extrusion resulting in the formation of Suppl. Table loops between CTCF sites [1,7,48]. However, chromatin folding rules, even the rules of loop extrusion, vary widely across species, resulting in some organisms exhibiting distinct patterns, like fountains in the fish chromatin[9], and loops between convergent gene pairs in *Dictyostelium*[30]. Comparative analyses of chromatin folding between species are typically constrained to closely related genomes [38,39,61–63], conserved syntenic regions [6,32,34,37,64] and homologous genes [33,35,65], or rely on highly abstract qualitative comparisons of Hi-C maps [16,17,36,66,67]. A significant advancement in the field came with the development of Akita [41], a deep learning-based neural network capable of predicting Hi-C data from DNA sequences. Akita was one of the first tools to successfully fold the mouse genome using a model trained on human data, revealing differences in the chromatin folding rules based on the observed discrepancies in predictions between species.

Continuing the trend of Akita's approach, we design Chimaera, a neural network that allows cross-species predictions at scale and easy interpretation of the learned mechanisms. Chimaera demonstrates good performance for Hi-C map reconstruction, especially for the organisms with high data resolution, and outperforms other existing models on the organisms with small genomes (Table 2). Notably, these results are achieved despite the fact that Chimaera: (1) uses only information about DNA and not epigenetics as input for Hi-C prediction, (2) might be biased against rare, underrepresented structures of the genome, and (3) uses reference genome as input and omit genomic variants that can affect genome folding (such as cell line-specific mutations in CTCFs).

Chimaera's specialty is not in the quality of the prediction of the Hi-C map but in the interpretability of the learned folding rules. We achieve interpretability by a characteristic **chimeric design** of our model based on an autoencoder. Firstly, the **Hi-C encoder** is trained to compress the Hi-C map snippet into a multi-dimensional vector. During training, the Hi-C encoder is exposed to various 3D genome architectures across the genome and learns various stereotypical patterns, such as insulation, TADs, loops, and fountains. As in semantic embeddings in natural language processing [68], the model learns the characteristic features of chromatin that can be then extracted from the embedded space and used, for example, for feature quantification.

Next, we (1) split the Hi-C autoencoder and retain its decoder part, (2) freeze the weights of the Hi-C decoder, and (3) connect it to the **DNA encoder** that is trained to predict the compressed Hi-C maps through the same multi-dimensional latent representation. We speculate that this strategy allows us to avoid fitting the noise of the data, and separate the task of learning the biological rules from the task of drawing the Hi-C map (see Table 2).

Moreover, the chimeric design allows us to utilize the semantics of the embedded space to **connect the stereotypical features of chromatin to the underlying DNA determinants**. As a proof of concept, we confirm the most prominent association "insulation - CTCF motif" in a wide range of species with both Chimaera's *in silico* mutagenesis, and *de novo* motif search. While the original Chimaera training on *Danio rerio* did not produce any association with CTCF, the adaptation of transfer learning from human data demonstrates the importance of CTCF motifs for the insulation pattern in zebrafish, in line with recent studies [69] (see Fig. 5e). Additionally, we confirm the correctness of the model by *in silico* mutating each CTCF instance in the mouse genome and demonstrating that the Chimaera-predicted maps look much more similar to the map generated in the *in vivo* rapid depletion of CTCF experiment [57].

Chimaera is capable of revealing species-specific associations, exemplified by insulation association with BEAF-32, M1BP (detected both by *in silico* mutagenesis and *de novo* search), and Cp190, dCTCF (detected by *in silico* mutagenesis) motifs in *Drosophila melanogaster*, which are known to act as insulators in this species.

Finally, Chimaera reveals the association of insulation with rich GC-content in *Mus musculus* and *Saccharomyces cerevisiae*. GC-content has been previously associated with promoter regions in vertebrates [55] and gene bodies of yeast [56], which might indicate the importance of active transcription for the formation of three-dimensional structure. While there is no final, universal explanation for the connection between insulation and active

transcription, some hypotheses on the potential mechanisms have been suggested [70,71]. In line with that, we observe that *in silico* insertions of genes play a critical role in the 3D organization in most of the studied species (Fig. 7).

To explore the associations with other 3D genome patterns, we consider formation of fountains, a recently reported architectural pattern, which looks as enriched interactions emanating from a single genomic locus [9,10]. With Chimaera's Evolutionary Search, we establish the association between fountains and GC-motifs in *Danio rerio* and *Arabidopsis thaliana*, which have not been previously associated with fountains but can provide important insight for the future mechanistic and functional interpretation of fountains.

Notably, the *de novo* motif search methods we implement — **Integrated Gradients**, **Gradient Descent**, and **Evolutionary Search** — do not always yield a single dominant association within the data. These methods can be run independently and complement each other in identifying significant associations. Both Gradient Descent and Evolutionary Search can be launched multiple times (see Methods), potentially producing varying results due to random nature of each approach, which may reflect the presence of multiple local optima in the search for DNA motifs that influence chromatin folding. At that, the Evolutionary Search produces both the DNA motif and the position-specific weight matrix that can be used to estimate the importance of each position.

However, for several organisms, no significant motifs were identified. This could be partially due to the low prediction quality of the models for these species. Interestingly, despite the high accuracy of the models for *Dictyostelium discoideum* and *Apis cerana*, no motifs were detected. This suggests that, in these species, the chromatin structure formation may not rely on sequence-specific DNA binding factors. For example, for *Dictyostelium discoideum*, the resulting interpretation of predictions (Fig. 7f) agrees well with the existing hypothesis that loops in its chromatin are formed at the sites of convergent transcription due to RNA polymerase moving along DNA and shifting extruder proteins to the ends of genes [30].

We perform *in silico* mutagenesis. The aim here is not to predict the consequences of real mutations but to interpret the obtained models. For example, the model may change its predictions after inserting only a part of some significant sequence if it has never seen this part separately in the train sample. In this case, the model may not predict the real effect of the mutation but will indicate the possible mechanism. At that point, we observe that mutating CTCF sites *in silico* has similar consequences as experimental degradation of CTCF.

Finally, we construct a **chromatin tree of life** based on Chimaera models trained for different species (Fig. 8). In this approach, each species-trained model serves as a proxy for the mechanism of chromatin organization in the nucleus of this species. The tree obtained from the correlation matrix of cross-predictions generally reflects the taxonomic relationships between the studied organisms, although with some exceptions. We speculate that this striking property of the chromatin tree of life reflects the conservation of chromatin-folding mechanisms in evolution and their gradual change with the growing evolutionary distance between species.

Why do we observe the species that do not follow conventional taxonomy based on their chromatin tree of life? Firstly, that might be due to the varying quality of Hi-C datasets in different species. With more consistently obtained Hi-C (Micro-C, or any other 3D genomic assay) datasets in multiple species, the effect of origin and batch will be mitigated and will allow for studying the relationships between chromatin folding mechanisms more unambiguously. Moreover, with higher resolution of Hi-C/Micro-C data available, the future Chimaera-like models might consider not only 3D genome interactions close to the diagonal, but long-range and inter-chromosomal interactions for the tree inference (circumventing our choice to 45-degree rotate Hi-C maps and potentially taking off-diagonal Hi-C signal into account).

Notably, we demonstrate significant improvement of the model quality when utilizing transfer learning of the model from the species with more high-quality datasets (Suppl. Fig. 6). Further exploration in the direction of transfer learning between species will allow improvements and, potentially, data imputation for species with lower quality of datasets.

Secondly, the difference between taxonomy based and chromatin based tree of life might be attributed to the diversity of studied tissues. For example, while we used datasets for *Danio rerio* early embryos and *Apis cerana* pupae, for *Arion vulgaris* and *Pomacea canaliculata,* we used highly differentiated muscle tissue of adult

animals, where the rules of folding can be, in principle, different than in early development. To resolve this problem, we look forward to future works that will dissect chromatin organization in different developmental stages across multiple species of non-model organisms, which are currently only limited.

Finally, the misfits in the chromatin tree of life might represent real evolutionary events when the mechanisms governing three-dimensional genome architecture have drastically changed, potentially with no opportunity to establish similarities with other species. In accordance with this hypothesis, the mechanisms of epigenetic regulation are known to vary substantially between species [72]. Loss of CTCF in the evolution of nematodes [73] might have been an event like that, forcing the 3D-genome organization of *Caenorhabditis elegans* to cluster together with amoeba and plants.

To sum up, Chimaera is a powerful tool for interpreting chromatin data. Our approach offers a new dimension to understanding the 3D genome organization, providing insights into how evolutionary pressure may have shaped the chromatin architecture. Chimaera utilizes a vast amount of Hi-C and Micro-C datasets published for multiple species, including non-model organisms, and yields insightful interpretation of associations between 1D and 3D genome patterns. The present study paves the way for future evolutionary studies of chromatin folding that might uncover the mechanisms that govern the speciation and emergence of regulation in evolution.

Data availability

Raw and processed Hi-C and Micro-C data was obtained from BioProject accessions PRJNA606649 (*Xenopus tropicalis*), PRJNA630123 (*Anopheles merus*), PRJNA318737 (*Aedes aegypti*), PRJNA665323 (*Culex quinquefasciatus*), PRJNA749654 (*Sarcoptes scabiei*), PRJNA683935 (*Archegozetes longisetosus*), PRJNA680311 (*Arion vulgaris*), PRJNA427478 (*Pomacea canaliculata*); BioSample accession SAMN13118423 (*Apis cerana*); GEO datasets GSE178982 (*Mus musculus* ESC), GSE129997 (*Mus musculus* cell cycle), GSE171396 (*Drosophila melanogaster*), GSE128568 (*Caenorhabditis elegans*), GSE151553 (*Saccharomyces cerevisiae*), GSE155503 (*Arabidopsis thaliana*), GSE195609 (*Danio rerio*), GSE247397 (*Dictyostelium discoideum*); 4DN dataset 4DNBSZOFFFM6 (*Homo sapiens*).

Source data for Fig. 5e is in Supporting data table 1, source data for Suppl. Fig. 8 is in Supporting data tables 2 and 3 . Preprocessed data for all studied organisms and trained models are posted online at OSF: https://osf.io/yf7cr.

Code availability

Chimaera code and illustrative examples are available at https://github.com/ashkolikov/chimaera/.

Author contributions

ASh: conceived the original idea, designed, implemented and tested the model, developed Chimaera toolkit. AxG: conceived the original idea, wrote the first draft, analyzed the data. MG: supervised bioinformatics and analysis. All the authors commented and edited the manuscript.

Acknowledgments

We are grateful to Prof. S.V. Razin (IGB RAS) and his lab (in particular, Dr. Sergey Ulianov) for sharing the preliminary data on *Dictyostelium discoideum* Hi-C and RNA-Seq, Dr. Daria Onichtchouk (Freiburg University) and Prof. S.V. Razin and their labs (in particular, Dr. Sergey Ulianov) for sharing the preliminary data on *Danio rerio* Hi-C with fountains.

AS and MG thank Prof. Andrey A. Mironov for the extensive discussions and the idea of building the chromatin-based tree of life. AG thanks Prof. Leonid Mirny, Prof. Geoff Fudenberg and Dr. Simon Grosse-Holz for the critical discussions at the early stages of the project development. AS thanks Anastasia Kashtanova for pre-processing some of the Hi-C data and bringing it to the attention of the team. AG and AS thank students Dmitry Skripka, Alexandr Yakushev and Alexandra Madorskaya for testing Chimaera API.

Aleksei Shkolikov and Mikhail Gelfand were supported by RSF (project No. 23-14-00136).

Figures

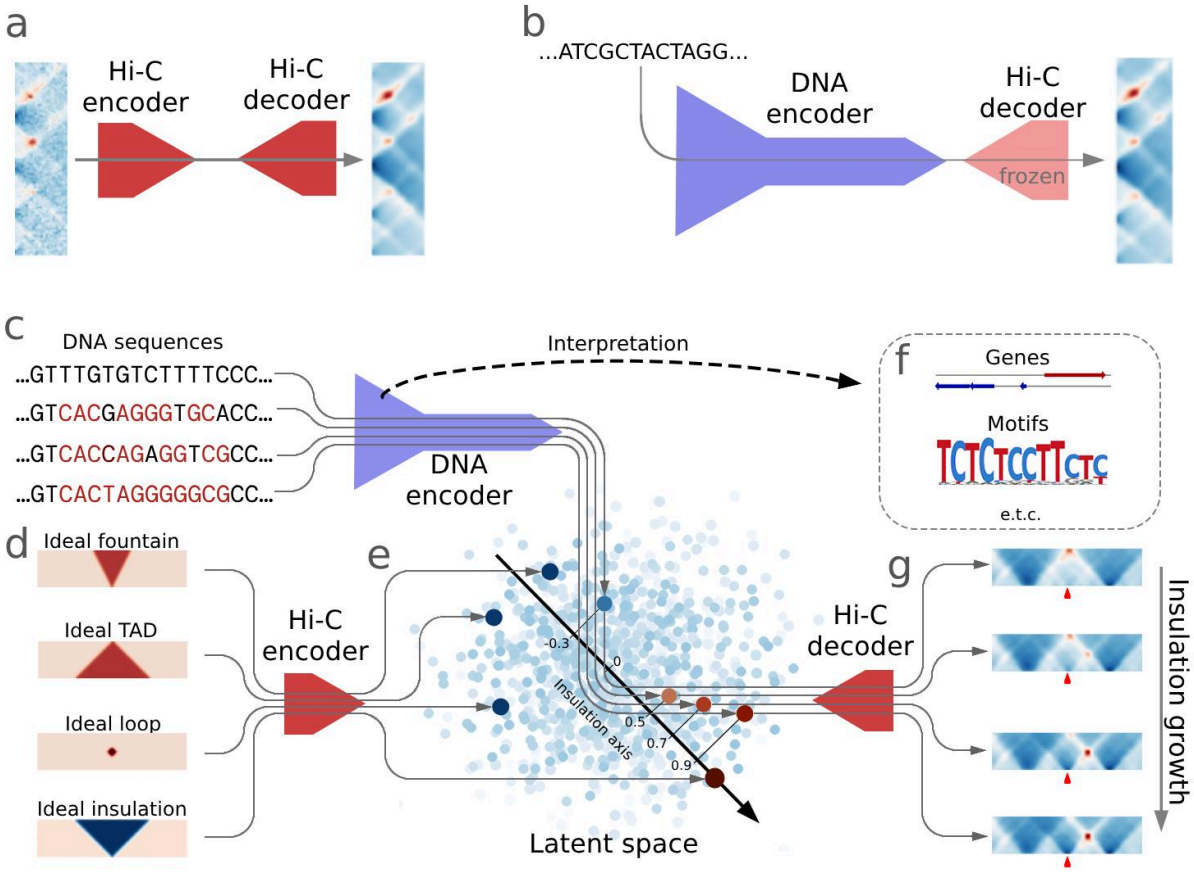


Figure 1. Summary of Chimaera, Convolutional neural network for Hi-C maps prediction using autoencoder for maps representation. Each gray line corresponds to the forward pass of the single input (DNA or Hi-C map) through the network. The schematics summarizes the training scheme (a, b) and three principal ways of using Chimaera in this work: Hi-C prediction from DNA sequence (c, g), pattern search and quantification (d, e), and interpretation of the DNA sequence / 3D genome pattern associations (f). For the full architecture, see Suppl. Fig. 1a. Real data is shown for human.

a. Training strategy for the Hi-C autoencoder. The network consists of two blocks: Hi-C encoder and Hi-C decoder that are trained simultaneously to denoise Hi-C maps.

b. Training strategy for the DNA encoder. DNA serves as input to predict latent representation of Hi-C map for corresponding genomic region. Weights of Hi-C decoder are unchanged during training (frozen). See Table 2 for comparison of different training strategies.

c. Example DNA sequences serving as input to the DNA encoder. Red letters represent parts of CTCF motif instances that happened to be in the genome. Top: sequence does not have any CTCF motif, bottom: sequence contains recognizable CTCF motif.

