Changes of nucleosome positioning and 3D chromatin organization in cell transitions

Christopher T. Clarkson

A thesis submitted for the degree of Doctor of Philosophy in Cell and Molecular Biology

School of Life Sciences

University of Essex
January 2020

Summary

The genome of a eukaryotic cell is stored inside the nucleus in a highly condensed form called chromatin. The basic unit of chromatin is the nucleosome. The positioning of nucleosomes on the DNA determines the accessibility of transcription factors (TFs) and other regulatory molecules. Beyond nucleosome positioning, the higher level of 3D chromatin architecture is constituted by relatively large loops of DNA such as topologically associating domains (TADs) which serve to insulate some loci from the rest of the genome. A major determinant of the chromatin domain boundaries is the architectural protein CTCF that binds to thousands of locations in the genome and changes the chromatin configuration during cell differentiation or cancer development. Chapter 1 of this thesis provides an overview of the field of chromatin folding and a premise of the questions that we later address. Chapter 2 is based on our paper [Clarkson et al. (2019) *Nucleic Acids Res.* **47**, 11181-11196] devoted to CTCF-nucleosome interplay at chromatin boundaries. It reports a new effect: the strength of CTCF binding to DNA is inversely proportional to the average distance between nucleosomes near its binding site. We found that a number of CTCF binding sites that remain bound during the differentiation of mouse embryonic stem cells maintain a relatively short distance between the neighbouring nucleosomes. Furthermore, we observed that CTCF binding sites occur in clusters at TAD boundaries, and proposed a new model of chromatin boundary formation through ordered, asymmetric nucleosome arrays. Chapter 3 documents my work on the connection between nucleosome positioning and chromatin state using machine learning. We developed a general methodology for this task and provided a proof of principle that it can work as a diagnostic tool classifying nucleosome positioning patterns to distinguish samples from peripheral blood of healthy individuals and patients with chronic lymphocytic leukaemia.

Acknowledgements

My PhD has been tumultuous and I could not have completed it without the help of the following people:

Firstly, thank you most of all to my supervisor Dr Vladimir Teif for the help and support that he gave me.

Thank you to Stuart Newman for all of his help with the cluster (apologies for the one or two times that I caused it to crash… although definitive proof that it was me has never been released- I'm just saying…).

Thank you to Graeme Thorn for his help with statistics and coding.

Thank you to Yevhen Vainshtein who wrote the code for the software NucTools and helped me to debug scripts many times.

Thank you to Karsten Rippe who led the CancerEpiSys consortium and allowed me to use their clinical data.

Thank you to Victor Zhurkin, for his timely advice on our CTCF paper.

Thank you to Emma Deeks and Ralph Samarista for the hard work on the CTCF paper.

Thank you to Hulkar Mamasuyupova for her advice on the CTCF paper and also her kind nature and demeanour in our lab.

Thank you to Elena Klenova for her highly useful input and advice on our paper.

Thank you to Viola Fanfani, Ravikiran Chimatapu, Hani Hagras and Giovanni Stracquadanio for the advice on machine learning.

Thank you to my girlfriend Ellie, who has been kind and supportive from the first day that I met her.

To Mum and Dad, your support, both financial and otherwise made my PhD possible- thank you for all the nights on the phone insisting that I eat well and sleep properly (it was a bit annoying at times but still- thanks.. Love you guys).

List of abbreviations

AUC- Area under the curve

CGI- CpG island

CLL- Chronic Lymphocytic Leukemia

CNN- Convolutional Neural Network

CSV- Comma separated values

CTCF- CCCCTC binding factor

CV- Cross validation

GAN- Generative adversarial network

mESC- mouse embryonic stem cells

ML- Machine learning

MEFs- Mouse embryonic fibroblasts

NPCs- Neural progenitor cells

NPS- Nucleosome positioning

NRL- Nucleosome repeat length

ROC- Reciever operator curve

SHAP- Shapely additive predictions

SNs- Strong nucleososomes

TAD- Topologically associated domain

TF- Transcription factor

TSS- Transcription start sites

TP-True Positive TN- True Negative FP- False Positive FN- False Negative

Table of Contents

Layperson overview

The nucleus of a cell contains 2 meters of DNA folded into a space that is ~1 μm diameter. There the genomic DNA exists in the form of chromatin – the complex of nucleic acids and proteins. This phenomenon allows for the instructions of the DNA code to be harnessed and for the different cell types to develop which will give rise to the tissues which ultimately make a full organism. Throughout my PhD I have studied a very distinct set of mechanisms of how chromatin conformation is involved in the process of cell differentiation.

One aspect of my PhD involved the way that the genome as a whole folds. In one cell type, such as an liver cell, the genome will be folded in a particular way whereas in a brain cell the genome will have folded differently to turn on/off the genes required to maintain the regulator pathways necessary for that cell. How, one might ask, can such different cellular behaviours result from a genome that is identical between these two cases?

On a small scale, one can study the chromatin complex up close and see what are known as nucleosomes. Nucleosomes are composed of proteins (known as histones) wrapped by DNA. We can think of nucleosomes as 'beads on a string'. They cover different parts of the genome and determine whether the constituent DNA is accessible. This is to say that if a region of chromatin has many nucleosomes (or beads) closely packed together – it cannot be accessed by the biological machinery that reads DNA and translates the instructions into RNA. Hence nucleosomes can determine which genes are 'read' and thus how the cell behaves.

If one were then to zoom out and take a 'bird's eye' perspective of chromatin, one would see that it is organised into regular loops each of which occupy a distinct area and are kept preferentially in contact with or away from other parts of the genome. While the full extent to which this

phenomenon is involved in governing cell behaviour is not clear, there is strong evidence that this organisation is by design and has at the very least some role in the process.

During my PhD I studied the way that the nucleosomes which decorate the chromatin on a lower level influence the higher order folding of the genome and how this may in turn affect cellular behaviour.

CHAPTER 1. LITERATURE REVIEW

1.1 Introduction

The genome of a eukaryotic cell is stored inside the nucleus in the form of a nucleoprotein complex called chromatin. The elementary unit of chromatin is the nucleosome. The nucleosome is composed of a histone octamer that is wrapped by 147 DNA base pairs (Luger *et al.* 1997). Nucleosome positioning in the genome is not random, and the positioning of nucleosomes determines the accessibility of DNA to enable the binding of transcription factors (TFs) and other proteins. Furthermore nucleosomes make it energetically unfavourable for the bound DNA to be transcribed (Chen *et al.* 2019). On a larger scale, different types of packing of nucleosome arrays lead to different genomic regions having varying chromatin density, protein abundances and gene expression. Chromatin has been historically classified into two types based on its compaction: these include regions called heterochromatin and euchromatin (Kresge, Simoni and Hill 2010). Heterochromatin signifies genomic regions where nucleosomes are arranged in a dense array and show low levels of expression, while the opposite is true for euchromatin. More than just two types of chromatin packing have now been discovered (Filion *et al.* 2010). Recent advances in sequencing technologies allowed genome wide characterisation of nucleosome positioning (Cremer and Cremer 2010) and long-range genomic contacts (Dekker and Mirny 2016). The extent to which nucleosome positioning affects higher order folding of the genome and is involved in cell-differentiation and disease is an area of extensive research.

1.2 Nucleosome positioning

1.2.1 Experimental measurement of nucleosome positions

There are many experimental methods to measure nucleosome positioning. Throughout this project our data mostly originating from three types of experimental data: MNase-seq, Mnase-assisted histone H3 ChIP-seq and chemical mapping, as explained below.

1.2.1.1 MNase-seq

The most commonly used technique for measuring for recording nucleosome positioning is MNase-seq. In this method the enzyme Micrococcal Nuclease (MNase) cuts the DNA between nucleosomes (Lai *et al.* 2018). The DNA that wraps the histone octamer core is then sequenced and mapped to the reference genome. The quality of the mapping can be assessed to check if the experimental results are viable. The concentration of MNase used in the experiment is critical as it can result in over and under digestion of the genome in different parts of the genome. Other important parameters include the temperature. This is because MNase preferentially cleaves AT-rich DNA which can lead to artificially long reads. This concept is illustrated in Figure 1 below.

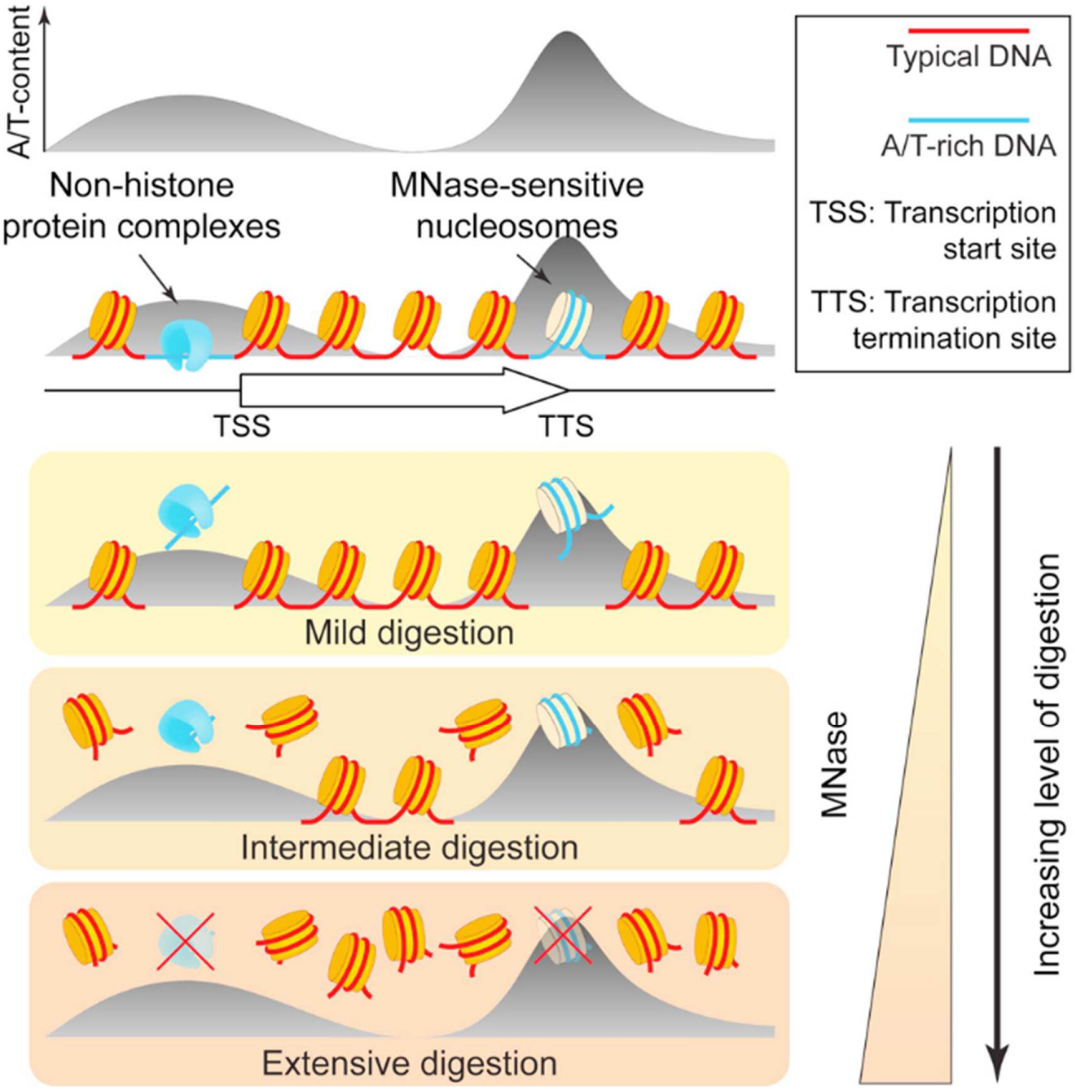


Figure 1: An illustration of the concept of how different parts of the genome have varying levels of sensitivity to MNase digestion and this can lead to information loss depending on the chosen concentration of the MNase. Adapted from (Lai *et al.* 2018).

1.2.1.2 MNase-assisted histone H3 ChIP-seq

One of the limitations of MNase-seq is that a number of protein complexes other than the histone octamer can protect DNA from MNase digestion and thus be mistaken for the nucleosomes. Therefore, some recent studies used an updated method called MNase-assisted histone H3 ChIP-

seq, which is an adaptation on the MNase-seq protocol where the chromatin is treated with MNase but also with an antibody against histone H3 to precipitate only the pieces of DNA associated with histone H3 (Wiehle *et al.* 2019).

1.2.1.3 Chemical mapping

While MNase-seq and its variations are widely used, this method has some disadvantages. First, MNase preferentially cleaves AT rich regions. Second, the cleavage of the DNA strand occurs at the border of the nucleosome bound DNA and it is not always possible to tell where the borders of the nucleosome begin and end. This is because different regions of the genome have variable levels of sensitivity to MNase digestion and this can result in over/ under digestion at different loci. To address these problems, several alternative approaches have been proposed. For example, in the method introduced by Voong and coauthors (2016), the histone residue in contact with the nucleosome dyad is mutated from Cysteine to Serine such that chemical cleavage can happen at the center of the nucleosome dyad (i.e. the center of the nucleosome covered DNA region). The resulting DNA reads then span between the centers of adjacent nucleosomes and can be mapped to the genome (Voong *et al.* 2016, 2017). However, this method also has a weakness, since many DNA-binding proteins other than histones contain Serine. Thus, DNA cuts can be introduced at those positions, resulting in false-positive nucleosome calls.

1.2.2 Determinants of genomic nucleosome positioning

One can distinguish the following four major determinants of nucleosome positions, which will be detailed in the next subsections:

- Intrinsic DNA sequence affinity of the histone octamer
- Chemical modifications of DNA or histones
- Competitive binding of nucleosomes and transcription factors
- ATP-dependent nucleosome repositioning by chromatin remodellers.

1.2.2.1 Sequence-dependent nucleosome positioning

Since the discovery of the nucleosome there has been a search for the best method to predict nucleosome positions (Segal *et al.* 2006; Teif and Clarkson 2019). DNA has a natural affinity for histones as its backbone is constituted by negatively charged phosphate residues and can be neutralised by histones which bear positive charges. Thus, nucleosomes can form at any location, but some locations are more likely to be selected for nucleosome formation for a number of reasons.

The history of the nucleosome positioning field started about 40 years ago with the finding that genomic DNA is characterised by ~10 bp periodicity of dinucleotide distribution, which theoretically could play a role in histone recruitment to specific sites in the DNA (Trifonov and Sussman 1980). This hypothesis was further explored in a subsequent study analysing 177 DNA molecules in an attempt to find the optimal sequence of base pairs to recruit a histone octamer (Satchwell, Drew and Travers 1986). Lowary & Widom elaborated on this work with an experiment to identify the optimal sequence of 147 bp for in vitro recruitment of the histone octamer (Lowary *et al.* 1998). A large pool of randomly generated sequences was assessed via salt-dialysis in competition with one another to attract a histone. Through this, the "Widom 601 sequence" was found to have the highest affinity for the histone octamer. This '601 sequence' gained prevalence as a strong nucleosome positioning element used in a number of following experiments. The cumulative evidence from these experiments confirmed the hypothesis that the

DNA sequence at least partially affects the positions of nucleosomes. This hypothesis was further supported in the post-genomic era, when genome-wide locations of nucleosomes have been mapped in many organisms and cell types (Flaus 2011). The theory of 'Strong Nucleosomes' (SNs) was proposed by Trifonov in 2012 (Trifonov and Nibhani 2015). This theory suggests that a certain class of longer DNA motifs exist which specifically attracts histone octamers both due to the local bend and due to additional interactions (Trifonov and Nibhani 2015). SNs were mapped in the *Arabidopsis Thaliana*, *Caenorhabditis elegans* and mouse genome, finding that they were particularly enriched in centromeric regions (Salih *et al.* 2015). It is currently believed that ~9% of nucleosomes are arranged consistently by the DNA sequence across different types of human cells (Gaffney *et al.* 2012).

The DNA nucleotide content affects the local bendability of the DNA sequence and hence its ability to wrap around the histone octamer. Several schemes exist for assessing nucleotide content of a piece of DNA and predict the energetic cost of its bending.

On a kilobase scale, genome-wide sequencing showed that GC-rich regions have higher nucleosome density, while A/T rich regions are more nucleosome depleted (Segal *et al.* 2006) yeast. AT-tracts have been proposed for the role of a nucleosome-stopping boundary due to their particular rigidity (Segal and Widom 2009). Large domains of GC-rich regions frequently coincide with constitutive heterochromatin with high nucleosome density. At the same time, many regulatory regions such as promoters and enhancers are GC-rich, but nucleosome depleted. The latter is especially true for so called CpG islands (CGIs), which are the "islands" of high-density unmethylated CpG di-nucleotides with preferably low nucleosome occupancy, situated in the "sea" of mostly methylated DNA with higher nucleosome density (Robertson 2005; Teif *et al.* 2014; Liu *et al.* 2018).

It is also known that the local conformation of the DNA affects its affinity for the histone octamer: if the minor groove is facing outwards, the histone octamer can be accepted and wrapped by the DNA (Flaus 2011). The local DNA bendability in general is a complicated function of the DNA sequence; a simple consideration that the period of the double helix is roughly 10 bp dictates that the nucleosome-wrapping DNA sequence should make continuous bends with about 10 bp periodicity in order to keep the major groove in contact with the histone octamer and the minor groove facing outwards. The latter can be favoured by having easily bendable elements (such as dinucleotides, trinucleotides or larger k-mers) incorporated in the double helix with 10-bp periodicity. Indeed, early studies have found ~10 bp periodicity of most dinucleotides in both chicken and yeast (Segal *et al.* 2006; Lantermann *et al.* 2010). However, the full extent to which nucleotide content is responsible for *in vivo* recruitment of histones, on a sub-kilobase level, is still under investigation (Chen *et al.* 2010). There are 6 categories of conformational parameters in that can be measured in the DNA sequence. These are: twist, shift, slide, roll, rise and tilt. It has been demonstrated that the known force constants that result from a given di-nucleotide pair in a given conformational change provide a measure of stiffness of the DNA (Lankaš *et al.* 2003). The local stiffness of a strand of DNA will then influence the likelihood of being wrapped by a histone (Deniz *et al.* 2011).

1.2.2.2 Effects of post-translational histone modifications on nucleosome positioning

Histones can be post-translationally modified at a number of positions. Post-translational modifications (PTMs) influence the conformation and subsequent function of a protein. Heterochromatin regions are known to have very low histone acetylation levels. One possible reason for this is that the negative acetyl group serves to negate the positive histone charge and weaken histone interaction with the negatively charged DNA phosphate backbone. On the other

hand, a typical heterochromatin modification H3K9me3 recruits heterochromatin protein 1 (HP1) which recruits more histone modifying factors to maintain and helps spreading the H3K9me3 modification (Wang *et al.* 2014).

1.2.2.3 Nucleosome competition with transcription factors

Histone octamers must compete with other proteins to bind a strand of DNA. The competitive binding of transcription factors is one of major regulators of nucleosome positioning. Experimental evidence has accumulated to show that the local concentration of transcription factors will determine their binding to a given DNA site (Karreth, Tay and Pandolfi 2014). Since one nucleosome can cover several TF binding sites, this competition can lead to cooperative effects (Miller and Widom 2003; Rippe 2010). The cooperative displacement of histones by TFs can be understood in analogy to the allosteric change that happens in haemoglobin, induced by the varying levels of oxygen concentration. Similarly, the histone affinity for DNA can be modelled with a sigmoid curve as a function of TF concentration, where the number of TF molecules increases to the point of saturation and the chance of nucleosome displacement increases (Mirny 2010). A specific example of nucleosome displacement via TF binding is that of Rap1. Rap1 has been shown to bind at TSS promoters of various genes and in, conjunction with the remodelling factor RSC in budding yeast, removes stable nucleosomes and subsequently recruits other TFs to regulate gene expression (Mivelaz *et al.* 2019). CTCF is another regulatory chromatin protein that has been shown to competitively bind to DNA against histones – the significance of this will be further discussed in '1.4.2 Transcription Factor Binding'.

1.2.2.4. ATP-dependent nucleosome repositioning by chromatin remodellers

Chromatin enzymes that actively reposition nucleosomes along the DNA are known as chromatin remodellers. A typical effect of a remodeller would be to remove a bound histone octamer from the DNA or to reposition it to change the local chromatin architecture. Several studies showed that many chromatin remodellers reposition nucleosomes in specific steps e.g. intervals of 10 bp (Xu and Olson 2010).

The mechanism by which nucleosomes are repositioned on the DNA is a topic of great interest. The most popular model to date is the 'loop recapture model' whereby the nucleosome is partially unwrapped in a step-wise fashion along each position in the nucleosome allowing the histone octamer to slide its position along the DNA (Schiessel *et al.* 2001). Examples of ATP-dependent chromatin remodellers are SWI/SNF and chromatin structure remodelling complex (RSC) (Längst and Becker 2004; Teif and Rippe 2009).

1.2.3 Predicting nucleosome positioning

Many algorithms exist for predicting nucleosome positions. These can be roughly categorized into "biophysical" (taking into account physical properties of DNA and histones) and "bioinformatical" (learning the rules of preferred nucleosome distributions without knowing details of molecular interactions), as detailed in our recent review (Teif and Clarkson 2019). Some of these algorithms are described below.

1.2.3.1 Classical bioinformatics algorithms

Several algorithms developed by Trifonov and co-authors implemented the 'Strong Nucleosomes' (SN) concept postulating that sequence can specifically recruit histones (Trifonov and Nibhani 2015). These algorithms study the base pair content of sequences and how far they deviate from the SN sequence templates. A sequence that is similar to these templates will be predicted to have a high nucleosome occupancy. Another type of algorithm uses di-nucleotide distributions of TA, TT, AA and GC and analyses them to predict the distribution of nucleosomes across a sequence of DNA based on the biophysical rules that have been shown to influence nucleosome formation (see above- '1.2.2 Determinants of nucleosome positioning') (Struhl and Segal 2013). Both of these types of algorithms were used to successfully position nucleosomes, coming to the conclusion that the experimentally validated '601 sequence' (discussed above in '1.2.2.1 Sequence-dependent nucleosome positioning') has a strong influence on the positioning of nucleosomes (van der Heijden *et al.* 2012).

1.2.3.2 Biophysical models

Several algorithms have been developed specifically to account for thermodynamically most probable nucleosome distribution along the DNA. For example, the interactive chromatin modelling (ICM) algorithm takes a sequence of DNA and, using di-nucleotide content, generates an energy level prediction at the different positions to predict favourable binding sites of nucleosomes. Disadvantages of such a strategy include that computation will take a long time and only relatively short sequences can be taken as input (Stolz and Bishop 2010). 'NucEnerGen' also studies di-nucleotide content and, using a model that was trained on the nucleosome occupancy of the yeast genome, calculates the formation energies of nucleosomes along the strand of DNA and

based on steric exclusion due to the local bendability of the DNA. Histone DNA interaction strength is assumed to be arbitrary herein (Locke *et al.* 2010). 'nuScore' is a program that, as opposed to previous algorithms which considered di-nucleotide content to make predictions, takes a vice versa approach and predicts the energy cost that the wrapping of a histone will implement on the DNA. This is done using a well characterised template of DNA. From this template, "stiffness constants" are derived based on sequence and these are then mathematically imposed on the input sequence to calculate the most favourable configuration of nucleosomes along the given strand of DNA (Tolstorukov *et al.* 2008).

1.2.3.3 Models that consider ATP-dependent chromatin remodellers

For the purpose of predicting nucleosome positioning under *in vivo* conditions, it is essential to consider the activity of ATP-dependent remodellers as the system allow for the dynamic repositioning of nucleosomes for dynamic regulation. This is because these enzymes serve the purpose of reducing the local energy barriers which prevent undesired manipulation of the chromatin under equilibrium conditions (Padinhateeri and Marko 2011; Beshnova *et al.* 2014). On top of considering the histone octamers to specific sequences at equilibrium, it is necessary for the simulation to allow the shifting of the nucleosomes by a set amount (Teif and Rippe 2009). This model predicts the putative nucleosome positions using dynamic programming to simulate a remodeller either shifting a nucleosome to the right/left (depending on the remodeller in question the repositioning step-size should be approximately known e.g. SWI/SNF ~50 bp) or removing the nucleosome completely, where each possibility is represented as a state and ultimately assigning a probability to a given 147 bp sequence of being bound by the nucleosome (Beshnova *et al.* 2014). Similarly to the model of Teif and Rippe, this model provides a simulation of nucleosome sliding,

evicting or a naked DNA being bound by a histone octamer in order to calculate the probability of a given sequence being bound. Also the ATP-dependent remodellers were modelled using experimentally determined enzyme rates (Padinhateeri and Marko 2011).

1.2.3.4 Models for nucleosome-TF competition

A number of theoretical models have been developed to take into account competitive binding of transcription factors and other chromatin proteins. Hence these algorithms can be used to calculate the theoretical positions of nucleosomes while considering the effect of transcription factors.

TFNuc is an algorithm, that considers the DNA as a one dimensional lattice. It estimates the TF binding constants with a provided PWM. It considers the phenomenon of partial unwrapping of the histone by TF/histone invasion at the entry/exit point of the nucleosome. This allows for a more dynamic simulation of *in vivo* nucleosome positioning and hence provides greater potential to more accurately predict the positioning configuration (Teif *et al.* 2013). Incorporating the relative accessibility of the given region into the model improves the accuracy (Kaplan *et al.* 2011). In a separate paper a biophysical model was used to assess nucleosome positioning in both *in vivo* and *in vitro* maps of the yeast what type genome, and it was found that TF binding was only a significant propellent of the system in promoter (nucleosome-free) regions, where the TFs had to compete with ATP-dependent remodellers rather than histones (Ozonov and van Nimwegen 2013).

