Albert-Ludwigs University Freiburg
Department of Computer Science
Bioinformatics Group

Master Thesis

Predicting Hi-C contact matrices using machine learning approaches

Author:
Ralf Krauth

Examiner:
Prof. Dr. Rolf Backofen

Second Examiner:
Prof. Dr. Ralf Gilsbach

Advisors:
Anup Kumar, Joachim Wolff

Submission date:
20.04.2021

Abstract

Harhar!

Zusammenfassung

Hohoho!

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe und alle Stellen, die wörtlich oder sinngemäß aus anderen Werken entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass die eingereichte Masterarbeit weder vollständig, noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens war oder ist.

Bad Krozingen, den 25. Januar 2021

Ralf Krauth

Contents

1 Introduction

In recent years, the three-dimensional organization of DNA has been shown to be a key factor for important processes in molecular biology. However, even with the most advanced experimental methods, it remains comparatively expensive to determine the spatial folding of DNA directly, so that current knowledge of three-dimensional DNA organization is still sketchy. In the last five years, several methods have thus been proposed to improve on this situation by determining DNA-DNA interactions *in-silico* All of these are using existing experimental data which is easier to obtain than spatial data, but correlates with spatial structure in certain ways. However, most current in-silico approaches have disadvantages and shortcomings, and the current thesis attempts to improve on these.

1.1 Spatial structure of DNA

In the late 1970s, Watson and Crick XXXdiscovered the now well-known double helix structure of DNA molecules. However, this is not the only relevant spatial structure of genomes. At larger scales, DNA molecules can be wound around certain proteins, so-called histones, forming DNA-protein complexes named nucleosomes. Several of these can further be compacted into fibres, which in turn can be "supercoiled" into the also well-known chromosomes. XXXhier muss das bild hin XXX

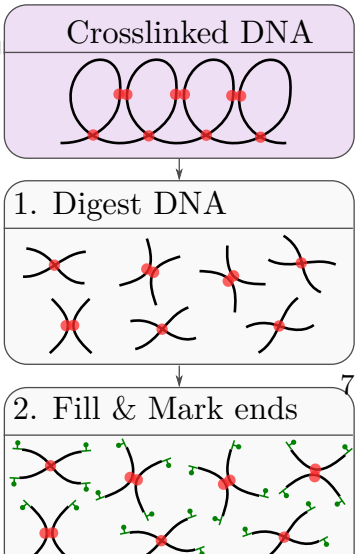
While the spatial structure of chromatin outlined above brings along compaction (fig. XXXfrom left to right), e. g. to make chromosomes fit into eucaryotic cell nuclei, it also has other functional implications. One well known effect of spatial structure is the regulation of gene transcription by establishing or releasing contacts between gene enhancers and the relevant promoters [1, 2]. Since enhancers can be up to a million basepairs away from the promoters, the two can only interact by means of spatial DNA structure. Many effects of spatial structure are still under investigation, two recent studies have for example found dependencies between chromatin conformation and cell differentiation in Drosophila Melanogaster [3] and investigated spatial structure dynamics during phase transitions in murine cells [4].

While the driving mechanisms behind the formation of spatial chromatin structures are partially still under research, certain proteins like CTCF or modified histones are already well known to mediate or prevent DNA-DNA contacts [5, 6, 7].

In the last two decades, DNA-sequencing-based techniques have increasingly been utilized to capture the spatial structure of DNA experimentally. The method of choice within this thesis is called Hi-C and shall be explained in the following section.

1.2 The Hi-C process for determining spatial DNA structu

The Hi-C process is an elaborate biochemical procedure for investigating the spatial structure of DNA by detecting DNA-DNA interactions within and across chromosomes. The original Hi-C workflow has been developed by Lieberman-Aiden et al. in 2009 [8] and is depicted in simplified form in fig. 1.



The typical input (In) to Hi-C consists of several millions of cells, which are treated chemically to fix existing DNA-DNA contacts, commoly using formaldehyde, before they are lysed. Next, the DNA is extracted and cut into fragments by certain restriction enzymes (1), usually HindIII or DpnII, and the cut ends are repaired with nucleotides, some of which are marked by biotin (2). The free ends are then joined (3) under conditions which prefer ligations among open ends over ligations between different fragments. Originally, such conditions were achieved by high dilution of the fragments in solvents, but especially this part of the protocol has been replaced by more efficient methods in later works [9, 10]. The ligated fragments are then purified and cut into shorter sequences, some of which contain biotinylated nucleotides and some not (4). The fragments of interest – the ones containing biotinylated nucleotides – are then selected by pulling down biotin (5), for example using magnetic tags, and subjected to paired-end DNA-sequencing (6). In the end, the output of the Hi-C lab process is a large number of short genomic "reads", which are subsequently processed in the bioinformatics part of the Hi-C protocol outlined in the following section.

On the software side of the protocol, the reads first need to be mapped to the corresponding reference genome. Here, only reads are kept where the "left" sequence (6)(a) uniquely maps to a different region of the reference genome than the "right" sequence (6)(b). These so-called chimeric reads are subjected to quality control, and those passing are counted as an interaction between the two genomic positions (1)(a) and (1)(b) to which the two ends belong. However, at reasonable read coverages, interactions cannot be counted per base pair. Instead, the reference genome is split into equally sized bins (or regions), and the reads are counted for those regions where they belong. The final outcome of a Hi-C experiment is then a sparse matrix, henceforth referred to as "Hi-C matrix", which records the interaction count for all possible pairs of regions in the reference genome.

For the thesis at hand, Hi-C matrices can abstractly be considered as sets of sparse, square matrices; one matrix for each chromosome. The individual elements $m_{i,j}$ of the matrices are counts of interactions between the bins with indices $i, j \in \{0, 1, ..., l\}$, where the last index l is given by the chromosome size cs and the binsize b in basepairs as $l = \lceil \frac{cs}{b} \rceil - 1$. Because interaction counts do not have a "direction", it holds that $m_{i,j} = m_{j,i}$, so Hi-C matrices are always symmetric.

A bin with index n then corresponds to the genomic region $p_n = [\, n \cdot b,\ (n+1) \cdot b\,)$, except the last bin l, where $p_l = [\, l \cdot b,\ cs\,]$, with l, b, cs as above. As an example, if $m_{0,10} = 22$ for a matrix with binsize $b = 25\,\text{kbp}$, this means that in the underlying experiment, 22 interactions have been counted between region p_0, ranging from 0 to 24 999 bp and region p_{10}, ranging from 250000 to 274 999 bp. Due to insufficient read coverage, $b = 1\,\text{bp}$ is impossible in practice. Instead, common values include $b \in \{1, 5, 10, 25, 50, 100, 1000\}\,\text{kbp}$.