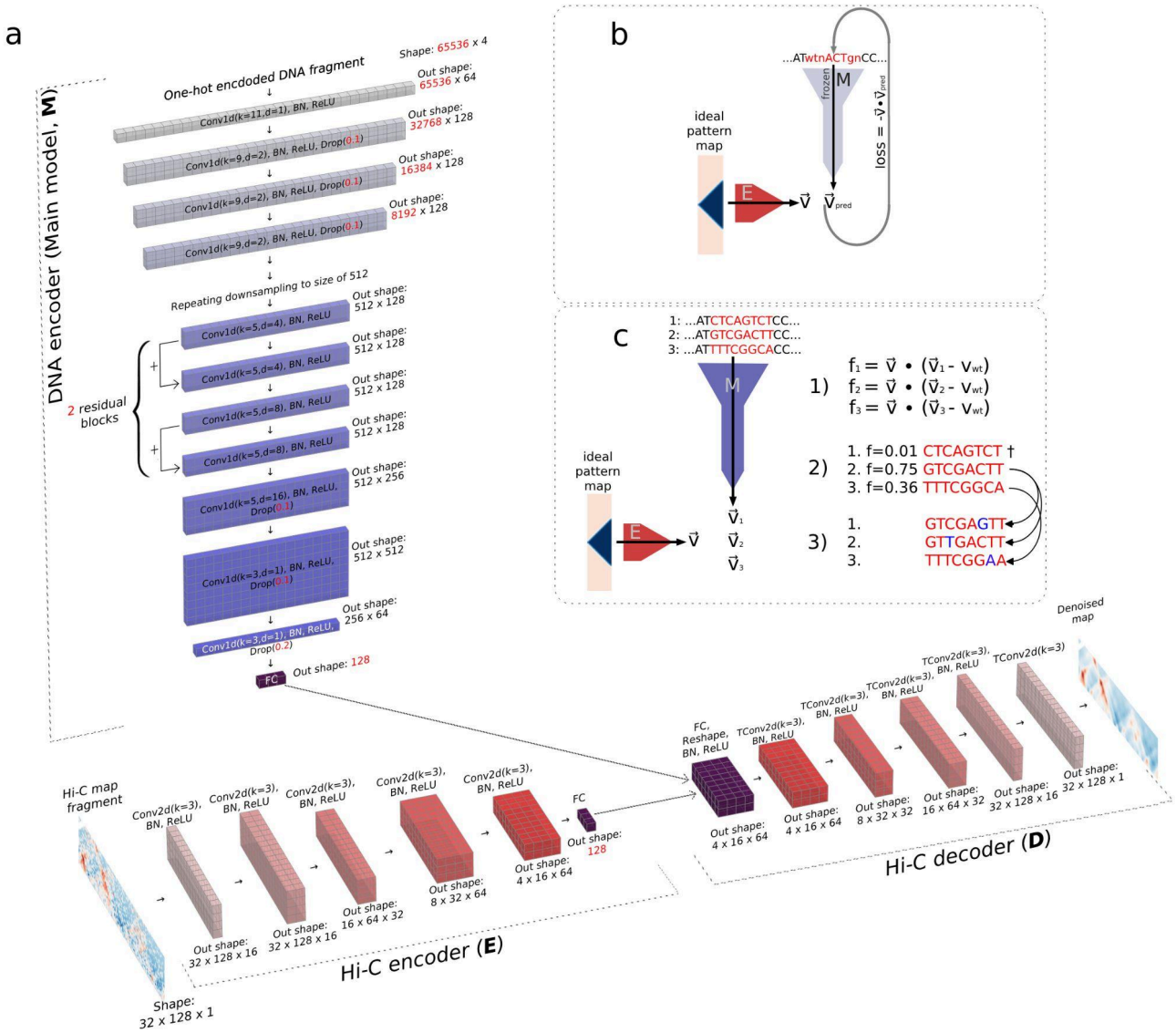
d. Example of patterns serving as input to the Hi-C encoder for feature calling and quantification. Note that we use 45-degree rotated Hi-C maps, where the horizontal axis corresponds to genomic coordinates and the vertical axis reflects genomic separation.

e. Latent space of autoencoder and results of projection into it. Light-blue dots: embeddings of each genomic region Hi-C.

From (d) to (e): each pattern is projected into a vector in the latent space. Note that TAD pattern is located in the distant area to the insulation pattern, while fountain and dot are located more in between. Note the location of the insulation vector used below for the quantification of the insulation pattern.
From (c) to (e): each sequence is projected into the vector in the latent space. In this example, we project each vector onto the vector of insulation to quantify the prominence of insulation at each genomic location. The length of the projection serves as a similarity measure of genomic location to an input pattern (here, insulation).

f. Maps predicted by Chimaera's Hi-C decoder based on latent representations (e) made by DNA encoder (c). Note how insulation grows in these examples, corresponding to the presence of CTCF motif in the input sequence (c).

g. Model interpretation. Extracting the information about DNA determinants of 3D genome organization, such as motifs of factors binding, and genes.



Supplementary Figure 1. Chimaera architecture (a) and interpretation methods (b,c). E – Hi-C Encoder, D – Hi-C Decoder, M – DNA Encoder, or Main model. frozen – the weights of the model block are locked and do not undergo updates at this point (highlighted only when relevant).

a. Model architecture. Parameters in black: the same for all species. Parameters in red: parameters that we varied and selected depending on the size of the input DNA/Hi-C fragment (see Table 3), and/or sample size. Conv1d/Conv2d – convolutional layers, TConv – transposed convolutional layer, BN – batch normalization, FC – fully connected, ReLU – rectified linear unit. Between convolutional layers we have used max pooling (see the change in dimensionality of the convolutional blocks).

(b, c). **Chimaera-based *de novo* calling of the DNA motifs corresponding to a given feature**: two approaches that we provide together with Chimaera, example for insulation pattern search.

b. **Gradient Descent in the space of DNA.** In this approach, we maximize the projection of the embedded vector (v_{pred}) to embed a given interaction pattern (v) in the successive iterations of the model run on DNA sequence. The weights of both Hi-C and DNA encoders are frozen, and only the input DNA sequence is changed.

c. Evolutionary Search in the space of DNA. As an input, we generate a set of random DNA sequences, propagate them through the DNA encoder, and then iteratively run steps (1-3): 1) — calculate the fitness function, 2) — select the best DNA sequences and let them "reproduce", 3) — randomly mutate the "reproduced" sequences.

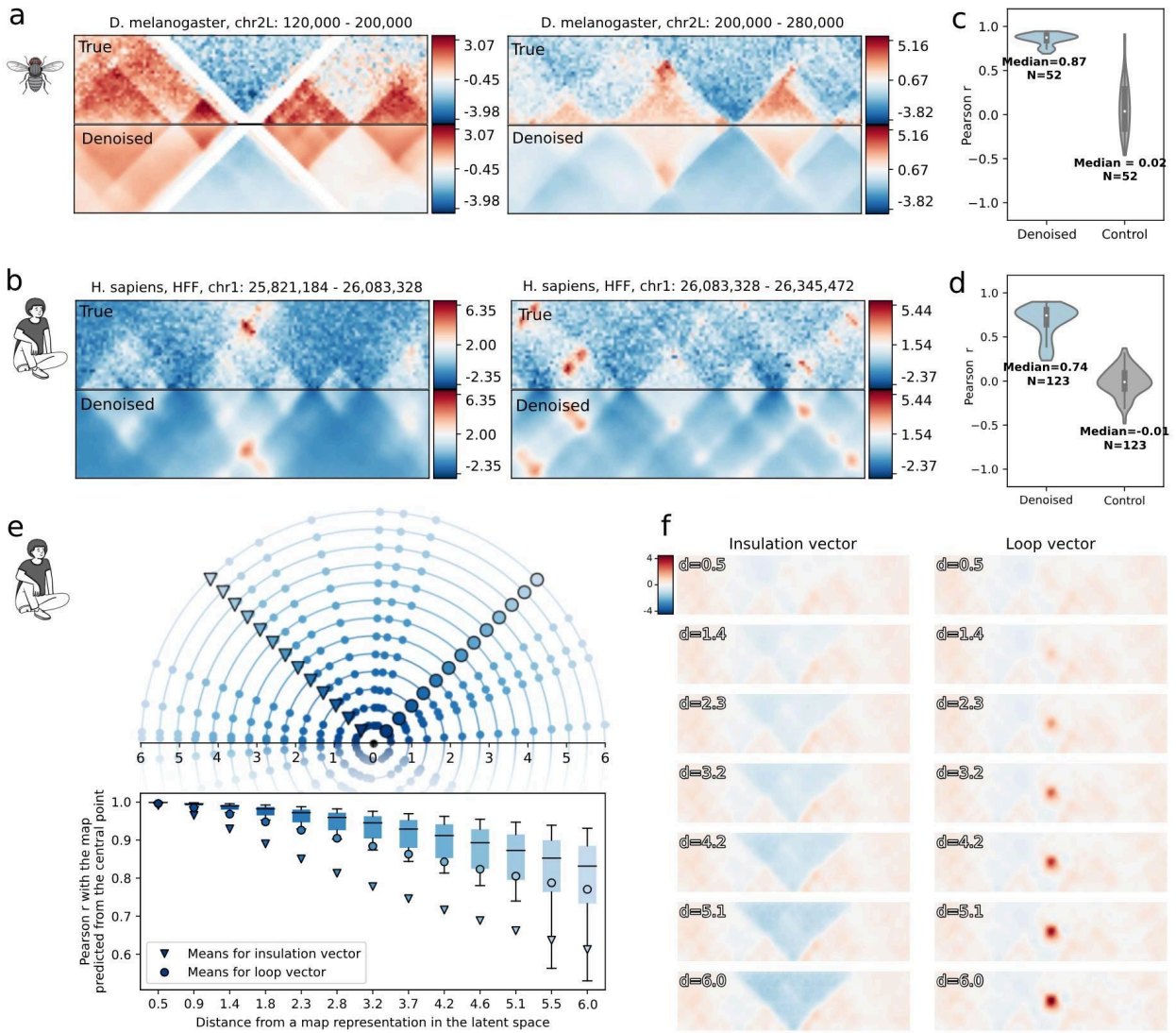


Figure 2. Key capabilities of the Hi-C encoder include denoising Hi-C maps (a-d) and generating a continuous, interpretable latent space (e-f).

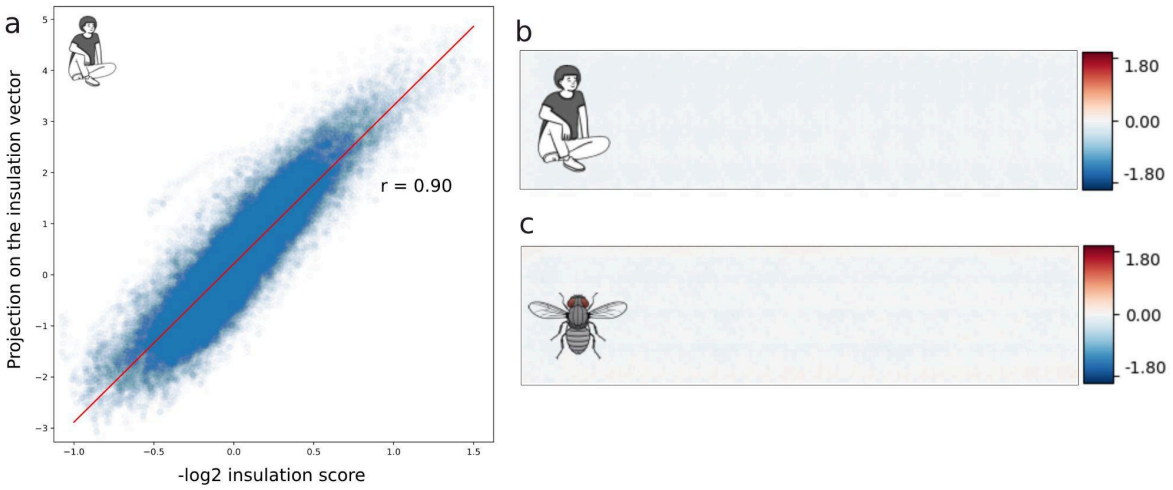
a, b. **Examples of noise reduction with Hi-C autoencoder for the maps of *D. melanogaster* (a) and *H. sapiens* (b).** The top half of each plot is the actual map, and the bottom half is the mirrored predicted one. The represented values are Z-score normalized observed over expected signal (true Hi-C/Micro-C or predicted by Chimaera).

c, d. **Pearson correlation on the test set between the original maps and the autoencoder denoised ones for *D. melanogaster* (c) and *H. sapiens* (d).** Controls: randomly paired true and predicted maps.

e. **Demonstration of latent space continuity.** Top: We started with the vector representations of the real Hi-C maps in the latent space of the Hi-C encoder (zero point). Next, we stepped away from each point with a fixed step size in a given direction (triangle – insulation direction, circle – loop direction). At each step, we decoded the maps for the resulting "shifted" vectors, and calculated similarity to the original. Bottom: distributions of the similarity of the "shifted" vectors to the originals.

f. Average outputs of shifts in the latent space. Left: shifts across the latent representation of the insulation pattern. Right: shifts across the latent representation of loop pattern. *d* – distance in the latent space from the

original points, as defined in (e).
Note that the shift towards positive values in the direction of the insulation pattern vector results in more pronounced insulation in the generated Hi-C maps. This observation goes in line with the fact intact genomic regions with higher projection onto the insulation vector have higher insulation score (see Suppl. Fig. 2).



Supplementary Figure 2. Properties of the latent space. Supporting information for the main Fig. 2.

a. Correlation plot of the projection on insulation vector against insulation score, example for human-trained Chimaera and human Hi-C maps. We plot the sign-flipped values of the traditional insulation scores [74], so that positive values of (-log-insulation scores) correspond to a more pronounced insulation pattern. Insulation scores were calculated for the whole genome by *cooltools* [50].

(**b-c**). Decoded Hi-C map corresponding to zero vector in the latent space. Note absence of any patterns and signal. Also note that the scale of the colorbar is much smaller than for main Fig. 2a-b.

b. For the human model.

c. For the *Drosophila* model.

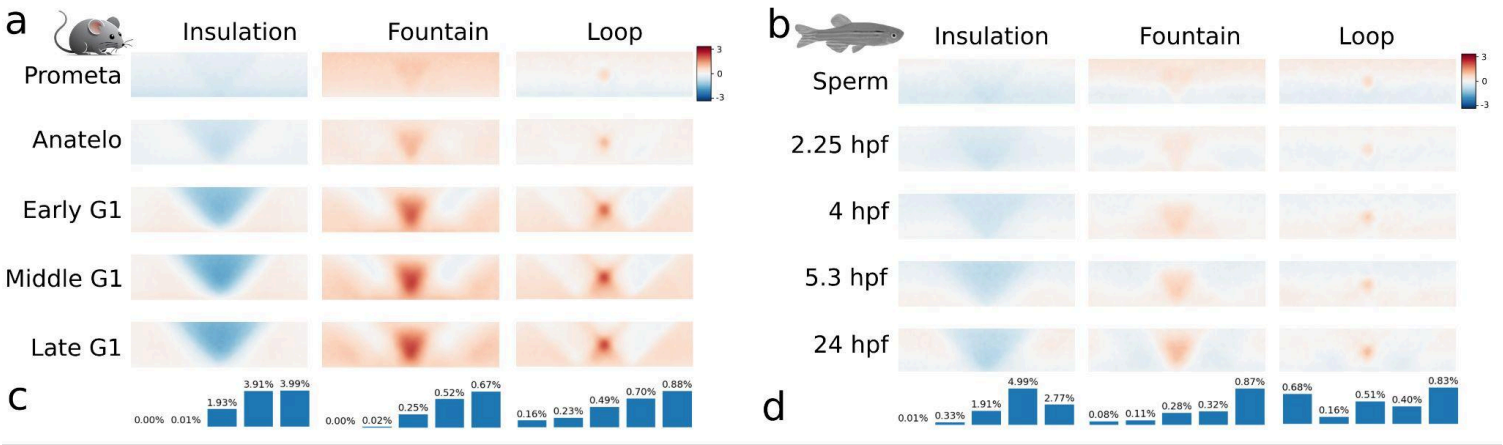
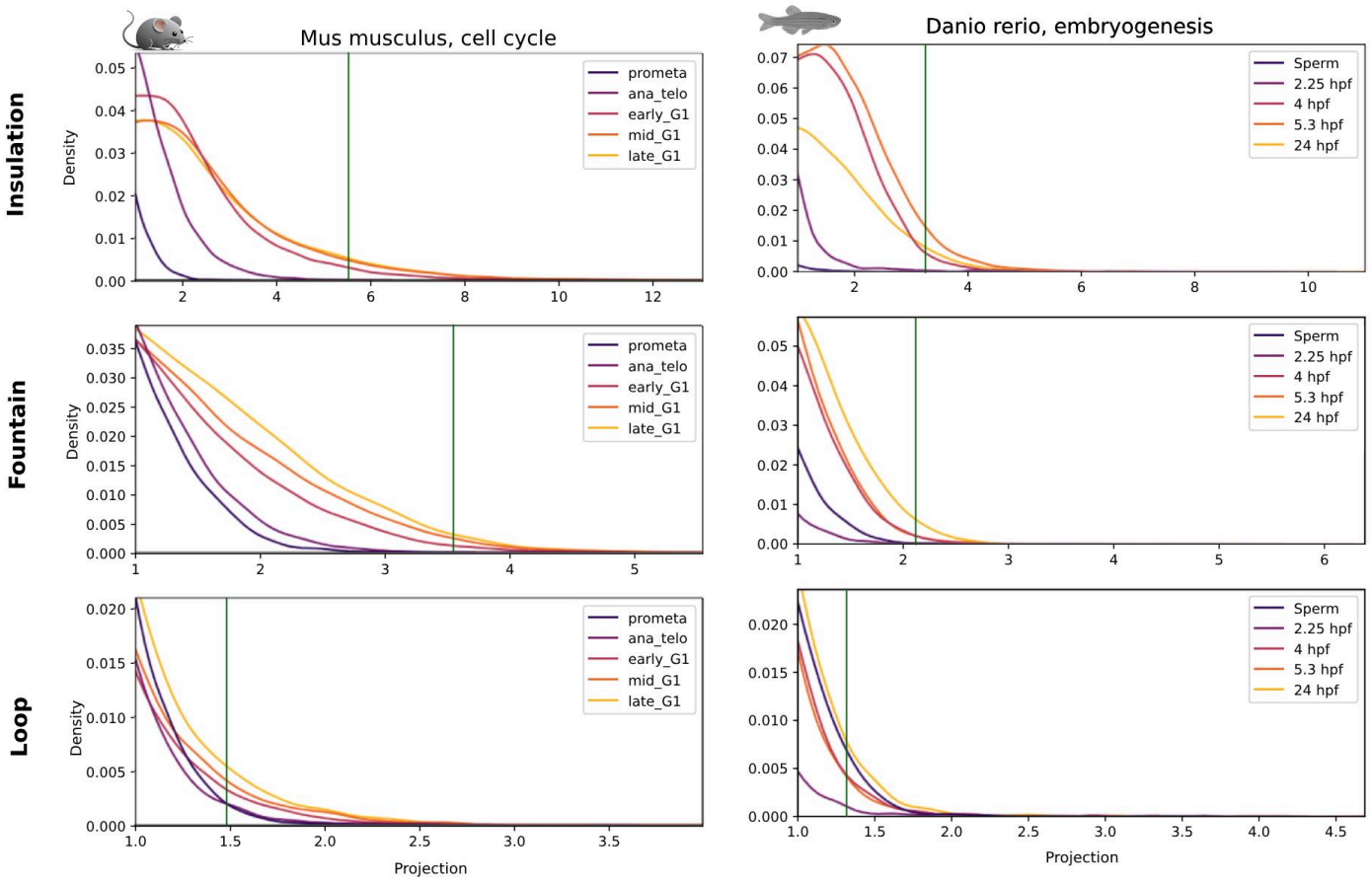


Figure 3. Chimaera quantification of the input patterns in the input Hi-C maps across two processes in two species: (a,c) *Mus musculus* cell cycle (G1E-ER4 cells, data from [51]), and (b,d) *Danio rerio* embryogenesis (whole-embryo, data from [9,75]). hpf — hours past fertilization. All results were obtained using the Hi-C encoder trained on mouse data.

a, b. Average pileup of the best findings of insulation, fountain, and loop pattern (from left to right) using latent space projections. Note the gradual appearance of all features in the time series Hi-C. Calling criterion: Kneedle algorithm (see Methods "Pattern calling in Hi-C maps").

c, d. Percentages of bins with projection larger than selected thresholds for each stage. See Suppl. Fig. 3 for illustration of the full distributions and cutoffs. The bars correspond to the stages in (a,b) and the numbers are shown above each bar. Note how insulation, fountain, and loop gradually emerge in both processes (although the dynamics can vary for different patterns). Calling criterion: top hits across all stages (see Methods "Pattern calling in Hi-C maps").