1.2.4 Machine-learning algorithms for nucleosome positioning

Machine learning (ML), is the use of mathematical models to make predictions and/or recognise patterns in a given dataset by fitting parameters to input data.

A more formal definition, as per Samuel (1959), describes ML as follows: “A computer program

is said to learn from experience E with respect to task T and some performance measure P, if its performance on T, as measured by P, improves with experience E" (Mitchell 2006; Géron 2019). Hence one feeds and algorithm with input data and it will adjust its parameters to find patterns in order to improve the accuracy of its predictions. ML can be used for a wide range of tasks such as email filtration (categorising emails as spam vs not spam) or diagnostic tests (identifying cancer in patients). Machine learning techniques can be separated into two general categories:

- Supervised training: Here the data are 'labelled'- this means that each datapoint fed to the algorithm will have an assigned category. The data are split into a 'training set' and a 'test set'. The algorithm will then learn the relationship between the labels and the input from the training data. Subsequently the validity of these learned relationships are assessed exposing the trained algorithm to the test data and comparing the output predictions to the real test labels (Géron 2019).
- Unsupervised training: Here the data are unlabelled and hence the labels/categories are learned by exposing the algorithm to raw, uncharacterised data and it will learn continuously and update the assigned labels with each new input (Géron 2019).

Examples of machine learning being applied to nucleosome positioning are: 'iNuc-PhysChem' - software that predicts nucleosomal sequences based on their local physico-chemical properties. This model was trained on a yeast genome dataset and predicts with 96% accuracy using a supervised algorithm (Chen *et al.* 2012b). Another model 'NuPoP' considers the interaction between nucleosomes. It does so by modelling inputting the linker lengths of chromatin into a Hidden Markov Model (HMM) and calculates the most likely configuration of nucleosomes along the DNA. This is an unsupervised algorithm and it does not perform as well as the above mentioned supervised method with 70% sensitivity (Xi *et al.* 2010).

1.2.4.1 Deep Learning/ Neural networks

Deep learning is a family of algorithms that have become popular for studying epigenomic data, including the locations of nucleosomes. The most basic type of deep learning model is a neural network. A neural network is a model inspired by the biological structure of the brain, whereby electrical input signals are carried through a network of biological neurons with synaptic terminals acting as filters. Throughout the ages of eukaryotic evolution, this system has proved to be a highly optimised way of recognising trends in complex data. Since the 1960's various pursuits have been made to impute the intuitive trend-recognising abilities of a brain into a computer with the goal of being able to study complex data such as e.g. financial markets, biological data etc. Hence an artificial neural network is a highly simplified simulation of a real biological neural network, designed to intake data in the input layer and output predictions through an end layer. Via an algorithm called forward and back-propagation, the network learns the rules of a given system to detect trends. Briefly, forward-propagation passes the signal from the input layer to the output layer which makes a prediction, the accuracy of this prediction is used as a reference to improve through back-propagation which adjust the weights of the network one layer at a time. In parallel, back-propagation, as a feature of dynamic programming, is also implemented in sequence alignment whereby genetic sequences are arranged to highlight their common patterns. Neural networks have proved extremely powerful in in accuracy by out-performing other machine-learning algorithms (Perkel 1988; Rampasek and Goldenberg 2016). Recently Di Gangi and coauthors used a deep neural network architecture to predict the binding of nucleosomes in *vivo* using the DNA sequence as input (Di Gangi, Lo Bosco and Rizzo 2018). This was done in 5 different organisms. The accuracy achieved was 89%.

There are many different neural network architectures. Two prevalent types of neural networks are 'convolutional neural networks' (CNNs) and 'long short-term memory networks' (LSTMs). CNNs are based on the cerebral cortex of the brain and are used mostly for image recognition. LSTMs are designed for short sequence recognition.

1.2.4.2 Making neural networks explainable

A common problem of deep learning algorithms is the lack of interpretability of the models. This refers to the fact that one cannot know how/why/on what basis a trained model makes a prediction. Some algorithms have explainable platforms such as fuzzy logic systems (whereby an input can be a member of multiple classes simultaneously). However, in the case of neural networks, they are often referred to as a "black box" algorithm, in reference to the fact that they accept input data and return highly accurate predictions, but there is little to no possibility of knowing the basis/ learned rules for how the prediction is made. This can cause problems with the progression of deep learning research. Without any underlying insight into the learned relationships on the input data, one cannot know what input features are most influential in the predictions being made. Generally as the predictive power of a model increases, its complexity also increases at the cost of explainability (Kuhn and Johnson 2013). There have been recent efforts to increase the interpretability of ML models. An example is the software SHAP (SHapley Additive exPlantions), where, in the case of neural networks, an input image is highlighted with colour indicating which features the model is most reacting to (Lundberg and Lee 2017).

Another possible way to learn what is important to a trained neural network is to use a generative adversarial network (GAN). Herein, the trained neural network (referred to as the discriminator) is exposed to fake images output from another neural network (referred to as a generator). The

weights of the discriminator are 'frozen' so that it cannot learn whereas the generator will be allowed keep producing fake images and learning from the feedback of the discriminator until it begins to 'fool' the discriminator. This can produce images that are emblematic of the learned features of the original trained model (Goodfellow *et al.* 2014). These concepts are illustrated below in Figure 2.

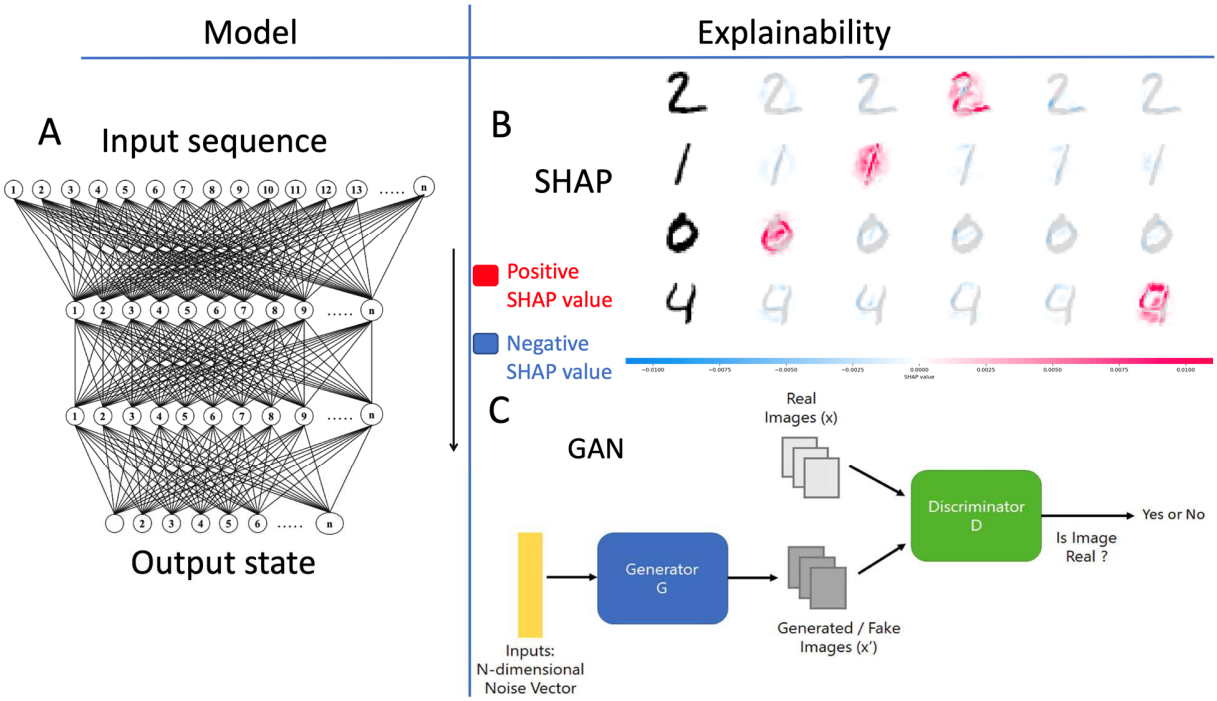


Figure 2. Overview of deep learning. (A) Shows a generic neural network is shown with input, hidden and output layers. The right panel represents techniques that can be used in an attempt to make neural network predictions interpretable. (B) Shows exemplary output of the software 'SHAP' which takes an input image and is highlights it with colour indicating which features the model is most reacting to (red for positive reaction and blue for negative). This gives an indication of how and why the model classifies a given input. (C) Shows a schematic layout of a GAN. The

discriminator is exposed to fake images output from the generator. The generator will keep producing fake images and learning from the feedback of the discriminator until it begins to 'fool' the discriminator. Figures adapted from (VanderPlas 2016; Shrikumar, Greenside and Kundaje 2017; Buscema et al. 2018).

1.3 Relationship between different levels of chromatin architecture

Chromatin has different nucleosome packing rates in different parts of the genome depending on the cell type. This is necessary for conveying the correct instructions to achieve a particular cell type. Dense packing of nucleosomes is characteristic of heterochromatin whereas euchromatin tends to be loose. Figure 3 illustrates the hypothesis that there could be a connection between lower level (nucleosome packing) and higher order (chromatin state, TAD formation, chromosome topology- explained below) chromatin architecture (Figure 3).

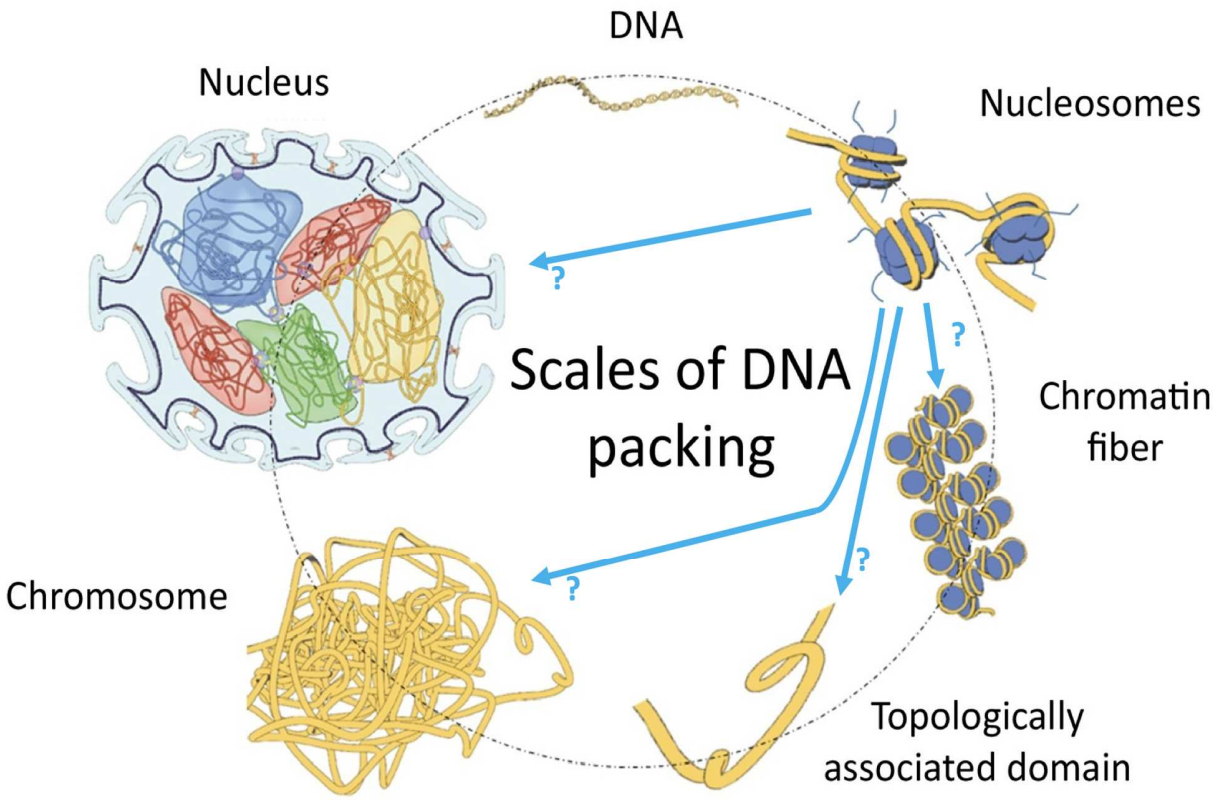


Figure 3. Illustration of scales of chromatin packing. Adapted to represent possible connections between different levels of chromatin organisation. Blue arrows with question marks represent the possible connections between lower and higher levels of chromatin organisation adapted from (Uhler and Shivashankar 2017).

The following subsections will be used to outline the different ways that one can characterise chromatin in terms of its nucleosome density and subsequently the higher order structures that could be affected by local organisation of nucleosomes.

1.3.1 Nucleosome repeat length

One of the classical characteristics of nucleosome packing in chromatin is the nucleosome repeat length (NRL), which is the average distance between the centers of neighbouring nucleosomes (Van Holde and Zlatanova 1996; Beshnova *et al.* 2014). Specifically, the NRL is calculated by using a histogram to document the inter-nucleosome distances, the resulting peaks are plotted as a function of themselves and the slope of the resulting best-fit line is the NRL. The NRL of different genomic regions depends on a number of factors including the size of linker histone proteins, and the interaction between nucleosomes. The NRL appears to be specific to the region in question- either due to DNA sequence-determined boundaries or due to competition with chromatin proteins (Beshnova *et al.* 2014). Figure 4 illustrates the workflow of calculating the NRL and how it can only be done with a sufficient amount of data (due to the need for reliable statistics).

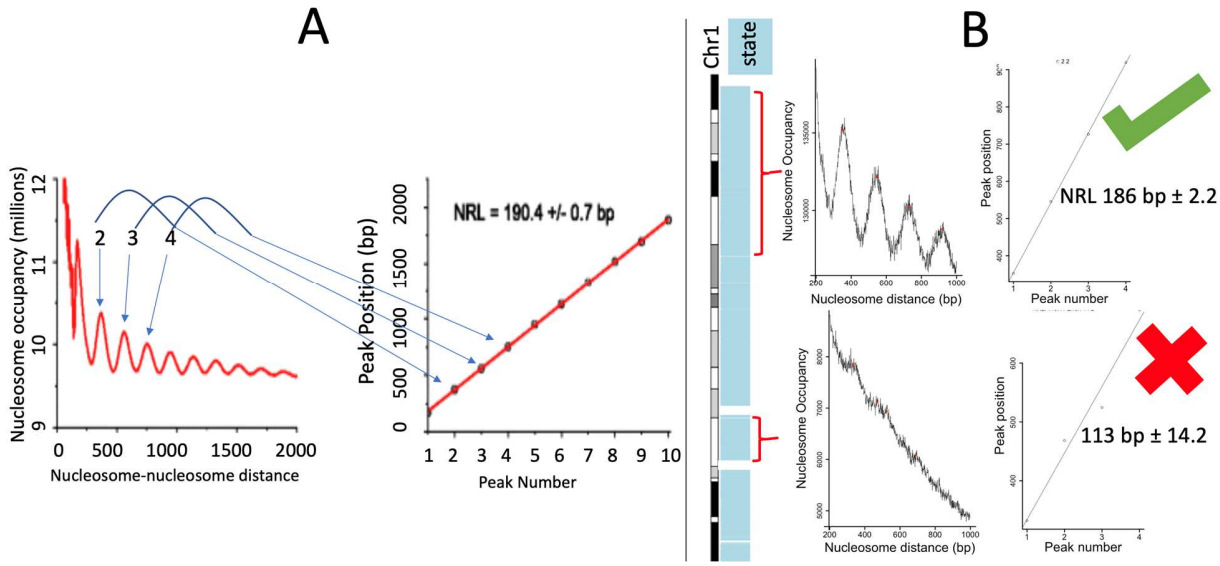


Figure 4. Calculating the NRL can be done with large portions of chromatin but is not feasible when using smaller segments. (A) A typical NRL calculation is done where a histogram documents the inter-nucleosome distances and the resulting peaks are plotted as a function of themselves. The slope of the resulting best-fit line is the NRL. (B) Illustrates how taking a relatively large portion of chromatin results in a viable calculation of the NRL with low error while scaling down to a lower resolution results in a non-viable calculation with high error.

As it is seen from Figure 4, in order to calculate reliably the NRL of a set of regions of interest, there must be an adequate number of loci to average across in order for the statistics to be viable. Furthermore the regions being studied should be > 500 bp as only ~2.5-3 nucleosomes can fit inside such a length of chromatin. Hence as one lowers resolution the viability of calculating the NRL declines. Thus the problem of linking local organisation of nucleosomes to higher order chromatin organisation is one that is worth investigating. Given the above- how then, can one find a relationship between local nucleosome organisation and higher order chromatin architecture? The following sections will outline the various other levels of chromatin architecture.

1.3.2 Chromatin states

Beyond the classical view of chromatin having two states (heterochromatin and euchromatin), more distinct types of chromatin have been identified in recent publications (Ernst and Kellis 2012; Roadmap Epigenomics Consortium *et al.* 2015). These newfound classifications not only refer to chromatin in terms of transcriptional activity but also the general functional role of said piece of chromatin. These classifications are based on what is known as 'chromatin state'. DNA methylation, histone modifications, TF binding and chromatin remodellers all together give rise to the structural and functional state of a piece of chromatin. Thus the term "chromatin state" refers to the combined effects of the chemical chromatin modifications and molecular composition at each given locus (Allis and Jenuwein 2016; Cavalli and Heard 2019).

Machine learning allows for the categorisation of chromatin into different states based on histone modifications and protein composition in order to study how the different regions spatially interact with one another (Ernst and Kellis 2012; Di Pierro *et al.* 2017). For example, the software 'ChromHMM' takes the location of a host of input features (including histone marks, metnylation and TF binding sites) for a set of regions and an arbitrary number of clusters that it must fit to the given input. The input data are binarized and a hidden markov model (HMM) is used to parse out the most prevalent combinations of marks present across the input loci. The resulting clusters can then be assigned different names based on putative functions inferred from the marks that are present at each of them. The software also returns a pairwise metric of how likely one state/ pre-assigned cluster is to transition into another (Ernst and Kellis 2012). Such studies while also taking into consideration the chromatin dynamics. An interesting study of this type was done where the software known as 'MCORE', which takes all of the reads of each studied histone mark into

account and creates a correlation profile plot between replicates, which observes the correlation value between the occupancies of each replicate as a function of distance along the genome. The profile that results yields information about the regions where marks are prevalent and the typical sizes of these domains. This made it possible to study the spreading of different states throughout differentiation (Molitor *et al.* 2017).

1.3.3 Large-scale chromatin organisation

3D chromatin organisation adds another layer of gene regulation. In particular, it is important to understand how chromatin folds and what brings together genomic loci that are far away along the 1D genomic coordinate. For example, gene promoters and enhancers can come together, modulating the expression of the gene in question. Many experimental techniques exist to map the locations of contacts between chromatin fibers. The most popular of these techniques is Hi-C, an experimental technology that allows genome-wide mapping of chromosomal contacts (Lieberman-Aiden *et al.* 2009). The application of Hi-C and related techinques gave rise to a number of new concepts, will be discussed below.

1.3.3.1 Topologically associated domains

Topologically associated domains (TADs) are genomic regions where the relative frequency of DNA-DNA contacts is increased. The boundaries between many TADs are demarcated by the chromatin protein CTCF. CTCF (CCCCTC binding factor) is a chromatin protein implicated in the facilitation of TAD formation (Hnisz, Day and Young 2016), as well as many other aspects of gene regulation (Phillips and Corces 2009; Herold, Bartkuhn and Renkawitz 2012; Nora *et al.* 2016; Schwarzer *et al.* 2016). A simplistic view of the function of TADs is that that they serve to

position specific genomic elements proximate to one another e.g. allowing enhancer/promoter interactions to regulate specific genes. However, the full extent to which this is true has not yet been resolved, which will be discussed below.

The relationship between gene regulation and TADs is unclear. While Stadhouders et al (2018) demonstrated that TAD formation and reorganisation precedes and/or occurs simultaneously with differential gene-expression during cellular transition, it is not certain that this is causing the changes in gene-expression. Other studies have investigated this – for example, Ghavi-Helm and coauthors demonstrated that mutagenic rearrangement of TAD borders did not affect the bulk of genome wide gene-expression (Ghavi-Helm *et al.* 2019). This suggests that most genes are not reliant on 3D organisation in their ability to be transcribed. Another school of thought is that the gene expression is what drives colocalization of different chromatin loci, and that TADs are a result rather than the cause of appropriate gene expression. This was suggested by the results of Hsieh et al. (2019). However, further investigation has suggested that the reality is intermediate of these two possibilities particularly with regard to cell transitions (discussed further below in '1.3.3.3 What governs the physical co-localisation between regions of the same chromatin state?', '1.3.3.4 Nucleosome clutches' and '1.4.4 Rewiring of 3D topology of the genome').

As to the formation of TADs- nucleosome placement may have a partial role in this. It has been shown experimentally that CTCF and histones compete to bind to the same DNA sites and hence could determine whether a TAD boundary is formed or not (Teif *et al.* 2012).

Another mechanistic example of how DNA sequence/nucleosome packing could affect higher order chromatin structures, is that the composition of local sequence of certain regions will influence the recruitment and positioning of histones and in turn the length of DNA linkers

between nucleosomes. The implication of this phenomenon on higher order chromatin architecture is that it significantly affects the stiffness of a length of chromatin and can affect processes such as looping and potential for TAD formation in the genome (Todolli *et al.* 2016).

1.3.3.2 Chromatin compartmentalisation

Hi-C technology has also made it possible to study the architecture of chromatin on a level above megabases and has led to the discovery of 'chromatin compartments' (Dekker and Mirny 2016). Dividing the genome into equal bins and studying the number of intra-chromosomal contacts, per bin, via principle component analysis (PCA), separates the chromatin into compartments showing a level of organisation that had not been previously seen (Lieberman-Aiden *et al.* 2009). From this analysis, the genome was revealed to be separated into A-compartments which contain active, transcribed genomic elements and B-compartments which contain gene-poor regions and are preferentially located at the nuclear lamina (i.e. periphery of the nucleus- such chromatin is referred to as Lamina Associated Domains (LADs)) (Lieberman-Aiden *et al.* 2009). In the mammalian genome, the size of TADs, generally do not surpass a megabase and hence that of chromatin compartmentalisation (Nora *et al.* 2016). While it was previously thought that TADs are involved in genome compartmentalisation, recent studies performed by Schwarzer et al. (2016) and Nora et al. (2016) showed that the genome-wide disruption of TAD boundaries via CTCF knockout does not change the overarching structure of A and B chromatin compartments. Instead, the results indicated that A and B compartments may be defined by something other than CTCF.

A plausible area to investigate to answer the question of what maintains the compartments upon CTCF removal is the chromatin state of the separate regions of the genome as it may well be causing the regions of a similar state to be attracted to one another (Kubo *et al.* 2017). DNA sequence must also be investigated in the future for involvement in genome compartmentalisation.

This is further discussed the next section.

1.3.3.3 What governs the maintenance of physical co-localisation between regions of the same chromatin state?

Previous studies have suggested that loci of similar chromatin regions in the same state are attracted to one another within the nucleus (Di Pierro *et al.* 2017). This may shed some light on how gene expression is regulated via 3D chromatin architecture. Jenkinson et al. demonstrated that DNA methylation shows stable or unstable levels across different loci in separate single cells, maintained separately within TAD boundaries (Jenkinson *et al.* 2017). This may contribute to both the maintenance of the regulatory function TAD and to its co-localisation with other genomic loci (Di Pierro *et al.* 2017; Jenkinson *et al.* 2017).

An interesting question to ask could be framed as such: what factor(s) in the chromatin governs (1) the 3D topological colocalization of loci (2) the insulation of regions of different chromatin state (3) the maintenance of the expression of genes in said regions?

This phenomenon could be the result of different levels of regulation including the following:

- Histone modifications that differentially decorate separate parts of the genome: Di Pierro et al. (2017) showed that it was possible to reproduce Hi-C conformation maps where states were inferred and imputed in a biophysical a model to predict genome folding. Furthermore it has been shown that protein HP1a promotes spatial segregation of heterochromatin from the rest of the genome (Larson *et al.* 2017).
- Inherent patterns in the DNA: Liu et al. (2018) showed that when dividing the genome into

CGI rich areas (referred to as forests) and CGI poor areas (referred to as prairies) differentially positioned in A/B compartments, on top of having different levels of transcriptional activity (Liu *et al.* 2018).

- Liquid-liquid phase separation (LLPS): Recent advances in experimental technology have shed new light on the self-organising nature of chromatin in maintaining "liquid droplets" which cause different parts of a genome to co-localise in a way that resembles how water and oil do not mix (Gibson *et al.* 2019; Sawyer, Sturgill and Dundr 2019). While the exact nature of LLPS has not been well formalised or demonstrated (Mir *et al.* 2019), progress in recent literature is being made. Of particular note is the work of Liang et al. (2019) and Larson et al. (2017) both of which demonstrated that differential concentration of factors such as HP1 in the nucleus give rise to heterochromatin formation and phase separation (Larson *et al.* 2017; Wang *et al.* 2019).