In the bioinformatics part of the Hi-C protocol, often just a small fraction of all reads fulfill the selection criteria outlined above, for example due to reads not being chimeric or uniquely mappable. This makes Hi-C a comparatively inefficient, slow and thus expensive process. For

example, the well-known dataset by Rao et al. [9] with matrix bin sizes down to 1 kbp, required several *billions* of reads being made.

Parts of fig. 1 and the process description above have been adapted from the preceding study project [11].

XXXmod input steps in graph XXXmark regions a, b throughout the graph

1.3 The ChIP-seq process for determining protein binding sites

The ChIP-seq process is a combination of Chromatin-Immunoprecipitation (ChIP) and DNA-sequencing, designed for investigating DNA-protein interactions [12, 13]. As with Hi-C, the input consists of a sufficient number of cells, which are first fixed with formaldehyde. The cells are then lysed, the DNA-protein structure is extracted and cut into fragments, usually by sonication (2). Next, specific antibodies are added, designed to bind only to a certain protein of interest (3). These antibodies are additionally equipped with a tag, for example a magnetic one, so that the DNA-protein-antibody structures can be precipitated, while fragments without antibodies are discarded (4). The proteins and antibodies are then removed (5), the DNA is purified and finally sequenced (6). Typically, a control experiment is performed together with the ChIP-seq process, which comprises all steps described above, except the immunoprecipitation (3),(4).

The outcome of the ChIP-seq lab process is again a bunch of short genomic sequences, which are then fed into the bioinformatics part of the pipeline.

On the software side of the process, the reads are filtered for quality and mapped to an appropriate reference genome. The number of mapped reads per genomic position can then be simply be counted. It is common to process reads from the control experiment in the same way as reads from the ChIP-seq experiment and then use special software to call "peaks", i. e. protein binding sites, at those genomic positions where the read count from ChIP-seq is statistically significant compared to the read count from the control group.

For the thesis at hand, the reads from the ChIP-seq experiment have been used directly and binned by taking the mean over all reads inside each bin. Contrary to Hi-C matrices, choosing a binsize of one is in principle possible for ChIP-seq experiments, but yields very sparse vectors. Thus, binsizes that evenly divide the binsizes of the corresponding matrices have been chosen for this thesis, namely 5 kbp and 25 kbp.

1.4 Motivation and goal of the thesis

Due to the high effort, Hi-C experiments have been conducted only by few labs and for comparatively few organisms and cell lines so far. For example, as of January 2021, Hi-C assays are available for just 26 of more than 150 human cell lines listed in the Encyclopedia of DNA elements [14, 15], and these have been produced by only five university labs.

However, investigations and experiments with the available Hi-C data have shown correlations between the spatial DNA

cell line	TF ChIP-seq	Histone ChIP-seq	Dnase-seq	Hi-C
K562	680	20	10	1
HepG2	668	15	3	1
A549	251	87	14	6
HEK293	231	6		
GM12878	211	15	3	7
MCF-7	155	18	8	
H1	88	55	3	
HeLa-S3	77	15	4	1
SK-N-SH	62	21	2	
IMR-90	16	34	2	2
HCT116	27	17	1	8
H9		35	7	

structure and the binding sites of certain transcription factors and histone modifications [9, 16]. Since these binding sites can be detected with the less costly ChIP-seq method, a computational method able to predict contact matrices from DNA-protein bindings would be very helpful.

The goal of this master thesis is thus to predict missing Hi-C data from existing ChIP-seq data, making use of the known correlations between the binding sites of transcription factors and histone modifications of the one hand and Hi-C interaction counts on the other hand. Because this is a wide field and lead time for a master thesis is limited, the present work will focus on machine learning techniques, which have proven a good choice for exploiting complex patterns and relationships in various kinds of data. XXXcitations needed More specifically, the goal of this thesis is to develop an easy-to-use machine learning approach for predicting Hi-C interaction matrices from ChIP-seq data, using standard in- and output formats with minimal pre- and postprocessing.

To underscore the usefulness of such an approach, table 1 to the side shows the first 16 human cell lines in ENCODE, sorted by total number of selected assays available. It is obvious that even otherwise well-investigated cell lines like HEK293, MCF-7 and H1 lack Hi-C data, while ChIP-seq data is mostly available. XXXdata for H1 is available in

2 Related work

In the last five years, several approaches have been presented to determine DNA-DNA interactions *in silico*, using existing data from various experiments. Section 2.1 gives an overview about these methods. Furthermore, some methods originally developed for image synthesis and similar tasks in computer vision might also be useful in the field of Hi-C matrix generation and are thus summarized in subsection 2.2. The section is then concluded by a short discussion of the existing work.

2.1 Methods for predicting DNA-DNA interactions and contact matrices

As of 2020, there is quite a body of existing work in the field of predicting DNA-DNA interactions, using various approaches and different types of input data.

Two conceptually similar methods have been proposed by Brackley et al. in 2016 and MacPherson et al. in 2018 [17, 18]. In both approaches, DNA is modeled as a "beads-on-a-string" polymer, and simulation techniques are employed to find energy-optimal spatial structures of these polymers. Apart from constraints derived from the molecule's DNA sequence itself, the models also consider spatial contact constraints derived from ChIP-seq experiments of chromatin factors which are known to mediate such DNA-DNA contacts. The interaction matrices derived from the simulations look interesting, but the paper from Brackley et al. [17] is unfortunately lacking a comparison with "true" experimentally measured Hi-C matrices, and the results from MacPherson et al. [18] seem inferior to most other ones presented in this section.

Another simulation-based method has been developed by di Pierro et al. in 2017 [19] and later extended by Qi and Zhang [20]. In both cases, a convolutional neural network (CNN) is trained to learn different "open" and "closed" chromatin states from 11 chromatin factors, and the predicted chromatin states are then taken as constraints for beads-on-a-string models. The difference between [19] and [20] lies mainly in the number of states considered and the simulation methods applied; the results are mathematically convincing in both cases.

A further approach using chromatin states is due to Farrè and Emberly [21]. Here, the conditional probability of two genomic regions being in contact, given their distance and the chromatin state around them, is estimated using Bayes' rule. In this case, the chromatin state – reduced to active or inactive – is derived from DNA adenine methyltransferase identification (DamID) signals of 53 chromatin factors using probabilistic methods [22]. The conditional probabilities on the right side of Bayes' rule are either computed from training data or estimated with different probabilistic approaches, too. While the predicted contact matrices do not look like real Hi-C matrices with this approach, highly interacting regions are still often well identifiable.