Supplementary Figure 3. Distributions of projections to the given input patterns for all the genomic regions of *Mus musculus* and *Danio rerio*. For each input pattern and species, the selected threshold is depicted as a vertical green line at each plot. Threshold choice: top hits across all stages (see Methods "Pattern calling in Hi-C maps"). The fraction of genomic bins above the selected threshold at each stage is shown in Fig. 3c,d.

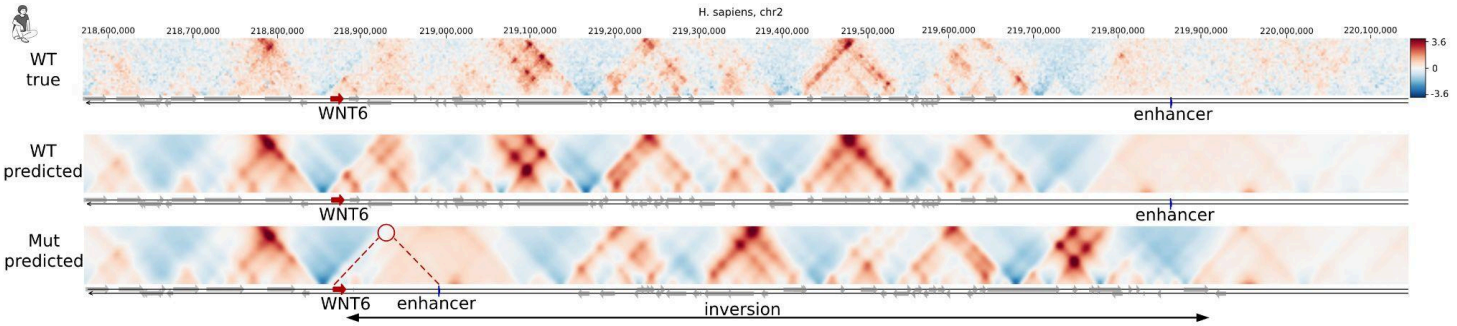
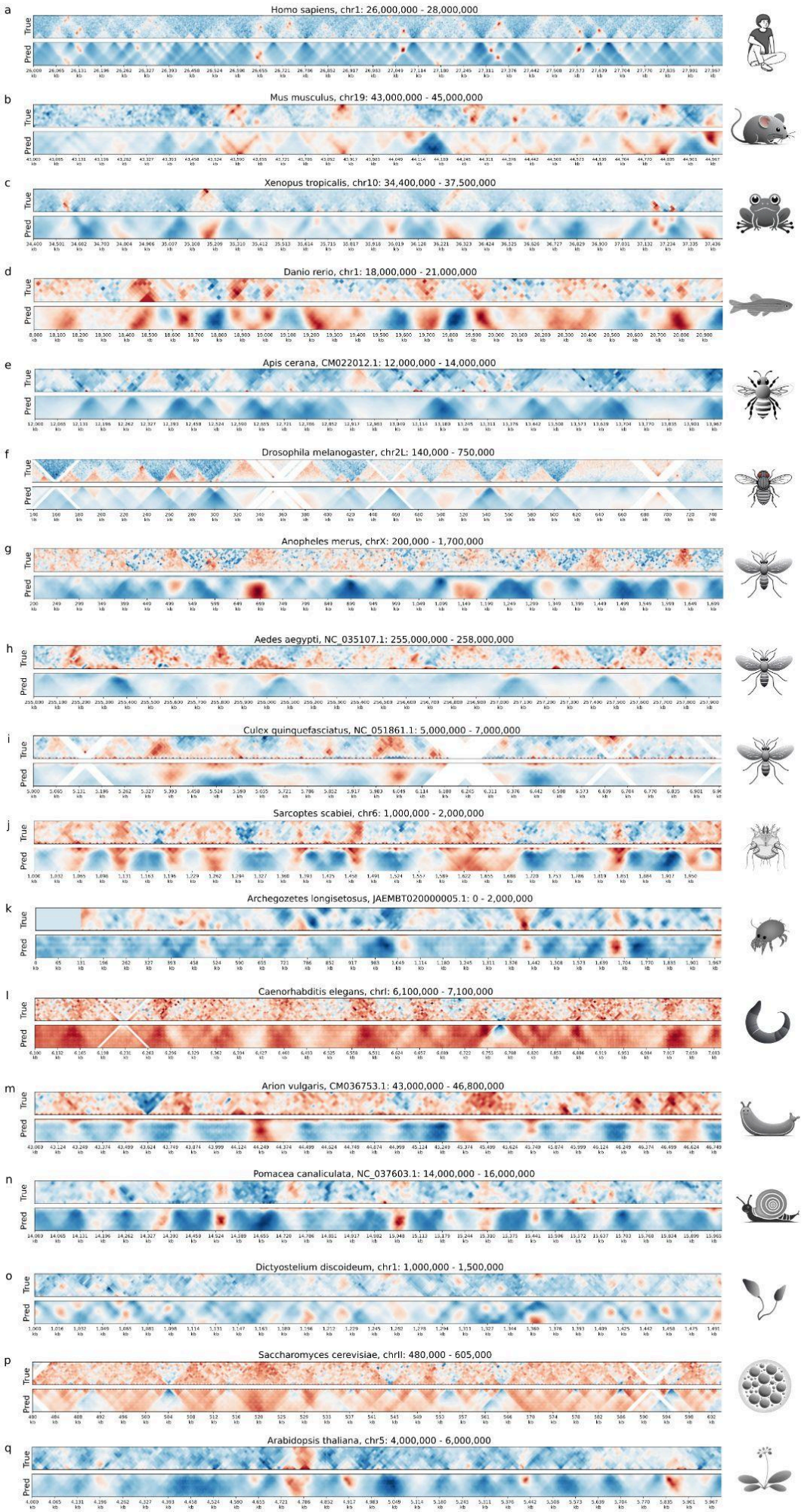
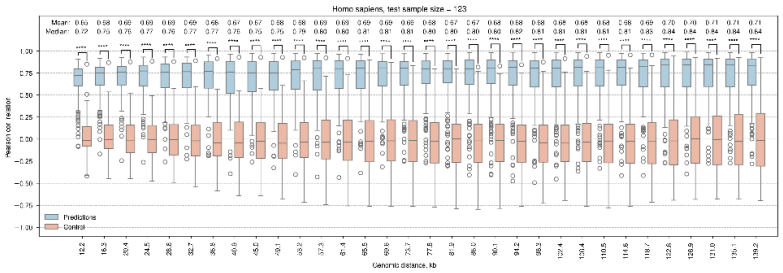
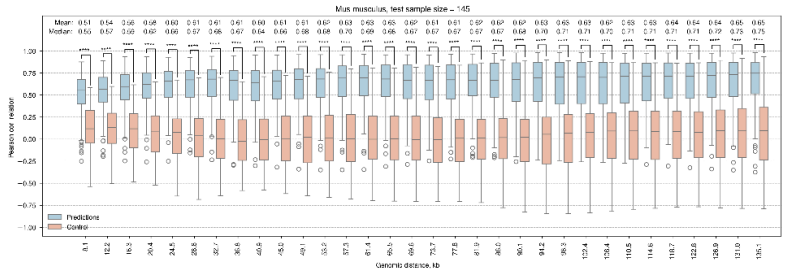


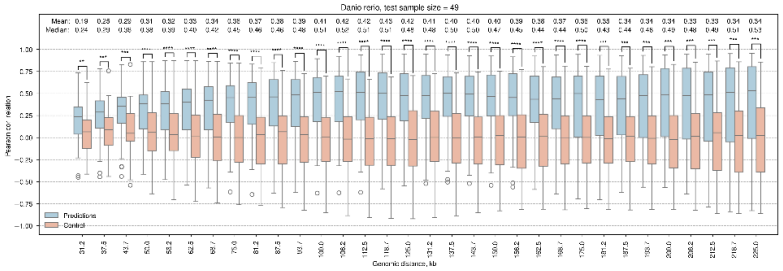
Figure 4. Prediction of the effect of structural variant that causes F-hand syndrome in development. Wild-type Micro-C for HFF6c cells from [54], the rearrangement coordinates based on [11,76]. The rearrangement that *in vivo* causes F-hand syndrome is an inversion that forces relocation of the enhancer closer in the linear genomic distance to the WNT6 gene promoter. However, chromatin organization of this rearrangement is not known (while prediction exists for other variants causing F-hand and limb malformations [42]). With Chimaera, we predict the effect of this rearrangement and observe the formation of the domain around the enhancer. This domain also includes WNT6, providing the basis for de-regulation of WNT6.

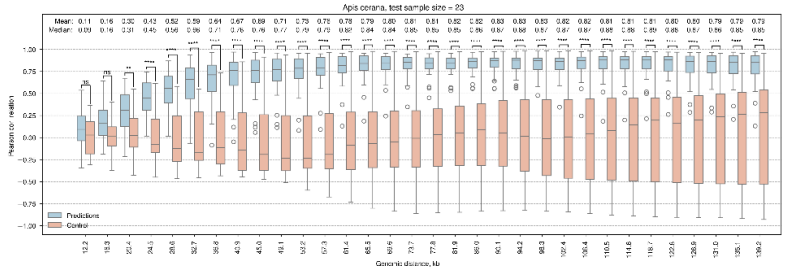

a
Homo sapiens, chr1: 26,000,000 - 28,000,000
b
Mus musculus, chr19: 43,000,000 - 45,000,000
c
Xenopus tropicalis, chr10: 34,400,000 - 37,500,000
d
Danio rerio, chr1: 18,000,000 - 21,000,000
e
Apis cerana, CM022012.1: 12,000,000 - 14,000,000
f
Drosophila melanogaster, chr2L: 140,000 - 750,000
g
Anopheles merus, chrX: 200,000 - 1,700,000
h
Aedes aegypti, NC_035107.1: 255,000,000 - 258,000,000
i
Culex quinquefasciatus, NC_051861.1: 5,000,000 - 7,000,000
j
Sarcoptes scabiei, chr6: 1,000,000 - 2,000,000
k
Archegozetes longisetosus, JAEMBT020000005.1: 0 - 2,000,000
l
Caenorhabditis elegans, chrI: 6,100,000 - 7,100,000
m
Arion vulgaris, CM036753.1: 43,000,000 - 46,800,000
n
Pomacea canaliculata, NC_037603.1: 14,000,000 - 16,000,000
o
Dictyostelium discoideum, chr1: 1,000,000 - 1,500,000
p
Saccharomyces cerevisiae, chrII: 480,000 - 605,000
q
Arabidopsis thaliana, chr5: 4,000,000 - 6,000,000
True
Pred

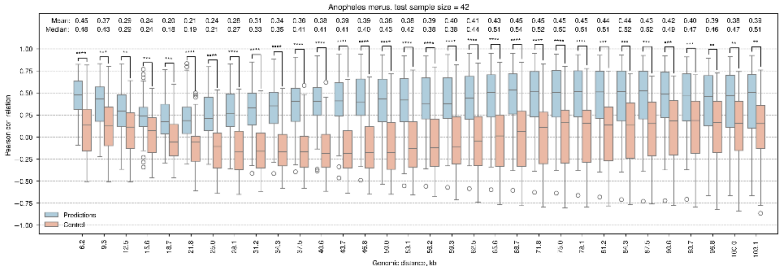
Supplementary Figure 4. Examples of Hi-C predicted maps based on DNA sequence for the species studied in this work. The top of each panel is a true input map, the bottom is a mirrored predicted one. From top to bottom: Micro-C for *Homo sapiens* HFFc6 cell line from [54], Micro-C for *Mus musculus* embryonic stem cells from [57], Hi-C for *Xenopus tropicalis* embryos from [77], Hi-C for *Danio rerio* embryos at 5.3 hours past fertilization from [9], Hi-C for *Apis cerana* from the pupae of drone pupae from [78], Micro-C for *Drosophila melanogaster* for nc14 stage embryos from [79], Hi-C for mosquitoes *Anopheles merus* from [38], *Aedes aegypti* from [80], *Culex quinquefasciatus* from [81], Hi-C for mites *Sarcoptes scabiei* from [82], *Archegozetes longisetosus* from [83], Hi-C for nematode *Caenorhabditis elegans* from [84], Hi-C for molluscs: muscle tissue of slug *Arion vulgaris* [85], and snail *Pomacea canaliculata* from [86], Hi-C for social amoeba *Dictyostelium discoideum* from [30], Micro-C XL for yeast *Saccharomyces cerevisiae* at 90 minutes after release from G1 from [87], and Hi-C from flowers of *Arabidopsis thaliana* from [88].

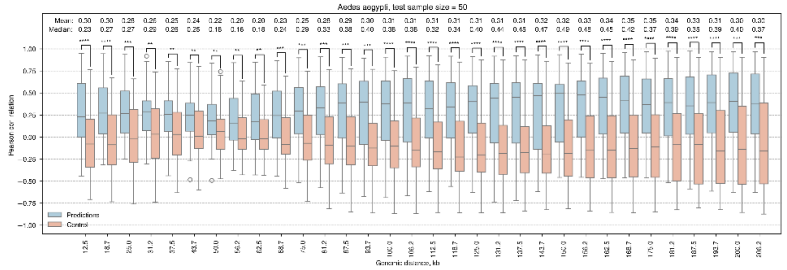

Homo sapiens, test sample size = 123
Mean: 0.65 0.68 0.68 0.69 0.69 0.69 0.68 0.67 0.67 0.68 0.68 0.68 0.67 0.67 0.68 0.68 0.68 0.68 0.68 0.68 0.68 0.68 0.68 0.69 0.70 0.70 0.71 0.71 0.71
Median: 0.72 0.75 0.76 0.77 0.76 0.77 0.76 0.75 0.75 0.79 0.80 0.80 0.81 0.81 0.80 0.80 0.80 0.80 0.81 0.81 0.81 0.81 0.81 0.83 0.84 0.84 0.84 0.84
Pearson cor relation
Predictions
Control
Genomic distance, kb

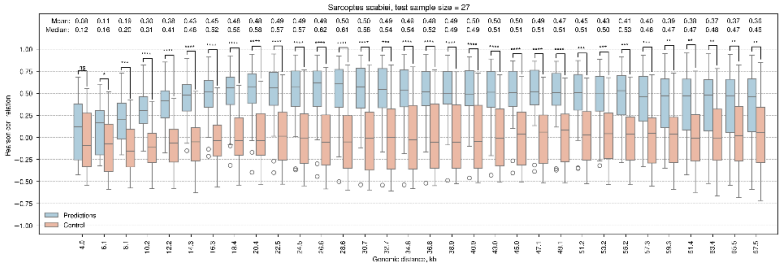

Mus musculus, test sample size = 145
Mean: 0.51 0.54 0.56 0.58 0.60 0.61 0.61 0.60 0.61 0.61 0.65 0.63 0.63 0.63 0.65 0.67 0.67 0.67 0.67 0.67 0.67 0.67 0.67 0.63 0.63 0.63 0.63 0.64 0.64 0.64 0.65
Median: 0.55 0.57 0.59 0.62 0.66 0.67 0.67 0.64 0.66 0.68 0.68 0.70 0.69 0.68 0.67 0.67 0.67 0.67 0.67 0.68 0.68 0.70 0.71 0.71 0.71 0.71 0.71 0.72 0.73 0.75
Pearson cor relation
Predictions
Control
Genomic distance, kb

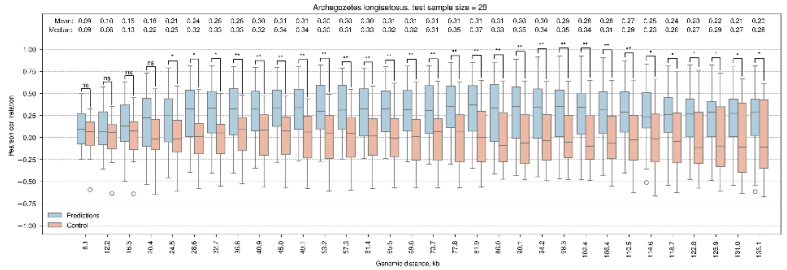

Danio rerio, test sample size = 49
Mean: 0.19 0.26 0.29 0.31 0.32 0.33 0.34 0.36 0.37 0.38 0.39 0.41 0.42 0.42 0.43 0.42 0.41 0.40 0.40 0.39 0.38 0.37 0.36 0.35 0.34 0.34 0.34 0.33 0.33 0.34 0.34
Median: 0.24 0.29 0.38 0.38 0.39 0.40 0.42 0.45 0.46 0.46 0.48 0.51 0.52 0.51 0.51 0.48 0.48 0.50 0.50 0.47 0.46 0.44 0.44 0.42 0.43 0.44 0.45 0.49 0.48 0.49 0.51 0.53
Pearson cor relation
Predictions
Control
Genomic distance, kb