1.3.3.4 Nucleosome clutches

The majority of experimental systems that study the 3D topology e.g. Hi-C, consider the genome in terms of the contact points between different loci. A different way of considering chromatin 3D topology is to look at it at the resolution of single nucleosomes. Ricci et al. (2015) performed stochastic optical reconstruction microscopy (STORM) to create a genome wide landscape of the 30 nm chromatin fiber. This experiment revealed that the nucleosomes of the genome are organised in what they deem 'clutches'- groups of nucleosome arrays that occur in varying sizes- with distinct regions that are depleted of nucleosomes. This result presents an interesting viewpoint of how chromatin regulation occurs. The structure of nucleosome clutches is illustrated in Figure 5. It shows how nucleosome organisation occurs *in vivo*. Given this, let's think further on how this

molecular structure could not only exist as a 1D sequence (as shown above) but also how the chromatin fiber could fold and bring different clutches into contact with one another and recapitulate the levels of chromatin architecture (discussed above, e.g. TADs, chromatin compartmentalisation etc.).

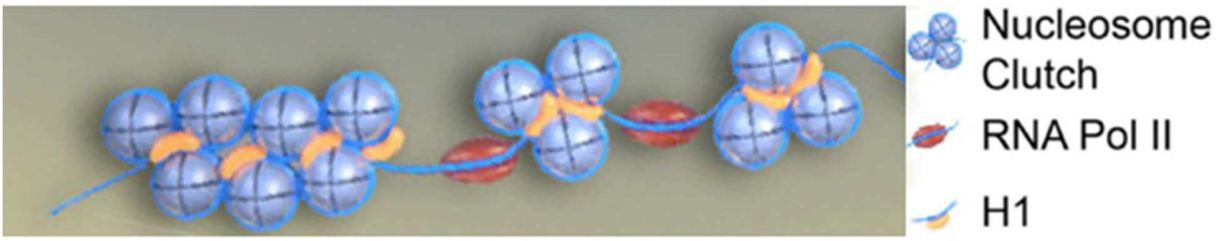


Figure 5. Illustration of nucleosome clutches adapted from (Ricci et al. 2015).

Another recent new technique for the analysis of 3D genome organisation called Micro-C is a 3C technology that documents contacts across the genome between all nucleosomes. Hsieh et al. (2019) performed Micro-C on the mESC genome. This experiment demonstrated that areas of chromatin exist where nucleosome occupied regions have strong contacts, called 'microTADs'. MicroTADs usually reflect key enhancer-promoter interactions. The borders that mark the ends of microTADs are often independent of CTCF and are nucleosome depleted.

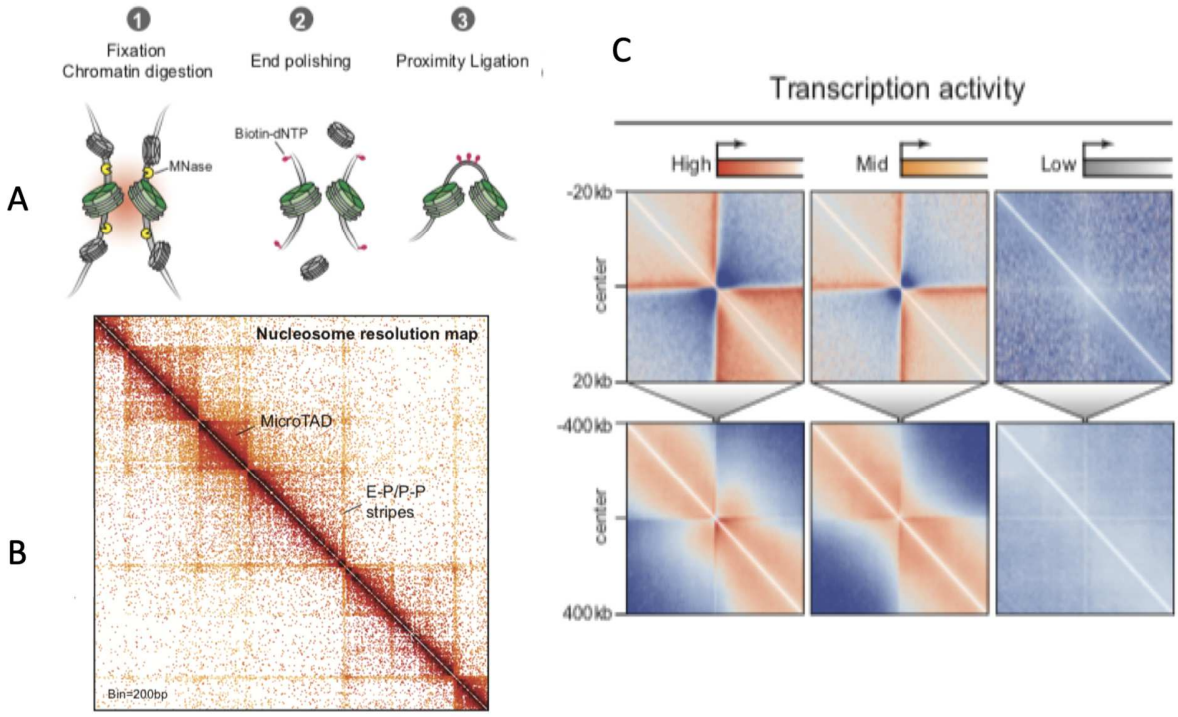


Figure 6. Schematic picture showing concept and results from Micro-C experiment done in the mESC genome. (A) Shows the experimental protocol where contact frequency between nucleosomes is documented. (B) Shows that Micro-C improves the resolution and accuracy with which frequent contact domains are identified (C) Shows that boundaries near genes with the highest levels of transcription have the highest contact frequency. Adapted from (Hsieh *et al.* 2019).

Furthermore the results from this experiment suggested that areas of chromatin where there is strong contact could be due to transcription rather than vice versa. While it is known that CTCF knockout does not significantly affect gene expression (Nora *et al.* 2016), Micro-C reveals that when Pol II transcription initiation was artificially inhibited the nucleosome contacts around enhancer and promoter regions were significantly depleted. This supports the view that gene expression may be a driver of chromatin colocalization (Hsieh *et al.* 2019). Thus, the combination of micro-C and STORM experiments suggests that the following phenomena occur:

- Chromatin contains sparse areas that are nucleosome depleted.
- These nucleosome depleted regions are the “fulcrums” on which the chromatin folds
- Gene expression is what drives colocalization of different chromatin loci between the nucleosome depleted areas.
- Chromosome compartments can be both a result and a consequence of gene expression.

Considering the above points, the following question arises: what maintains the boundaries between different state regions in the genome? These regions are usually depleted of nucleosomes, but is this due to something inherent in the DNA sequence or is it the competitive binding of transcription factors? Indeed in the above mentioned study TFs are shown to be enriched at the found microTAD borders. The results of Stadhouders et al. (2018) support this finding further in showing that TFs are not only enriched at TAD borders but also are influential in TAD reorganisation. Or could it be that there is a natural depletion of nucleosomes at these borders which provides an opening for TF binding? If so, then the sequence of these regions is a likely candidate that causes the depletion of nucleosomes at chromatin boundaries. This is discussed further in the next section '1.3.3.5 Chromatin boundaries’.

1.3.3.5 Chromatin boundaries

A chromatin boundary can be loosely defined as a locus where there is a sharp change in the state of a region of chromatin (‘state’ here refers to the histone mark profiles). These borders can be either ‘fixed’ or ‘negotiable’ (Wang *et al.* 2014). The set of rules that dictate the formation of a boundary is not clear. Previous studies have suggested that CTCF binding sites are boundaries that maintain and insulate loci of different states from one another (Herold, Bartkuhn and Renkawitz

2012). Figure 7 summarises two other types of experiments that show the implication of CTCF in chromatin boundary formation.

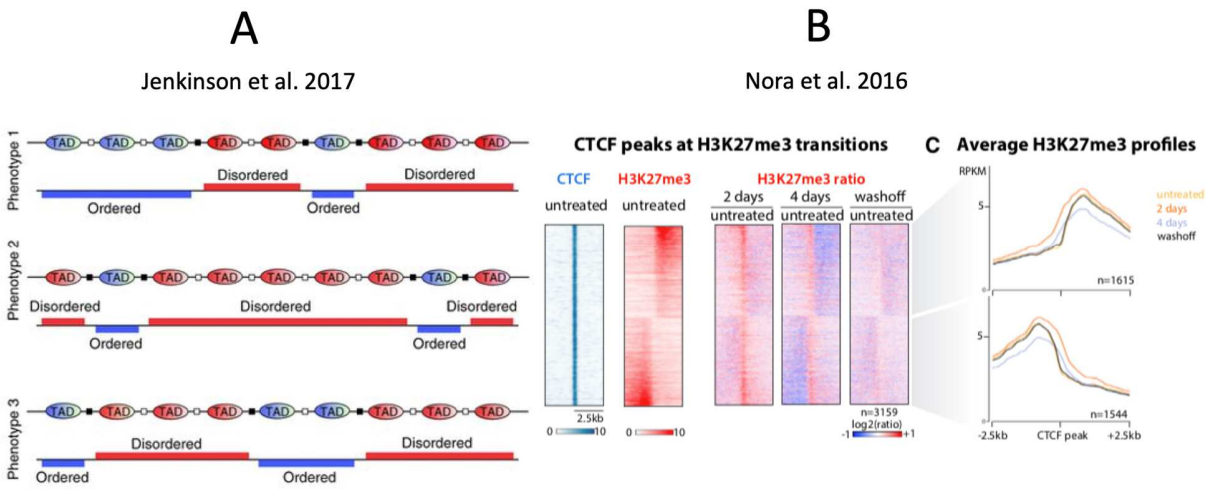


Figure 7. Illustration of the potential role of CTCF in maintaining TAD boundaries. A) Differential methylation landscapes are maintained between TAD borders (Jenkinson et al. (2017). B) The effect of CTCF knock out on the genome-wide H3K27me3 levels (Nora et al. 2016).

Jenkinson et al. (2017) (Figure 7A) showed that TAD boundaries coincided with state transitions in terms of methylation levels of different loci (Jenkinson *et al.* 2017). However, in the experiment done by Nora et al. (2016) where CTCF was knocked out genome wide, the repressive mark H3K27me3 did not show spreading beyond the lost CTCF binding sites (Figure 7B). Hence the following questions arise:

- Does CTCF function play a role in the insulation of chromatin loci with different states and if so, to what extent?

- Is this effect locus specific, i.e. different for all CTCF binding sites?
- If CTCF removal does not prevent the spreading of differential chromatin states, could there be something inherent in the sequence that is acting as the boundary in its absence?

1.3.3.6 Overarching questions about chromatin architecture and regulation

As demonstrated above, the past studies of chromatin architecture and function have been very extensive, involving many different aspects. There are still many gaps in our understanding. Considering all of the above (section "4. The various levels of chromatin architecture at different resolutions and the relationships between them", the following questions arise:

- What is the relationship between local nucleosome organisation and higher order structures, e.g. chromatin states?
- What are the criteria for the formation of a chromatin boundary?
- Are these boundaries a result of the DNA sequence or TF binding?
- Is nucleosome depletion necessary for boundary formation?
- What role does CTCF have in these phenomena – both local nucleosome organisation and subsequent higher order architecture?

A final question that is worth asking with all of the above in mind is: how does this affect cellular transitions such as differentiation and cancer development? These themes will be reviewed in the following section.

1.4 Chromatin changes during cell differentiation and other cell transitions

The above sections discussed various levels of genomic regulation. This section will discuss how regulatory processes are involved in cellular differentiation. As an embryonic cell differentiates, changes occur in the structure of chromatin. This includes the redistribution of nucleosome positions, TF binding positions, histone marks and changes of 3D genome organisation.

1.4.1 Nucleosome positioning

As mentioned in the Introduction, a nucleosome positioned on a piece of DNA makes it energetically unfavourable for the covered region to be transcribed. This is important for gene regulation. Teif et al. (2012) showed that, at the transcription start sites (TSSs) and transcription termination sites (TTSs) have distinct nucleosome profiles depending on the level of transcription. These experiments were done in mESCs, NPCs and MEFs for the purpose of telling if this relationship changed between early and late cell life (Teif *et al.* 2012).

1.4.2 Transcription factor binding

The competitive binding of TFs and histones to DNA was discussed in section 1.2.3 above. The consequence of this phenomenon is that nucleosomes that cover an important regulatory loci in early cell life could be displaced by TFs to initiate cellular pathways allowing for differentiation. Teif et al. (2012) studied the profiles of nucleosomes around TF binding sites in mESCs, NPCs and MEFs. In doing so they were able to classify TFs into 4 different classes:

- TFs that were in NDRs in ESCs and remained in NDRs throughout differentiation.
- TFs that were in NDRs in ESCs but became preferentially occupied by nucleosomes after differentiation.

- TFs that were in regions that were generally depleted of nucleosomes but showed a small peak at the center where the TF was bound and showed a further increase in nucleosome occupancy upon differentiation.
- TFs that position themselves on top of nucleosomes rather than spatially occluding them (TFs such as Sox2, Nanog and Oct4). These are known as "pioneer TFs" that can initiate differentiation pathways. Pioneer TFs are capable of binding chromatin that is bound by nucleosomes and recruit other TFs to trigger a cascade of differential pathways (Zaret and Carroll 2011; Meers, Janssens and Henikoff 2019).

From the above studies, CTCF stood out as an architectural protein that spatially excludes nucleosomes. Furthermore nucleosomes were regularly positioned around CTCF binding sites. This effect was preserved throughout differentiation but only at places where CTCF remained bound. CTCF sites that were lost upon differentiation became occupied by a nucleosome but the regular phased organisation of nucleosomes either side of the binding site was lost (Teif *et al.* 2012).

With regard to higher-order chromatin architecture, experiments done by Stadhouders et al. (2018) allowed to visualise the wiring of the genome in B-cells at various timepoints while the cells transitioned to induced pluripotency. These data were then used to show that clusters of TF binding sites are present at TAD borders and compartment borders and are important for the formation of new TADs during the cell transition (Stadhouders *et al.* 2018). Furthermore, Avsec et al. (2019) demonstrated patterns in the spacing between TF binding sites at enhancer DNA to facilitate enhancer activation and subsequent gene regulation (Avsec *et al.* 2019).

1.4.3 Formation of heterochromatin domains

Heterochromatin regions can expand throughout the process of differentiation. This process is important during mammalian cell differentiation as a repressive mark such as H3K9me3 can spread to cover genes with developmental functionality (Wang *et al.* 2014). The mechanism by which heterochromatin spreads is an area of intense investigation. A common mechanism may not exist and can depend on the type of heterochromatin mark that is in question (Wang *et al.* 2014). In the case of H3K9me3-dependent heterochromatin, a main molecular player is heterochromatin protein 1 (HP1). In the experiments of Hathaway et al (2012), HP1 was selectively recruited to cause the spreading of H3K9me3, the kinetic effects were observed and this allowed to study the rate at which heterochromatin spreads in different regions. These could be classified into 2 categories, based on the width that the spreading effect occurred, as 'large' and 'small'. Furthermore, after being artificially established, these epigenetic landscapes were inherited through several generations of mice (Hathaway *et al.* 2012). Larson et al. (2017) showed that HP1a promotes spatial segregation of the genome and this gives rise to heterochromatin formation. They also showed that nucleosomes preferentially partition within the segregated loci where HP1a was enriched, but molecules associated with active chromatin showed no preference with regard to these regions (Larson *et al.* 2017). These results are further supported by Wang et al. (2019) who showed that HP1 and histone methyltransferase Suv39H1 interact with H3K9me2/3 marks to promote phase separation and induce heterochromatin formation (Wang *et al.* 2019).

Thus heterochromatin forms in particular domains and co-localises with other heterochromatin while euchromatin colocalises with other active regions. The full extent to which this phenomenon is involved in 3D genome organisation is still under investigation (see above "4.3.3 What governs the maintenance of separation between regions of different chromatin state?").

1.4.4 Rewiring of 3D topology of the genome

The idea that TADs bring enhancers and promoters together as a means of switching genes on and off has been a topic of much debate recently. While Ghavi-Helm et al. (2019) showed that mutation of TADs did not affect the majority of gene expression, Stadhouders et al. (2018) showed that, throughout differentiation, gene expression never precedes topological rearrangement- the two phenomena either occur simultaneously (in the case of 56% of TADs) or gene expression changes after rearrangement (in the case of 44% of TADs)- which suggests genomic regulation via TADs (Stadhouders *et al.* 2018). Another possibility is one that is intermediate of the 2 above schools of thought, namely that TAD organisation and topological rewiring is essential for a relatively distinct subset of genes with regard that are uniquely required for an individual cell-type. For example, it was shown in the case of haematopoietic stem and progenitor cells, knockout of cohesin results in aberrant activation of inflammatory pathways and a reduced ability to differentiate (Cuartero *et al.* 2018). Hence the full extent to which TADs are involved in chromatin regulation and differentiation across all cell types is still under investigation.

1.4.5 Changes of chromatin state/boundaries/conformation during cell differentiation

The process of the change of 3D chromatin conformation is an important aspect of cell differentiation. Figure 8 illustrates two recent important studies of chromatin conformational change during cell transitions performed by the lab of Marc Marti-Renom (Stadhouders *et al.* 2018; Vilarrasa-Blasi *et al.* 2019). Both of these analyses were based on Hi-C data, investigating the A-B chromatin transition. Panel A shows the change in composition of A and B comparments at important regulatory regions during cell induced pluripotency (which is so to speak the reverse of

differentiation). Panel B, in addition, demonstrates how the A/B chromatin distribution changes in cancer.

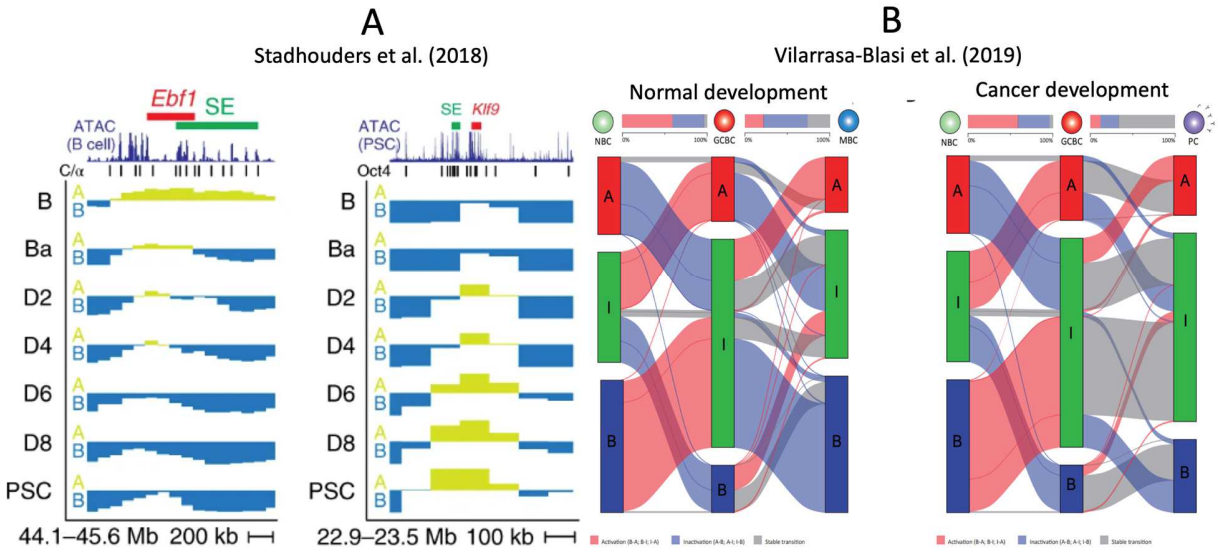


Figure 8. Illustration of the change that happens across the genome in terms of transition between A and B chromatin during cell-lineage development. A) The metric for A vs B chromatin around example genomic regions in mESCs, demonstrating how two different genes are progressively activated/inactivated as a cell is induced into pluripotency. Blue corresponds to low density (compartment B) and green to high density of contacts (compartment A). Each row corresponds to a different stage of differentiation (adapted from Stadhouders et al. (2018)). B) Data from Vilarrasa-Blasi et al. (2019), showing the number of regions that transition from compartment in state A to I (intermediate of A and B) to B in the case of normal and cancer development.

On top of the studies of the 3D architecture, Vilarassa-Blasi et al. (2019) (Figure 8B) also investigated the change in ChromHMM-configured states during cell development from a Naïve B-cell to a germinal center B-cell (GCBC). In this study they allowed for a third category to exist between A and B chromatin, referred to as 'intermediate' (I). Across development they monitored

how the coverage of chromatin state of the regions that transitioned from intermediate to A and B. These investigations are schematically depicted below in Figure 9. Specifically in panel B, the authors were studying these effects in a candidate region containing genes that had been previously implicated in cancer. This was done in an effort to show how both the chromatin state and topological conformation change in this region between cancer vs healthy conditions. These analyses do not suggest whether chromatin state (in the form of heterochromatin spreading) or 3D rewiring of the genome is the driver of the change in cell type. From the above, the following questions stand out: are boundaries that maintain the state of a locus being changed/corrupted during the process of differentiation? Also, does heterochromatin spread/contract in conjunction with this demonstrated rewiring?

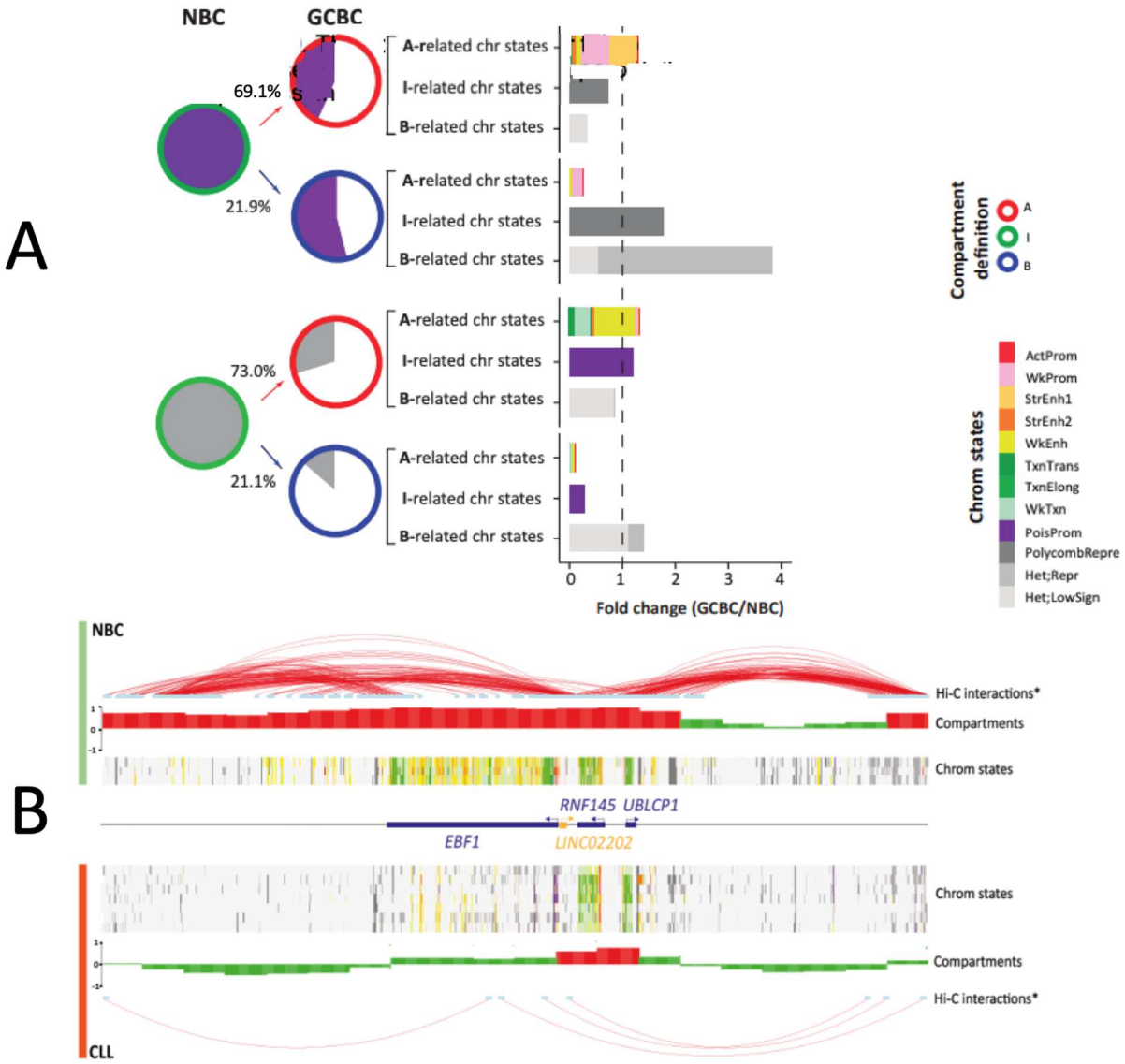


Figure 9. Chromatin state changes during development. (A) The percentage coverage of different chromatin state regions changes while transitioning to A or B chromatin from intermediate chromatin. (B) Hi-C contact map for an example region comparing B-cells in shows that in conjunction with TAD rewiring the state of chromatin changes going from a Naïve B-cell to a cancerous one. Adapted from (Vilarrasa-Blasi *et al.* 2019).