Three further approaches in the field make use of random forests. 3DEpiLoop by Bkhetan and Plewczynski and Lollipop by Kai et al., both from 2018, use a random forest classifier to predict DNA loops, but differ in input data and preprocessing [23, 24]. While 3DEpiLoop is using only ChIP-seq data of histone modifications and transcription factors [23], Lollipop additionally takes ChIA-PET-, RNA-seq- and DNase-seq-data, CTCF motif orientation and loop length as inputs [24]. Both approaches show good coincidence of predicted loops with experimental data, but their output is binary and rather sparse. Contrary to these two, the third random-forest-based approach, HiC-Reg by Zhang et al. from 2019, allows predicting real-valued Hi-C interaction

matrices directly [25]. To this end, it employs random forest regression to predict interactions between two genomic "windows", using ChIP-seq data of 13 chromatin factors and the genomic distance of the two windows. The published results for five human cell lines look interesting.

Another recent method which investigated decision-tree-based algorithms is due to Martens et al. (2020) [26]. Here, gradient boosted decision trees, logistic regression and neural networks were used to predict highly interacting chromatin regions and TAD domain boundaries from histone modifications and CTCF ChIP-seq data. However, in this setup, the neural network approach yielded the best, overall acceptable results of the three approaches, but again in form of a binary classifications. A neural-network approach with comparable input data, but without the restriction to binary classifications has been presented by Farrè et al. in 2018 [27]. Here, a one-dimensional convolutional filter is used to convert ChIP-seq data from 50 chromatin factors into a one-dimensional chromatin vector, which is then processed by a dense neural network (DNN). This allows predicting real-valued Hi-C interaction matrices, which resemble the general structure of experimentally derived matrices quite well.

While the approaches discussed so far have either modeled DNA as a "beads-on-a-string" polymer or not used it explicitly at all, there are also several machine-learning approaches which directly consider DNA sequence without a need for polymer modeling. In 2019, Singh et al. presented SPEID [28], an approach to predict promoter-enhancer interactions from DNA sequence, using a combination of CNNs, a recurrent network (LSTM) and a DNN. The results match well with experimental data, but are limited to promoter and enhancer loci by design, disallowing predictions of complete Hi-C contact matrices. Other researchers have tried to design similar methods without such limitations. The work by Peng from 2017 [29] is an extension of SPEID, based on a 2016 preprint [30], additionally taking into account a "middle sequence" between enhancer- and promoter sequences, CTCF motif counts within the sequences and genomic distance between two sequence snippets. However, the network lacks generalization, i. e. the results are only good in training regions [29, figs. 4, 5]. A conceptually similar method to the one by Peng [29], but with a different neural network design has been presented by Schreiber et al., also in 2017, named Rambutan [31]. It accepts DNA-sequence, DNase-seq data and distance between two genomic loci as inputs and then uses a combination of CNNs and a DNN to predict whether the given two loci interact or not. Unfortunately, it is difficult to decide whether the results of Schreiber et al. are useful for the task at hand, since the evaluation is done only by statistical means and no actual Hi-C matrices have been published. The original paper [31] also contains a known error and seems not to have appeared in a peer-reviewed journal in improved form yet. A probably more promising method working on DNA sequence, Akita, has been published by Fudenberg et al. in 2020 [32]. It is based on two rather involved convolutional neural networks. While the first one, "trunk", processes one-dimensional, one-hot encoded DNA sequence input through convolutional filters, the second one, "head", converts one-dimensional representations to 2D, further processes the data with convolutional filters and enforces symmetry. Although Fudenberg et al. initally seemed to focus on determining the influence of DNA modifications on spatial structure [33], predicting complete Hi-C matrices is an integral part of their work, and a large number of images of Hi-C matrices from the test set has been published alongside the article. The predicted matrices often hardly look like experimental Hi-C matrices, but mostly still indicate highly interacting regions quite well.

A further method by Schwessinger et al. [34] also makes use of DNA sequence and additional epigenetic data for its predictions, but is conceptually different from the ones presented so far.

Here, ChIP-seq tracks are initially used to train a CNN on the relationship between sequence and the corresponding chromatin factors. The weights of this first network are then used to seed another convolutional neural network, which is responsible for predicting DNA-DNA interactions from DNA sequence. In this case, the results were blurry, but generally in good accordance with experimental Hi-C data.

Yet another machine-learning concept based on sequence data, Samarth, has been proposed by Nikumbh and Pfeifer in 2017 [35]. Here, a support vector machine is trained on 5C data, using a specific representation for DNA sequence called oligomer distance histograms. The results showed acceptable consensus with experimental Hi-C data and allowed some interesting conclusions about the importance of certain k-mers for DNA folding. However, the applicability for the task at hand is hard to assess, because none of the predicted matrices used for statistical evaluation has been published alongside the paper.

2.2 Image synthesis techniques from computer vision

With the advent of deep learning, both the number of opportunities and the demand for using machine learning techniques in image synthesis tasks have risen, as recently summarized by Tsirikoglou et al. [36]. Since Hi-C contact matrices can be seen as are square grayscale images, such techniques can also be relevant for this thesis.

Probably one of the first applications of computer vision methods for Hi-C matrices was presented by Liu and Wang in 2019 [37]. Here, deep convolutional neural networks – modified from established image super-resolution networks XXX– have been used to enhance low-resolution Hi-C matrices.

Another technique from computer vision that has been transferred to Hi-C matrices are conditional generative adversarial networks (GANs), invented by Goodfellow, Mirza and colleagues in 2014 [38, 39]. Again, several works have employed this comparatively new and involved method to increase the resolution of given Hi-C matrices, including the ones by Liu and colleagues [40], Hong et al. [41] and Dimmick et al. [42]. In general, all three works feature the typical GAN setup with two competitive networks – a generator, which is trained to produce realistic high-resolution Hi-C matrices from its low-resolution inputs, and a discriminator, which tries to distinguish real Hi-C matrices from generated ones and thus serves as a "critique", an additional loss function, for the generator. While the convolutional building blocks for the discriminators and the residual building blocks for the generators are conceptually similar in all cases, the three works differ in the number of building blocks used and the activation functions applied within the blocks. Furthermore, Dimmick et al. and Hong et al. include additional loss functions beyond standard generator- and adversarial losses. This includes perceptual losses derived from other (pre-trained) neural networks to obtain "visually good" Hi-C matrices and total variation loss to suppress noise while maintaining edges. The method by Dimmick et al. outperformed the others for most test cases, but it is also the most elaborated.