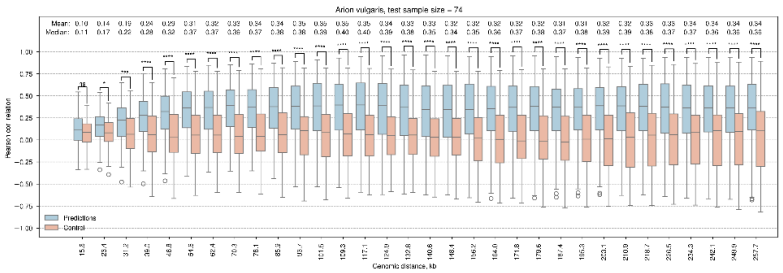

Apis cerana, test sample size = 23
Mean: 0.11 0.16 0.32 0.42 0.52 0.59 0.64 0.67 0.69 0.71 0.73 0.75 0.76 0.78 0.79 0.80 0.81 0.81 0.82 0.82 0.83 0.83 0.82 0.82 0.81 0.81 0.81 0.80 0.80 0.79 0.79 0.79
Median: 0.00 0.16 0.31 0.45 0.56 0.66 0.71 0.76 0.79 0.77 0.79 0.79 0.82 0.84 0.84 0.85 0.85 0.86 0.86 0.86 0.87 0.87 0.87 0.87 0.87 0.88 0.88 0.88 0.88 0.87 0.86 0.85
Pearson cor relation
Predictions
Control
Genomic distance, kb

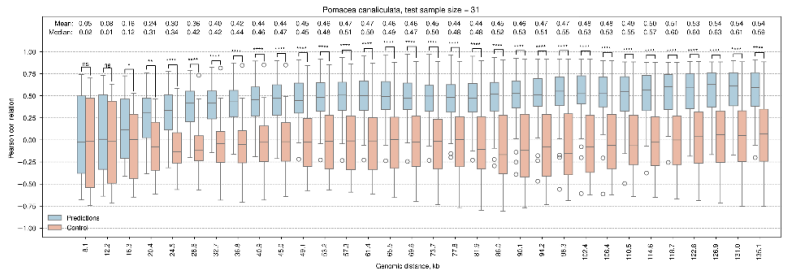

Anopheles merus, test sample size = 42
Mean: 0.45 0.37 0.25 0.24 0.20 0.21 0.24 0.28 0.31 0.34 0.36 0.38 0.39 0.38 0.38 0.39 0.40 0.41 0.43 0.45 0.45 0.45 0.45 0.44 0.44 0.43 0.42 0.40 0.39 0.38 0.38
Median: 0.49 0.43 0.23 0.21 0.18 0.19 0.21 0.27 0.33 0.35 0.41 0.41 0.41 0.40 0.45 0.42 0.38 0.38 0.41 0.51 0.51 0.52 0.52 0.51 0.51 0.52 0.52 0.47 0.46 0.47 0.47 0.51
Pearson cor relation
Predictions
Control
Genomic distance, kb

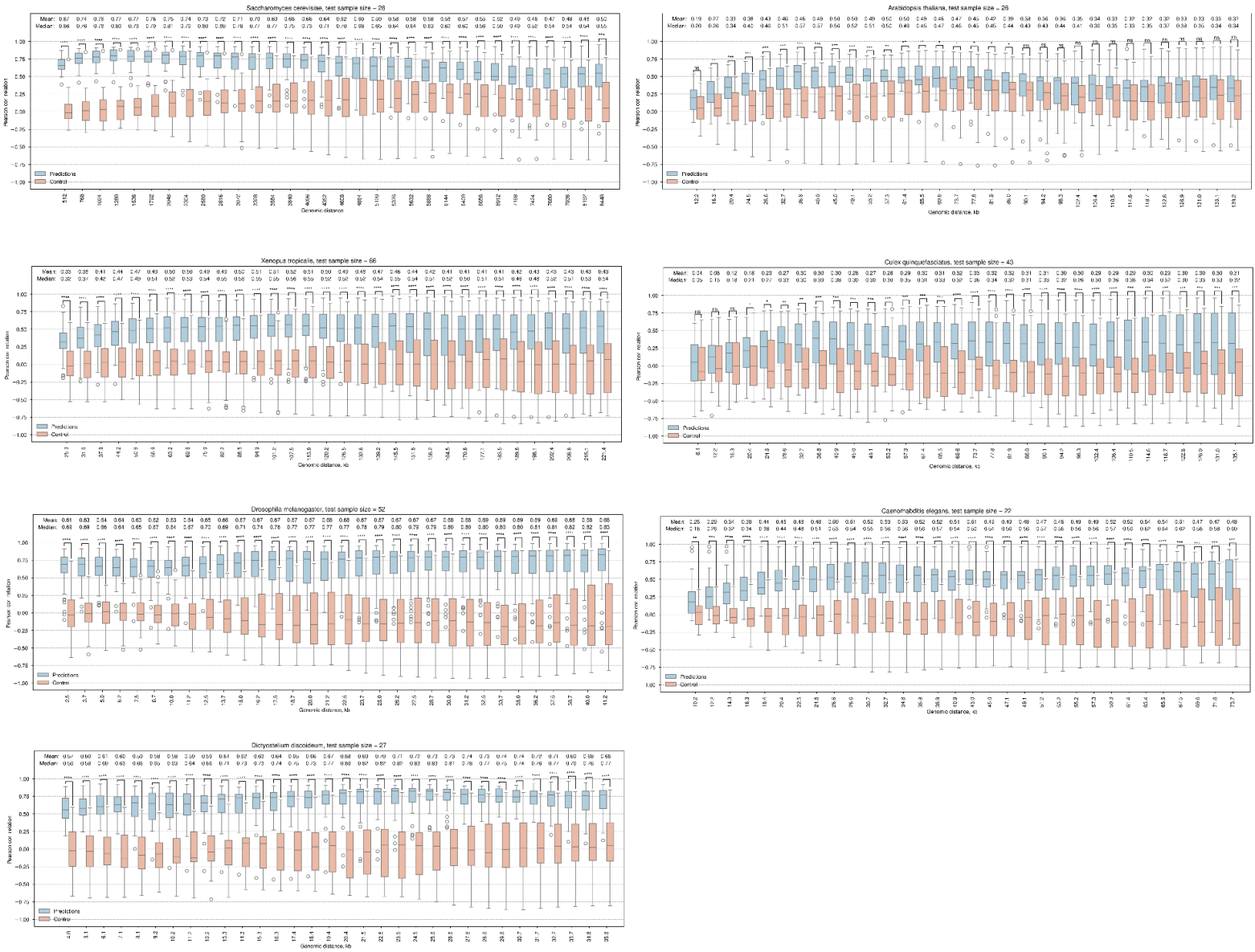

Aedes aegypti, test sample size = 50
Mean: 0.30 0.30 0.28 0.26 0.25 0.24 0.22 0.20 0.20 0.23 0.25 0.28 0.29 0.30 0.30 0.31 0.31 0.31 0.31 0.31 0.31 0.32 0.32 0.33 0.34 0.35 0.35 0.34 0.33 0.31 0.30 0.30
Median: 0.23 0.27 0.27 0.28 0.28 0.25 0.16 0.16 0.18 0.21 0.29 0.32 0.38 0.40 0.38 0.38 0.32 0.34 0.40 0.41 0.43 0.47 0.46 0.46 0.45 0.42 0.37 0.38 0.33 0.39 0.40 0.37
Pearson cor relation
Predictions
Control
Genomic distance, kb


Sarcoptes scabiei, test sample size = 27
Mean: 0.09 0.11 0.19 0.20 0.38 0.45 0.45 0.48 0.48 0.49 0.49 0.50 0.50 0.49 0.49 0.50 0.50 0.50 0.49 0.47 0.45 0.43 0.41 0.40 0.39 0.38 0.37 0.37 0.36
Median: 0.12 0.16 0.22 0.31 0.41 0.46 0.52 0.56 0.57 0.57 0.62 0.61 0.56 0.54 0.52 0.49 0.49 0.51 0.51 0.51 0.51 0.50 0.53 0.48 0.47 0.47 0.48 0.47 0.45
Pearson cor relation
Predictions
Control
Genomic distance, kb

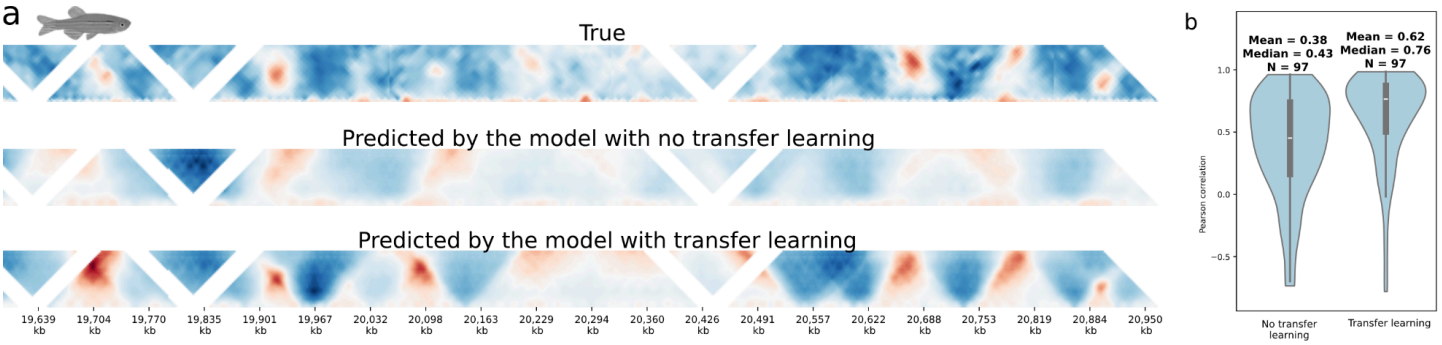

Archagozetes longisetosus, test sample size = 29
Mean: 0.09 0.10 0.15 0.18 0.21 0.24 0.26 0.26 0.30 0.31 0.32 0.30 0.30 0.31 0.31 0.31 0.31 0.31 0.31 0.30 0.30 0.29 0.28 0.28 0.27 0.25 0.24 0.23 0.22 0.21 0.22
Median: 0.09 0.09 0.13 0.22 0.25 0.32 0.35 0.35 0.32 0.34 0.32 0.31 0.30 0.32 0.32 0.31 0.35 0.37 0.33 0.35 0.34 0.35 0.34 0.31 0.29 0.23 0.26 0.27 0.29 0.27 0.28
Pearson cor relation
Predictions
Control
Genomic distance, kb


Arion vulgaris, test sample size = 74
Mean: 0.10 0.14 0.19 0.24 0.29 0.31 0.32 0.32 0.34 0.35 0.35 0.35 0.34 0.33 0.33 0.32 0.32 0.32 0.32 0.31 0.31 0.32 0.32 0.33 0.33 0.34 0.34 0.34 0.34
Median: 0.11 0.17 0.22 0.28 0.32 0.37 0.38 0.37 0.38 0.38 0.38 0.40 0.40 0.39 0.38 0.35 0.34 0.35 0.36 0.37 0.38 0.38 0.38 0.39 0.39 0.36 0.37 0.37 0.36 0.36 0.36
Pearson cor relation
Predictions
Control
Genomic distance, kb


Pomacea canaliculata, test sample size = 31
Mean: 0.05 0.08 0.18 0.24 0.30 0.36 0.40 0.42 0.44 0.44 0.46 0.46 0.47 0.47 0.46 0.46 0.46 0.44 0.44 0.46 0.46 0.47 0.47 0.48 0.48 0.49 0.50 0.51 0.52 0.54 0.54 0.54
Median: 0.02 0.01 0.12 0.31 0.34 0.42 0.42 0.44 0.46 0.47 0.45 0.48 0.51 0.50 0.49 0.47 0.50 0.48 0.48 0.52 0.53 0.51 0.55 0.53 0.53 0.55 0.57 0.60 0.60 0.63 0.61 0.58
Pearson cor relation
Predictions
Control
Genomic distance, kb



Supplementary Figure 5. Detailed metrics of model predictions from Suppl. Fig. 4



Supplementary Figure 6. Transfer learning for model improvement for *Danio rerio*. Hi-C data for, *Danio rerio* adult brain from [89]. Pre-training was done on human HFFc6 Micro-C data [54].

a. Hi-C maps for a test genomic region of *Danio rerio*: top – true data, middle – predicted by a model without transfer learning, bottom – with transfer learning, where the model was first trained on human data.

b. Pearson correlation on the test set between the original maps and the predicted by the models for *Danio rerio* trained without (left) and with transfer learning (right).

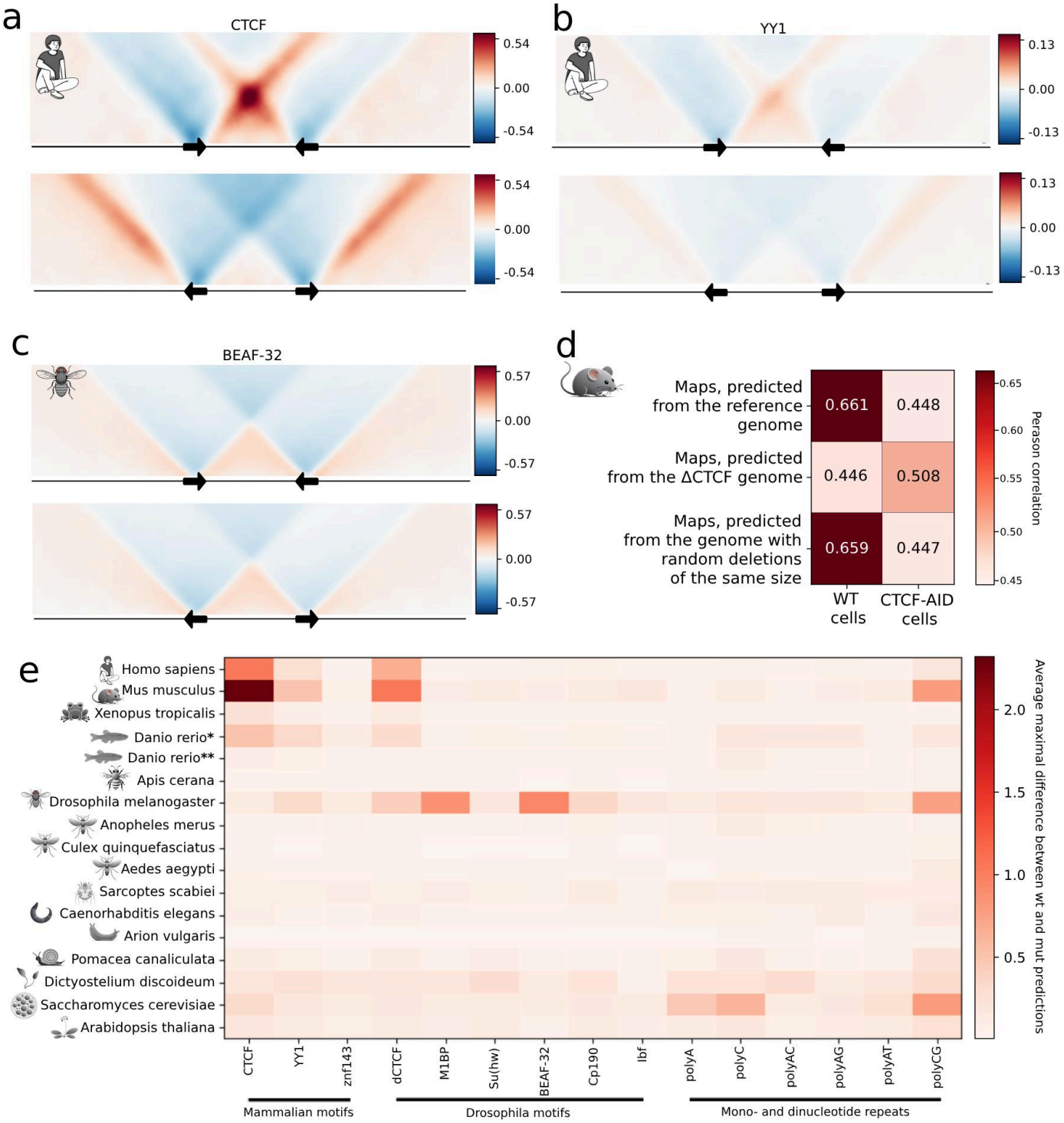

a
CTCF
0.54
0.00
-0.54
b
YY1
0.13
0.00
-0.13
c
BEAF-32
0.57
0.00
-0.57
d
Maps, predicted from the reference genome
Maps, predicted from the ΔCTCF genome
Maps, predicted from the genome with random deletions of the same size
0.661
0.448
0.446
0.508
0.659
0.447
WT cells
CTCF-AID cells
Pearson correlation
0.65
0.60
0.55
0.50
0.45
e
Homo sapiens
Mus musculus
Xenopus tropicalis
Danio rerio*
Danio rerio**
Apis cerana
Drosophila melanogaster
Anopheles merus
Culex quinquefasciatus
Aedes aegypti
Sarcoptes scabiei
Caenorhabditis elegans
Arion vulgaris
Pomacea canaliculata
Dictyostelium discoideum
Saccharomyces cerevisiae
Arabidopsis thaliana
CTCF
YY1
znf143
dCTCF
M1BP
Su(hw)
BEAF-32
Cp190
Ibf
polyA
polyC
polyAC
polyAG
polyAT
polyCG
Mammalian motifs
Drosophila motifs
Mono- and dinucleotide repeats
Average maximal difference between wt and mut predictions
2.0
1.5
1.0
0.5

Figure 5. *In silico* mutagenesis shows importance of known DNA motifs for Chimaera predictions in diverse species. ΔCTCF – genome with *in silico* deletion of CTCFs. AID – auxin-induced degradation. wt – wild-type genome, mut – genome with *in silico* mutagenesis by insertion. * – *Danio rerio* fine-tuned model on adult brain cells [89] with transfer learning from human, ** – *Danio rerio* model on embryo cells [9] without transfer learning.