1.5 Aims

As demonstrated above, the chromatin research field is already very advanced, but several challenges still represent a bottleneck for connecting the different levels genome organisation. In particular, the use of the classical NRL calculation technique is limited to large genomic regions (or a large number of smaller regions). There is no method that can connect 1D and 3D chromatin organisation for an arbitrary size of a genomic region. It is also still not clear how CTCF organises nucleosomes, and how it defines chromatin boundaries and 3D genome topology. In line with these open challenges, I sought to answer the following questions:

- Can nucleosome positioning be used to predict higher order state such as chromatin state?
- How does the competitive binding between TFs and histones to DNA affect nucleosome array organisation?
- What is the function of these phenomena in the context of cellular transitions (either cancer development or cellular differentiation)?

To address these questions, I aim to develop a computational description of the following:

1) The relationship between CTCF-dependent chromatin boundaries and nucleosome organisation, as well as the role of this effect in cell differentiation.
2) The prediction of the chromatin state based on nucleosome positioning, and the framework of applying this method to cancer diagnostics.

In line with these aims, in the next chapters I show the following:

- In Chapter 2 I investigate the relationship between the strength of CTCF binding and nucleosome organisation near its binding site. We report a new effect: the strength of CTCF binding to DNA is inversely proportional to the average distance between nucleosomes

near its binding site. We also assessed the effect of the DNA sequence surrounding the CTCF motif on the nucleosome organisation and showed that, contrary to previous assumptions, nucleosomes form asymmetric arrays near CTCF. We also found that CTCF binding sites occur in clusters around TAD boundaries. We assessed how these effects modulate 3D genome architecture in terms of chromatin boundary formation and maintenance and how this can affect the process of differentiation.

- Chapter 3 documents my work on the connection between nucleosome positioning and the chromatin state using machine learning. I studied the relationship between the chromatin state of a locus and the constituent nucleosome positioning patterns. We used machine learning to classify segments of DNA, labelled by their chromatin state, using nucleosome positioning as input. Then I applied this technique as a diagnostic tool to nucleosome positioning data from cancer patients with chronic lymphocytic leukemia (CLL) and healthy people.

CHAPTER 2. CTCF-dependent chromatin boundaries formed by asymmetric nucleosome arrays with decreased linker length

2.1 Abstract

This chapter is based in the paper (Clarkson *et al.* 2019). All supplementary figures are in Appendix 5.1. The CCCTC-binding factor (CTCF) organises the genome in 3D through DNA loops and in 1D by setting boundaries isolating different chromatin states, but these processes are not well understood. Here I investigate chromatin boundaries in mouse embryonic stem cells, defined by the regions with decreased Nucleosome Repeat Length (NRL) for ~20 nucleosomes near CTCF sites, affecting up to 10% of the genome. I found that the nucleosome-depleted region (NDR) near CTCF is asymmetrically located >40 nucleotides 5'-upstream from the center of CTCF motif. The strength of CTCF binding to DNA and the presence of cohesin is correlated with the decrease of NRL near CTCF, and anti-correlated with the level of asymmetry of the nucleosome array. Individual chromatin remodellers have different contributions, with Snf2h having the strongest effect on the NRL decrease near CTCF and Chd4 playing a major role in the symmetry breaking. Upon differentiation, a subset of preserved, common CTCF sites maintains asymmetric nucleosome pattern and small NRL. The sites which lost CTCF upon differentiation are characterised by nucleosome rearrangement 3'-downstream, with unchanged NDR 5'-upstream of CTCF motifs. Boundaries of topologically associated chromatin domains frequently contain several inward-oriented CTCF motifs whose effects, described above, add up synergistically.

2.2 Introduction

Nucleosomes are positioned along the genome in a non-random way (Lai and Pugh 2017; Baldi 2019; Teif and Clarkson 2019), which is critical for determining the DNA accessibility and genome organisation (Maeshima, Ide and Babokhov 2019). A particularly important nucleosome positioning signal is provided by CTCF, an architectural protein that maintains 3D genome architecture (Merkenschlager and Nora 2016; Nora *et al.* 2016; Rao *et al.* 2017) and can organise up to 20 nucleosomes in its vicinity (Fu *et al.* 2008; Dunham *et al.* 2012; Teif *et al.* 2014; Wiehle *et al.* 2019) (Figure 1A). Most other TFs do not possess such nucleosome-organising potential (Figure S1). CTCF has ~100,000 potential binding sites in the mouse genome. Usually there are ~30,000-60,000 CTCF sites bound in a given cell type, which translates to about 1 million of affected nucleosomes (up to 10% of the mouse genome) (Chen *et al.* 2012a; Wang *et al.* 2012; Yue *et al.* 2014; Wiehle *et al.* 2019). CTCF is able to act as an insulator between genomic regions with different chromatin states, but how exactly this is achieved is not known. Here I explore molecular mechanisms of the insulator boundary formation by CTCF through rearrangement of surrounding nucleosome arrays.

One of the ways to characterise genomic nucleosome distribution is through an integral parameter called the nucleosome repeat length (NRL), defined as the average distance between the centers of adjacent nucleosomes. NRL can be defined genome-wide, locally for an individual genomic region or for a set of regions. The local NRL is particularly important, since it reflects different structures of chromatin fibers (Routh, Sandin and Rhodes 2008; Bascom, Kim and Schlick 2017; Nikitina *et al.* 2017; Risca *et al.* 2017; Bass *et al.* 2019). Ever since the discovery of the nucleosome (Kornberg 1974; Olins and Olins 1974) there have been many attempts to compare NRLs of different genomic regions (Lohr, Tatchell and Van Holde 1977; Gottesfeld and Melton

1978; De Ambrosis *et al.* 1987) and it has been established that genome-wide NRL changes during cell differentiation (Weintraub 1978; Bradbury 1989). Recent sequencing-based investigations showed that active regions such as promoters, enhancers and actively transcribed genes usually have shorter NRLs and heterochromatin is characterised by longer NRLs (Sun, Cuaycong and Elgin 2001; Liu *et al.* 2015; Baldi *et al.* 2018; Chereji *et al.* 2018). While in yeast it is possible to link NRL changes to the action of individual chromatin remodellers (Zhang *et al.* 2011; Hennig *et al.* 2012; Möbius *et al.* 2013; Ocampo *et al.* 2016; Kubik *et al.* 2019), in mouse or human regulatory regions are very heterogeneous and it is difficult to come up with a set of definitive remodeller rules determining their effect on NRL (De Dieuleveult *et al.* 2016; Giles *et al.* 2019).

We previously showed that in mouse embryonic stem cells (ESC), NRL near CTCF is about 10 bp smaller than genome-wide NRL (Teif *et al.* 2012, 2014). Our analysis demonstrated that purely statistical positioning of nucleosomes near CTCF boundaries would not be enough to explain genome-wide NRL shortening near bound CTCF observed experimentally; also, the effects of strong nucleosome-positioning DNA sequences, while compatible with the observed NRL, are limited to a small number of CTCF sites (Beshnova *et al.* 2014). A very recent study has investigated the effect of Snf2 and Brg1 remodellers on NRL in ESCs, suggesting Snf2 as the primary player (Barisic *et al.* 2019). However, other factors may be at play as well. Thus, it is still unclear what determines the NRL near CTCF and how different CTCF sites are distinguished from each other e.g. during cell differentiation. Furthermore, recent studies have shown that CTCF can act as a boundary element between different chromatin states (e.g. DNA methylation) linearly spreading along the genome (Jenkinson *et al.* 2017; Wiehle *et al.* 2019), but the mechanistic explanation for such a function is not immediately clear from the better established role of CTCF

in 3D chromatin looping. Here I address these problems using available experimental datasets in ESCs and their differentiated counterparts.

I show below that the boundaries of nucleosome arrays are encoded in extended DNA regions >200 bp long enclosing individual CTCF motifs. Furthermore, the strength of CTCF binding provides a single "code" that determines the value of NRL near CTCF, the level of asymmetry of CTCF-dependent nucleosome array boundaries, and eventually serves as a guide for larger-scale chromatin rearrangements during cell differentiation.

2.3 Materials and Methods

2.3.1 Experimental datasets

Nucleosome positioning, transcription factor and chromatin remodeller binding datasets were obtained from the Gene Expression Omnibus (GEO), Short Read Archive (SRA) and the ENCODE web site as detailed in Table ST1. NRL calculations near CTCF in ESCs were performed using the MNase-seq dataset from (Voong *et al.* 2016). NRL calculations near 18 stemness-related proteins in ESCs shown in Figure 1C and S3 were performed using the chemical mapping dataset from (Voong *et al.* 2016). NRL calculations in NPCs and MEFs were based on the MNase-seq datasets from (Teif *et al.* 2012). MNase-assisted H3 ChIP-seq from (Teif *et al.* 2014) was used for demonstrative purposes in the phasogram calculation in Figure 1B and aggregate profiles in Figure S9. A more detailed list of datasets used in each figure is provided in Table ST1. Coordinates of genomic features and experimental maps of transcription factor and remodeller binding in ESCs were obtained from published sources as detailed in Table ST1. The coordinates of loops described in (Bonev *et al.* 2017) were kindly provided by the authors in a BED file aligned to the mm10 mouse genome and converted to mm9 using liftOver (UCSC Genome Browser).

2.3.1.1 Data processing

For nucleosome positioning, raw sequencing data were aligned to the mouse mm9 genome using Bowtie allowing up to 2 mismatches. For all other datasets I used processed files with genomic coordinates downloaded from the corresponding database as detailed in Table ST1. Where required, coordinates were converted from mm10 to mm9 since the majority of the datasets were in mm9. TF binding-sites were extended from the center of the site to the region [100, 2000]. In order to find all nucleosomal DNA fragments inside each genomic region of interest, the bed files containing the coordinates of nucleosomes processed using the NucTools pipeline (Vainshtein, Rippe and Teif 2017) were intersected with the corresponding genomic regions of interest using BedTools (Quinlan 2014).

2.3.2 Binding site prediction

Computationally predicted TF binding sites were determined via scanning the mouse mm9 genome with position frequency matrices (PFMs) from the JASPAR2018 database (Khan *et al.* 2018) using R packages TFBSTools (Tan and Lenhard 2016) and GenomicRanges (Lawrence *et al.* 2013). A similarity threshold of 80% was used for all TFs in order to get at least several thousand putative binding sites. In the case of MYC, I used matrix MA0059.1 defined in Homo sapiens, since its matrix MA0147.2 defined in Mus musculus returned a significantly smaller number of sites. For all other TFs I used default JASPAR matrices provided for Mus musculus.

2.3.3 Separation into forward and backward facing CTCF motifs

I used TFBSTools (Tan and Lenhard 2016) to search on the 5'-3' prime strand for forward facing CTCF motifs using the JASPAR matrix MA0139.1 and the 3'-5' strand for motifs that are backwards facing ones. An alternative calculation using RSAT (Castro-Mondragon *et al.* 2017) with the same matrix led to similar results.

2.3.4 Calculation of aggregate nucleosome profiles

Aggregate nucleosome profiles were calculated using NucTools with single-base pair resolution (Vainshtein, Rippe and Teif 2017). The calculation taking into account CTCF motif directionality was done as follows: in the case, if the motif is on the plus strand, the region [-1000, 1000] near CTCF also starts left to right, whereas for the minus strand the position of the region was mirrored with respect to the middle of the CTCF site.

2.3.5 Stratification of TF-DNA binding affinity

In the case of experimentally determined binding sites of CTCF, I stratified 33,880 sites, reported by the mouse ENCODE consortium (Yue *et al.* 2014), into five equally sized quintiles according to their ChIP-seq peak height reported in the original publication. In the case of computationally predicted TF sites, I started with 111,480 sites found by scanning the mouse genome with TFBStools using JASPAR matrix MA0139.1 with 80% similarity threshold, and split them into five equal quintiles based on their TRAP score (Roider *et al.* 2007). The TRAP score is proportional to the binding probability of CTCF for a given site. In order to calculate the TRAP score I extended CTCF motifs by 30 nucleotides in both directions and used tRap implementation

of the TRAP algorithm in R with default parameters (https://github.com/matthuska/tRap). In the calculations involving CTCF motif directionality (Figures 5-7) I first arranged predicted sites by the TRAP score into quintiles, and after that intersected them with the experimental ChIP-seq peaks of CTCF. Only motifs overlapping with sites that were experimentally detected by ChIP-seq in at least one mouse cell type were retained (including datasets from ENCODE (Yue *et al.* 2014), GSE27944 (Martin *et al.* 2011), GSE96107 (Bonev *et al.* 2017), GSE114599 (Wiehle *et al.* 2019)), and these were further filtered to exclude CTCF sites separated by less than 1000 bp from annotated transcription start sites (TSSs), which removed about 10% of CTCF sites. TSSs were taken from the Genomatics Eldorado database (Genomatix GmbH), which were defined using the following criteria: taken as the 5' ends of the cDNA from the tags of CAGE experiments. After these filtering steps I obtained the following numbers of sites in the binding strength quintiles Q1 to Q5: 3,596 (Q1); 3,782 (Q2); 6,776 (Q3); 14,776 (Q4); 16,860 (Q5).

2.3.6 Phasogram calculation

The "phasograms" representing the histograms of dyad-to-dyad or start-to-start distances were calculated with NucTools. When paired-end MNase-seq was used, dyad-to-dyad distances were calculated using the center of each read as described previously (Vainshtein, Rippe and Teif 2017). When chemical mapping data was used, this procedure was modified to use the start-to-start distances instead, because in the chemical mapping method the DNA cuts happen at the dyad nucleosome locations. The phasogram was then used for the NRL calculation as explained in Figure 1B. The NRL is defined by the slope of the line connecting the phasogram peaks; this line is determined by linear fitting, taking into account only the phasograms where ANOVA P-value for the slope determination is below 0.05.

2.3.7 Selection of the location of the region near CTCF for NRL calculations

We noticed that NRL near CTCF depends critically on the distance of the region of NRL calculation to the binding site summit (Figure S2). While the phasograms for regions [100, 2000] and [250, 1000] near the summits of the experimental CTCF sites, which both exclude the CTCF site, are quite similar to each other, a region that includes the CTCF peak summit [-500, 500] is characterised by a very different phasogram. However, the latter phasogram is an artefact of the effect of the interference of two "waves" of distances between nucleosomes: one wave corresponds to the distances between nucleosomes located on the same side of CTCF, and the second wave corresponds to distances between nucleosomes located on different sides from CTCF. The superposition of these two waves results in the appearance of additional peaks (Figure S2A). A linear fit through all the peaks given by the interference of these two waves gives NRL=155 bp, but this value does not reflect the real prevalent distance between nucleosomes (Figure S2B). We thus selected the region [100, 2000] for the following calculations. Below, all NRLs refer to regions [100, 2000] near the summits of TF binding sites, unless specified otherwise. We would like to note that the effect explained above means that some of the previous publications reporting NRL near CTCF may need to be re-evaluated, because the summit of CTCF site needs to be always excluded from the genomic region for robust NRL calculations; otherwise, the apparent NRL is unrealistically small. I checked that this artefact at least does not affect NRL calculations near TSS (Figure S2C). Once the region location with respect to the CTCF site is fixed, the phasograms are not significantly affected by the choice of the nucleosome positioning dataset (Figure S2D). In the following calculations in ESCs I used the high-coverage MNase-seq and chemical mapping datasets from (Voong *et al.* 2016).

2.3.8 Automated NRL determination from phasograms

Studying many phasograms proved cumbersome when manually picking the peak locations in a non-automated way. To circumvent this problem, I developed an interactive applet called NRLcalc based on the Shiny R framework (http://shiny.rstudio.com), to allow one to interactively annotate each phasogram such that the NRL could be calculated conveniently. NRLcalc allows one to select a smoothing window size to minimise noise in the phasograms. A smoothing window of 20 bp was used in our calculations. The applet also provides the Next and Back button to allow the user to go through many phasograms, as well as intuitive user interface to load and save data.

To account for the fact that NRL was determined by manual point picking and hence could be subject to human error, a subset of the calculated profiles were used to verify that my own point picking and that of Vlad's were consistent. If the last peak in a profile was not particularly strong, the decision to include it in the calculation or not was based on whether it significantly changed the result from that of the stronger points (Figure S19).

2.3.8.1 Analysis of RNA expression near CTCF

RNA-seq data was downloaded from the GEO GSE98671 (Nora *et al.* 2016) and mapped with TopHat (Trapnell *et al.* 2012) to the mm9 genome. The mapped BAM files were converted to BED format with BEDOPS (Neph *et al.* 2016). The numbers of RNA reads aligning 1,000 bp up- and downstream of CTCF motifs were calculated using BedTools (Quinlan 2014), requiring at least 1bp intersection. The effect was tested for while including and excluding all Ensembl genes.

2.3.9 TAD analysis

TAD coordinates in ESCs and NPCs reported by Bonev et al. (Bonev *et al.* 2017) for the mm10 genome were converted to mm9 using liftOver. TADs defined as common, lost and gained upon ESC to NPC transition were determined using BedTools' command intersect with parameter -wc. TADs with the rate of overlap between ESCs and NPCs >90% were considered common; those appearing in ESCs and NPCs with an overlap rate <90% were defined as lost and gained correspondingly. The aggregate profiles of CTCF motifs around TAD boundaries were calculated with HOMER (Heinz *et al.* 2010) at a bin resolution of 5000 bp.

Our software *NRLcalc* is available at https://github.com/chrisclarkson/NRLcalc

2.4 Results

2.4.1 Setup of NRL calculations

Let us base our NRL calculations on the "phasogram" algorithm introduced previously (Liu *et al.* 2015; Vainshtein, Rippe and Teif 2017). The idea of this method is to consider all mapped nucleosome reads within the genomic region of interest and calculate the distribution of the frequencies of distances between nucleosome dyads. This distribution typically shows peaks corresponding to the prevalent distance between two nearest neighbour nucleosomes followed by the distances between next neighbours. The slope of the line resulting from the linear fit of the positions of the peaks then gives the NRL (Figure 10B). To perform bulk calculations of NRLs for many genomic subsets of interest I developed software NRLcalc, which loads the phasograms computed in NucTools (Vainshtein, Rippe and Teif 2017) and performs linear fitting to calculate the NRL (see Methods).

2.4.2 Each TF is characterised by a unique NRL distribution near its binding sites

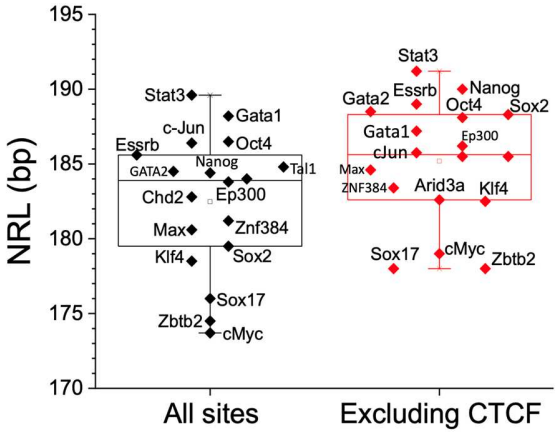
For example, I used a recently reported chemical nucleosome mapping dataset (Voong *et al.* 2016) to calculate NRLs in the region of up to 2000bp from the center of the binding site excluding the central 100 bp (hereafter referred to as region [100, 2000]) for 18 stemness-related TFs whose binding has been experimentally determined in ESCs using ChIP-seq (Figure 10C). This analysis revealed that the proximity to CTCF binding sites unanimously reduced the NRL near these sites. When I filtered out TF binding sites that overlap with CTCF binding sites in ESCs, the NRLs for each individual TF increased (Figure 10C). On the other hand, TF binding sites that overlap with CTCF had significantly smaller NRLs (Figure S3).

2.4.3 The strength of CTCF binding correlates with NRL decrease in the adjacent region

To dig deeper into the relationship between CTCF and local chromatin conformation, I split CTCF sites into 5 quintiles of increasing binding strength. Two metrics were used as a means of quantifying CTCF binding strength: i) Experimentally determined CTCF binding sites in ESCs were split into 5 quintiles based on the height of the ChIP-seq peaks reported by the mouse ENCODE consortium (Yue *et al.* 2014). ii) Theoretically predicted binding sites defined by scanning the mouse genome using TFBStools (Tan and Lenhard 2016) with the 19-bp CTCF motif (JASPAR MA0139.1) (Khan *et al.* 2018) were split into 5 quintiles based on their calculated TRAP score that is proportional to the probability of CTCF binding to a given site (50) (Roider *et al.* 2007) (see Methods). In each case, the calculation of the NRL was performed in the region [100, 2000] near CTCF binding sites using MNase-seq data (Voong *et al.* 2016). These calculations revealed a smooth decrease of NRL as the strength of CTCF binding increased in the case of both

used metrics (Figure 10D). In addition, I used the chemical nucleosome mapping dataset (Voong *et al.* 2016) to compare the CTCF quintiles in terms of the distribution of nucleosome dyad-to-dyad distances, which also revealed that stronger CTCF binding is associated with smaller NRLs (Figure S4). Thus, the CTCF-dependent NRL decrease is a general, dataset-independent effect. Note that chemical mapping-based NRLs should not be directly compared with MNase-seq ones due to the inherent peculiarities of the chemical mapping experiment that were observed noticed previously (Vainshtein, Rippe and Teif 2017); below I will use only MNase-seq and MNase-assisted histone H3 ChIP-seq datasets for nucleosome mapping.

We then asked, whether the same effect on NRL is observed for CTCF's binding partner cohesin. Cohesin is a ring-shaped complex that slides along one or two DNA double helices until it meets CTCF, thus extruding DNA loops (Fudenberg *et al.* 2016). Cohesin shows regular nucleosome phasing around it even when not associated with CTCF (Figure S5), thus it is interesting whether it has a similar effect on NRL. Cohesin does not have its own DNA sequence preferences, but we can still stratify mapped cohesin locations in terms of the strength of binding using ChIP-seq of cohesin's component SMC1 and sorting its occupancy peaks into quintiles based on their height. Figure 10E shows that, similarly to CTCF, cohesin sites are characterised by the local NRL decrease as cohesin's binding strength increases. However, the effect of cohesin's binding strength on NRL is weaker than that for CTCF, and almost disappears if only the cohesin sites that do not contain CTCF motifs are considered (Figure 10E). On the other hand, the bound CTCFs that do not overlap with bound cohesin in ESCs still display a pronounced effect of CTCF binding strength on NRL (Figure 10F). This effect was also recapitulated for CTCF sites residing at least 10,000 bp outside of annotated TSSs (Figure S6), showing that it was not caused by protein coding gene transcription.


195
190
185
180
175
170
NRL (bp)
Stat3
Gata1
c-Jun
Oct4
Essrb
Nanog
GATA2
Tal1
Chd2
Ep300
Max
Znf384
Klf4
Sox2
Sox17
Zbtb2
cMyc
Stat3
Gata2
Essrb
Nanog
Oct4
Sox2
Gata1
Ep300
cJun
Max
ZNF384
Arid3a
Klf4
Sox17
cMyc
Zbtb2
All sites
Excluding CTCF

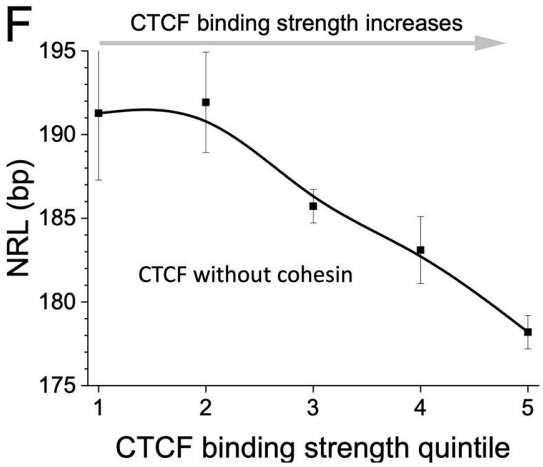

F
CTCF binding strength increases
195
190
185
180
175
NRL (bp)
CTCF without cohesin
1
2
3
4
5
CTCF binding strength quintile

*Figure 10. CTCF-dependent decrease of the nucleosome repeat length (NRL). (**A**) Average nucleosome profile based on MNase-seq from* (Voong *et al.* 2016) *around CTCF binding sites in ESCs determined by ChIP-seq* (Yue *et al.* 2014). *This profile is calculated without taking into account the directionality of CTCF binding. (**B**) Illustration of the 'phasogram' method of NRL calculation for the region [100, 2000] from the center of experimental CTCF sites measured in ESCs. The calculation of frequencies of nucleosome dyad-to-dyad distances is followed by the linear regression of the peak positions (insert). (**C**) NRLs calculated near binding sites of 18 stemness-related chromatin proteins in ESCs in the region [100, 2000] from the summit of TF binding ChIP-seq peak, using chemical nucleosome mapping data from* (Voong *et al.* 2016). *Left: all TF binding sites; right: TF binding sites which do not intersect with CTCF. Open squares show the average NRL value based on all these TFs. The full list of experimental ChIP-seq datasets used in this calculation is provided in Supplementary Table ST1. (**D**) Dependence of NRL on the strength of CTCF binding based on experimental ChIP-seq peaks from mouse ENCODE* (Voong *et al.* 2016) *stratified into binding strength quintiles by the heights of peaks (black line) and computationally predicted CTCF sites obtained by scanning the mouse genome with TFBStools using >80% similarity for JASPAR matrix MA0139.1 stratified into binding strength quintiles by their TRAP score (red line). (**E**) NRL near bound cohesin, split into 5 quintiles based on the heights of experimental ChIP-seq peaks of the cohesin subunit SMC1* (Sun *et al.* 2019), *calculated separately for all cohesin sites (black) and cohesin sites that do not contain CTCF motifs (red). (**F**) The same as (D), but only for experimental CTCF peaks that do not overlap with SMC1 peaks. The error bars correspond to the standard deviation of the linear fit across the peaks of the phasogram as explained in panel B.*

Using the same procedure I investigated NRL near other chromatin proteins. Firstly, we considered 497 TFs which have position weight matrices in JASPAR2018 (Khan *et al.* 2018), and for each of them calculated NRL in the region [100, 2000] from the TF motif as a function of the DNA-binding strength predicted for a given TF. This analysis revealed that for TFs other than CTCF, the NRLs did not reveal a monotonic function of their binding strength (see Figure 11 for examples of TFs relevant to stem cells). I also performed a similar calculation for chromatin remodellers that have been experimentally profiled in ESCs, asking whether NRL in the region [-1000, 1000] near remodeller depends on the height of the corresponding remodeller peak (Figure S7). These calculations did not reveal NRL dependence on the binding strength as in the case of CTCF or cohesin. Thus, CTCF and cohesin are unique proteins whose DNA binding strength is anticorrelated to the NRL value.