2.3 Discussion of existing work

Independent of chosen techniques, several of the methods shown above only allow predictions restricted to certain loci (e. g. promoters and enhancers) or yield binary predictions in the sense

of “interaction” or “no interaction” between certain loci [23, 24, 26, 28]. These methods are thus not directly suitable for the task at hand, but would require further develoment.

The chromatin-modeling based approaches [17, 18, 19, 20] allow indirect derivation of Hi-C matrices by performing sufficient simulation passes and estimating contacts from the resulting model ensembles. Depending on the chosen chromatin model – which seems not straightforward [43, 44] – and the number of constraints involved, the required computations can be expensive. However, the results still seem slightly inferior compared to other methods mentioned in section 2.1, maybe because not all constraints for chromatin modeling are known or cannot be considered in the models yet.

Three of the DNA-sequence based methods mentioned above also allow direct prediction of Hi-C matrices [31, 32, 34]. At first look, it is surprising that learning spatial structure from DNA-sequence actually works, because there seems no obvious correlation between sequence and structure – instead, 3D conformation can vary considerably for different cell lines of the same organism, which all share the same DNA. On the other hand, the chosen artificial networks might be able to figure out binding sites of relevant proteins from DNA sequence, which *do* have a correlation with spatial structure. This is especially true for the method by Schwessinger [34], where the network is seeded by training on exactly such binding sites. While the methods by Schreiber and Schwessinger use secondary inputs and therefore can – at least partially – adapt to cell lines sharing the same DNA, the method by Fudenberg focuses on DNA and is thus expected to produce the same outputs for all cell lines of a given organism. All three methods require comparatively deep networks which are expensive to train. The sequence-based method by Nikumbh et al. [35] is using a support vector machine, which is generally easier to train, but the adaptability to different cell lines is expected to remain problematic due to the chosen concept.

The random-forest-based method by Zhang et al. [25] is directly targeted at the goal of this thesis, since it directly predicts Hi-C matrices, is not limited to certain loci and can adapt to different cell lines using corresponding ChIP-seq data. However, this approach has extensively been investigated in two previous study projects at the university of Freiburg, and the results could not be reproduced for unknown reasons [11, 45].

The dense neural network approach by Farrè et al. [27] is also compatible with the goals of this thesis. The published results are visually and statistically convincing, and both the presented network and the training process offer opportunities for amendments, which will be explored in section 3.1.

Since all of the image synthesis methods presented above in section 2.2 require existing Hi-C matrices for training *and* prediction, none of them is directly suitable for the task at hand. However, some of the concepts can still be used to develop novel approaches, which will be discussed in section 3.2.

3 Advancing predictions of Hi-C interaction matrices

In the following subsections, two conceptually different approaches towards the goal of the thesis, predicting Hi-C matrices from ChIP-seq data, will be explored. While the first approach is a dense neural network based on work by Farré et al. [27], the second is a novel method based on conditional generative adversarial networks.

3.1 Dense Neural Network approach

In their 2008 paper [27], Pau Farré, Alexandre Heurtau, Olivier Cuvier and Eldon Emberly propose a combination of a 1D convolutional filter with a three-layer dense neural network which already fulfills most goals of this thesis with some exceptions regarding data formats and preprocessing. Here, we try to build on the success of their method by extending the comparatively simple neural network in various ways, modifying the binsizes of the Hi-C matrices and changing the learning process.

3.1.1 Basic network setup

The basic network setup taken over from Farré et al. [27] is shown in simplied form in XXX.

Since the network implements a supervised learning technique, it requires two kinds of inputs for training – ChIP-seq data of the chosen chromatin factors and target Hi-C matrices for each training chromosome. The target matrices are just taken as submatrices of size $w \times w$ with fixed windowsize w, centered at the diagonal of the original Hi-C Matrix, cf. section 4.1.1. The ChIP-seq data is taken as $3w \times n$ subarray of the original array, cf. section 4.1.2, whereby the middle w bins are aligned with the position of the submatrix, the first w bins correspond to the left flanking region and the last w bins correspond to the right flanking region of the current submatrix region. Training samples are then obtained from the input data by sliding the input windows along the diagonal of the target Hi-C matrix, XXX.

Within the network, a 1D convolutional filter compresses the $3w \times n$ inputs to a 1D vector of size $3w \times 1$, and three dense layers further process the compressed input. The number of neurons in the last dense layer corresponds to the number of bins in the upper triangular part of the target submatrix, i. e. it consists of $(w \cdot (w + 1))/2$ neurons.

Training of the network is performed by minimizing the mean squared error of the predicted matrix versus the target Hi-C matrix using stochastic gradient descent. The technical details are given in section 4.2 and the initial results for our basic setup are shown in section 5.1.1.

The network shown above is quite simple, and immediately offers some opportunities for expansion, partially already proposed in the original paper [27]. These will be explored below.

3.1.2 Modifying the convolutional part of the network

One starting point for modifying the neural network is its convolutional part.

With only a single 1D convolutional filter in one layer, there is little chance of capturing complex relationships between Hi-C interaction counts and more than one of the chromatin features. For this reason, an extended "longer" network has been created, comprising three 1D convolutional filter layers with 16, 8 and 4 filters, respectively. This is still a comparatively low number of layers and filters, but the choice seems justified in order to avoid overfitting to the low-dimensional input.

Next, a "wider" network was created, featuring the same setup as the basic network except the width of the filter kernel, which was set to 4 instead of 1. The idea here is to allow the network to capture correlations between Hi-C interaction counts and chromatin features which span more than one bin. The actual number has been kept low, since at binsize $b = 25\,\text{kbp}$, 4 bins already correspond to 100 kbp. Of course, increasing filter width and using more filters can also be combined, hopefully allowing the "wider-longer" network to capture both correlations spanning more than one bin and more than one chromatin feature.