(**a-c**). Average effect of insertion of two tandems with sites into a large number of sequences. Top: convergent orientations, bottom: divergent orientations. Averaging over n=50 randomly picked genomic regions (from Chimaera validation/test set).

a. Insertions of CTCF sites into the human genome.

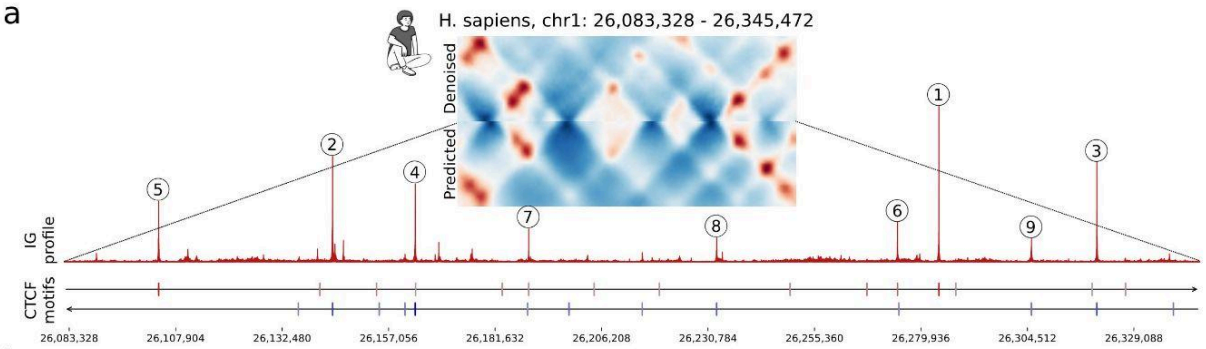
b. Insertions of YY1 sites into the human genome.

c. Insertions of BEAF-32 sites into the *D. melanogaster* genome.

d. Average correlations of predictions with the true Micro-C of WT and CTCF-AID cells. Top row: predictions based on the reference genome, middle row: reference genome with *in silico* deleted CTCF sites (ΔCTCF), bottom row: reference genome with randomly selected regions deleted (same length as CTCF sites). Note how ΔCTCF-genome predictions are more similar to CTCF-AID Micro-C. Results for chromosome 19, which was not used for Chimaera training.

e. Average maximum changes in predictions caused by multiple insertions of sites with motifs of architectural factors and simple repeats (columns) into the reference genomes of different species (rows). Averaging over n=20 randomly picked genomic regions (from Chimaera validation/test set).

a



b

	Found by integrated gradients	Found by gradient descent	Found by evolutionary search
Homo sapiens	CTCF (pr = 2e-9, ps = 1e-7)	CTCF (pr = 3e-14, ps = 3e-14)	CTCF (pr = 2e-10, ps = 2e-9)
Mus musculus	CTCF (pr = 6e-8, ps = 2e-8)	poly-CpG (pr = 4e-16, ps = 3e-7)	CTCF (pr = 3e-15, ps = 3e-15)
Xenopus tropicalis	CTCF (pr = 6e-5, ps = 7e-8)		
Danio rerio	* CTCF (pr = 2e-6, ps = 2e-8)	** CTCF (pr = 4e-10, ps = 4e-7)	** GC-rich (pr = 4e-13, ps = 1e-7)
Drosophila melanogaster			**unknown**(pr = 5e-11, ps = 4e-10) M1BP (pr = 9e-14, ps = 2e-15) BEAF-32 (pr = 6e-6, ps = 9e-9)
Saccharomyces cerevisiae		GC-rich (pr = 7e-16, ps = 0.16)	GC-rich(pr = 2e-13, ps = 0.17)
Arabidopsis thaliana			GC-rich (pr = 1e-7, ps = 0.34)

— found using projections on **insulation** vector — found using projections on **fountain** vector

Reference motifs from JASPAR database:

CTCF (H. sapiens) M1BP (D. melanogaster) BEAF-32 (D. melanogaster)

Figure 6. *De novo* motif search, three methods developed in this work: Integrated Gradients (IG), Gradient Descent, and Evolutionary Search. Motif impact significance: pr – P-value for the control with randomized motifs of the same length, ps – P-value for the control with shuffled motifs (see Methods "*Significance of motifs impact by in silico mutagenesis"*). * – *Danio rerio* fine-tuned model on adult brain cells [89] with transfer learning from human, ** – *Danio rerio* model on embryo cells [9] without transfer learning.

a. Example of Integrated Gradients profile for a region of the reference genome. Note that peaks correspond to CTCF sites detected in the same region (bottom tracks). Brightness of the color at the bottom panel refers to the CTCF site score based on its motif.

b. Table of motifs detected by three methods. Note that for Gradient Descent and Evolutionary Search we searched specifically for motifs associated with insulation (blue signs) or fountains (red signs). Empty cells indicate no motif found by a given method. The height of each letter in the motif logo represents the information content of the position in the position-specific weight matrix based on the method output. Note that Evolutionary Search and Integrated Gradients position weights can be interpreted as the importance for the prediction, but should be taken with caution as Evolutionary Search usually overestimates the importance of each position relative to the true motif (see CTCF example in the table).

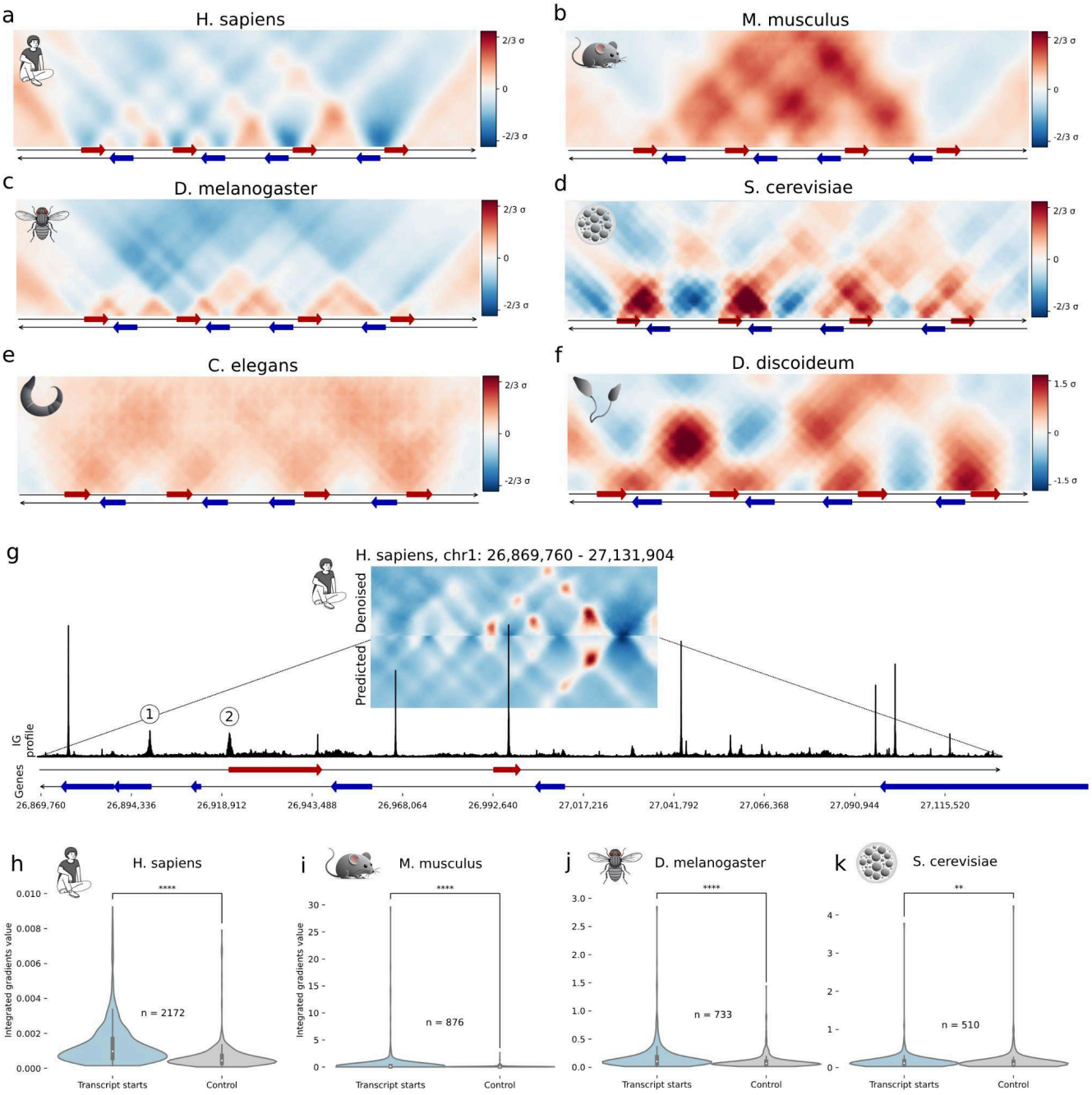
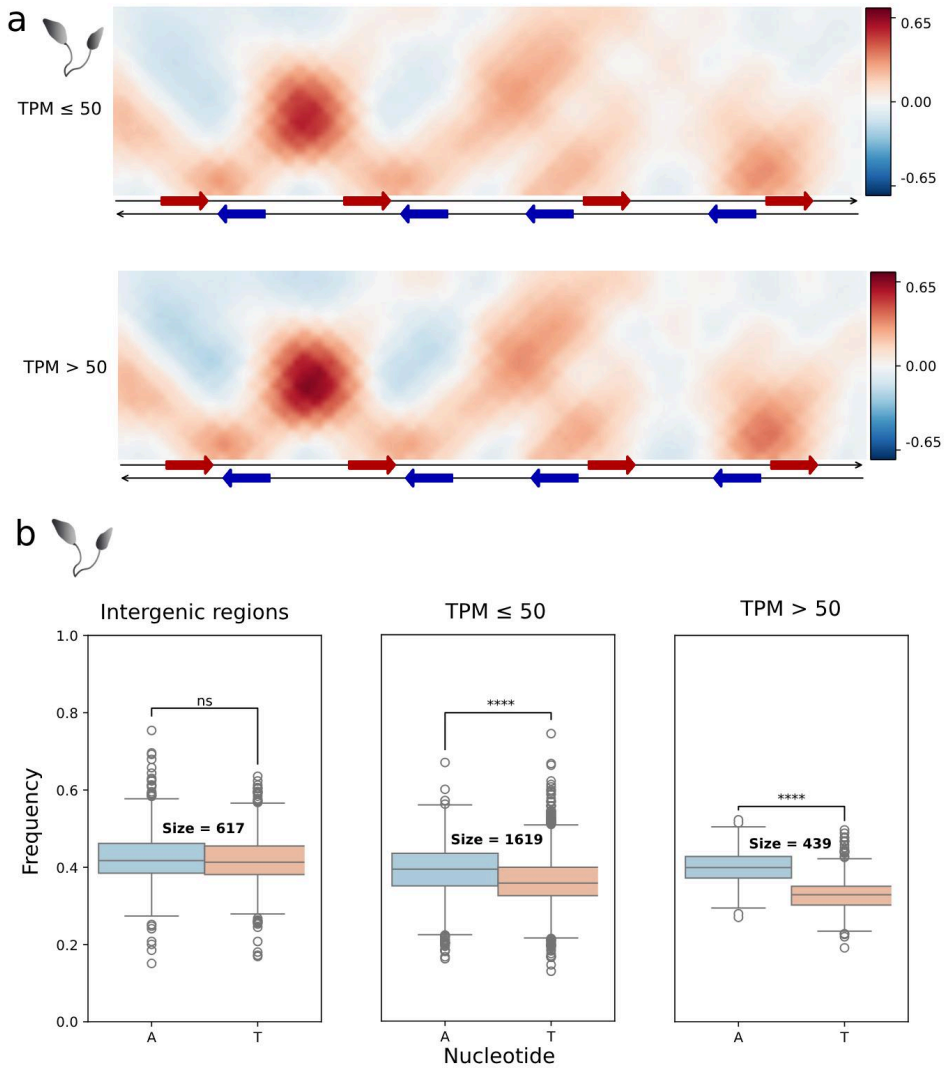


Figure 7. Effect of gene location on model predictions. Gene annotation based on Ensemble database [90] for all species and dictyBase for *Dictyostelium* [91].

(**a-f**). Averaged predictions of Chimaera models based on *in silico* constructed sequences containing genes in specified orientation. Genes are shown by arrows at the bottom of each panel. The color scale for each heatmap represents standard deviations of train samples, calculated for each organism independently.

g. Example of an integrated gradients profile for a region of the human reference genome, based on a model trained for human. Broad peaks of integrated gradients are present at the promoters of some genes (genes are shown with arrows at the bottom).

(**h-k**). Comparison of average integrated gradients values around promoters against randomly picked genomic regions of the same size (control). P-values are for two-sided Mann-Whitney.



Supplementary Figure 7. Gene expression in *Dictyostelium* affects model predictions. We took genes with different levels of expression at the vegetative state of *Dictyostelium* development [30], and tested their properties with Chimaera (a) and nucleotide content (b). TPM – transcripts per million.

a. *In silico* generated sequence with genes with low (TPM<50, top) and high (TPM>50, bottom) expression level. Note how the loops and other patterns are more pronounced for the highly expressed genes, although the model was not informed on the expression level.

b. Testing the hypothesis that nucleotide content is different for genomic regions with different expression properties: A and T nucleotide frequencies at intergenic regions and at the genes with low and high expression levels.

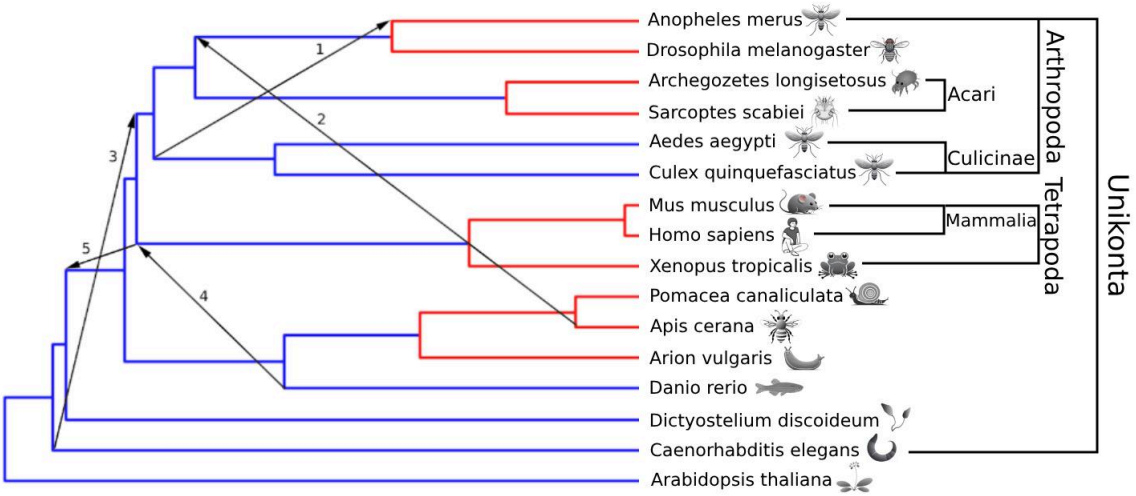
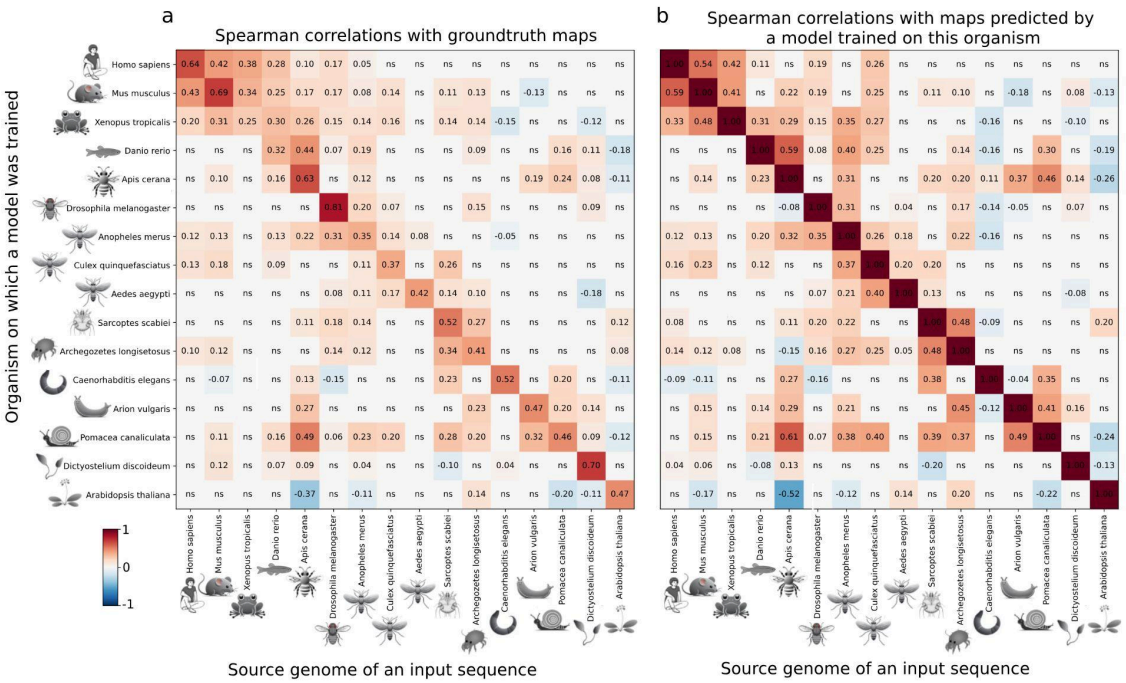


Figure 8. Cluster tree of studies species based on correlation matrix of cross-predictions of models trained for different species. Average of tables presented in Suppl. Fig. 8a-b was used for tree inference (UPGMA). We excluded *Saccharomyces* from this analysis due to very small bin size (512 bp) relative to other species (from 1 Kb in *Drosophila*, see Table 3).