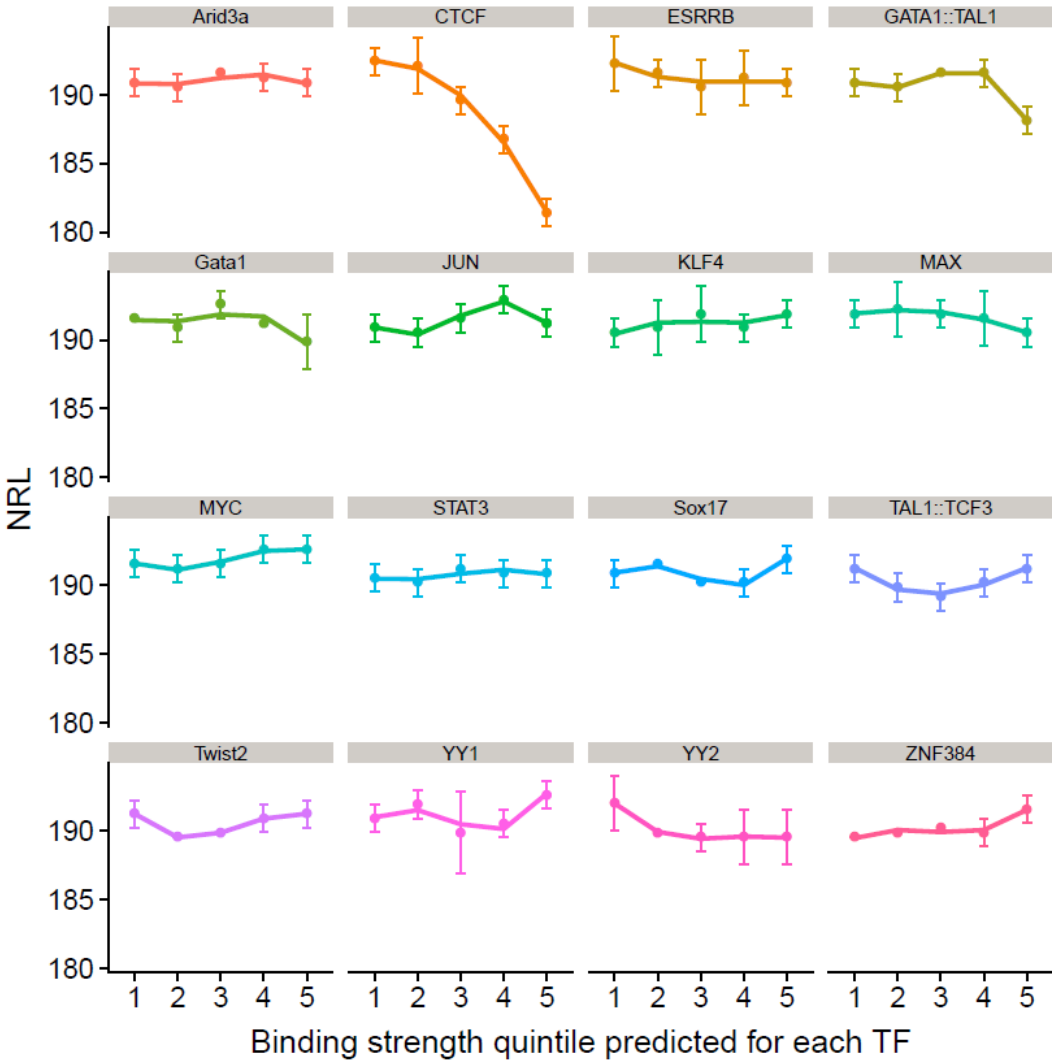


Figure 11. Proteins other than CTCF and cohesin do not show the relationship between DNA-binding strength and NRL near their binding sites. Sixteen representative TFs related to stem cells are shown (similar calculations were performed for 497 TFs listed in JASPAR2018). TF bindings sites used in this analysis were predicted computationally by scanning the mouse genome using TFBStools with the 80% motif similarity cut off and then stratified into five binding strength quintiles based on the TRAP score (see Materials and Methods).

2.4.4 The strength of CTCF-DNA binding correlates with GC and CpG content

In order to understand the physical mechanisms of NRL decrease near CTCF, we considered a number of genomic features and molecular factors that could potentially account for the NRL decrease near CTCF (Figure 12). A number of previous observations suggested that the ability of CTCF sites to retain CTCF during cell perturbations is related to the surrounding GC and CpG content (Pavlaki *et al.* 2018; Wiehle *et al.* 2019). Our calculations performed here provide more detail on this effect, showing that the strength of CTCF binding is correlated with GC content around CTCF sites (Figure 12A), and that the probability for a given site to be located in a CpG island monotonically increases with the CTCF binding strength (Figure 3B). It is worth noting that the CTCF motif itself is GC-rich, which corresponds to the central peak in Figure 3A, but the effects mentioned above extend to distances >1000 bp from CTCF motif. Furthermore, the CTCF site location inside CpG islands was associated with a significantly decreased NRL in comparison with all CTCF sites (Figure 12D).

2.4.4.1 The strength of CTCF-DNA binding correlates with the probability of a given site to be inside cis-regulatory elements and domain boundaries

Another potential hypothesis is that the small NRL near CTCF could be because CTCF sites are in active regions (promoters, enhancers, etc.) which have a smaller NRL in comparison to genome-average based on previous studies (Valouev *et al.* 2011; Baldi *et al.* 2018). Our analysis performed here demonstrated that there is a positive correlation between the strength of CTCF binding and the probability that it is inside a promoter region (Figure 3C). I also used recently published coordinates of topologically associated domains (TADs) and promoter-enhancer loops in ESCs (Bonev *et al.* 2017) and showed that there is a correlation between the strength of CTCF binding

and the probability that it forms a boundary of TADs and even higher correlation for the boundaries of loops (Figure 12C). Furthermore, the NRL near CTCF sites was smaller if these sites were inside borders of loops or TADs, while the NRL value went up if all known regulatory regions were excluded (Figure 12D).

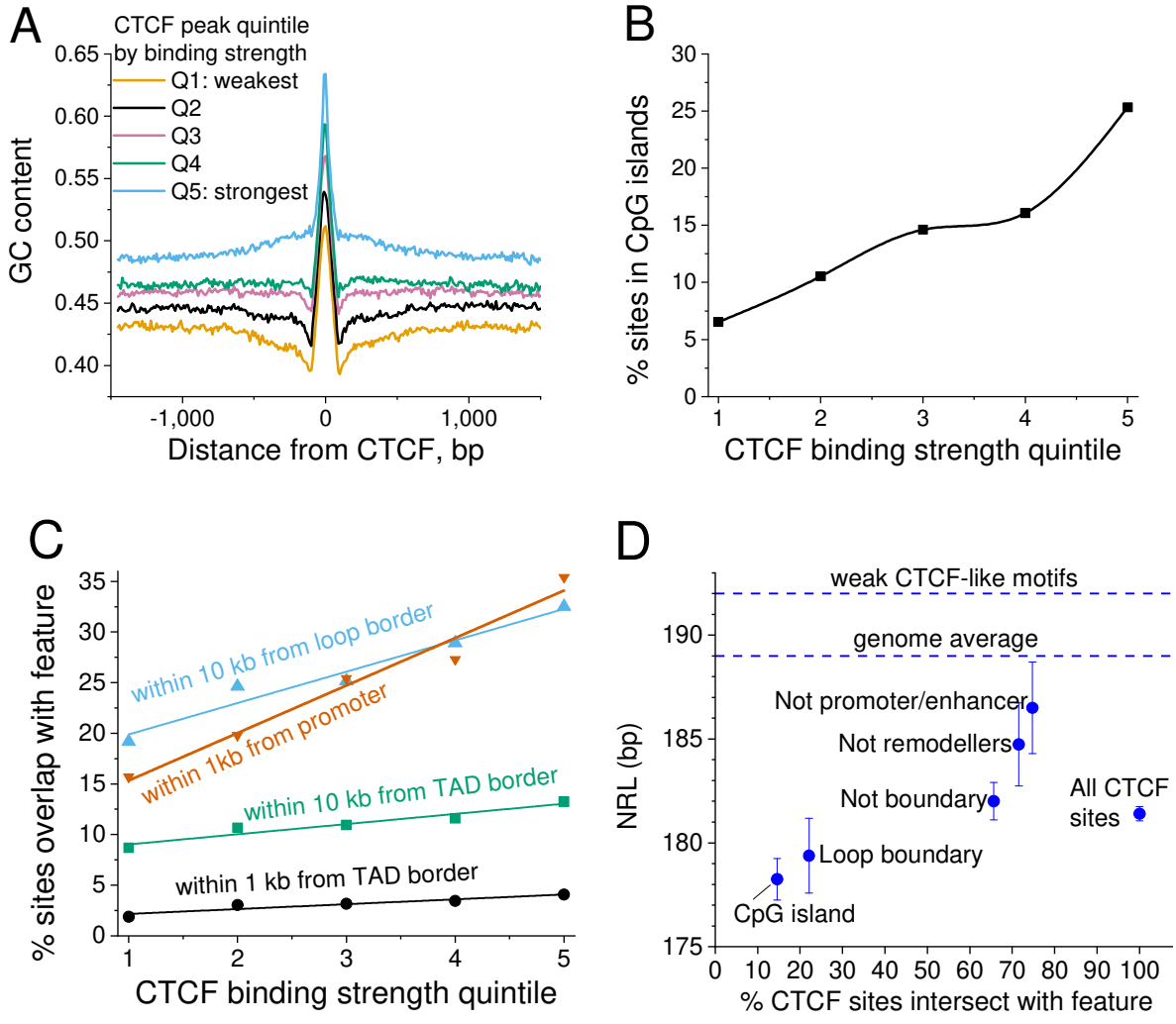


*Figure 12. Genetic features correlating with the experimental strength of CTCF binding. (**A**) CTCF binding sites split into quintiles based on their binding strength are characterized by increasing GC content as CTCF binding strength increases. (**B**) The stronger CTCF binding site*

*the higher is the probability that it is located in a CpG island. (**C**) The stronger CTCF binds the higher the probability that it is located in a promoter or forms a boundary of TADs or enhancer-promoter loops. (**D**) NRLs for the following subsets of CTCF sites: all sites bound in ESCs; inside chromatin loop boundary; outside of boundaries of loops and TADs; inside CpG islands; outside of chromatin remodeller peaks; outside of promoters and enhancers. The top horizontal dashed line corresponds to the weak CTCF-like motifs from Figure 1D. Vertical bars show the standard deviation.*

2.4.5 Remodeller-specific effects on NRL near CTCF

Active nucleosome positioning is determined by chromatin remodellers, but the rules of action of individual remodellers are not well defined. In order to clarify remodeller effects on NRL decrease near CTCF, we processed all available remodeller ChIP-seq datasets in ESCs and plotted the percentage of CTCF sites overlapping with remodeller ChIP-seq peaks (Figure 13A). This analysis showed that the stronger CTCF binds, the higher the probability that a given CTCF binding site overlaps with remodellers. A particularly large percentage of CTCF sites overlaps with peaks of remodellers Chd4, EP400, Chd8 and BRG1, with Chd4 being the top CTCF-related remodeller. I also performed similar analysis for three different TFs: CTCFL, Oct4 and c-Jun (Figure S8). CTCFL (also known as BORIS), shares a number of sites with CTCF, and unsurprisingly BORIS and CTCF have similar preferences for remodellers. On the other hand, Oct4, which is highly expressed in ESCs, showed a qualitatively similar effect of increasing co-binding with remodellers as its DNA sequence-determined binding strength increases, but the top Oct4-associated remodeller was BRG1 rather than Chd4. As a negative control, I considered c-Jun, which is not a

stem cell TF. As expected, for c-Jun binding sites the percentage of intersection with remodeller peaks did not depend on the predicted strength of c-Jun binding to DNA (Figure S8).

Next I set to derive systematic rules of remodeller effects on NRL near CTCF (Figure 13B). By comparing NRLs near CTCF sites overlapping and non-overlapping with each remodeller, we learned that Brg1 has no detectable effect (based on two independent Brg1 datasets), and Snf2h has the strongest effect. The effect of other remodellers on NRL near CTCF is increasing in the order BRG1 ≤ Chd4 < Chd6 < Chd1 ≤ Chd2 ≤ EP400 ≤ Chd8 < Snf2h (Figure 13B).

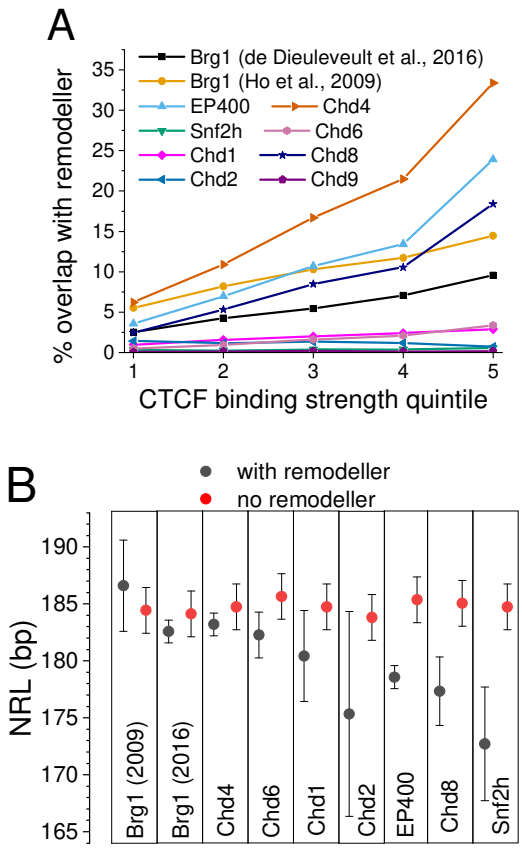


*Figure 13. Effects of different chromatin remodellers on the value of NRL near CTCF. (**A**) The stronger CTCF binds the higher is the probability that it is co-enriched with different chromatin remodellers indicated on the fig- ure. The enrichment was defined as the ratio of CTCF sites*

*overlapping with ChIP-seq peaks of a given remodeller to the total number of CTCF sites in a given quintile. (**B**) NRLs calculated near CTCF sites that overlap (black) and do not overlap (red) with ChIP-seq peaks of eight chromatin remodellers experimentally mapped in ESCs. Remodeller names are indicated on the figure. Two Brg1 datasets reported in 2009 and 2016 are taken from separate publications,* (Ho *et al.* 2009) *and* (De Dieuleveult *et al.* 2016) *respectively.*

2.4.6 CTCF motif directionality introduces asymmetry in adjacent nucleosome distribution

All our calculations above were performed without considering the directionality of the CTCF motif. For example, Figure 10A shows a symmetric pattern of nucleosome occupancy around CTCF, which arises due to averaging of different patterns around CTCF motifs in the direction of the plus and minus strand. Now let us always orient the CTCF motif in the same way, left to right (5' to 3'), and refer to positions in 5' direction from the CTCF motif as "upstream" and 3' direction as "downstream". Using this setup, I calculated aggregate profiles of nucleosome occupancy around CTCF by aligning all regions in 5' to 3' direction of the CTCF motif defined by the JASPAR matrix (MA0139.1). In these calculations only CTCF motifs located in ChIP-seq defined peaks in at least one mouse cell type were considered. Furthermore, we excluded CTCF sites that are located inside annotated promoters (see Methods).

Figure 14A shows the aggregate profiles of MNAse-seq nucleosome occupancy (Voong *et al.* 2016) around CTCF in ESCs taking into account the motif directionality. Here, the wave-like pattern of the nucleosome occupancy around CTCF sites reveals strong asymmetry. To the best of our knowledge this is the first report of such a pronounced nucleosome asymmetry around CTCF motifs. Counterintuitively, the weaker the CTCF binding, the stronger is the asymmetry. Such an asymmetry is similar to what is usually observed near promoters, except that w have excluded

from this calculation CTCF sites that overlap with promoters. This effect was also confirmed using MNase-assisted H3 ChIP-seq dataset (Figure S9) and plotted the occupancy of RNA Pol II around CTCF (Figure 14B). Pol II occupancy shows CTCF-dependent enrichment, which increases with the increase of CTCF binding strength. Weak CTCF sites which have the strongest asymmetry are devoid of Pol II. Thus, the asymmetry of nucleosome occupancy near CTCF is similar to the asymmetry observed for promoters, but these are not promoters and not related to Pol II-transcribed non-coding regions.

2.4.6.1 CTCF is not due to Pol II-dependent transcription

The most striking feature of the asymmetric nucleosome profiles near CTCF is that the deepest point of the nucleosome-depleted region is shifted about 41 bp "upstream" in 5' direction from the center of the CTCF motif. This is different from what is usually assumed based on symmetric profiles such as in Figure 10A. Interestingly, the first strong nucleosome peak at 105 bp "downstream" in 3' direction from CTCF appears similarly for all CTCF site quintiles, whereas the next peak at 165 bp "downstream" in 3' direction from CTCF is extremely sensitive to the CTCF binding strength. There are also several other nucleosome occupancy peaks that display strong sensitivity to the CTCF binding strength.

2.4.6.2 The CTCF-dependent peak of nucleosome occupancy 3'-downstream of CTCF can be attributed to Chd4

In order to determine the structural origin of the nucleosome occupancy peak at 165 bp from the CTCF motif I calculated aggregate profiles of all chromatin remodellers using their ChIP-seq binding datasets in ESCs (Figure S10). Interestingly, we see in Figure S10 that the remodellers

position themselves between nucleosomes. Chd4 is the only remodeller characterised by a CTCF-dependent peak at position +165 bp (Figure 14C- see also Figure S20 and S21 which make this effect clearer). The peak of Chd4 at this location is quite pronounced, which is consistent with Chd4 being the top CTCF-associated remodeller (Figure 13A). Thus, Chd4 plays an important role in establishing the asymmetry of nucleosome positioning, while it does not affect the NRL value per se (Figure 13B). On the other hand, another remodeller Snf2h affects the value of NRL and the regularity of the nucleosome near CTCF (see Figure S11, plotted using the recent Snf2h knockout data (Barisic *et al.* 2019)).

CTCF creates asymmetric nucleosome arrays; cohesin symmetrises them. Next I investigated the interplay between CTCF and cohesin in relation to the asymmetry of nucleosome arrays. Cohesin's subunits Rad21 and SMC1 bind quite symmetrically with respect to the CTCF motif (Figure S12) and they have a dramatic effect on the symmetry of nucleosome arrays around CTCF (Figure 14D). Our calculations showed that for all CTCF binding strength quintiles, CTCFs which are not co-bound with cohesin create asymmetric and less regular nucleosome arrays, whereas CTCFs co-bound with cohesin create more symmetric and more regular arrays of nucleosomes (Figure 14D).

2.4.6.3 *The value of NRL in the region 3'-downstream of the CTCF motif linearly depends on the CTCF binding strength*

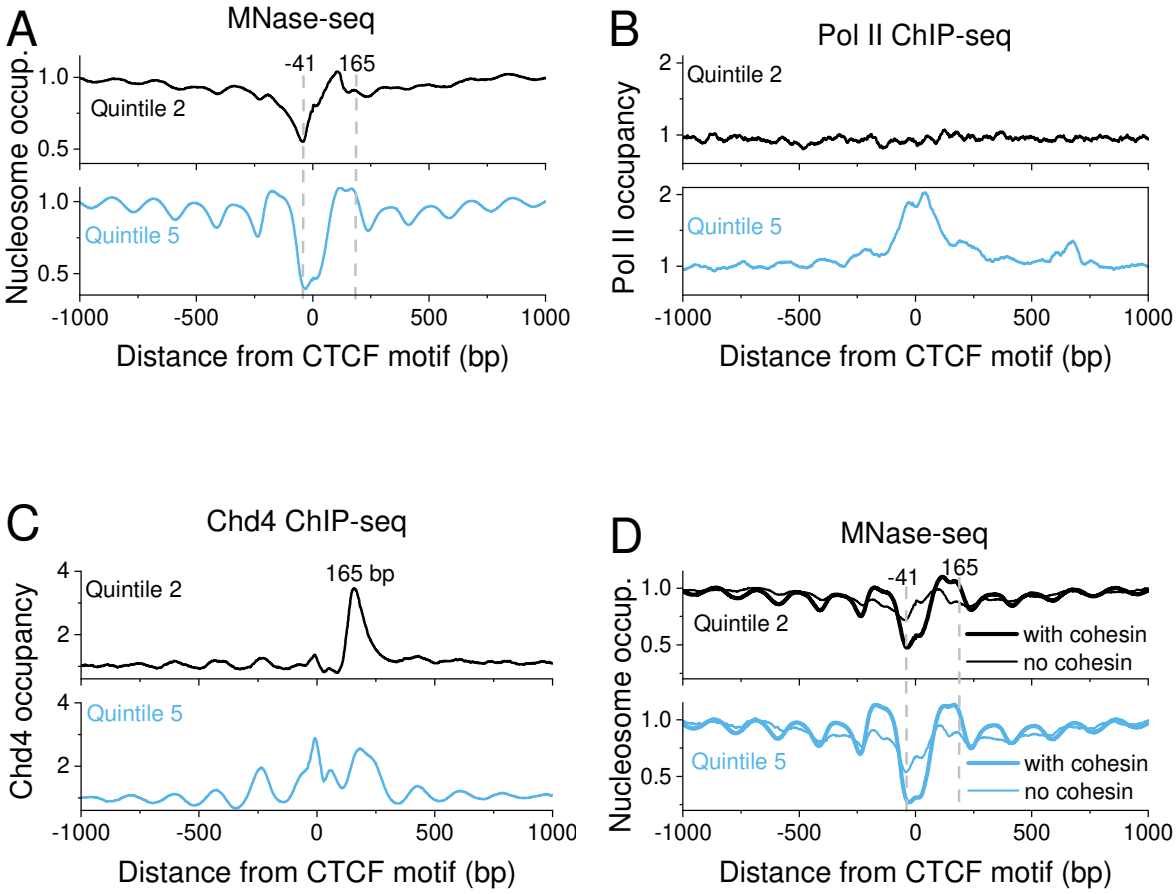
The effect of CTCF motif directionality introduces a significant correction to the NRL dependence on the CTCF binding strength that we found above (Figure 14E and F). When performing NRL calculations separately for the region [100, 2000] 3'-downstream and region [-2000, -100] 5'-upstream from the center of the CTCF motif, we noticed that the most regular behaviour is observed 3'-downstream, where the effect can be described by a linear dependence (Figure 14F).

I also checked whether the appearance of the nucleosome occupancy peak 165 bp downstream of CTCF is the main determinant of the NRL decrease (this may not be clear as there are only 2 binding strengths to compare in Figure 14A- hence an alternative Figure S20 is provided to make this effect clearer). The recalculation of the NRL in the interval [300, 2000] 3'-downstream from CTCF showed that while the NRL decrease is less steep, it still follows the same trend (Figure S13). This is because the the nucleosome at +165 bp is discluded from the calculation.

2.4.7 The asymmetric nucleosome depletion 5'-upstream of CTCF/CTCFL motifs is encoded in DNA repeats and may be linked to their transcription

Next we calculated the average nucleotide distribution around CTCF sites used above taking into account the orientation of CTCF motifs. This revealed an unexpected nucleotide pattern in the extended region near CTCF (Figure 15A). The nucleosome depletion in the region around -41 bp upstream of CTCF is associated with a decrease of GC content. This is consistent with previous observations that high AT-content and in particular poly(dA:dT)-tracts have strong nucleosome-excluding properties (Segal and Widom 2009). It is worth noting that the CTCF motif used in our calculations is just 19 bp, but the length of the highly structured area near CTCF is more than 200 bp. This means that the CTCF motif is frequently encountered as part of a much larger DNA sequence organisation, some type of sequence repeats that are primarily responsible for the establishment of the asymmetric boundaries around CTCF. Indeed, 50% of the CTCF motifs used in our calculations in Figures 5 and 6 overlapped with repeats defined by the UCSC Genome Browser repeat masker. Furthermore, the percentage of repeats given by the repeat masker shows a very structured profile with an extended region (>200 bp) near CTCF strongly enriched with repeats (Figure 15B).