Another approach to potentially improve the predictions that goes somewhat into the direction of the "wider" network has been proposed, but not implemented by Farré et al. in their paper [27]. As already noted in section 1.3, the ChIP-seq data can usually safely be binned at smaller binsizes than Hi-C data due to the nature of the process. This can be exploited to capture finer details in the ChIP-seq data without a need for higher (training-)matrix resolutions. To this end, the initial network can be generalized by binning the ChIP-seq data at k times the bin size of the matrices, whereby $k \in \mathbb{N}^{\geq 1}$. This yields an input data size of $k \cdot 3w \times n$, which is then again compressed to a $3w \times 1$ vector by a 1D convolutional filter with kernel size n and strides k. For practical reasons, $k = 5$ was chosen for the thesis at hand, and the results for binsizes $b_{mat} = 25\,\text{kbp}$, $b_{feat} = 5\,\text{kbp}$ are shown in XXX. The technical details are to be found in section 4.2.3. Note, for $k = 1$, one would again get the basic network described in section 3.1.1.

3.1.3 Modifying the loss function

- MSE is known to produce blurry images in regression tasks
- others have used TV loss, Perception loss to improve on this
- SSIM can also be used as to improve perception, but caution needed
- additionally, insulation-score-based approach to get sharper boundaries at highly interacting regions

3.1.4 Modifying binsize and windowsize

- rationale: smaller binsizes - predict smaller structures, larger binsizes - predict larger structures
- then combine the predictions (e.g. sum them up, take the mean or even use a NN)

- use trapezoids, i.e. capped larger submatrices and flankingsize smaller than windowsize
- rationale: larger windowsize without increasing training time too much
- train on matrices/features with different binsizes in the same training run (e.g. two matrices with 25k, 50k and bin at 25k, 50k or 5k, 10k)

3.1.5 DNA sequence as an additional network input branch

- use DNA as an additional input
- rationale a): allow the network to figure out true binding sites in conjunction with cs data
- rationale b): given the success of pure DNA based methods, allow the network to find yet unknown sequence structure correlations
- probably not the most important subsection, leave it out in case of time problems

3.2 Hi-cGAN approach

3.2.1 Pix2Pix as a generic generative network

Describe the basic setup of pix2pix and what it is currently used for Pix2Pix using 2D input, but we have n times 1D. Bad. 2 ideas how to come around, using DNN like Farre et al, potentially pretrained, or using new CNN. Maybe mention restriction of 2^x for image size

3.2.2 Using a DNN for 1D - 2D conversion

- use network like Farre et al for converting inputs to 2D
- rationale: this nw has been shown to create comparatively good Hi-C matrices
- rationale: pix2pix could then improve them
- rationale: transfer learning can be employed

3.2.3 Using a CNN for 1D - 2D conversion

- use somewhat deep CNN for conversion
- rationale: the DNN approach didn't work
- rationale: downsampling like in image processing, just in 1D

4 Method details

4.1 Input data

For the thesis at hand, data from human cell lines GM12878, K562, HMEC, HUVEC and NHEK was used. The exact data sources and data processing will be outlined in the following subsections 4.1.1 and 4.1.2

4.1.1 Hi-C matrices

Hi-C data due to Rao et al. [9] was downloaded in .hic format from Gene Expression Omnibus under accession key GSE63525. Here, the quality-filtered "combined_30" matrices were taken, which contain only high-quality reads from both replicates.

Next, matrices at 5 kbp binsize were extracted and converted to cooler format using `hic2cool` and subsequently coarsened to resolutions of 10, 25, 50 and 100 kbp using `cooler coarsen`.

Contrary to the work from Farré et al. [27], which is using a distance-based normalization, and many others in the field which are using ICE- or KR-normalization, these matrices have not been normalized for the thesis at hand because no benefit of doing so was found during the study project [11].

In the cGAN approach outlined in section 3.2, the matrices are essentially treated as images and were thus scaled to a value range of [0...1] in 32-bit floating point format.

4.1.2 ChIP-seq data

For this thesis, ChIP-seq data for 13 chromatin features and DNaseI-seq data was used, cf. table 2. Here, ChIP-seq data for the 13 features was downloaded in .bam format from the ENCODE project XXXfor both replicate one and two, and DNaseI-seq data was downloaded in .bam format from XXX; the download links are also given in table 2.

The data were then converted to bigwig format, which is more convenient to handle, and the replicates were merged into one bigwig file. Pseudocode for the full conversion process is given in XXX.

- ENCODE as source
- Scaling to [0...1] for processing
- refrain from whitening, etc. – the data is no image with inherent correlations

feature name	download link
CTCF	
DNaseI	
H3k27ac	
H3k27me3	
H3k36me3	
H3k4me1	
H3k4me2	
H3k4me3	
H3k79me2	
H3k9ac	
H3k9me3	
H4k20me1	
Rad21	
Smc3	

Table 2: chromatin features used for the thesis

4.2 Dense Neural Network approach

4.2.1 Basic setup

In the basic setup, the chromatin features were binned to the same binsize b_{feat} as the training matrices using pybigwig XXXand stacked into a $l \times n$ array, where $l = \left\lceil \frac{cs}{b_{feat}} \right\rceil - 1$ with chromsize cs and $n = 14$ the number of ChIP-seq experiments. Hi-C matrices were taken as provided by the cooler format, cf. section 4.1.1, i. e. as $(l \times l)$-matrices with l as above, since by design $b_{mat} = b_{feat}$ and cs is a constant.

To create training samples for the neural network, subarrays of the feature array of size $(3w_{mat} \times n)$ were cut out such that the i-th training sample corresponded to the subarray containing the columns i, $i+1$, $i+2$, $\ldots$, $i+3w_{mat}$ of the full array. Sliding the window along the array with stepsize one obviously yields $N = l - 3w_{mat} + 1$ training samples. The corresponding Hi-C matrices were then cut out along the diagonal of the original matrix as submatrices with corner indices $[j,\ j]$, $[j,\ j+w_{mat}]$, $[j+w_{mat},\ j+w_{mat}]$, $[j+w_{mat},\ j]$ in clockwise order, where $j = i + w_{mat}$. The idea here is that the first 0, 1, $\ldots$ w_{mat} columns of each feature sample form the left flanking region of the training matrix, the next $w_{mat}+1$, $w_{mat}+2$, $\ldots$ $2w_{mat}$ correspond to the matrix' region and the last $2w_{mat}+1$, $2w_{mat}+2$, $\ldots$ $3w_{mat}$ columns form the right flanking region.

Because Hi-C matrices are symmetric by definition, it is enough to use the upper triangular part of the matrices, including the diagonal, and the neural network's last layer thus also needs exactly the corresponding number of output neurons, which is $\frac{w_{mat} \cdot (w_{mat}+1)}{2}$. Note that the size of the left and right flanking regions was chosen as w_{mat} in accordance with [27], but generally needs not be the same as the size of the training matrix. Figure 2 exemplarily shows the sample generation process for a (15×15)-matrix with $w_{mat} = 4$ and $n = 3$ chromatin features. In this case, five training samples would be generated – the one encircled in green and four more to the right.