Right: General taxa groups.

Red branches: Robust clades that are reproducible between clustering by different clusterization approaches (agglomerative clustering with different linkages on Suppl. Fig. 8a and Suppl. Fig. 8b).

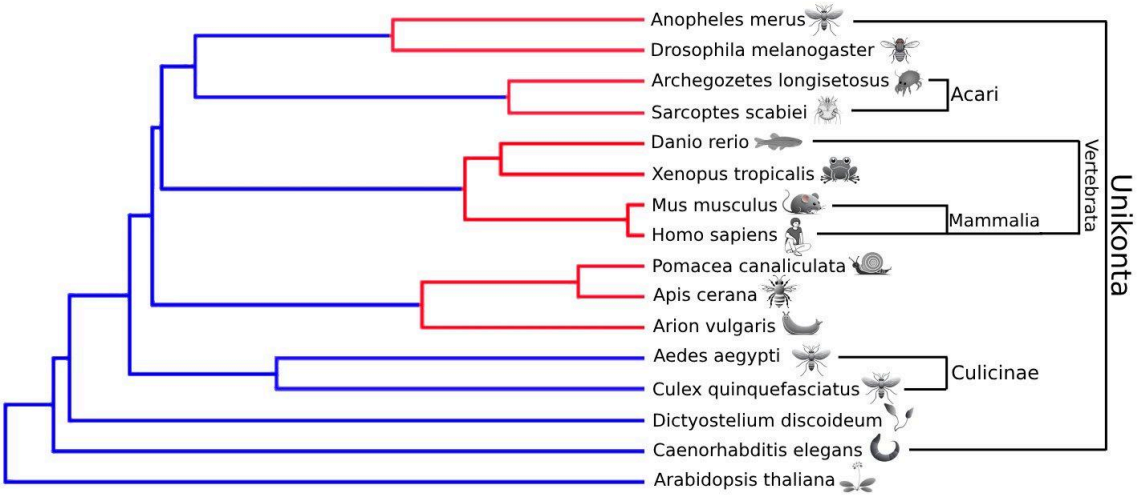
Black arrows: Tree pruning and grafting that would result in perfect taxonomic tree: 1 — moving two mosquitoes to the dipteran cluster, 2 — moving bee to the resulting insect cluster, 3 — moving nematode to the resulting *Ecdysozoa* cluster, 4 — moving fish to the vertebrate cluster, 5 — moving vertebrates out of the resulting cluster of protostomes.



Supplementary Figure 8. Cross-prediction correlation matrices. Supporting information for Figure 8. ns – correlations that are not statistically significant.

a. Correlations between true Hi-C maps at genomic loci of different species (columns) and predictions made by the models trained for different species (rows). Note that the values at the diagonal reflect how well the model was able to predict the Hi-C maps in the species where it was trained.

b. Correlations between Hi-C maps predicted by a model trained on a given species (columns) and Hi-C maps predicted by the models trained for different species (rows). Note that now all the diagonal values are equal to 1.



Supplementary Figure 9. Cluster tree with *Danio rerio* model based on transfer learning from human data. Similar to Fig. 8. Note that *Danio rerio* is now correctly placed with other vertebrates, however, two mosquitos (Culicinae subfamily) are now separated from the rest of insects.

Supplementary Information

Species	Median Pearson correlation coefficient of full true and predicted maps	Maximum median Pearson correlation coefficient of map contacts at a certain distance (Hi-C map diagonal)	Distance at which the correlation maximum is achieved, Kbp	Significance of difference from control (randomly paired true and predicted maps)	High-resolution dataset used for training of autoencoder if transfer learning was adapted
Homo sapiens	0.78	0.84	140	****	
Mus musculus	0.67	0.75	135	****	
Xenopus tropicalis	0.51	0.56	110	****	
*Danio rerio**	0.52	0.54	220	***	*H. sapiens*
Arion vulgaris	0.39	0.41	120	****	*D. melanogaster*
Pomacea canaliculata	0.54	0.63	130	****	*H. sapiens*
Apis cerana	0.79	0.89	120	****	*D. melanogaster*
Drosophila melanogaster	0.75	0.83	40	****	
Anopheles merus	0.51	0.54	70	****	*D. melanogaster*
Culex quinquefasciatus	0.32	0.36	110	***	*D. melanogaster*
Aedes aegypti	0.37	0.49	150	****	*D. melanogaster*
Sarcoptes scabiei	0.53	0.62	30	****	

Archegozetes longisetosus	0.35	0.37	80	**	
Caenorhabditis elegans	0.53	0.64	65	****	
Saccharomyces cerevisiae	0.67	0.81	2	****	
Dictyostelium discoideum	0.73	0.83	25	****	
Arabidopsis thaliana	0.42	0.57	40	***	*X. tropicalis*

Supplementary Table 1. Quality of predictions on test samples for different models for different species after a fixed number of epochs. If Hi-C data quality is low, the training may not converge at all without Hi-C decoder transfer, and if data resolution is high enough, using Hi-C decoder trained on another organism doesn't affect prediction quality (column 6). * – For *Danio rerio*, the results are shown on embryonic data, without transfer learning (pre-training the full model on human data), see Suppl. Fig. 6 for details.

Model variant	Model	*S. cerevisiae*	*H. sapiens*
1	Adapted Akita (modified for rotated maps)	0.49	**0.48**
2	Chimaera with Akita head	0.50	0.44
3	Chimaera with Hi-C-decoder trained on other organism data	0.56	**0.48**
4	Chimaera model omitting pre-training of Hi-C-decoder	0.47	0.41
5	Chimaera with pre-trained Hi-C-decoder trained to predict raw maps	0.52	0.42
6	Chimaera with pre-trained Hi-C-decoder trained to predict denoised maps	**0.59**	**0.49**

Supplementary Table 2. Pearson correlation between true non-denoised and predicted maps from the validation samples after a fixed number of training epochs for different versions of the models on *S. cerevisiae* and *H. sapiens* data. Model variant #4 is a control that demonstrates the effectiveness of pre-training of Hi-C decoder prior to DNA encoder, potentially due to simplifying the Hi-C map "drawing rules" (see Discussion). Model variant #5 is a control that tests prediction of raw maps instead of denoised maps, which works slightly worse potentially due to contribution of unlearnable noise in the data (see Discussion). Model # 6 is a final variant used throughout the paper.

Species	Fragment size	Hi-C map resolution	Train set size	~Genome size, Mbp	Genome assembly (name or NCBI Assembly Accession)
Homo sapiens	262144	2048	8424	3100	hg38
Mus musculus	262144	4000	9287	2700	mm10
Xenopus tropicalis	405000	15000	3151	1500	xenTro10
Danio rerio	400000	20000	2861	1400	danRer11
Arion vulgaris	500000	25000	2247	1500	GCA_020796225.1
Pomacea canaliculata	262144	10000	1216	440	GCF_003073045.1
Apis cerana	262144	10000	483	215	GCA_011100585.1
Drosophila melanogaster	80000	1000	1322	175	dm6
Anopheles merus	200000	5000	930	294	GCF_017562075.2
Culex quinquefasciatus	262144	10000	1752	573	GCF_015732765.1
Aedes aegypti	400000	10000	2818	1300	GCF_002204515.2
Sarcoptes scabiei	131072	5000	282	57	GCA_020844145.1
Archegozetes longisetosus	262144	10000	308	141	GCA_018873245.2
Caenorhabditis elegans	131072	4096	670	100	ce10
Saccharomyces cerevisiae	16384	512	583	12	sacCer3
Dictyostelium discoideum	65536	2000	406	33	GCA_000004695.1
Arabidopsis thaliana	262144	10000	328	119	TAIR10

Supplementary Table 3. Input data properties for Chimaera training

Methods

Data collection

In total, we have collected data for 17 organisms (see Suppl. Table 3): Micro-C for *Homo sapiens* HFFc6 cell line from [54], Micro-C for *Mus musculus* embryonic stem cells from [57] (also used Micro-C data with structural depletion of structural proteins from the same study and for cell cycle stages from [51]), Hi-C for *Xenopus tropicalis* embryos from [77], Hi-C for *Danio rerio* embryos at 5.3 hours past fertilization from [9], Hi-C for *Apis cerana* from the pupae of drone pupae from [78], Micro-C for *Drosophila melanogaster* for nc14 stage embryos from [79], Hi-C for mosquitoes *Anopheles merus* from [38], *Aedes aegypti* from [80], *Culex quinquefasciatus* from [81], Hi-C for mites *Sarcoptes scabiei* from [82], *Archegozetes longisetosus* from [83], Hi-C for nematode *Caenorhabditis elegans* from [84], Hi-C for molluscs: muscle tissue of slug *Arion vulgaris* [85], and snail *Pomacea canaliculata* from [86], Hi-C for social amoeba *Dictyostelium discoideum* from [30], Micro-C XL for yeast *Saccharomyces cerevisiae* at 90 minutes after release from G1 from [87], and Hi-C from flowers of *Arabidopsis thaliana* from [88].

Hi-C data preparation

Hi-C snipping. We extracted specific regions, or "snippets," from the Hi-C interaction matrices for model input. This approach provided focused sections that emphasized local chromatin interactions, optimizing computational efficiency in downstream processing.
Snippet positions were determined by the model training scheme, as described below: the map was segmented into constant-sized fragments (see Suppl. Table 3), shifted along the genome with a stride equal to half the fragment size.

Iterative correction. When snipping, we obtained Hi-C/Micro-C contact frequencies normalized by iterative correction [47], which (1) removes bin-specific mappability, amplification and other biases, (2) normalized interactions and allows interpretation as contact frequencies.

Log-transformation. To further stabilize the variance of contact frequencies, we applied a log transformation. This adjustment reduces the impact of high-frequency outliers and makes the distribution of contact frequencies more amenable to neural network training. Note that we aimed to avoid obtaining non-interpretable values for Hi-C elements with zero contacts, and added a pseudocount of 10^{-3} to all the elements in the matrix before log-transformation.

Observed-over-expected normalization. To account for the tendency of contact frequency to decrease sharply with increased genomic distance, we normalized observed log-transformed Hi-C contact interactions by subtracting the expected interactions for each genomic separation distance. Expected interactions were calculated as mean log-transformed average values for each genomic separation for each chromosome. This observed-over-expected transformation helps reveal local interaction patterns that may otherwise be overshadowed by distance-dependent effects.

Normalization by standard deviation. Next, we divided all the values by standard deviation of the signal for each chromosome.

Treating unmapped regions. We used two approaches to treat missing data in the input Hi-C maps. First, we detected unmapped bins as those that failed to be iteratively corrected by *cooler* [92] with default parameters. Then, we have interpolated those bins by *cooltools*-based bilinear interpolation [50]. This approach was used for training the Hi-C autoencoder.
Alternatively, for the training that requires more rigor (such as DNA encoder), we have excluded the unmapped bins from while training and evaluation to avoid learning Hi-C artifacts instead of the true interaction signal. For

that, we have dynamically modified the loss function (see below) and zeroed out the unmappable regions of the Hi-C maps.

45-degree rotation. To minimize the influence of noisy long-range interactions, we rotated each normalized Hi-C map by 45 degrees by *scipy* [93] *ndimage.rotate* using spline interpolation of 0 order (no smoothing), effectively rearranging interaction patterns to emphasize local contacts. This transformation (i) excludes long-range interactions beyond half the fragment size and (ii) resizes the Hi-C map to dimensions of 128×32, standardizing the inputs and outputs across all models. This approach improves the model's focus on essential chromatin interaction structures while minimizing noise from distant genomic contacts. For datasets with worse data quality, we applied 1-st order spline interpolation.

DNA data preparation

To match the DNA sequence data with Hi-C interaction matrices and optimize it for model training, we performed several preparatory steps to standardize input format and address resolution mismatches:

One-Hot encoding. DNA sequences were encoded in a one-hot format, where each nucleotide (A, T, C, G) is represented by a unique binary vector. Nucleotides with ambiguous base calls were replaced with "N" (zero one-hot encoded vector), ensuring that unresolved regions do not contribute to the sequence representation.

Choice of DNA fragment matching Hi-C map. DNA sequence was taken from the region centered at the corresponding Hi-C snippet location. The size of the DNA window was larger than those covered by Hi-C to allow the model to learn from the context of the input Hi-C interactions.
The choice of offset size was species-specific, and guided by balancing computational efficiency and predictive quality. Larger offset windows improve alignment accuracy but increase memory consumption and training time. Therefore, the final offset size was optimized to provide high predictive quality while managing memory and processing constraints effectively. The offset size was capped at no more than half the Hi-C window size, allowing for adequate coverage while limiting excessive data expansion.

Chimaera implementation

We implemented Chimaera with Python framework pytorch [94]. Chimaera consists of two parts: Hi-C autoencoder and DNA encoder that are trained separately (Suppl. Fig. 1a).

The first component is a *Hi-C autoencoder* designed to produce both a denoised Hi-C map and its latent representation. This 2D convolutional network includes an encoder that transforms the input image into a multi-dimensional latent vector, and a decoder that reconstructs a denoised image from this latent representation. We have max pooling operations between the sequential layers of the network and added two fully connected layers at the transition to and from latent space (see Suppl. Fig. 1a, horizontal component). We used 128 dimensions for most species, and 96 dimensions for the species with relatively small train samples.

The second part is *DNA encoder*, a model that converts the nucleotide sequence into a multi-dimensional vector. It is a 1D convolutional network with residual blocks. Again, we used max pooling between some convolutional layers and added a fully connected layer at the end (see Suppl Fig. 1a, vertical component). The number of dimensions of the latent space matched the choice of number of dimensions for the Hi-C autoencoder.

We implement an API that abstracts the Chimaera model creation and training. Model requires cool-formatted [92] Hi-C/Micro-C files as input, and DNA sequence in fasta format. The model then allows the choice of Hi-C map resolution and fragments size. The problem of parameter choice is simplified by the Chimaera module performing data quality control and visualization.

Next, the user can set the hyperparameters for the Hi-C and DNA encoders, including: the number and the size of convolutional filters, the number of residual blocks, the percentage of dropout, and many more. We have three ready-to-use presets that we recommend for different sizes of the genomes:

(1) “Small” that is designed to be trained on the small sample sizes that are risky to achieve overfitting. It has higher dropouts, no residual blocks, and 96-dimensional latent space.

(2) “Middle” and (3) “Big” are both designed for larger sample sizes that are expected to result in less overfitting. The dropout rates are smaller, the latent space dimensionality is 128 and residual blocks are used (4 in “Middle” and 8 in “Big”).

Chimaera’s API includes model training, validation, visualization of the results and interpretation of the model as separate modules. The API and the examples of usage are available at GitHub link: https://github.com/ashkolikov/chimaera/

Hi-C autoencoder training

The Chimaera workflow began with the training of an *Hi-C autoencoder* (Fig. 1a). Weights are initialized using Kaiming initialization [95].

Model setup. As input, it takes Hi-C/Micro-C pre-processed snippets. Loss function is a Mean Squared Error (MSE) between original and reconstructed map, with Kullback–Leibler (KL) divergence regularization. Regularization penalizes the difference of the latent space representation of the Hi-C autoencoder and standard normal distribution, which makes our encoder variational and is a prerequisite for achieving continuity of the latent space [96]. Note that when calculating the loss function between the true and predicted maps, we used maps with interpolated unmapped bins (see “Hi-C data preparation”).

Model training. Chimaera network weights were optimized by Adam gradient descent [97]. We stopped training of the model when the plateau of the validation quality metrics was achieved.

Whole-genome model application. Once the Hi-C encoder was trained, we ran it on all snippets of Hi-C maps and obtained (1) latent representations and (2) reconstructed maps across the genome of each species. We later used this information for training DNA encoder, model quality control, and interpretation.

DNA autoencoder training

After pre-training the Hi-C autoencoder, we fixed the weights of the *Hi-C decoder* and stacked it to the *DNA encoder* (Fig. 1b). Weights of DNA encoder were also initialized using Kaiming initialization [95]. As input, the DNA encoder takes a one-hot encoded sequence of the reference genome. During training, we used Adam gradient descent [97].

Loss function. Importantly, we set the loss function as MSE between a DNA-based predicted Hi-C map and an autoencoder-reconstructed Hi-C map for the same snippet. The rationale behind this choice is that decoded Hi-C maps after Hi-C decoder (1) contain only the information that is possible to pass through a lower-dimensional vector, and (2) are denoised from irrelevant random fluctuations in data coverage and dropouts. In Chimaera API, we retain an option for a user to use raw Hi-C maps as ground truth for training. However, this option has demonstrated to have poor performance (see Suppl. Table 1, rows 5-6). Notice that in both cases evaluation metrics were calculated with raw maps.
For clarity and uniform interpretation of the model quality, we run validation on the original Hi-C maps, without applying Hi-C decoding.
In contrast to Hi-C autoencoder training, when calculating the loss function between the true (decoded) and predicted maps, we masked out interpolated unmapped bins (see “Hi-C data preparation”) — for these pixels the loss function was set to 0, and for the remaining ones it was reweighted to preserve the average value. In this way, the model learned to predict only pixels obtained from experimental data, ignoring possible interpolation artifacts.