Another interesting finding shown in Figure 6C and D is that when subjected to a separate de novo motif discovery, for each binding strength quintile, the strongest quintile 5 was associated with the classical CTCF motif (JASPAR MA0139.1), whereas a weak quintile 2 was associated with CTCFL (BORIS) defined by the JASPAR matrix MA1102.1. To the best of our knowledge this is the first indication that CTCF and CTCFL may have different effects on nucleosomal organisation (Figure 16A).



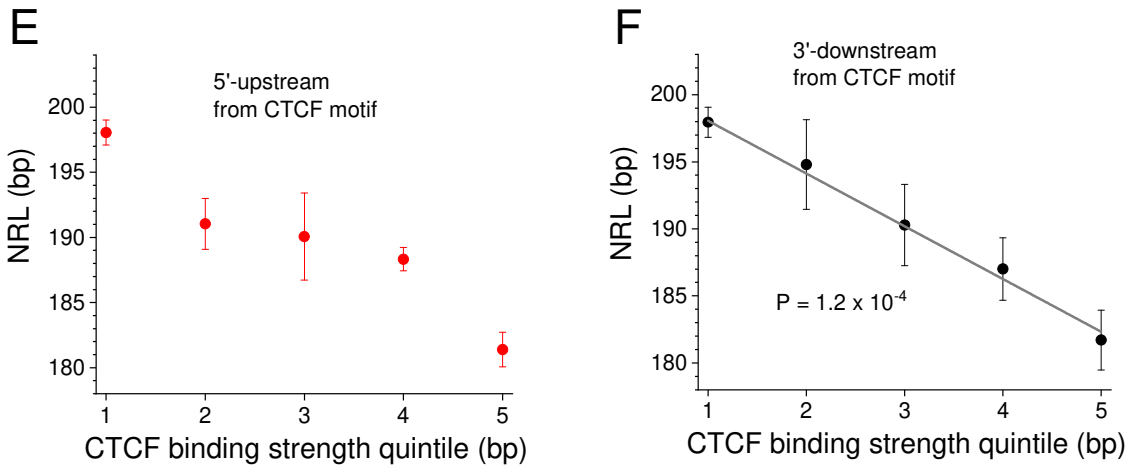


*Figure 14. Combined effects of CTCF motif directionality and binding strength on nucleosome positioning. (**A**) Aggregate nucleosome profiles based on MNase-seq (42) around CTCF motifs outside promoters which coincide with experimentally verified binding sites in at least one mouse cell types, taking into account the DNA strand directionality. The strong peak at 105 bp from the center of CTCF motif appears for all CTCF quintiles. On the other hand, the nucleosome peak at position 165 is sensitive to the strength of CTCF binding and increases as the strength of CTCF binding increases from weak binding at quintile 2 to strong binding at quintile 5. (**B**) CTCF binding outside of promoters is associated with CTCF-dependent Pol II enrichment. In the weakest CTCF quintile there is no Pol II enrichment, so the promoter-like nucleosome occupancy near CTCF is not due to Pol II. (**C**) The binding of Chd4 (and not any other experimentally profiled remodeller) shows a CTCF dependent peak at 165 bp, coinciding with the nucleosome occupancy peak. (**D**) Nucleosome positioning based on MNase-seq, as in panel A, but CTCF sites are split into those that overlap with the cohesin subunit SMC1 (thick line) and do not overlap with SMC1 (thin line). E and F) NRL as a function of CTCF binding strength quintile corrected for the CTCF motif directionality. (**E**) NRL calculated in the region [−2000, 100] in 5′ direction ('upstream') of*

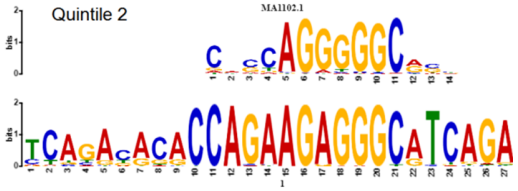
*the center of CTCF motif. (**F**) NRL calculated in the region [100, 2000] in 3' direction ('downstream') of the center of CTCF motif. In the latter case NRL dependence of CTCF binding strength can be fitted as a straight line (t-test $P = 1.2 \times 10^{-4}$).*

We have also checked whether the nucleosome depletion 5'-upstream of CTCF is related to transposon transcription. Using coordinates of ChIP-seq peaks of RNA Pol III determined previously in ESCs (Carrire *et al.* 2012), we found that 33% of co-localisations of Pol III and 17% of co-localisations of SINE repeats and Pol III overlapped with our CTCF motifs. Thus, not only the DNA repeats are responsible for the AT-rich region 5'-upstream of CTCF, but also their Pol III-dependent transcription may be linked to the asymmetric nucleosome depletion pattern.

2.4.7.1 CTCF binding directly affects expression of adjacent RNA

In order to investigate quantitatively the effect of CTCF on RNA expression, I plotted the normalised amount of total RNA reads within [-1000, 1000] from CTCF as a function of CTCF binding strength (Figure 15E). It showed that the strong CTCF binding correlates with the weaker expression of neighbouring RNA (P = 1.2e-11). There was no significant asymmetry in RNA expression up- or downstream of the CTCF motif. This is exclusively in coding and non-coding regions to test if the effect was specific to CTCF near genes. The effect was consistent in either case.

C



D

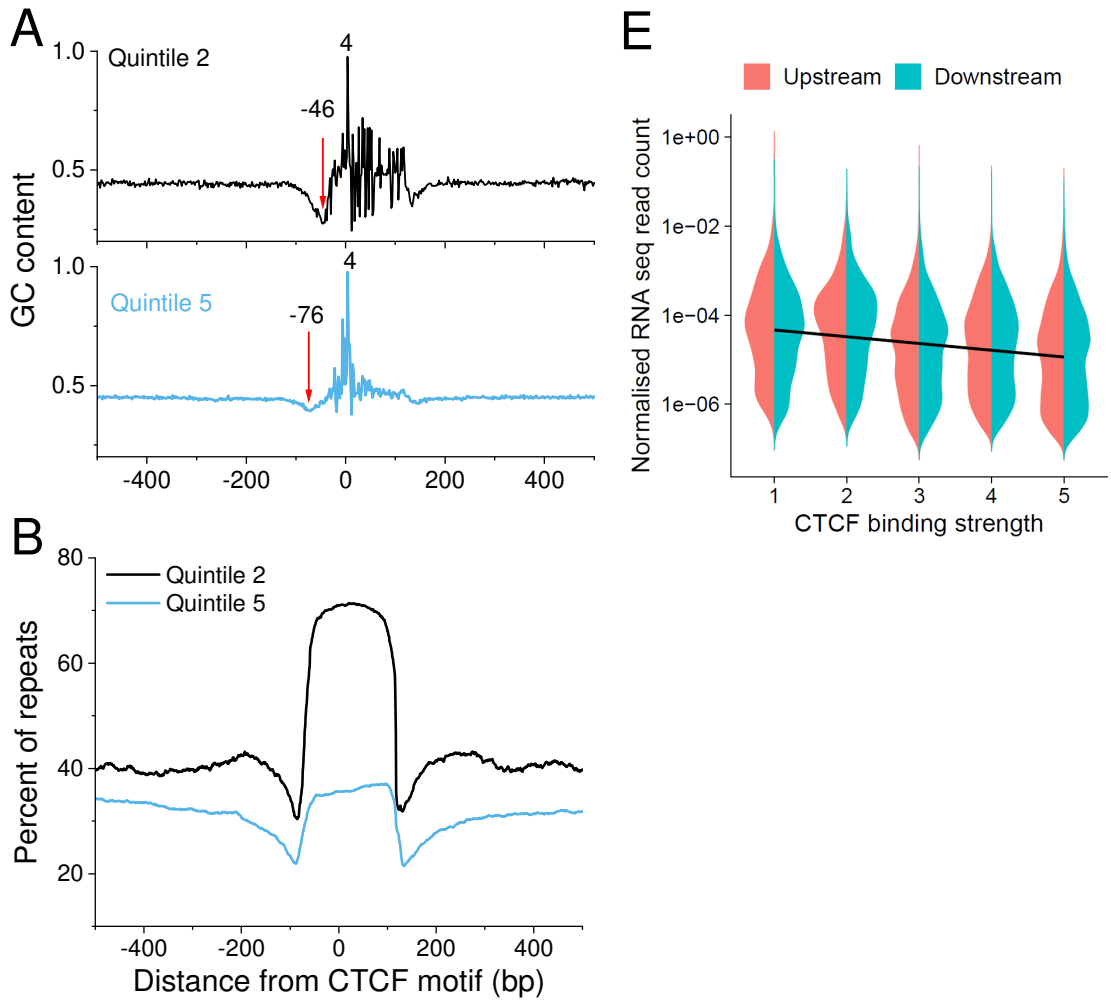


*Figure 15: Effects of the nucleotide content around CTCF sites. (**A**) Average GC content around CTCF motifs for CTCF binding strength quintiles 2 and 5. (**B**) The percentage of repeats determined by the USCS Genome Browser's Repeat Masker as a function of the distance from the middle of CTCF motifs. (**C**) The sequence of the consensus motif in quintile 2 with the smallest P-value. The best TF match for the quintile 2 consensus motif is CTCFL (Boris) (JASPAR MA1102.1). (**D**) The sequence of the consensus motif in quintile 5. The quintile 5 consensus sequence contains the classical CTCF motif (JASPAR MA0139.1). (**E**) Violin plot showing the numbers of RNA reads expressed from the regions [−1000; 0] and [0; 1000], respectively upstream (red) and downstream (blue) of CTCF binding sites, as a function of CTCF binding strength. The straight line is a linear fit through all the points, showing a general decrease of the number of RNA reads*

as CTCF binding strength increases (P = 1.2e−11). The linear fits performed separately across 'downstream' or 'upstream' regions are not distinguishable.

2.4.8 Nucleosome-depleted boundaries 5'-upstream of CTCF motif are preserved even if binding CTCF is lost during cell differentiation

Next I compared nucleosome positioning around CTCF motifs upon differentiation of ESCs to neural progenitor cells (NPCs), as well as in the differentiated mouse embryonic fibroblasts (MEFs) using MNase-seq data from (Teif *et al.* 2012) and CTCF ChIP-seq data from (Bonev *et al.* 2017) (Figure 16A). Notably, stronger CTCF binding to DNA increases the probability that a given site will remain bound upon differentiation. This suggests that the sequence-dependent strength of CTCF binding can act as the "CTCF code", determining which CTCF sites are retained and lost upon differentiation (and thus how the 3D structure of the genome will change). Our further analysis revealed that common CTCF sites that are present in all three states are characterised by quite minor asymmetry of nucleosome organisation (Figure 16B). On the other hand, CTCF sites that are lost upon ESC differentiation to NPCs and MEFs have more profound asymmetry of the nucleosome pattern around them (Figure 16C and D). Upon differentiation both in NPCs and MEFs, the array of nucleosomes 3'-downstream of the CTCF motif is shifted to cover the CTCF site. It is worth noting that nucleosome positioning in this region is only partly CTCF-dependent. For example, inside the [-100, 100] region around CTCF, the percentage of nucleosomes covering the CTCF motifs that lost CTCF upon differentiation changes from 47% to 60% upon ESC to MEF transition, and from 42% to 54% upon ESC differentiation to NPC. Interestingly, the nucleosome-depleted region 5'-upstream of CTCF still remains open upon differentiation. The latter effect was

also confirmed for the case of CTCF sites that are not bound by CTCF in ESCs and become bound in MEFs (Figure S15).

2.4.8.1 Common CTCF sites preserve local nucleosome organisation during ESC differentiation

Then, I set to determine the functional consequences of the NRL decrease near CTCF. NRL near bound CTCF on average increases as the cells differentiate from ESCs to NPCs or MEFs (Figure 16E and S16). However, common CTCF sites resist this NRL change, suggesting that CTCF retention at common sites upon differentiation preserves both 3D structure and nucleosome patterns at these loci. As we have established previously (Figure 14F), the effect of the active CTCF-dependent NRL decrease is mostly pronounced in the region 3'-downstream of CTCF motifs. The NRL increase near CTCF upon cell differentiation is also mostly in the 3'-downstream region (Figure S17).

2.4.9 Directed CTCF motifs mark TAD boundaries

Our previous calculations were performed at the level of boundaries formed by single CTCF motifs. However, in some cases chromatin boundaries are created by cumulative action of several CTCF motifs located not far from each other. In particular, our calculations showed that CTCF motifs oriented toward the inner part of TAD are centered at the TAD boundaries, whereas the outward-looking CTCF motifs are enriched at the outer side of the boundaries (Figure 16F). TADs that were lost upon differentiation demonstrate a smaller enrichment of CTCF motifs near them (Figure S18), which suggests that CTCF motifs at functionally important chromatin boundaries may act additively. Thus, the effects of individual CTCF motifs described above can be summed up at a region of up to several kb, to act synergistically at the boundaries between large chromatin domains.

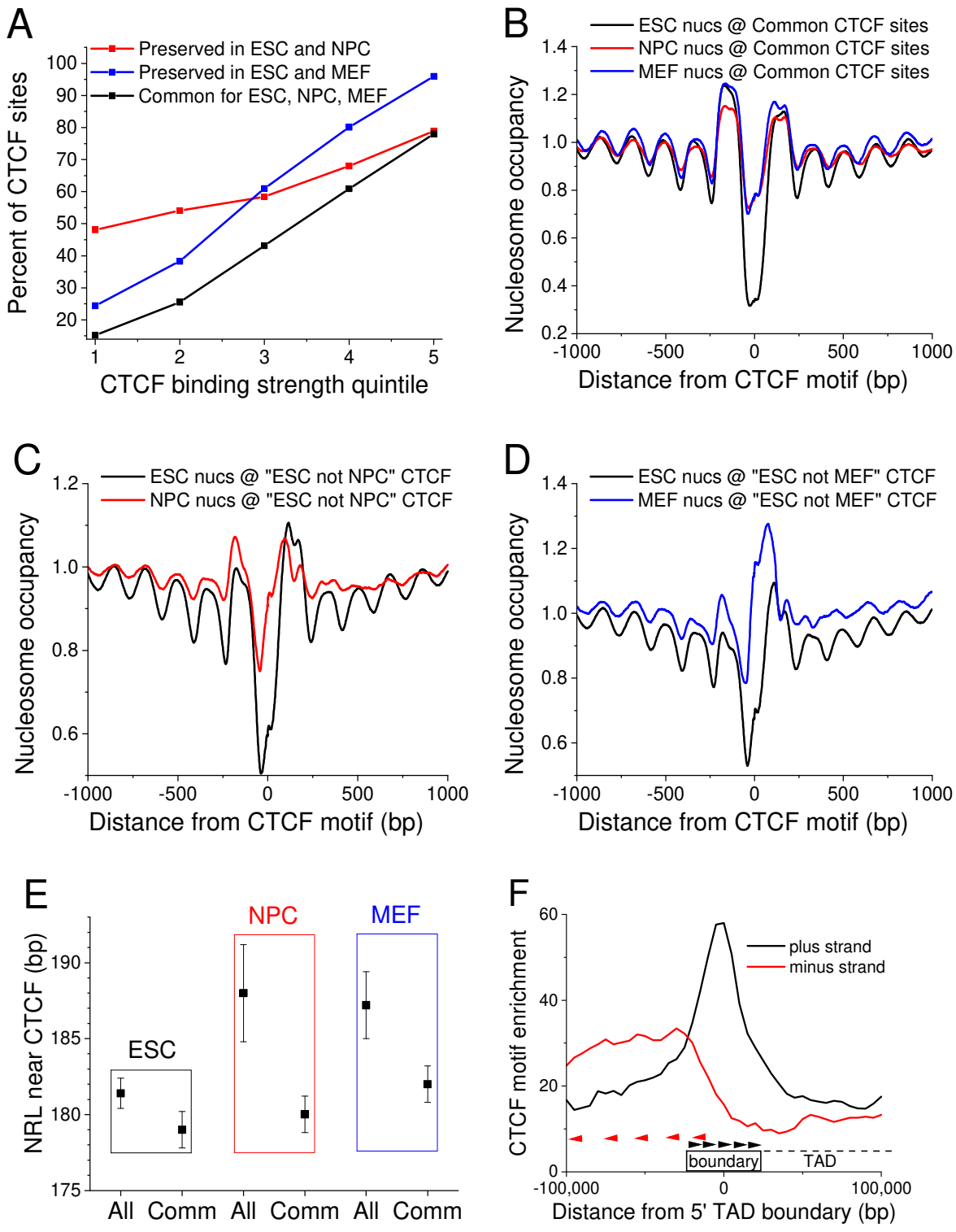

A
Preserved in ESC and NPC
Preserved in ESC and MEF
Common for ESC, NPC, MEF
Percent of CTCF sites
CTCF binding strength quintile
B
ESC nucs @ Common CTCF sites
NPC nucs @ Common CTCF sites
MEF nucs @ Common CTCF sites
Nucleosome occupancy
Distance from CTCF motif (bp)
C
ESC nucs @ "ESC not NPC" CTCF
NPC nucs @ "ESC not NPC" CTCF
Nucleosome occupancy
Distance from CTCF motif (bp)
D
ESC nucs @ "ESC not MEF" CTCF
MEF nucs @ "ESC not MEF" CTCF
Nucleosome occupancy
Distance from CTCF motif (bp)
E
ESC
NPC
MEF
NRL near CTCF (bp)
All Comm All Comm All Comm
F
plus strand
minus strand
CTCF motif enrichment
boundary
TAD
Distance from 5' TAD boundary (bp)

*Figure 16. Effects of asymmetric CTCF-dependent boundaries in stem cell differentiation. (**A**) The fraction of CTCF sites preserved upon differentiation of ESCs to NPCs and MEFs as a function of CTCF binding strength. CTCF sites preserves in all these three cell types are termed 'common'. (**B**) Nucleosome occupancy in ESCs (black), NPCs (red) and MEFs (blue) around CTCF sites common between ESC, NPC and MEF, calculated taking into account CTCF motif directionality. (**C**) Nucleosome occupancy around 'ESC not MEF' sites that are present in ESCs (black line) but lost in MEFs (red line) taking into account CTCF motif directionality. (**D**) Nucleosome occupancy around 'ESC not NPC' sites that are present in ESCs (black line) but lost in NPCs (red line) taking into account CTCF motif directionality. Note that in differentiated cells a nucleosome is being positioned to cover the 'lost' CTCF sites, but nucleosome depletion on the left of CTCF is still preserved. (**E**) NRLs in region [100, 2000] from CTCF's experimental binding site summit calculated without taking into account the motif directionality. Upon differentiation average NRL near CTCF increases (denoted 'All'), but common CTCF sites keep the smallest NRL (denoted 'Comm'). (**F**) Enrichment of the strongest CTCF motifs (5th quintile) near 5′-boundaries of TADs in ESC, calculated separately for CTCF motifs oriented 5′-to-3′ (black) and 3′-to-5′ (red). The TAD is located to the right from the 5′-boundary. The arrows show an example of CTCF motif distribution for an individual region.*

2.5 Discussion

I developed a new NRLcalc methodology to investigate nucleosome rearrangement and NRL changes near TF binding motifs distinguished by their orientation and binding strength. The

application of this method to CTCF and cohesin binding sites revealed a number of new effects (Figure 17).

Firstly, I found that contrary to previous assumptions, the nucleosome arrangement near CTCF motifs is asymmetric and to a large degree hard-wired in the sequence of the DNA region >200 bp long including the CTCF motif (Figure 14A and 15A). The asymmetry in this case is not just a consequence of heterogeneity of nucleosome distributions around subsets of sites (Kundaje *et al.* 2012), but is a generic feature across all CTCF sites. The nucleosome-depleted region, which was previously believed to coincide with the CTCF binding site (Beshnova *et al.* 2014; Teif *et al.* 2014), is actually shifted 5'-upstream of CTCF motif (Figure 14). This nucleosome depletion is associated with AT-rich DNA sequence repeats which may disfavour nucleosome formation (Segal and Widom 2009) and introduce bending of the double helix near CTCF (Nichols and Corces 2015; Ghirlando and Felsenfeld 2016). The effect of CTCF is modulated by its binding partner cohesin, which symmetrises the nucleosome arrays when it co-binds with CTCF (Figure 5D).

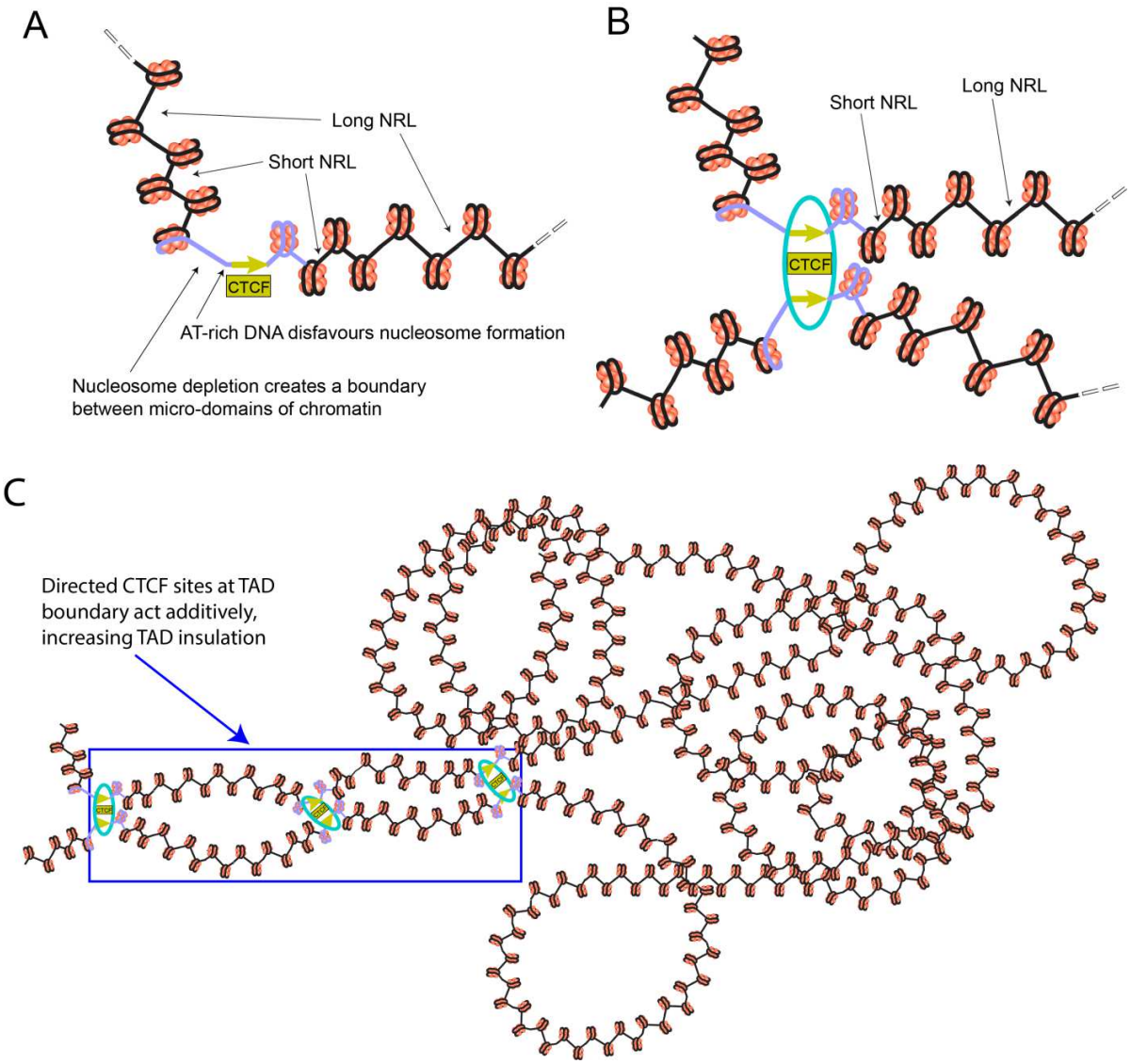


*Figure 17: Schematic illustration of the effect of CTCF binding strength and motif orientation on the nucleosome arrangement in a single genomic region (**A**), at the base of a loop (**B**), and as part of a chromatin boundary containing several CTCF motifs (**C**). An extended DNA region including CTCF motif is enriched with repetitive sequences that define the mechanical properties of this region as a chromatin boundary (shown in violet colour)—see Figures 5A, 6A, D and Supplementary Figure S4. The region 5'-upstream of CTCF motif contains AT-rich sequences that*

disfavour nucleosome formation and may account for DNA bending in the complex with CTCF. Such regions can be due to DNA repeats such as SINEs, some of which are transcribed by Pol III that interact with CTCF. In analogy to the coding gene transcription the region 5'-upstream of the CTCF motif is depleted of the '-1' nucleosome. In the region 3'-downstream of CTCF motif chromatin remodellers including Chd4 and Snf2h determine the regularity of the nucleosome array. The nucleosomes located close to CTCF are separated by shorter linkers and nucleosomes further away from CTCF are separated by longer linkers, reaching the genome-average linker length at distances where CTCF effects disappear (corresponding to NRL change from ~180 bp near strong CTCF sites to ~190 bp genome-average, see Figure 3D). The cohesin ring is represented by the cyan ellipse. In the chromatin boundary containing several CTCF motifs, the effects described above for individual CTCFs may add up to increase chromatin domain insulation through the construction of special nucleosome array packing at the boundary, physically preventing interactions between adjacent TADs.