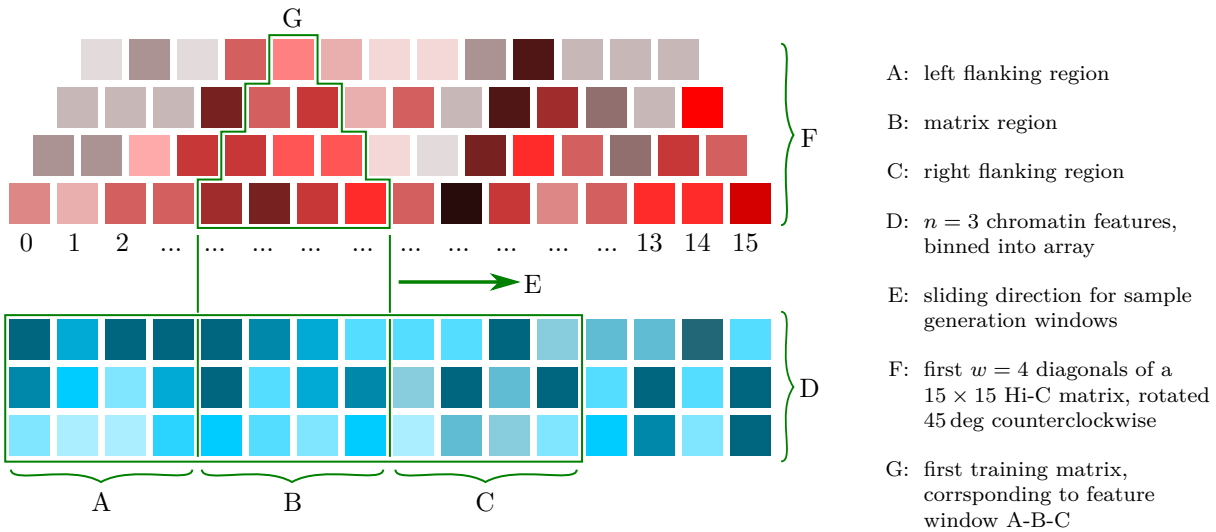


Figure 2: Sample generation process

Since the neural network G with its weights ω described above is meant for supervised learning, one needs to find weights ω^* such that the mean squared error L_2 between the predicted submatrices $M_s = G_s(\omega)$ and the training submatrices T_s becomes minimal for $s \in (1, 2, \ldots, N)$. Formally, one needs to optimize

$$\omega^* = arg\,min_\omega L_2(\omega) = arg\,min_\omega \frac{1}{N} \sum_{s=0}^{N} (M_s - T_s)^2 \tag{1}$$

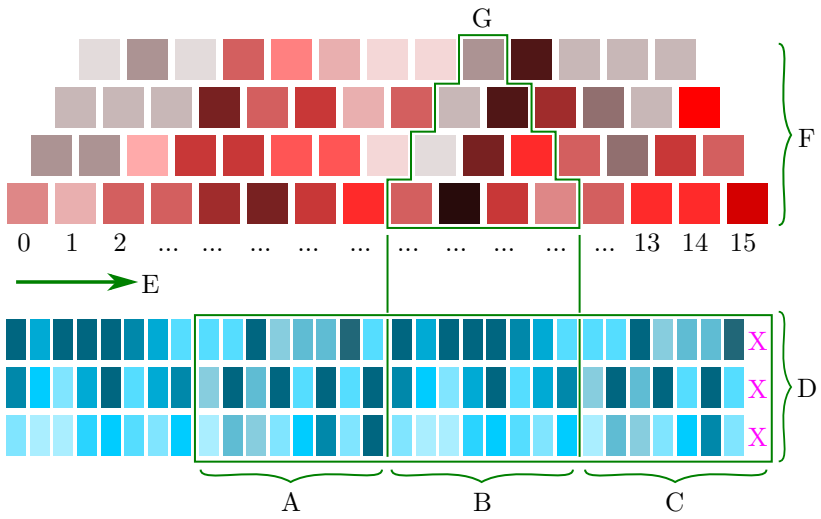
For the thesis at hand, stochastic gradient with learning rate $1e^{-5}$ was used to find w^*. Following [27], optimization was performed on minibatches of 30 samples, assembled randomly from the N training samples to prevent location-dependent effects and improve generalization. The last batch was dropped, if $N/30 \notin \mathbb{N}$.

The network and its learning algorithm was implemented in python using tensorflow deep learning framework with keras XXXcitation, see XXXrepository.

- nr neurons
- activations
- initialization
- optimizer, learning rate
- refrain from mirroring, little benefit for doubling input sample numbers
- train- and test sample preparation
- rebuilding the predicted matrix.

4.2.2 Modifying kernel size, number of filter layers and filters

maybe not needed



A: left flanking region

B: matrix region

C: right flanking region

D: $n = 3$ chromatin features, binned into array

E: sliding direction for sample generation windows

F: first $w = 4$ diagonals of a 15×15 Hi-C matrix, rotated 45 deg counterclockwise

G: first training matrix, corrsponding to feature window A-B-C

Figure 3: Generalized sample generation process for $k = 2$

4.2.3 Generalization of feature binning

The binning procedure described above in section 4.2.1 mostly also works if the binsize relation $k = \frac{b_{mat}}{b_{feat}}$ is not equal to one.

The training matrices remain unchanged, i. e. $(l \times l)$-arrays, from which training submatrices of size $w_{mat} \times w_{mat}$ can be extracted. With $k \in \mathbb{N}^{>1}$, one bin on the matrix diagonal corresponds to k bins of the feature array, so the feature windowsize needs to be k times the submatrix windowsize, $w_{feat} = k \cdot w_{mat}$. However, the number of bins along the matrix diagonal is generally not k times the number of bins in the feature array, see eq. (2).

$$l_{feat} = \left\lceil \frac{cs}{b_{feat}} \right\rceil - 1 = \left\lceil \frac{cs}{k \cdot b_{mat}} \right\rceil - 1 \neq k \cdot \left(\left\lceil \frac{cs}{b_{mat}} \right\rceil - 1 \right) = k \cdot l_{mat} \tag{2}$$

For the training process, this discrepancy has been resolved by simply dropping the last training sample, if the feature window belonging to it has missing bins. For the prediction process, the feature array has been padded with zeros on its upper end. This ensures that no errors are introduced into the training process by imputing values, but keeps the size of the predicted matrix consistent with the training matrix binsizes.