Data augmentation. Hi-C data, by design, is not strand-specific; the Hi-C map for the forward strand mirrors that of the reverse complement strand when rotated 180 degrees. To prevent strand bias, we implemented a random reversal of data points (reverse Hi-C along with the corresponding reverse complement DNA). This approach also effectively doubles the data pool for augmentation purposes.

Data segmentation and splitting

Segmentation of the chromosomes into fragments. Since the input fragments were parts of contiguous chromosomes, they could be sliced with overlaps. At the same time, feeding the same sequences into the model, but in different contexts, would help to better resist overfitting. To do this, each training epoch included several sub-epochs, after each of which the genome marking of the training sample was shifted by a small step. We chose a step of 1/12 of the fragment length, which improved the metrics on the validation sample by up to 20%.
For the final prediction of the models, predictions were made for both direct and reverse sequences, and the results were averaged. Empirically, this averaging approach enhanced prediction accuracy across all organisms. Low-quality fragments with mappability less than 50-95% (depending on the input data quality for each species) were removed.

Train-test-validation splitting. Fragments for test and validation samples were taken from full chromosomes or contiguous parts of long chromosomes separated for this purpose. We ensured that the train and test/validation sets never contained overlapping Hi-C or DNA segments.
For species with high-resolution abundant data, we reserved the whole chromosomes as test/validation. For the species with more limited data, we selected continuous parts of long chromosomes (to ensure at least 30 fragments) for the test/validation sample. To ensure no leakage between train and test, we implemented an internal check function in Chimaera that ensures that no overlapping fragments happen to be both in train and test/validation.
The split between test and validation was 2:1.

Model performance evaluation. Throughout this work, we used two Chimaera performance metrics, (1) Pearson correlation between whole Hi-C maps for a genomic fragment (most of the quality controls in the paper if not mentioned otherwise), (2) Pearson correlation at specific distance (for Suppl. Table 1, 2 and Suppl. Fig. 4.2).

Pearson's correlation at specific distance measured the agreement between true and predicted maps for each genomic separation (in other words, rows of 45-degrees rotated Hi-C matrices). This allowed us to study the model performance for different genomic separations, select the characteristic distances where the model performs the best, and avoid the issue of Hi-C having worse quality and signal-to-noise ratio at higher genomic separations. To ensure there is no correlation between random pairs of map fragments, we controlled by calculating the correlations between predicted maps and randomly picked maps from the dataset. Significances for each distance were calculated using the two-sided Mann-Whitney test and p-values were adjusted using Benjamini-Hochberg procedure.

Robust predictions for validation purposes. For visualization and scores in the Suppl. Tables, we run robust predictions by Chimaera, when we run multiple predictions of the neighboring regions and average overlaps between them for the final output. Robust predictions usually have higher quality (by ~10%), but were not used for the hyperparameter choice nor for the criterion of the model training.

Pattern calling in Hi-C maps

Selection of the genomic regions for pattern calling. For pattern calling, we used the whole genome Hi-C maps. We did not separate into train and test for this task, as we were interested in the whole-genome presence of certain patterns.

Genome scanning for latent representation evaluation. We next applied Chimaera to the snippets of the Hi-C maps, with the fragment size as required by the model (see Suppl. Table 3) and the stride equal to 8 bins in the transformed Hi-C matrix dimensions (after 45-degree rotation, see “Hi-C data preparation” section). Each snippet served as an input for Hi-C encoder, and the output was recorded.

Obtaining “ideal” known patterns. First, we constructed “ideal” Hi-C patterns: insulation, loop and fountain. We started with the *in silico* Hi-C matrix filled with zeros. The size of the matrix was set to the size of the Chimaera input (128×32).

For the insulation pattern, we positioned a right triangle filled with negative values, with its apex at the bottom (in genomic separation coordinates) centered within the snippet (in genomic position coordinates). The insulation in the ideal pattern is thus centered in the middle of the snippet (see Fig. 1d).
For the loop pattern, we placed the rhombus with diagonal size=8 at the position equal to half of the window size (in genomic separation coordinates).
For the fountain pattern, we positioned a triangle filled with positive values, with its apex at the bottom (in genomic separation coordinates) centered within the snippet (in genomic position coordinates). The angle at the bottom-facing apex was 45 degrees, reflecting the growth of the fountain spread with genomic separations [9].
For the TAD pattern, we positioned a right triangle filled with positive values, with its apex at the top (in genomic separation coordinates) centered within the snippet (in genomic position coordinates).

Obtaining the latent representation of the "ideal" known patterns. The "ideal" known patterns were input into the Chimaera Hi-C encoder, and the latent representations were obtained.

Similarity of latent representations. To measure the similarity between the latent representation of Hi-C snippets from real Hi-C maps and the "ideal" patterns, we projected the latent vector of the real Hi-C representation onto the latent vector of the "ideal" pattern. This results in a number (measure of similarity) of Hi-C signal in genomic position to the "ideal" pattern. Combined with the whole-genome scanning, this results in a genomic track of similarity to a given pattern.

Thresholding of similarity track. To obtain a list of "called" instances of patterns present in Hi-C, we applied a thresholding of similarity track. We used two approaches to choosing the threshold:
(1) *Kneedle algorithm*. First, we sorted projections of all genomic bins and found the elbow point of the resulting plot for values above the median using Kneedle algorithm [98]. Next, we applied this threshold and considered all the genomic regions with values higher than this value as a "hit". This allowed us to automatically detect insulation, fountains, and loops in the Hi-C maps in development of species and cell cycle (Fig. 3 a-b), and then plot average patterns in these processes.
(2) *Top hits across all stages*. As an alternative to the Kneedle algorithm, we applied a biology-based approach for setting similarity track thresholds. Previous studies estimated the frequency of TAD boundaries and fountains: approximately one fountain occurs every ~500 Kb in the zebrafish genome around 5.3 hpf, and insulation events occur every 200-500 Kb in mammalian cells in asynchronous conditions [7,9,99]. Based on this, we assumed that the number of features per dataset should align with these estimates. For each dataset, we defined regions with high similarity scores as "hits" based on the similarity distribution across all stages (e.g., throughout the mouse cell cycle): insulation regions in the top 2%, fountains in the top 0.3%, and loops in the top 0.5%. These thresholds, shown in Suppl. Fig. 3, were applied in Fig. 3c-d.

De novo **DNA motif search with Integrated Gradients**

Integrated gradients [100] is a method that allows obtaining information about the positions in the input data that most significantly affect the model prediction. Integrated gradient is calculated based on the inputs and the weights of the model, with the higher values corresponding to higher influence.

We implemented the Integrated Gradients for the *DNA encoder* model, and defined it as follows:

$$(x_i - x_i') \times \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m} \times (x - x'))}{\partial x_i} \times \frac{1}{m}$$

where: i – is a position in the input data (DNA nucleotide position),
x_i – is an initial value in the given position (DNA base),
x' – is a base level (no DNA input, empty input),
m – is the number of interpolation levels,
k – is a given interpolation level, and
F – is a neural network model (fully differentiable).
We set the number of interpolation levels to 20.

The resulting values of integrated gradients were used to detect the importance of the genomic position on the prediction (see Fig. 6a, 7g).

To select the most impactful regions for the prediction by the integrated gradients, we selected the maximal value for each genomic fragments. We then fetched a 50 bp sequence around the peak and set profile values for it to zero. Then this procedure was repeated 15 times for each DNA fragment. Then the resulting 50 bp sequences were fed into the MEME algorithm to find motifs.

De novo DNA motif search with Gradient Descent

Gradient Descent is a method for searching the DNA motif associated with a specific pattern of Hi-C maps. This method is based on the latent representations of Hi-C maps obtained by the Hi-C encoder. See Suppl. Fig. 1b for schematics of Gradient Descent motif search.

(i) As an input for this method, we provide a map with an ideal pattern in its center (for example, insulation). This map is converted to its latent representation by Hi-C encoder.

(ii.a) We then take a random DNA fragment from the genome (from the test sample), take a one-hot encoded representation of it, and replace the central 100 bp of it (*target window*) with uniformly distributed nucleotide frequencies. This central 100 bp region will be later modified to optimize the similarity of the resulting pattern in the predicted Hi-C matrix from this input.

(ii.b) We apply softmax activation to contrast the values in the DNA matrix and maintain the sum of each column equal to 1.

(iii) We then take the DNA encoder (note that the weights of the neural network are frozen at this stage), and provide it with the one-hot encoded DNA from (ii). This results in the latent representation of DNA sequence in the same space as the ideal pattern representation from (i).

(iv) Next, we calculate the loss function for Gradient Descent: a projection of the DNA-based latent vector onto the ideal Hi-C-based latent vector with inverted sign. We backpropagate the resulting loss function through the network (without modifying the weights) but update the values in the target window of the input DNA.

(v) We repeat the steps (ii)-(iv) 50 times (after confirming qualitative convergence of the resulting target window) and store the target window after applying the softmax function.

(vi) We run 50-100 re-initializations of the Gradient Descent for a given species and different DNA regions, and run a MEME-like algorithm (see Methods below) to find motifs in the input sequences.

De novo DNA motif search with Evolutionary Search

Evolutionary Search is another targeted motif search method, implemented as a genetic algorithm based on *in silico* mutagenesis. See Suppl. Fig. 1b for the schematic of Evolutionary Search for motifs.

(i) A map with an ideal pattern centered within it is provided as input. This map is then converted into its latent representation using the Hi-C encoder, equivalent to Step (i) in Gradient Descent (see above).

(ii.a) We generate 200 random short sequences (10-20 bp) using a uniform nucleotide distribution. The sequence length is chosen based on the organism, and both short and long sequences were tested for each species.

(ii.b) For each short sequence, a DNA fragment is selected from the test sample, and multiple copies of the short sequence are inserted at the center of this fragment (10 copies with a 20-nucleotide interval). Each “insert” is thus embedded within a random DNA context from the real genome.

(iii) Similar to Step (iii) of Gradient Descent, we then take the DNA encoder, and provide it with the one-hot encoded DNA sequences from Step (ii). This results in the latent representation for each input DNA sequence in the same space as the ideal pattern representation from Step (i).

(iv) *Estimate fitness.* We calculate projections of the DNA-based latent vectors onto the ideal Hi-C-based latent vector and raise this metric to the 10th power to accentuate differences. The resulting values serve as fitness scores for each short sequence. Standard normalization is applied to the fitness score distribution to maintain consistent selection pressure.

(v) *"Reproduction".* "Offspring" sequences are generated based on fitness scores. We normalize the fitness scores from Step (iv) by their sum to use them as sampling probabilities. Then, 180 sequences are sampled with replacement, allowing sequences with higher fitness scores to have more "offspring" and lower-fitness sequences to have fewer or none.

(vi) *Mutation.* Each "offspring" sequence undergoes random mutations, with a 5% chance of substitution at each position.

(viii) The top 20 sequences from the previous generation, ranked by fitness, are carried over to the next generation. This ensures that highly fit sequences are retained in the population, preventing loss of the best candidates.

(ix) The new sample of 200 short sequences then serves as input for the next iteration, cycling through Steps (iii) to (vii). This process repeats for a set number of epochs or until the mean fitness reaches a plateau.

(x) Finally, a position frequency matrix (PFM) is constructed from the resulting sample, summarizing the evolutionary search outcome.

Tasks on DNA motifs and *in silico* mutagenesis

In this work, we implemented and used a wide variety of the tools for working and DNA sequence and DNA motifs:

(1) obtain the known DNA motifs from the JASPAR database [101] as the position frequency matrix (PFM) – see Fig. 6 bottom,

(2) convert the position frequency matrix (PFM) to a positional weight matrix (PWM) by applying the log2 transformation to the background-normalized frequencies with a pseudocount, assuming equal nucleotide contribution at each position,

(3) retrieve consensus sequence from PFM and sample sequences based on a given PFM – for *in silico* mutagenesis,

(4) insert sequence into genomic DNA sequence in different orientations and combinations – for *in silico* mutagenesis,

(5) MEME[102]-based *de novo* motif search in a set of DNA sequences – for Integrated Gradients and Gradient Descent motif search, see Fig. 6,

(6) Tomtom[103]-based comparison of motifs against those from JASPAR database [101],

(7) manipulate the PFMs: reverse complement, randomize frequencies, shuffle positions.

These tools were applied consistently throughout the manuscript.

PWM-based deletion of CTCF sites from the genome. To remove CTCF sites from the genome for predicting maps of cells with depleted CTCF (Fig. 5d), we searched for the vertebrate CTCF PFM from the JASPAR database [101]. We selected a threshold giving a number of sites in the genome comparable to reported numbers of CTCF protein molecules bound to the mouse genome [104].

Significance of motifs impact by in silico mutagenesis. For each *de novo* found motif (see Fig. 6), we tested its significance by *in silico* inserting the sequences sampled from the motif into the genome and predicting the effect of insertion. Each insertion into the test region was in 10 copies with the 20 bp step.
For a given motif, we predicted both wild type and mutated sequences and calculated the difference between predicted maps. We then calculated the maximal absolute value of the difference map, and used it as a metric of the impact. This procedure was repeated 50 times with randomly picked genomic regions.

As a control, we used two types of randomizations: a shuffled motif (ps values in Fig. 6) and a random motif of the same length (pr values in Fig. 6). For each control procedure was the same and gave 50 values of a control metric. To calculate significance, the two-sided Mann-Whitney test on the observed and control samples was used.

Tasks on genes

In a similar approach to motif-based predictions via *in silico* mutagenesis, we developed a method for assessing gene-based effects. Using genome annotations from Ensembl [90] and RefSeq[105], we defined the following regions based on transcript annotations:
1. Promoters: regions around transcription start sites (TSS), spanning from -200 bp upstream to +100 bp downstream.
2. Gene bodies: regions from TSS to transcription end sites (TES), including both exonic and intronic sequences.
3. Intergenic regions: sequences between gene end and start annotations.

Promoter regions were used for collecting the statistics for integrated gradients in Fig. 7h-k.

Gene-based in silico mutagenesis. To design and test the sequences for assessing the effect of gene positioning in the DNA (Fig. 7a-b, Suppl. Fig. 7), we implemented and run the following steps:

(i) We created a background DNA sequence by concatenating random intergenic regions until reaching the desired fragment length, producing a gene-free sequence.

(ii) This “background” sequence was used to predict 3D genome organization with Chimaera, serving as a baseline for later gene-insertion comparisons.

(iii) To insert genes, we set a target gene length, usually one that allows for eight insertions within the fragment (see setup in Fig. 7a) and closely matches the species' median gene length. We randomly selected eight genes within 20% of the specified length.

(iv) These selected gene sequences were then inserted into the background sequence, replacing non-gene DNA.

(v) Using the modified sequence, we predicted its 3D genome organization with Chimaera, generating a “target” structure to compare with the baseline from (ii).

(vi) We subtracted the baseline map from the target map to quantify the effect of gene insertion on 3D organization (“effect maps”).

(vii) Steps (i)-(iv) were repeated 256 times, and the resulting effect maps were averaged (see Fig. 7a-f, Suppl. Fig. a).

This gene-based *in silico* mutagenesis approach enabled us to estimate the influence of specific gene insertions on genome architecture.

Cross-predictions

Selection of genomic regions for cross-predictions. For each species in the analysis, we picked ~5-20 Mbp-long continuous genomic regions from the test sample, prioritizing the regions with the minimum number of unmapped bins. Note that Chimaera predicts the Hi-C maps for the fragments from these regions, with fragment sizes ranging from 65 to 500 Kb (see Suppl. Table 3). Thus, we segmented these selected genomic regions into fragments of size specific to each model.

Cross-predictions run. We then input the selected DNA regions from each species to the models trained on other species. Since the region of each species was continuous, and we ran the prediction with the stride equal to half the fragment size, we were able to reconstruct the Hi-C signal for the whole DNA region (below certain genomic separations).

Model output adjustments to enable cross-species comparisons. Because the fragment lengths accepted by the models varied across organisms, the predicted maps were trimmed and scaled to a uniform size. Consequently,

S. cerevisiae was excluded from the analysis, as its fragment size was significantly smaller than that of the others.

Cross-species comparisons. We then calculated Spearman correlations between the maps predicted by each model: (1) against the maps predicted by the model trained on a specific organism, and (2) against the actual Hi-C maps of the corresponding organisms.

Building the tree. The resulting two correlation matrices were symmetrized by averaging with their transposes, and diagonal elements were set to one. The matrices were then averaged and subtracted from 1. The final matrix was used to construct a tree via hierarchical clustering with average linkage.