The asymmetric nucleosome-depleted regions near CTCF resemble the pattern observed near TSS, and the corresponding effect of NRL decrease as the gene activity increases (Figure S14). Importantly, this effect is observed even for CTCF sites that are separated by more than 10,000 bp from the nearest annotated TSS (Figure S6). Thus, the effects reported here are not directly related to gene transcription by Pol II. However, they may be linked to transcription of transposons such as Pol III-dependent SINE repeats. Several publications suggested an important role of transposons in the evolution of CTCF sites (Bourque *et al.* 2008; Schmidt *et al.* 2012; Choudhary *et al.* 2018; Zhang *et al.* 2018; Kentepozidou *et al.* 2019), and also it is known that mouse SINE B2 repeats can act as insulators (domain boundaries) per se (Lunyak *et al.* 2007). In addition, our data suggests

that CTCF may play active role in transposon functioning as transcribed units separating nucleosome arrays. This is in line with recent reports about transcribed transposons associated with CTCF sites (Zhang *et al.* 2019). Interestingly, previous publications reported that TFIIIC binds to RNA Pol III at tRNA genes and acts as a barrier against the spreading of heterochromatin (Simms *et al.* 2008)– this barrier function can be now re-interpreted in light of our results on the association of CTCF with Pol III as well as Pol II outside of gene promoters (Figure 5B). The importance of repetitive DNA sequences in the formation of chromatin boundaries near CTCF is further strengthened by the possibility of non-consensus TF binding in these regions (Afek *et al.* 2014). Unexpectedly, the effect of CTCF on the expression of RNA from adjacent locations is short-range repression, which becomes stronger as CTCF binding increases (Figure 15E).

I also showed that the asymmetry of the nucleosome signatures depends on the DNA-defined strength of CTCF binding and may be in addition determined by the CTCF/BORIS competition, because "weak" CTCF binding sites are enriched with the CTCFL recognition motif (Figure 15). BORIS has been previously proposed to interfere with CTCF binding, and our results further substantiate its role in the "CTCF code" (Bonev *et al.* 2017) that defines differential CTCF/BORIS binding. Previous studies have shown that CTCF and BORIS are produced by genes from a common ancestor and that they can compete for the same motifs. Upon differentiation, BORIS can displace CTCF at distinct loci. Upon differentiation, BORIS can displace CTCF at distinct loci in gonad cells (in mammals). While it has also been shown to be aberrantly expressed in cancer cells (Hore, Deakin and Marshall Graves 2008). Hence it could be possible that the weaker CTCF binding sites are in fact BORIS sites that will be displaced when the cell differentiates.

Secondly, I found that the NRL decrease near CTCF is correlated with CTCF-DNA binding affinity (Figure 10D, F and 14F). This result goes significantly beyond previous observations that

the CTCF binding strength is related to a more regular nucleosome ordering near its binding site (Vainshtein, Rippe and Teif 2017; Owens *et al.* 2019) and may have direct functional implications. Strikingly, the variation of NRL as a function of CTCF binding affinity can be as large as ~20 bp (the difference between NRL near the weakest CTCF-like motifs and the strongest CTCF-bound sites). Cohesin has a similar effect, but it is pronounced only when cohesin co-binds with CTCF. None of other DNA-binding proteins showed such behaviour (Figure 11). This uniqueness of CTCF can be explained by the large variability of its binding affinity through different combinations of its 11 zinc fingers that allows creating a "CTCF code" (Fang *et al.* 2015; Nichols and Corces 2015; Lobanenkov and Zentner 2018). The effect of the NRL dependence on CTCF binding strength is most profound 3'-downstream of CTCF motifs, where it can be approximated by a linear function (Figure 14F). This strong nucleosome patterning downstream but not upstream of CTCF is comparable to that of transcription start sites (TSSs) of protein-coding genes. In analogy, this effect could provide an additional argument that this may be linked to the transcription of non-coding repeats enclosing CTCF including Pol III-dependent SINEs.

Thus, our data suggests that the nucleosome arrangement near CTCF is defined by an active, remodeller-dependent process. Therefore, we analysed the contributions to this process by each of 8 chromatin remodellers that have been experimentally profiled in ESCs (Figure 13). We found that Snf2h has a major role in NRL decrease near CTCF, consistent with previous studies of Snf2H knockout in HeLa cells (Wiechens *et al.* 2016) and ESCs (Barisic *et al.* 2019). In accord with the latter study, we observed that BRG1 has no detectable effect on NRL near CTCF, although it may be still involved in nucleosome positioning near TAD boundaries (Rasim Barutcu *et al.* 2017). Our investigation also identified Chd8 and EP400 as regulators of NRL near CTCF (Figure 13B, S10). These findings are consistent with the previous investigations that showed that Chd8 physically

interacts with CTCF and knockdown of Chd8 abolishes the insulator activity of CTCF sites required for IGF2 imprinting (Ishihara, Oshimura and Nakao 2006). One can hypothesise that this kind of insulator activity of CTCF is related to the boundary created by the nucleosome-free region 5'-upstream of the CTCF motif reported here, which may physically prevent the spreading of DNA methylation and other epigenetic modifications. Interestingly, our analysis revealed that the main chromatin remodeller responsible for the asymmetry of the nucleosome array near CTCF is Chd4. We show that Chd4 is both the top CTCF-associated remodeller (Figure 13A) and the sole remodeller responsible for the CTCF-dependent nucleosome occupancy peak 3'-downstream of the CTCF motif (Figure 5C). This finding may be important in the context of recent studies indicating that Chd4 is increasing the nucleosome density at regulatory regions (Bornelöv *et al.* 2018).

The third major finding of this work concerns the effects of CTCF motif directionality and binding strength on nucleosome rearrangement during cell differentiation. Our calculations showed that the binding affinity is a good predictor for a given CTCF site being preserved upon cell differentiation (Figure 16A). This may be used as a foundation for the "CTCF code" determining its differential binding as the cell progresses along the Waddington-type pathways. A specific subclass of common CTCF sites preserved upon cell differentiation tends to keep a small NRL, while the average NRL near all CTCF sites increases due to the active nucleosome repositioning 3'-downstream of CTCF motifs (Figure 16). A previous study reported a related distinction of common versus non-common CTCF sites based on the distance between the two nucleosomes downstream and upstream of CTCF (Snyder *et al.* 2016). The preservation of NRL for common CTCF sites may give rise to a new effect where differential CTCF binding defines extended regions which do not change (or change minimally) their nucleosome positioning. Unexpectedly,

the nucleosome-depleted region 5'-upstream of the CTCF motif remains even after CTCF depletion from a given site during differentiation. These nucleosome-depleted regions can have important functional roles, including the preservation of chromatin states while CTCF-dependent loops are dynamic and frequently break and reform throughout the cell cycle (Hansen *et al.* 2017). For example, if the spreading of some chemical modifications of DNA or histones along the genomic coordinate requires enzymes cooperatively binding to the adjacent nucleosomes, then the consistent lack of a nucleosome at a given location can stop the propagation of the "epigenetic wave".

Finally, our finding of the asymmetry of CTCF-dependent chromatin boundaries at the scale of several nucleosomes may provide the missing mechanistic explanation for the asymmetry of chromatin boundaries at the scale of hundreds to thousands of nucleosomes reported recently (Barrington *et al.* 2019; Nanni *et al.* 2019). As I showed, TAD boundaries often contain several directed CTCF motifs (Figures 16F, S18). One can speculate that in this case the effects of individual CTCF sites accumulate, leading to the formation of a specific, asymmetric and 3D-structured nucleosome organisation at TAD boundary (schematically represented in Figure 17C). Such additivity of individual CTCF motifs could explain previous observations where the removal of part of the DNA sequence responsible for the boundary does not lead to the complete loss of TAD insulation (Barutcu *et al.* 2018). In general, the asymmetric nucleosome organisation near CTCF reported here can be particularly interesting in light of the ongoing debate on the functional roles of chromatin boundaries in gene regulation.

CHAPTER 3. Nucleosome positioning is predictive of chromatin states

3.1 Abstract

The positioning of nucleosomes in the genome is an area of major research as it plays a role in determining the regions in the genome that are transcribed and those that are not. Genomic regions that are tightly packed by histones show little or no transcriptional activity (heterochromatin) while regions that are free of histone wrapping are loose and accessible to the transcriptional machinery (euchromatin) and this has major implications in cell-differentiation and cancer. It is not fully known what determines where exactly a nucleosome positions itself on the DNA fiber. Various attempts to derive an understanding of this phenomenon have been made in the form of predictive models. Various strategies exist to predict nucleosome positioning all are far from achieving perfect accuracy across an entire genome (Segal *et al.* 2006). Beyond nucleosome organisation is a higher level of chromatin architecture. The combined effects of the presence of multiple chromatin modifications and/or transcription factors (TFs) at a genomic locus gives rise to a distinct chromatin conformation and gene activity (referred to as the chromatin state). A problem that has not been solved yet is the relationship between the small-scale chromatin structure (at the level of nucleosomes) and the higher order chromatin state. Here we propose a hypothesis that chromatin loci of different states have distinct patterns in how the constituent nucleosomes are positioned relative to one another. I use a convolutional neural network (CNN) to predict chromatin state based on the nucleosome positioning signature (NPS). I perform both binary and multi-label classification to achieve 60-99% accuracy depending on the state in question, suggesting that different states have different levels of regularity in their patterns of constituent nucleosome organisation. I then also use this same technique to study nucleosome positioning data

from cancer patients and healthy individuals. Subsequently I use visualisation techniques to isolate the patterns that are being recognised by the CNN and study these outputs.

3.2 Introduction

In recent years, new light has been shed on the way that gene-expression and cellular identity are regulated via chromatin. One problem that has evaded a solution, however, is what is the set of rules that govern nucleosome positioning and to which extent do these rules influence gene expression pathways and cellular identity. What determines that in a given genomic locus containing an important regulatory region, in one cell-type, will have nucleosomes closely packed together while in the case of a different cell-type, that same locus can have a different organization of its constituent nucleosomes? Previous research has not managed to distill a clear set of rules in terms of what DNA sequences most attract nucleosome formation (Segal *et al.* 2006; Salih *et al.* 2015), and furthermore, the DNA sequence only partially influences nucleosome positioning. Hence it is useful to study large portions of the genome to look for general trends in terms of how nucleosome organization occurs. Regions that have a high density of bound nucleosomes show little transcriptional activity (heterochromatin) whereas those that have a low density of nucleosomes show high transcriptional activity (euchromatin). Beyond this broad view of chromatin, one can further divide chromatin into subtypes known as chromatin states.

3.2.1 Chromatin state

The concept of a chromatin state aims to combine the effects of covalent chromatin modifications and presence of different protein combinations at each genomic locus. The attitude herein is that the simultaneous presence of multiple separate modifications at the same locus will give rise to a

segment of chromatin with specific properties such as conformation and expression rate. A genomic locus with independent properties is said to have 'state'. Hence it is now possible to categorise chromatin into different states and subsequently study the implications in genomic regulations (Ernst and Kellis 2012; Di Pierro *et al.* 2017).

3.2.2 Relationship between nucleosome organisation and chromatin state

The rules via which nucleosomes are positioned on the DNA fiber are not fully clear (Segal *et al.* 2006; Salih *et al.* 2015). However it is known that different regions of the genome, depending on the chromatin state have different densities of nucleosome organisation. This is a trend that emerges when averaging over many regions. A measurement that can to characterise the rate of nucleosome packing in chromatin is the nucleosome repeat length (NRL).

3.2.3 Inferring NRL from MNase-seq and other types of data

The NRL is the average distance between the centers of neighboring nucleosomes (Beshnova et al. 2014). As explained in Figures 4 and 10B, a brief outline of the NRL calculation procedure is the following: (1) the experimental coordinates of positioned nucleosomes are used to plot the frequencies of distances between nucleosome centers (2) then a linear fit through the peak summits is made, and the slope of the resulting best-fit line gives the NRL.

The most typical experimental protocol for finding the positions of all nucleosomes in the genome is MNase-seq (detailed above). Another protocol is chemical mapping, which possesses key advantages over the MNase-seq protocol and thus gives enhanced precision of nucleosome placement. However chemical mapping is not as readily available in all organisms as the more common MNase-seq (Zentner and Henikoff 2012; Voong *et al.* 2016).

3.2.4 Calculating NRL for small genomic regions is challenging

Calculating NRL is relatively straightforward when dealing with long portions of chromatin e.g. the length of a chromosome. However as one moves to lower resolutions, studying the chromatin fiber in greater detail, it becomes difficult to consistently have a reliable measure of NRL in relatively short pieces of chromatin, due to the small number of nucleosomes (see Figure 4). Hence for small segments of chromatin, a connection between the patterns of nucleosome positioning and the higher order state of the chromatin has never been demonstrated.

3.2.5 Machine learning used to characterise different types of chromatin

Given the above problem, we sought to develop a pipeline that could reliably detect the relationship between nucleosome positioning and higher order chromatin architecture. We wanted to see if the state of a segment of chromatin could be predicted, solely from the distribution of constituent nucleosomes. This was done in order to test a hypothesis that nucleosomes are positioned differently relative to one another in a way that is manifestant of chromatin state. The concept of a pattern of nucleosome positioning that is distinct to a specific type of chromatin will be referred to as ‘nucleosome positioning signature’ (NPS) from here on. We used a convolutional neural network (CNN) as a predictor of chromatin state based on the NPS. This concept is illustrated below in Figure 18.

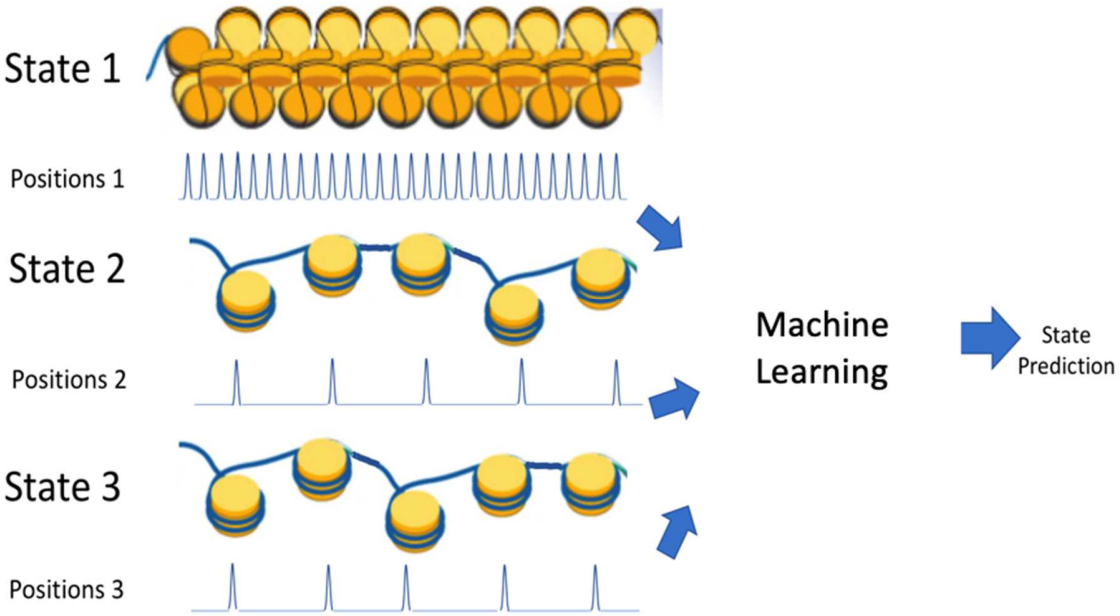


Figure 18. An illustration of the general concept to predict chromatin state using nucleosome positioning. The nucleosome coordinates are converted into a signal which can be interpreted by an ML model which is then used to predict the chromatin state of each input region.

Figure 18 above provides a general outline of the concept that was used in this project, namely that chromatin state could be predicted via the signal derived from NPS. It is worth noting that the chromatin state label is the result of a predictive model and not something that has been experimentally observed. From this point of view, the experiment is flawed. However, all chromHMM datasets used in this study were taken from independent publications where the combination of marks in each state were diverse so that we could be sure that the different labels were biologically relevant.

The input data used in the above illustrated pipeline came in three formats, which are listed as follows: (1) standard MNase-Seq data (2) chemical mapping data which, as mentioned above (see

1.2.1.3 Chemical mapping) has enhanced precision in nucleosome placement. (3) We also applied our technique to H3 assisted ChIP-Seq data from humans (see 1.2.1.2 MNase-assisted histone H3 ChIP-seq). Depending on the state in question, we achieved 60-99% accuracy in this task. We performed both binary classification (predicting whether a segment of chromatin was of a particular state or not) and multi-label classification (which of many states an input chromatin segment belongs to).

3.2.6 The pipeline adapted for use as a preliminary cancer diagnostic tool

After demonstrating that machine learning could be used to predict chromatin state in all of the above mentioned data types, we also applied our technique to cancer diagnostics. I used the dataset of MNase-assisted ChIP-seq with an antibody against histone H3, which came from peripheral blood B-cells from cancer patients (CLL) and healthy individuals (Mallm et al., 2019). In this case I adapted the pipeline to predict whether a segment of chromatin is derived from a cancerous or non-cancerous sample.

3.3 Methods

3.3.1 Sourcing of ChromHMM data

3 separate ChromHMM datasets were used in this project: (1) 15-state model dataset for mESCs sourced from (Bogu *et al.* 2016), used in "3.3.6.1 Binary Classification". (2) 20-state model dataset for mESCs sourced from (Juan *et al.* 2016) used in "3.3.6.2 Multi-label Classification". (3) In the case of the human CLL data, I used the 12-state dataset from (Mallm *et al.* 2019). The latter dataset was used in section "3.3.6.1 Binary Classification".

3.3.2 Mapping

All data were mapped using Bowtie (Langmead *et al.* 2019). In the case of mouse data, we mapped to the mm9 genome and for human data we mapped to the hg19 genome. The mapped data were then converted to BED files using the NucTools pipeline (Vainshtein, Rippe and Teif 2017). In the case of paired-end MNase-assisted H3 ChIP-seq the BED files contained the coordinates of the nucleosome start and end positions. In the case of chemical mapping the BED files contained the coordinates of the dyads of the two adjacent nucleosomes. MNase-seq and chemical mapping data for mouse embryonic stem cells were from Voong et al. (2016). MNase-assisted H3 ChIP-Seq data for B-cells from peripheral blood of healthy people and CLL patients were sourced from the CancerEpiSys consortium (Mallm *et al.* 2019).

3.3.3 NRL calculation

The estimation of NRL was done via the calculation of phasograms from MNase-Seq data. The "phasograms" representing the histograms of dyad-to-dyad or start-to-start distances were calculated with NucTools (Vainshtein et al., 2017) (script "nucleosome_repeat_length.pl") using default parameters. Dyad-to-dyad distances were calculated using the center of each MNase read in all regions of interest. The most common distances were picked and plotted as a function of themselves and the resulting best fit line gave a slope which was the NRL. This was done for each chromatin state to initially get a sense of how the nucleosomes were generally organized in each individual state.

3.3.4 Nucleosome positioning representation for ML

The general strategy to represent the positions of nucleosomes in such a way that they could be fed to a ML model is as follows:

In the case where nucleosome reads were unique and non-overlapping, a binary sequence was used to encode the nucleosome center positions. At each bp coordinate where the center of a nucleosome was positioned, '1' was placed and '0' was placed at all other bp positions. This resolution was used to ensure as we wanted to test for the possibility that even small changes of nucleosome position could change the chromatin state. This process was done at all locations of the state of interest in a sliding window. This concept is illustrated in Figure 19. The code used in this calculation is available in Appendix 5.2.2.

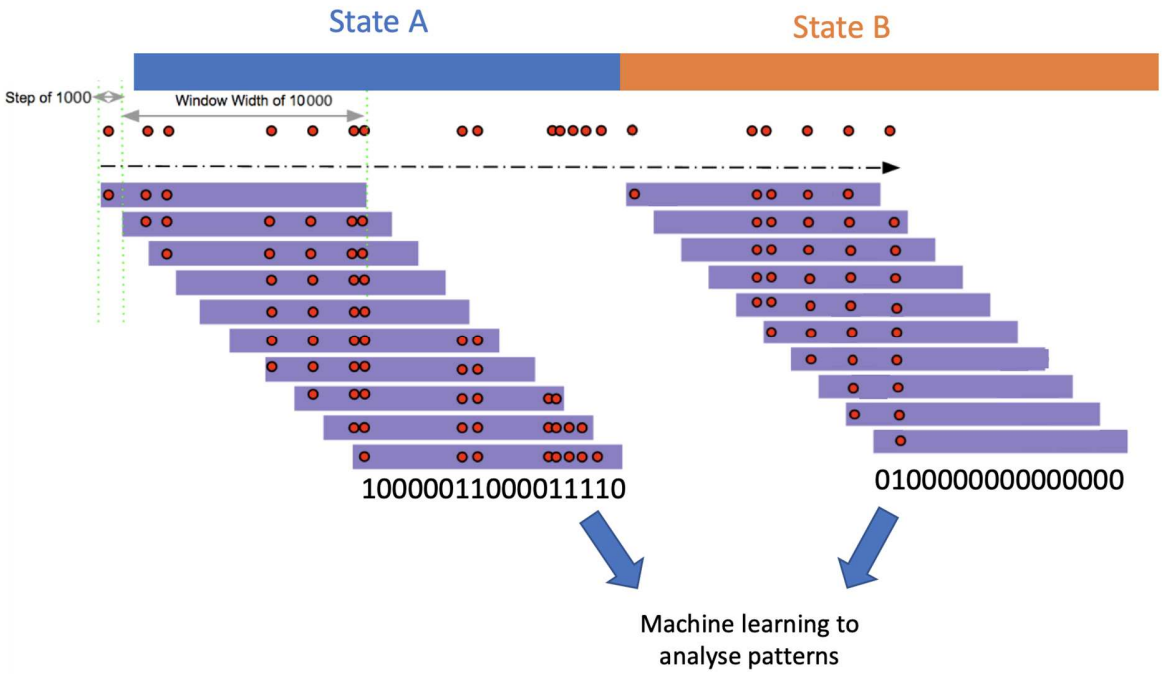


Figure 19. A scheme of the nucleosome positioning data pre-processing to prepare a dataset for machine learning in the case of unique nucleosome positions derived from chemical mapping. The top bar represents a portion the reference genome coloured by separate chromatin states, the red dots represent the position of nucleosome centers, the purple bars represent the rows of data recorded in a sliding window moving along the genome. '1' was used at each nucleotide position where the center of a nucleosome was located and '0' was placed at all other nucleotide positions. This was done across the whole genome with a sliding window, separately for each different state.

Thus the nucleosome positioning was imputable in a ML model. The above case shows what the input rows would look like if the nucleosomes reads were unique and non-overlapping. However, it should be noted that this is not the case for the majority of datasets that were used in this project. In the cases where there could be more than one nucleosome read at a given position, a non-binary count number was used to represent the number of nucleosome reads at each bp. In the sections below, it will be noted which of these 2 representation strategies is used in each case.

3.3.4.1 Representation of MNase-seq and MNase-assisted histone H3 ChIP-seq data

When using paired-end MNase-seq data, we identified nucleosome centers by averaging between the start and end of each paired-end nucleosome read coordinate. There could be many nucleosome centers at a given point. Hence, each bp position was assigned a value equal to the number of occurrences of a nucleosome center at that point. In the case of paired-end MNase-assisted histone H3 ChIP-seq data, the data were also prepared in the same way as above in the case of the MNase-seq data.

3.3.4.2 Representation of chemical mapping data

Chemical mapping data was studied in two forms: (1) filtered dataset with inferred non-overlapping nucleosome positions and (2) unfiltered dataset with raw mapped paired-end reads, both from (Voong et al., 2016). In the case of the filtered data, the nucleosome locations were unique and hence '1' was placed at each nucleosome center. These filtered data were downloaded from GEO (GSE82127), in a text file (GSM2183909_unique.map_95pc.txt- made using the software NCPScore (Xi *et al.* 2014)) compiled by Voong et al. In the case of the raw mapped paired-end reads, for each bp coordinate a value equal to the number of mapped nucleosome-read centers was assigned. Since the chemical mapping protocol cleaves at the center of the nucleosome- the start of the mapped paired-end read marks the nucleosome center (dyad), hence this coordinate was used when encoding nucleosome positions rather than the average between the start and end (as was used in the case of MNase data).