Figure 3 shows the generalized training process with a (15×15)-training matrix and $k = 2$. If $14 \cdot b_{mat} + b_{feat} \geq cs > 14 \cdot b_{mat} + 1$ holds for the chromosome size cs in the example, then the number of matrix bins will be $= 15$, while the number of chromatin feature bins will be $29 \neq 2 \cdot 15$. In this case, the 5th sample would be dropped for training, while a column with (three) zero bins would be added for to the feature array for prediction so that the resulting matrix would have the desired size of 15×15.

4.2.4 Custom loss function based on TAD insulation score

- computation details, custom layer
- full insulation score takes too long, simplified

4.2.5 Combination of mean squared error, perception loss and TV loss

maybe not needed

4.3 HiC-GAN approach

4.3.1 Using a DNN for 1D-2D conversion

4.3.2 Using a pre-trained DNN for 1D-2D conversion

4.3.3 Using a CNN for 1D-2D conversion

5 Results

5.1 Dense Neural Network approaches

5.1.1 Initial results for comparison

5.2 HiC-GAN approaches

6 Discussion and Outlook

Approach by Farre et al also working for human data. Changes might bring benefits in certain cituations, but are not a big improvement.

cGAN seems promising, but the discriminator remains to improve. Improvements still not good enough to beat transfer GM12878-K562 in many cases.

If possible, compare to other approaches. Maybe we can use the one by Zhang et al. for which data from study project exists, but it might take too long.

References

[1] Andrea Smallwood and Bing Ren. "Genome organization and long-range regulation of gene expression by enhancers". In: *Current Opinion in Cell Biology* 25.3 (June 2013), pp. 387–394. DOI: 10.1016/j.ceb.2013.02.005.

[2] David U. Gorkin, Danny Leung and Bing Ren. "The 3D Genome in Transcriptional Regulation and Pluripotency". In: *Cell Stem Cell* 14.6 (June 2014), pp. 762–775. DOI: 10.1016/j.stem.2014.05.017.

[3] Keerthi T. Chathoth and Nicolae Radu Zabet. "Chromatin architecture reorganization during neuronal cell differentiation in Drosophila genome". In: *Genome Research* 29.4 (Feb. 2019), pp. 613–625. DOI: 10.1101/gr.246710.118.

[4] Haoyue Zhang et al. "Chromatin structure dynamics during the mitosis-to-G1 phase transition". In: *Nature* 576.7785 (Nov. 2019), pp. 158–162. DOI: 10.1038/s41586-019-1778-y.

[5] Jennifer E. Phillips and Victor G. Corces. "CTCF: Master Weaver of the Genome". In: *Cell* 137.7 (June 2009), pp. 1194–1211. DOI: 10.1016/j.cell.2009.06.001.

[6] Jennifer E. Phillips-Cremins et al. "Architectural Protein Subclasses Shape 3D Organization of Genomes during Lineage Commitment". In: *Cell* 153.6 (June 2013), pp. 1281–1295. DOI: 10.1016/j.cell.2013.04.053.

[7] Jesse R. Dixon et al. "Chromatin architecture reorganization during stem cell differentiation". In: *Nature* 518.7539 (Feb. 2015), pp. 331–336. DOI: 10.1038/nature14222.

[8] E. Lieberman-Aiden et al. "Comprehensive Mapping of Long-Range Interactions Reveals Folding Principles of the Human Genome". In: *Science* 326.5950 (Oct. 2009), pp. 289–293. DOI: 10.1126/science.1181369.

[9] Suhas S.P. Rao et al. "A 3D Map of the Human Genome at Kilobase Resolution Reveals Principles of Chromatin Looping". In: *Cell* 159.7 (Dec. 2014), pp. 1665–1680. DOI: 10.1016/j.cell.2014.11.021.

[10] Houda Belaghzal, Job Dekker and Johan H. Gibcus. "Hi-C 2.0: An optimized Hi-C procedure for high-resolution genome-wide mapping of chromosome conformation". In: *Methods* 123 (July 2017), pp. 56–65. DOI: 10.1016/j.ymeth.2017.04.004.

[11] Ralf Krauth. *Improving predictions of Hi-C matrices from ChIP-seq data*. Tech. rep. Albert-Ludwigs Universität Freiburg, 2020. URL: https://github.com/MasterprojectRK/HiCPrediction.

[12] D. S. Johnson et al. "Genome-Wide Mapping of in Vivo Protein-DNA Interactions". In: *Science* 316.5830 (June 2007), pp. 1497–1502. DOI: 10.1126/science.1141319.

[13] Gordon Robertson et al. "Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing". In: *Nature Methods* 4.8 (June 2007), pp. 651–657. DOI: 10.1038/nmeth1068.

[14] ENCODE Project Consortium. "An integrated encyclopedia of DNA elements in the human genome". In: *Nature* 489.7414 (Sept. 2012), pp. 57–74. DOI: 10.1038/nature11247. URL: https://www.encodeproject.org.

[15] Carrie A. Davis et al. "The Encyclopedia of DNA elements (ENCODE): data portal update". In: *Nucleic Acids Research* 46.D1 (Nov. 2017), pp. D794–D801. DOI: 10.1093/nar/gkx1081.

[16] Boyan Bonev and Giacomo Cavalli. "Organization and function of the 3D genome". In: *Nature Reviews Genetics* 17.11 (Oct. 2016), pp. 661–678. DOI: 10.1038/nrg.2016.112.

[17] Chris A. Brackley et al. "Predicting the three-dimensional folding of cis-regulatory regions in mammalian genomes using bioinformatic data and polymer models". In: *Genome Biology* 17.1 (Mar. 2016). DOI: 10.1186/s13059-016-0909-0.

[18] Quinn MacPherson, Bruno Beltran and Andrew J. Spakowitz. "Bottom–up modeling of chromatin segregation due to epigenetic modifications". In: *Proceedings of the National Academy of Sciences* 115.50 (Nov. 2018), pp. 12739–12744. DOI: 10.1073/pnas.1812268115.

[19] Michele Di Pierro et al. "De novo prediction of human chromosome structures: Epigenetic marking patterns encode genome architecture". In: *Proceedings of the National Academy of Sciences* 114.46 (Oct. 2017), pp. 12126–12131. DOI: 10.1073/pnas.1714980114.

[20] Yifeng Qi and Bin Zhang. "Predicting three-dimensional genome organization with chromatin states". In: *PLOS Computational Biology* 15.6 (June 2019). Ed. by Jian Ma, e1007024. DOI: 10.1371/journal.pcbi.1007024.