References

1. Fudenberg, G. *et al.* Formation of Chromosomal Domains by Loop Extrusion. *Cell Rep.* **15**, 2038–2049 (2016).
2. Gschwind, A. R. *et al.* An encyclopedia of enhancer-gene regulatory interactions in the human genome. *bioRxivorg* (2023) doi:10.1101/2023.11.09.563812.
3. Rinzema, N. J. *et al.* Building regulatory landscapes reveals that an enhancer can recruit cohesin to create contact domains, engage CTCF sites and activate distant genes. *Nat. Struct. Mol. Biol.* **29**, 563–574 (2022).
4. Kane, L. *et al.* Cohesin is required for long-range enhancer action at the Shh locus. *Nat. Struct. Mol. Biol.* **29**, 891–897 (2022).
5. Adams, N. M. *et al.* Cohesin-mediated chromatin remodeling controls the differentiation and function of conventional dendritic cells. *bioRxiv* (2024) doi:10.1101/2024.09.18.613709.
6. Dixon, J. R. *et al.* Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature* **485**, 376–380 (2012).
7. Rao, S. S. P. *et al.* A 3D map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell* **159**, 1665–1680 (2014).
8. Vian, L. *et al.* The Energetics and Physiological Impact of Cohesin Extrusion. *Cell* **173**, 1165–1178.e20 (2018).
9. Galitsyna, A. *et al.* Extrusion fountains are hallmarks of chromosome organization emerging upon zygotic genome activation. *bioRxiv* (2023) doi:10.1101/2023.07.15.549120.
10. Guo, Y. *et al.* Chromatin jets define the properties of cohesin-driven in vivo loop extrusion. *Mol. Cell* **82**, 3769–3780.e5 (2022).
11. Lupiáñez, D. G. *et al.* Disruptions of topological chromatin domains cause pathogenic rewiring of

gene-enhancer interactions. *Cell* **161**, 1012–1025 (2015).

12. Solovei, I. & Mirny, L. Spandrels of the cell nucleus. *Curr. Opin. Cell Biol.* **90**, 102421 (2024).
13. Sexton, T. *et al.* Three-dimensional folding and functional organization principles of the Drosophila genome. *Cell* **148**, 458–472 (2012).
14. Hou, C., Li, L., Qin, Z. S. & Corces, V. G. Gene density, transcription, and insulators contribute to the partition of the Drosophila genome into physical domains. *Mol. Cell* **48**, 471–484 (2012).
15. Feng, S. *et al.* Genome-wide Hi-C analyses in wild-type and mutants reveal high-resolution chromatin interactions in Arabidopsis. *Mol. Cell* **55**, 694–707 (2014).
16. Szabo, Q., Bantignies, F. & Cavalli, G. Principles of genome folding into topologically associating domains. *Sci Adv* **5**, eaaw1668 (2019).
17. Rowley, M. J. *et al.* Analysis of Hi-C data using SIP effectively identifies loops in organisms from C. elegans to mammals. *Genome Res.* **30**, 447–458 (2020).
18. Isiaka, B. N. *et al.* Cohesin forms fountains at active enhancers in C. elegans. *Genomics* (2023).
19. Kim, J., Wang, H. & Ercan, S. Cohesin organizes 3D DNA contacts surrounding active enhancers in C. elegans. *bioRxivorg* (2024) doi:10.1101/2023.09.18.558239.
20. Liu, N. Q. *et al.* Rapid depletion of CTCF and cohesin proteins reveals dynamic features of chromosome architecture. *Molecular Biology* (2021).
21. Flyamer, I. M. *et al.* Single-nucleus Hi-C reveals unique chromatin reorganization at oocyte-to-zygote transition. *Nature* **544**, 110–114 (2017).
22. Ulianov, S. V. *et al.* Order and stochasticity in the folding of individual Drosophila genomes. *Nat. Commun.* **12**, 41 (2021).
23. Wang, Q., Sun, Q., Czajkowsky, D. M. & Shao, Z. Sub-kb Hi-C in D. melanogaster reveals conserved characteristics of TADs between insect and mammalian cells. *Nat. Commun.* **9**, 188 (2018).
24. Ulianov, S. V. *et al.* Active chromatin and transcription play a key role in chromosome partitioning

into topologically associating domains. *Genome Res.* **26**, 70–84 (2016).

25. Bing, X. *et al.* Chromosome structure in Drosophila is determined by boundary pairing not loop extrusion. *Elife* **13**, (2024).
26. Matthews, N. E. & White, R. Chromatin Architecture in the Fly: Living without CTCF/Cohesin Loop Extrusion?: Alternating Chromatin States Provide a Basis for Domain Architecture in Drosophila. *Bioessays* **41**, e1900048 (2019).
27. Ramírez, F. *et al.* High-resolution TADs reveal DNA sequences underlying genome organization in flies. *Nat. Commun.* **9**, 189 (2018).
28. Nègre, N. *et al.* A comprehensive map of insulator elements for the Drosophila genome. *PLoS Genet.* **6**, e1000814 (2010).
29. Brandão, H. B. *et al.* RNA polymerases as moving barriers to condensin loop extrusion. *Proc. Natl. Acad. Sci. U. S. A.* **116**, 20489–20499 (2019).
30. Zhegalova, I. V., Ulianov, S. V., Galitsyna, A. A. & Pletenev, I. A. Convergent gene pairs restrict chromatin looping in Dictyostelium discoideum, acting as directional barriers for extrusion. *bioRxiv* (2024).
31. Chapard, C. *et al.* Transcription promotes discrete long-range chromatin loops besides organizing cohesin-mediated DNA folding. *bioRxiv* 2023.12.29.573667 (2023) doi:10.1101/2023.12.29.573667.
32. Vietri Rudan, M. *et al.* Comparative Hi-C reveals that CTCF underlies evolution of chromosomal domain architecture. *Cell Rep.* **10**, 1297–1309 (2015).
33. Gómez-Marín, C. *et al.* Evolutionary comparison reveals that diverging CTCF sites are signatures of ancestral topological associating domains borders. *Proc. Natl. Acad. Sci. U. S. A.* **112**, 7542–7547 (2015).
34. Krefting, J., Andrade-Navarro, M. A. & Ibn-Salem, J. Evolutionary stability of topologically

associating domains is associated with conserved gene regulation. *BMC Biol.* **16**, 87 (2018).

35. Kovina, A. P. *et al.* Evolution of the genome 3D organization: Comparison of fused and segregated globin gene clusters. *Mol. Biol. Evol.* **34**, 1492–1504 (2017).
36. Hoencamp, C. *et al.* 3D genomics across the tree of life reveals condensin II as a determinant of architecture type. *Science* **372**, 984–989 (2021).
37. Samborskaia, M. D. *et al.* Cumulative contact frequency of a chromatin region is an intrinsic property linked to its function. *PeerJ* **8**, e9566 (2020).
38. Lukyanchikova, V. *et al.* Anopheles mosquitoes reveal new principles of 3D genome organization in insects. *Nat. Commun.* **13**, 1960 (2022).
39. Renschler, G. *et al.* Hi-C guided assemblies reveal conserved regulatory topologies on X and autosomes despite extensive genome shuffling. *Genes Dev.* **33**, 1591–1612 (2019).
40. Lonfat, N. & Duboule, D. Structure, function and evolution of topologically associating domains (TADs) at HOX loci. *FEBS Lett.* **589**, 2869–2876 (2015).
41. Fudenberg, G., Kelley, D. R. & Pollard, K. S. Predicting 3D genome folding from DNA sequence with Akita. *Nat. Methods* **17**, 1111–1117 (2020).
42. Zhou, J. Sequence-based modeling of three-dimensional genome architecture from kilobase to chromosome scale. *Nat. Genet.* **54**, 725–734 (2022).
43. Schwessinger, R. *et al.* DeepC: predicting 3D genome folding using megabase-scale transfer learning. *Nat. Methods* **17**, 1118–1124 (2020).
44. Hinton, G. E. & Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks. *Science* **313**, 504–507 (2006).
45. Mikolov, T. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).
46. Ji, Y., Zhou, Z., Liu, H. & Davuluri, R. V. DNABERT: pre-trained Bidirectional Encoder

Representations from Transformers model for DNA-language in genome. *Bioinformatics* (2021).

47. Imakaev, M. *et al.* Iterative correction of Hi-C data reveals hallmarks of chromosome organization. *Nat. Methods* **9**, 999–1003 (2012).
48. Lieberman-Aiden, E. *et al.* Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science* **326**, 289–293 (2009).
49. Polovnikov, K. E. *et al.* Crumpled polymer with loops recapitulates key features of chromosome organization. *Phys. Rev. X.* **13**, (2023).
50. Open2C *et al.* Cooltools: enabling high-resolution Hi-C analysis in Python. *bioRxiv* 2022.10.31.514564 (2022) doi:10.1101/2022.10.31.514564.
51. Zhang, H. *et al.* Chromatin structure dynamics during the mitosis-to-G1 phase transition. *Nature* **576**, 158–162 (2019).
52. Jurafsky, D., James, H. & Martin, N. J. Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition. *Upper Saddle River* (2000).
53. Linder, J., Srivastava, D., Yuan, H., Agarwal, V. & Kelley, D. R. Predicting RNA-seq coverage from DNA sequence as a unifying model of gene regulation. *bioRxiv* 2023.08.30.555582 (2023) doi:10.1101/2023.08.30.555582.
54. Krietenstein, N. *et al.* Ultrastructural Details of Mammalian Chromosome Architecture. *Mol. Cell* **78**, 554–565.e7 (2020).
55. Deaton, A. M. & Bird, A. CpG islands and the regulation of transcription. *Genes Dev.* **25**, 1010–1022 (2011).
56. Fenouil, R. *et al.* CpG islands and GC content dictate nucleosome depletion in a transcription-independent manner at mammalian promoters. *Genome Res.* **22**, 2399–2408 (2012).
57. Hsieh, T.-H. S. *et al.* Enhancer-promoter interactions and transcription are maintained upon acute

loss of CTCF, cohesin, WAPL, and YY1. *bioRxiv* 2021.07.14.452365 (2021) doi:10.1101/2021.07.14.452365.

58. Schreiber, J., Nair, S., Balsubramani, A. & Kundaje, A. Accelerating in silico saturation mutagenesis using compressed sensing. *Bioinformatics* **38**, 3557–3564 (2022).

59. Nair, S., Shrikumar, A., Schreiber, J. & Kundaje, A. fastISM: performant in silico saturation mutagenesis for convolutional neural networks. *Bioinformatics* **38**, 2397–2403 (2022).

60. Bailey, T. L. & Elkan, C. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **2**, 28–36 (1994).

61. Corbo, M., Damas, J., Bursell, M. G. & Lewin, H. A. Conservation of chromatin conformation in carnivores. *Proc. Natl. Acad. Sci. U. S. A.* **119**, e2120555119 (2022).

62. Sandoval-Velasco, M. *et al.* Three-dimensional genome architecture persists in a 52,000-year-old woolly mammoth skin sample. *Cell* **187**, 3541–3562.e51 (2024).

63. Yang, Y., Zhang, Y., Ren, B., Dixon, J. R. & Ma, J. Comparing 3D genome organization in multiple species using Phylo-HMRF. *Cell Syst.* **8**, 494–505.e14 (2019).

64. Lazar, N. H. *et al.* Epigenetic maintenance of topological domains in the highly rearranged gibbon genome. *Genome Res.* **28**, 983–997 (2018).

65. Kaaij, L. J. T., van der Weide, R. H., Ketting, R. F. & de Wit, E. Systemic loss and gain of chromatin architecture throughout zebrafish development. *Cell Rep.* **24**, 1–10.e4 (2018).

66. Rowley, M. J. *et al.* Evolutionarily conserved principles predict 3D chromatin organization. *Mol. Cell* **67**, 837–852.e7 (2017).

67. Sexton, T. & Cavalli, G. The role of chromosome domains in shaping the functional genome. *Cell* **160**, 1049–1059 (2015).

68. Grand, G., Blank, I. A., Pereira, F. & Fedorenko, E. Semantic projection recovers rich human knowledge of multiple object features from word embeddings. *Nat. Hum. Behav.* **6**, 975–987

(2022).

69. Franke, M. *et al.* CTCF knockout in zebrafish induces alterations in regulatory landscapes and developmental gene expression. *Nat. Commun.* **12**, 5415 (2021).

70. Guérin, T. M., Barrington, C., Pobegalov, G., Molodtsov, M. I. & Uhlmann, F. An extrinsic motor directs chromatin loop formation by cohesin. *EMBO J.* (2024) doi:10.1038/s44318-024-00202-5.

71. Banigan, E. J. *et al.* Transcription shapes 3D chromatin organization by interacting with loop extrusion. *Proc. Natl. Acad. Sci. U. S. A.* **120**, e2210480120 (2023).

72. Yi, S. V. Epigenetics research in evolutionary biology: Perspectives on timescales and mechanisms. *Mol. Biol. Evol.* **41**, (2024).

73. Heger, P., Marin, B. & Schierenberg, E. Loss of the insulator protein CTCF during nematode evolution. *BMC Mol. Biol.* **10**, 84 (2009).

74. Crane, E. *et al.* Condensin-driven remodelling of X chromosome topology during dosage compensation. *Nature* **523**, 240–244 (2015).

75. Wike, C. L. *et al.* Chromatin architecture transitions from zebrafish sperm through early embryogenesis. *Genome Res.* **31**, 981–994 (2021).

76. Li, R., Liu, Y., Li, T. & Li, C. 3Disease Browser: A Web server for integrating 3D genome and disease-associated chromosome rearrangement data. *Sci Rep* **6**, 34651 (2016).

77. Niu, L. *et al.* Three-dimensional folding dynamics of the Xenopus tropicalis genome. *Nat. Genet.* **53**, 1075–1087 (2021).

78. Wang, Z.-L. *et al.* A Chromosome-Scale Assembly of the Asian Honeybee Apis cerana Genome. *Front. Genet.* **11**, 279 (2020).

79. Batut, P. J. *et al.* Genome organization controls transcriptional dynamics during development. *Science* **375**, 566–570 (2022).

80. Contessoto, V. G. *et al.* Interphase chromosomes of the Aedes aegypti mosquito are liquid

crystalline and can sense mechanical cues. *Biophys. J.* **122**, 20a (2023).

81. Ryazansky, S. S. *et al.* The chromosome-scale genome assembly for the West Nile vector Culex quinquefasciatus uncovers patterns of genome evolution in mosquitoes. *BMC Biol.* **22**, 16 (2024).
82. Yang, G.-Y. *et al.* Comparative genomics of Sarcoptes scabiei provides new insights into adaptation to permanent parasitism and within-host species divergence. (2022) doi:10.22541/au.165052994.44357848/v1.
83. Brückner, A., Barnett, A. A., Bhat, P., Antoshechkin, I. A. & Kitchen, S. A. Molecular evolutionary trends and biosynthesis pathways in the Oribatida revealed by the genome of Archegozetes longisetosus. *Acarologia* **62**, 532–573 (2022).
84. Anderson, E. C. *et al.* X chromosome domain architecture regulates Caenorhabditis elegans lifespan but not dosage compensation. *Dev. Cell* **51**, 192–207.e6 (2019).
85. Chen, Z., Doğan, Ö., Guiglielmoni, N., Guichard, A. & Schrödl, M. Pulmonate slug evolution is reflected in the de novo genome of Arion vulgaris Moquin-Tandon, 1855. *Sci. Rep.* **12**, 14226 (2022).
86. Liu, C. *et al.* The genome of the golden apple snail Pomacea canaliculata provides insight into stress tolerance and invasive adaptation. *Gigascience* **7**, (2018).
87. Costantino, L., Hsieh, T.-H. S., Lamothe, R., Darzacq, X. & Koshland, D. Cohesin residency determines chromatin loop patterns. *bioRxiv* (2020) doi:10.1101/2020.06.11.146902.
88. Zhong, Z. *et al.* DNA methylation-linked chromatin accessibility affects genomic architecture in Arabidopsis. *Proc. Natl. Acad. Sci. U. S. A.* **118**, e2023347118 (2021).
89. Yang, H. *et al.* A map of cis-regulatory elements and 3D genome structures in zebrafish. *Nature* **588**, 337–343 (2020).
90. Harrison, P. W. *et al.* Ensembl 2024. *Nucleic Acids Res.* **52**, D891–D899 (2024).
91. Fey, P., Dodson, R. J., Basu, S. & Chisholm, R. L. One stop shop for everything Dictyostelium:

dictyBase and the Dicty Stock Center in 2012. *Methods Mol. Biol.* **983**, 59–92 (2013).

92. Abdennur, N. & Mirny, L. A. Cooler: scalable storage for Hi-C data and other genomically labeled arrays. *Bioinformatics* **36**, 311–316 (2020).
93. Virtanen, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272 (2020).
94. Paszke, A. *et al.* Automatic differentiation in PyTorch. (2017).
95. He, K., Zhang, X., Ren, S. & Sun, J. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *arXiv [cs.CV]* (2015).
96. Kingma, D. P. & Welling, M. Auto-Encoding Variational Bayes. *arXiv [stat.ML]* (2013).
97. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv [cs.LG]* (2014).
98. Satopaa, V., Albrecht, J., Irwin, D. & Raghavan, B. Finding a 'kneedle' in a haystack: Detecting knee points in system behavior. in *2011 31st International Conference on Distributed Computing Systems Workshops* (IEEE, 2011). doi:10.1109/icdcsw.2011.20.
99. Forcato, M. *et al.* Comparison of computational methods for Hi-C data analysis. *Nat. Methods* **14**, 679–685 (2017).
100. Sundararajan, M., Taly, A. & Yan, Q. Axiomatic attribution for deep networks. *arXiv [cs.LG]* (2017) doi:10.48550/ARXIV.1703.01365.
101. Rauluseviciute, I. *et al.* JASPAR 2024: 20th anniversary of the open-access database of transcription factor binding profiles. *Nucleic Acids Res.* **52**, D174–D182 (2024).
102. Bailey, T. L. & Elkan, C. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Mach. Learn.* **21**, 51–80 (1995).
103. Gupta, S., Stamatoyannopoulos, J. A., Bailey, T. L. & Noble, W. S. Quantifying similarity between motifs. *Genome Biol* **8**, R24 (2007).
104. Prickett, A. R. *et al.* Genome-wide and parental allele-specific analysis of CTCF and cohesin DNA

binding in mouse brain reveals a tissue-specific binding pattern and an association with imprinted differentially methylated regions. *Genome Res* **23**, 1624–1635 (2013).

105. O'Leary, N. A. *et al.* Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res* **44**, D733–45 (2016).