[21] Pau Farré and Eldon Emberly. "A maximum-entropy model for predicting chromatin contacts". In: *PLOS Computational Biology* 14.2 (Feb. 2018). Ed. by Alexandre V. Morozov, e1005956. DOI: 10.1371/journal.pcbi.1005956.

[22] Jian Zhou and Olga G. Troyanskaya. "Probabilistic modelling of chromatin code landscape reveals functional diversity of enhancer-like chromatin states". In: *Nature Communications* 7.1 (Feb. 2016). DOI: 10.1038/ncomms10528.

[23] Ziad Al Bkhetan and Dariusz Plewczynski. "Three-dimensional Epigenome Statistical Model: Genome-wide Chromatin Looping Prediction". In: *Scientific Reports* 8.1 (Mar. 2018). DOI: 10.1038/s41598-018-23276-8.

[24] Yan Kai et al. "Predicting CTCF-mediated chromatin interactions by integrating genomic and epigenomic features". In: *Nature Communications* 9.1 (Oct. 2018). DOI: 10.1038/s41467-018-06664-6.

[25] Shilu Zhang et al. "In silico prediction of high-resolution Hi-C interaction matrices". In: *Nature Communications* 10.1 (Dec. 2019). DOI: 10.1038/s41467-019-13423-8.

[26] Laura D. Martens et al. "Identifying regulatory and spatial genomic architectural elements using cell type independent machine and deep learning models". In: (Apr. 2020). DOI: 10.1101/2020.04.19.049585.

[27] Pau Farré et al. "Dense neural networks for predicting chromatin conformation". In: *BMC Bioinformatics* 19.1 (Oct. 2018). DOI: 10.1186/s12859-018-2286-z.

[28] Shashank Singh et al. "Predicting enhancer-promoter interaction from genomic sequence with deep neural networks". In: *Quantitative Biology* 7.2 (June 2019), pp. 122–137. DOI: 10.1007/s40484-019-0154-0.

[29] Rui Peng. *Predicting High-order Chromatin Interactions from Human Genomic Sequence using Deep Neural Networks*. Tech. rep. Carnegie Mellon University, 2017. URL: https://www.ml.cmu.edu/research/dap-papers/F17/dap-peng-rui.pdf.

[30] Shashank Singh et al. "Predicting Enhancer-Promoter Interaction from Genomic Sequence with Deep Neural Networks". In: *bioRxiv* (Nov. 2016). DOI: 10.1101/085241.

[31] Jacob Schreiber et al. "Nucleotide sequence and DNaseI sensitivity are predictive of 3D chromatin architecture". In: *bioRxiv* (Jan. 2017). DOI: 10.1101/103614.

[32] Geoff Fudenberg, David R. Kelley and Katherine S. Pollard. "Predicting 3D genome folding from DNA sequence with Akita". In: *Nature Methods* 17.11 (Oct. 2020), pp. 1111–1117. DOI: 10.1038/s41592-020-0958-x.

[33] Geoff Fudenberg, David R. Kelley and Katherine S. Pollard. "Predicting 3D genome folding from DNA sequence". In: *bioRxiv* (Oct. 2019). DOI: 10.1101/800060.

[34] Ron Schwessinger et al. "DeepC: Predicting chromatin interactions using megabase scaled deep neural networks and transfer learning". In: *bioRxiv* (Aug. 2019). DOI: 10.1101/724005.

[35] Sarvesh Nikumbh and Nico Pfeifer. "Genetic sequence-based prediction of long-range chromatin interactions suggests a potential role of short tandem repeat sequences in genome organization". In: *BMC Bioinformatics* 18.1 (Apr. 2017). DOI: 10.1186/s12859-017-1624-x.

[36] A. Tsirikoglou, G. Eilertsen and J. Unger. "A Survey of Image Synthesis Methods for Visual Machine Learning". In: *Computer Graphics Forum* 39.6 (Sept. 2020), pp. 426–451. DOI: 10.1111/cgf.14047.

[37] Tong Liu and Zheng Wang. "HiCNN2: Enhancing the Resolution of Hi-C Data Using an Ensemble of Convolutional Neural Networks". In: *Genes* 10.11 (Oct. 2019), p. 862. DOI: 10.3390/genes10110862.

[38] Ian Goodfellow et al. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani et al. Vol. 27. Curran Associates, Inc., 2014, pp. 2672–2680. URL: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

[39] Mehdi Mirza and Simon Osindero. *Conditional Generative Adversarial Nets*. 2014. arXiv: 1411.1784 [cs.LG].

[40] Qiao Liu, Hairong Lv and Rui Jiang. "hicGAN infers super resolution Hi-C data with generative adversarial networks". In: *Bioinformatics* 35.14 (July 2019), pp. i99–i107. DOI: 10.1093/bioinformatics/btz317.

[41] Hao Hong et al. "DeepHiC: A generative adversarial network for enhancing Hi-C data resolution". In: *PLOS Computational Biology* 16.2 (Feb. 2020). Ed. by Ferhat Ay, e1007287. DOI: 10.1371/journal.pcbi.1007287.

[42] Michael C. Dimmick, Leo J. Lee and Brendan J. Frey. "HiCSR: a Hi-C super-resolution framework for producing highly realistic contact maps". In: *bioRxiv* (Feb. 2020). DOI: 10.1101/2020.02.24.961714.

[43] Kai Huang, Vadim Backman and Igal Szleifer. "Interphase chromatin as a self-returning random walk: Can DNA fold into liquid trees?" In: (Sept. 2018). DOI: 10.1101/413872.

[44] Artemi Bendandi et al. "Chromatin Compaction Multiscale Modeling: A Complex Synergy Between Theory, Simulation, and Experiment". In: *Frontiers in Molecular Biosciences* 7 (Feb. 2020). DOI: 10.3389/fmolb.2020.00015.

[45] Andre Bajorat. *Hi-C Predictions based on protein levels*. Tech. rep. Albert-Ludwigs Universität Freiburg, 2019. URL: https://www.bioinf.uni-freiburg.de/Lehre/Theses/TP_Andre_Bajorat.pdf.

Acronyms

ChIA-PET chromatin interaction analysis by paired-end tag sequencing.

ChIP-seq chromatin immunoprecipitation followed by sequencing.

CNN convolutional neural network.

DamID DNA adenine methyltransferase identification.

DNN dense neural network.

GAN generative adversarial network.

LSTM long-short-term memory.

TAD topologically associating domain.