3.3.5 Parallelisation of data preparation

The size of the BED file datasets (containing nucleosome positions) that were used in this project ranged from ~10-100GB. Hence the task of extracting the nucleosome reads that were in a region of interest in a way that was scalable and fast was not straightforward. The strategy used to make the comma-separated value (CSV) files that marked nucleosome center positions is illustrated in figure 20 below:

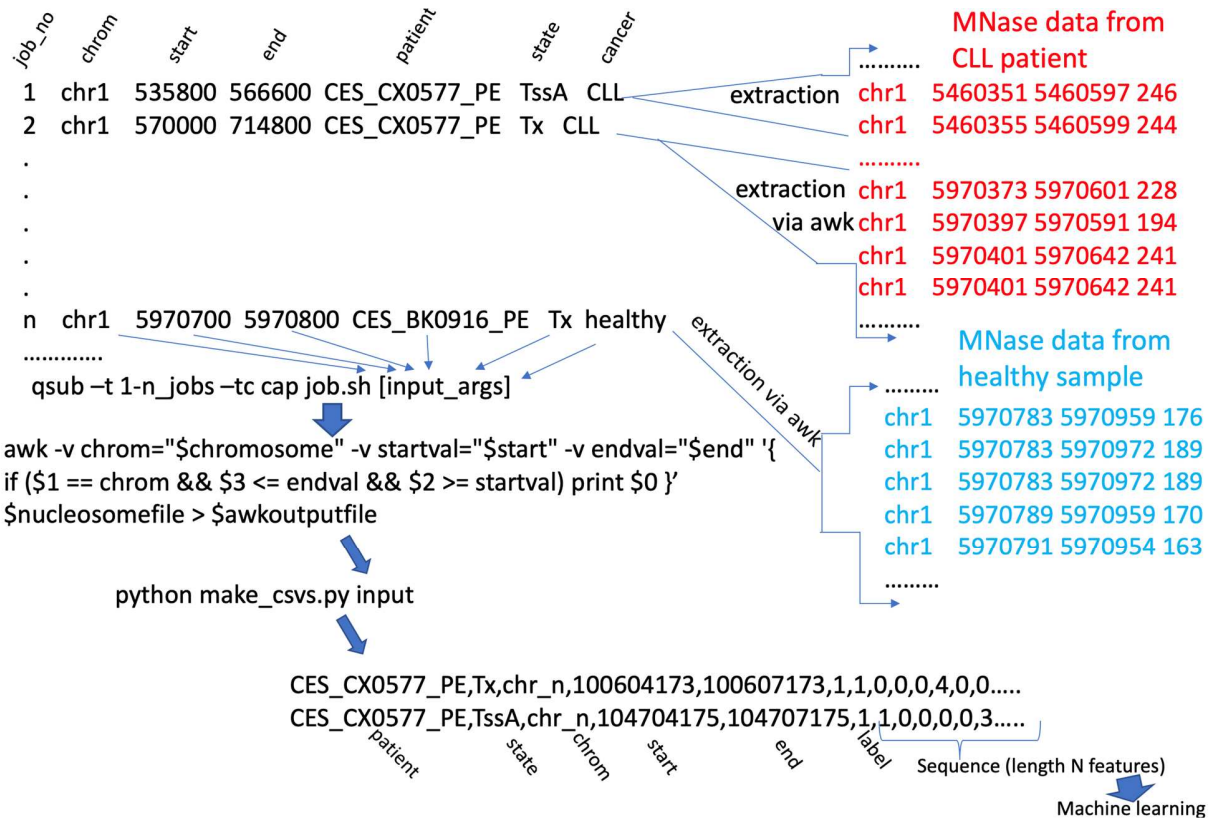


Figure 20. The pipeline used to make CSV files encoding the nucleosome positions. For each region of interest with a start and end position on a particular chromosome, the input BED file contains the coordinates of all nucleosomes that constitute that region. The locations of the nucleosomes in an individual region were extracted from the initial BED file to a temporary BED file for each region. Hence the locations of interest were submitted, in a bash script, as a series of parallel jobs to a computer cluster. In this way, the process of making the CSV files was parallelised by submitting all of the jobs at once, with a cap on the number of jobs that could run simultaneously. The CSV columns contained the chromosome coordinates, the patient number (if the data is CLL in origin), the chromatin state of the region, the label (binary or multi-label) and the final columns were the sequence to be studied with ML.

Figure 20 describes how CSV files marking nucleosome positions were made for each individual locus. The CSV files were made in parallel by submitting the combinations of locations to a cluster at once using the command '*qsub -t 1-N_jobs -tc cap job.sh*' where '-t' inputs the range of jobs and '-tc' inputs the cap on the number of jobs that could run simultaneously. When all of the CSV files were made for all individual regions they were all concatenated into one large CSV file to be studied with machine learning. See the code in Appendix 5.2.2.

3.3.6 Machine learning. Data preparation and analysis

3.3.6.1 Preparing real vs null data for binary classification

Binary classification was performed in order to assess whether a given segment of the genome of a particular chromatin state could be distinguished from the rest of the genome using its NPS. To prepare a suitable dataset for this task, we required regions that were from the state of interest (to provide rows where the label was '1') and regions that were not (to provide rows labelled '0'). To do this, the library BedTools (Quinlan 2014) was used to randomly shuffle the BED coordinates of a given chromatin state in a way that the resulting shuffled regions could be from anywhere in the genome but the original state. Subsequently, rows of data were recorded in a sliding window in the real state regions and the shuffled regions (see above "3.3.3 Nucleosome positioning representation" and "3.3.4 Parallelisation of data preparation") and the resulting rows were labelled with a '1' or '0' accordingly.

3.3.6.2 Reading and shaping of CSV input

The input data for machine learning were read into a python environment using the library pandas (McKinney and Team 2015). Subsequently they were reshaped using the library NumPy (Oliphant and Millma 2006). An extra dimension of 1 was given to the X input such that tensorflow could accept it as a 3D tensor. See the code in Appendix 5.2.2.

3.3.6 Classification

3.3.6.1 Binary classification

Binary classification was used when predicting whether an input segment was of a particular chromatin state or not. 'Of a particular state' here means that an input segment was from a bona fide region of the state in question (label=1), whereas 'not present' means that the input was from a randomly picked region that was positioned anywhere but the state in question (label=0) (see above '3.3.5.1 Preparing real vs null data for binary classification'). The ChromHMM data used for this task was that of the 15-state model (Bogu *et al.* 2016). The states used for this classification task were '12_heterochromatin', '13_heterochromatin', '14_heterochromatin' and '1_elongation'. This is because each of these states covered enough of the genome to provide substantial datasets to allow for sufficient training and accurate predictions. These states varied widely in terms of their percentage coverage of the genome. Hence the quality of chromatin state identification was assessed with different resolutions for each of these states. The lengths of the input chromatin segments to be classified for each state was as follows: '12' and '13_heterochromatin', covered the majority of the genome, and hence the segments were taken with a sliding window of width 10,000 bp, and a sliding step of 1,000 bp. '14_heterochromatin' covered much less of the genome than '12' and '13' and hence the stretches of this state were generally much smaller- thus the

recorded segments were taken in a sliding window of width 3,000 bp with a sliding step of 300 bp. '1_Elongation' covered small, relatively rare portions of the genome. These segments were taken using a sliding window of width 1000 bp and a sliding step of 100bp.

With regard to the CLL part of the project, binary classification was used for two separate tasks: (1) to predict whether a segment was of a particular chromatin state or not, in the same way as above (2) to predict whether a segment was from a cancerous sample or not. For both cases the segments were taken in a sliding window of width 3,000 bp with a sliding step of 300 bp.

The model used in all cases of binary classification was a simple 1 layer CNN composed using tensorflow and keras (Abadi *et al.* 2015). The model architecture was as follows: Convolutional layer (100 filters), a variety of convolutional windows, a max pooling layer and a sigmoid layer. The optimiser chosen was 'rmsprop'. See the code in Appendix 5.2.2.

3.3.6.2 Multi-label classification

Multi-label classification was used in the cases when multiple chromatin states were a possibility and hence rather than predicting whether an input nucleosome sequence *was* of a particular state or not, the task was to predict *which* state an input segment was. For this task, rather than the 15-state ChromHMM dataset used in the case of binary classification (see above), a 20-state dataset was used (sourced from (Juan *et al.* 2016)). This was done as the state regions in this dataset were much more evenly sized than in the case of 15-state dataset, which was a necessity as all inputs needed to be the same length in order for a multi-label system to be feasible. Only states that had a diverse combination of marks were selected. Hence it was possible in this way to have a dataset with sufficient numbers of input sequences from multiple different types of chromatin.

The input segments for this task were taken with a sliding window of width of 3000 bp with a sliding step of 300 bp.

For the model design, a similar architecture as in the case of binary classification was used for multilabel classification, except the final layer had the same number of neurons as there were categories- in such way that the separate neurons would activate depending on what the predicted classification label was. See the code in Appendix 5.2.2. See also '3.3.5.7 Grid search to see the how an optimal set of parameters were selected for the multi-label classifier'.

3.3.6.3 Handling of imbalanced datasets

To handle imbalanced data (i.e. datasets with more '1' labelled rows than '0' or vice versa), in the case of binary classification, 'undersampling' was used. This means that to ensure that the datasets were balanced, it was determined if there were more rows with the label '1' than with the label '0' or vice versa, then depending on which category had more rows, a number that was similar to that of the lesser category was randomly sampled from the larger category. This meant that there were equal numbers of '0' and '1' labels. See the code in Appendix 5.2.2.

3.3.6.4 10-fold cross-validation

The sklearn library (Pedregosa *et al.* 2011) was used to split the data into training and test data and redo this across 10 rounds of cross-validation. See the code in Appendix 5.2.2.

3.3.6.5 Performance assessment

In the case of binary classification receiver operator curves (ROCs) were used to assess performance. A ROC was calculated in each round of cross validation and the performance was

assessed by the average AUC +/- the standard error of all of the AUCs across the different ROCs. The code was adapted from the tutorial found at this link: (https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc_crossval.html). See the code in Appendix 5.2.2.

In the case of multi class labelled systems precision-recall curves were used to stratify the different states in terms of their recognisability. Precision and recall were calculated using sklearn commands which employ the equations: precision=TP/TP+FP; recall=TP/TP+FN. This was again done in the same way as the above ROC calculations, where 10 rounds of CV were done and a precision-recall curve was calculated for each round and the average curve was plotted. See the code in Appendix 5.2.2.

3.3.6.6 Training the model

Models were fit to the input data with a batch size of 100 and 3 epochs (training iterations). See the code in Appendix 5.2.2.

3.3.6.7 Grid search used to find optimal model parameters

A grid search was performed to find the best set of parameters to maximize the accuracy of the multi-label classifier. The grid was composed of 74 distinct sets of parameters. The parameters that were varied in the grid were: the size of the convolutional window (500-2000 in increments of 500), the size of the max_pooling window (60, 75, 100), the activation function of the last layer of the model ('sigmoid', 'softmax', 'relu'), the dropout rate used in the final layer (20%), and the optimizer used ('rmsprop', 'adam'). The grid was then looped over and for each set of parameters 10 fold CV was performed and a precision recall curve (averaged across all rounds of CV) was

plotted for each set of parameters in the grid. From this, the best set of parameters was judged to be the following: convolutional window width: 2000 bp; the size of the max_pooling window: 75 bp; the activation function of the last layer of the model: 'softmax'; the dropout rate in the final layer: 20%; the optimizer: 'adam'. See the code in Appendix 5.2.2.

3.3.7 Finding characteristic NPSs using explainable AI

In the case of SHAP, the sequence of interest was input into the command 'DeepExplainer' and contrasted against the background of 100 randomly selected inputs. This output is a 3D matrix which then needed to be reshaped to a 1D vector that could then be plotted (Shrikumar, Greenside and Kundaje 2017). See the code in Appendix 5.2.2.

3.3.8 Finding characteristic NPSs using GANs

The GAN generator was composed of 3 layers of convolutional layers that took a random noise vector as input and produced a sequence that the discriminator could take as input. The discriminator was composed of the trained model described above in "3.3.5.1 Binary Classification" (but with one difference- the chosen optimizer was 'Adam' with a learning rate of 0.0002 – this was to be consistent with the tutorial code available in the links below). The GAN was run on a loop for 500 epochs (rounds). During each epoch, the following steps occurred: (1) 500 random noise vectors were input to the generator which then produced 500 fake sequences (2) 500 real nucleosome inputs were randomly selected from a dataset derived from the filtered chemical mapping data (3) the discriminator classified the 500 fake sequences and 500 real ones and the classification accuracy was recorded for that epoch. (4) The loss value for the training of the discriminator and generator was also recorded. (5) The statistical distance between the

distributions of number of nucleosomes in each sequence in the fake and real inputs was recorded via the Wasserstein distance algorithm which was a way of keeping track of how similar to the real input the generated sequences were. After 500 rounds of this process the loss profiles, classification accuracy and Wasserstein distance for each epoch was plotted. The loss profile numbers are an indication of how far from the label the prediction is. Hence the loss profile at each epoch can be interpreted as how much learning the generator/discriminator is doing and hence if the loss of one model is high the loss of the other model will be low- indicating that the model with high loss is 'losing' to the other. The code used was adapted from 2 primary example scripts on github which can be found at: (https://github.com/eriklindernoren/Keras-GAN/blob/master/gan/gan.py, https://github.com/KordingLab/lab_teaching_2016/blob/master/session_4/Generative%20Adversarial%20Networks.ipynb).

3.4 Results

3.4.1 NRL measurement in different chromatin states

As a preliminary analysis I calculated NRLs across 15 different chromatin states in mouse embryonic stem cells (ESCs) using the states annotations from Bogu et al. (2016). This chromHMM dataset is currently the "golden standard" this type of data for mESCs because the predicted states are so diverse in their marks and this allows for meaningful biological functions to be inferred in the case of each state (Bogu *et al.* 2016). These results are documented below in Figure 21:

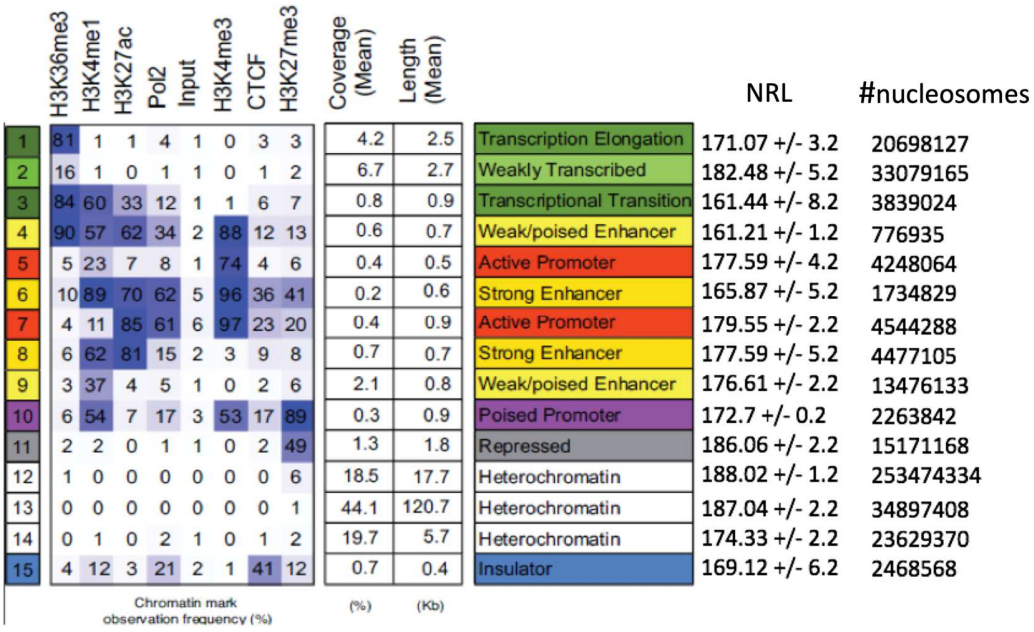


Figure 21. NRLs measured across different chromatin states. 15 chromatin states were assigned to different regions in the genome based on the input presence of various histone marks or of CTCF. The left-most panel documents the abundance of each studied mark in each state. The second and third columns show the amount (%) that each state covers the genome and the average length of those regions (kilobases). A functional label was given to each state in the right column to provide context as to the inferred biological function of these regions. These data were taken from Bogu et al. 2016. The NRL and total number of nucleosomes were calculated for each state and are shown in the two outer-right columns.

From the above Figure 21, interesting differences in NRL are measured in similarly labelled states. See that the states labelled 'Strong Enhancer' have NRLs of 165 and 177- this becomes interesting in light of the fact that CTCF is present in one and not in the other which is in line with the findings of Chapter 2 above. Furthermore, the states labelled promoters have more

consistent NRLs that are similar to those seen of the strongest binding CTCF sites seen above in Chapter 2. One could speculate that the similar arrangement of nucleosomes seen at promoters and that shown in the above paper is consistent and that such a strict conformational arrangement is required for transcription initiation at promoters in a way that can not vary like in the case of enhancers.

I used convolutional neural networks to distinguish trends in the patterns of nucleosome positioning between differently labelled lengths of chromatin. This will be outlined in the following section.

3.4.2 Preparing nucleosome positioning data for machine learning

I took the positions of nucleosomes from 2 different types of experimental data (MNase-seq and chemical mapping), see 3.3.1 Data Preparation. These were prepared in such a way that they could be fed to the CNN: a binary sequence was used where the nucleosome positions were ordered and subsequently a '1' was used at each nucleotide position where the center of a nucleosome was positioned and '0' was placed at all other nucleotide positions. Binary sequences of a particular length were taken from everywhere in the genome, in a sliding window, where the said state occurred (see details in the Methods section). This resulted in rows of data where the label was '1' i.e. rows documenting loci where the chromatin state of interest was present. To provide corresponding '0' labelled rows, the locations of the given state were shuffled randomly and the same sliding window operation was used to mark the locations of nucleosomes in loci where the chromatin state was not present.

3.4.3 NPS-based classification of chromatin states using machine learning (ML)

3.4.3.1 Binary ML Classification of chromatin states

To provide a proof of concept in determining whether the chromatin state of a locus could be identified by the NPS, binary classification datasets were prepared by taking the locations of the chromatin state of interest, labelling these data as '1' and generating randomly sampled regions from anywhere in the genome where the state in question was not present and labelling these data as '0'. This provided a dataset of rows labelled as '1' and '0' for when the state was/wasn't present and allowed one to test if the NPS of a state could be reliably detected against noise, as demonstrated in Figure 22 below.

The performance in identifying the chromatin state from NPS varied widely across states. This would suggest that certain chromatin states have a more regular distinct NPS than others. In particular, the active state '1_Elongation' is regularly recognizable across both datasets.

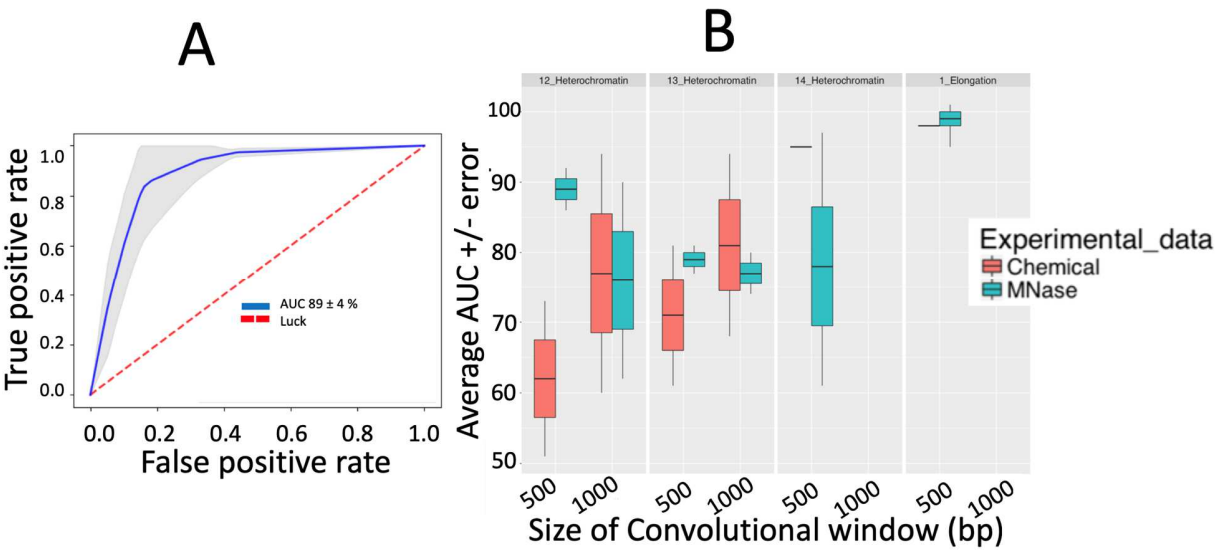


Figure 22. The performance in binary classification was assessed by the area under the curve of receiver operator curves. (A) An examplary ROC curve is shown. The blue line represents the

average performance (89% area under the curve (AUC)) across 10 rounds of cross-validation the shaded area represents the standard error (4%) (B) The spread of ROC performance rates (given by the AUC) across all studied states are plotted on in boxplots with the average AUC marked by the middle line and the maximum and minimum marking the average AUC plus and minus the error. For each state, the different convolutional windows that were used are marked on the x-axis. Red corresponds to the chemical mapping and green to MNase-seq data.

Figure 22 above documents the spread of performance rates (AUCs from different ROCs that resulted form 10-fold cross-validation). The active state '1_Elongation' shows consistent degree of distinguishability in the case of both MNase and chemical mapping. One might speculate that this is in line with what is shown in Figure 21 where the active states generally have a lower NRL than heterochromatin states- which means that the NPS may be more regular in these regions.

3.4.3.2 Binary ML classification of chromatin states across cancer patients

After having established the ML classification pipeline in mouse ESCs, I applied this approach to nucleosome positioning data in human. The experimental dataset was determined by the CancerEpiSys consortium in B-cells taken from peripheral blood from CLL cancer patients and healthy individuals (Mallm *et al.* 2019). When assessing the identifiability of a chromatin state from NPS in humans, I decided first to train the ML model on one patient and then test it on another patient. In total I included six healthy and six CLL samples as detailed in Figure 23 below.

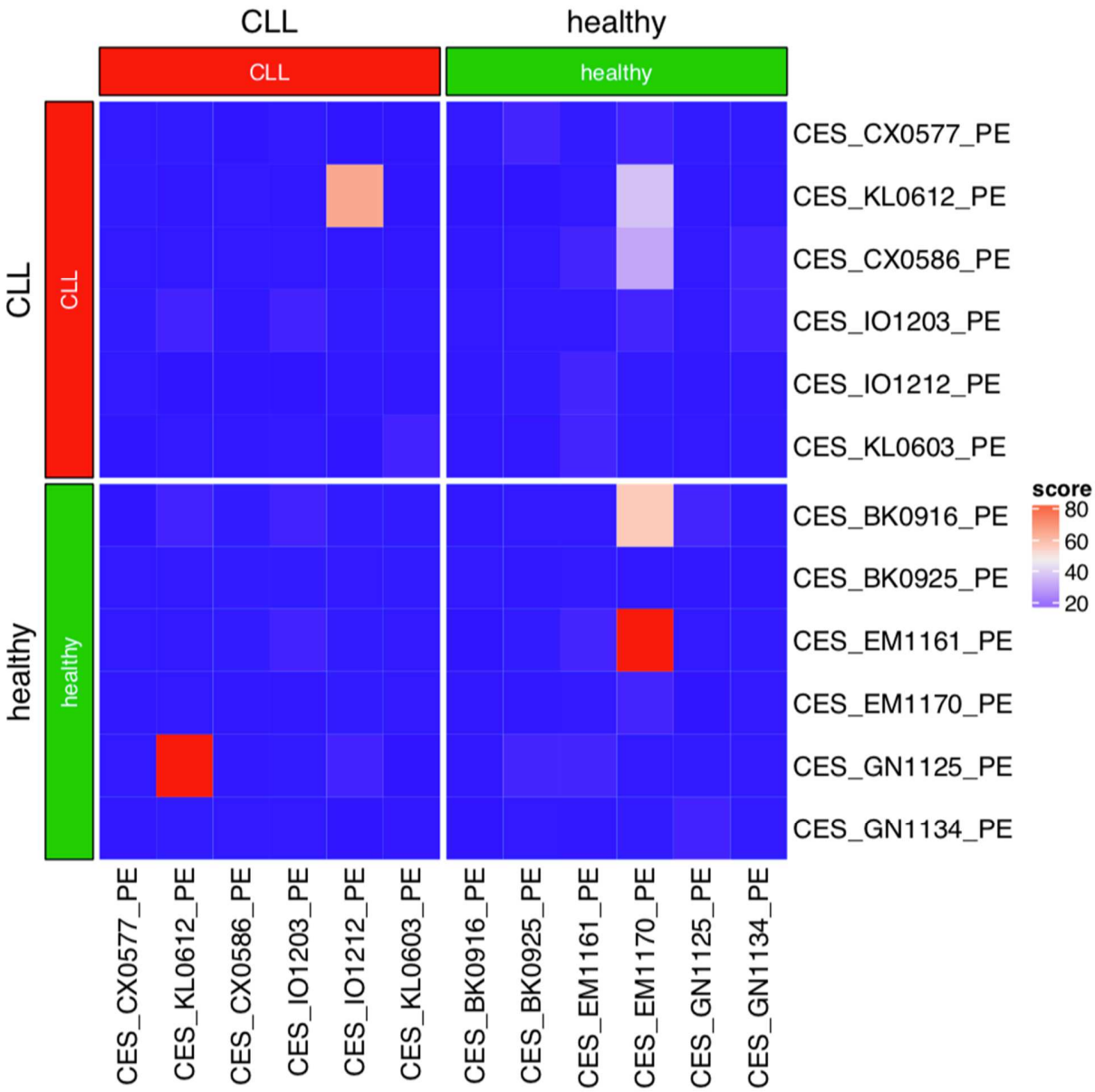


Figure 23. A heatmap documenting how well the presence of the state 'Tx' (transcriptional elongation) could be identified when training on data from one individual and predicting on data from another. The x-axis labels mark the individuals on which training was performed and the y-axis marks the individuals on which testing was performed. The intensity of the colours represents the prediction accuracy (AUC) divided by the standard error of each of the ROCs across 10 rounds of cross-validation (see Methods).

Figure 23 above, documents the consistency (average AUC-ROC normalised by the spread of AUCs across 10-fold CV) with which a chromatin state can be identified when training in one patient and testing in another. The performance across all patients in terms of identifying the state of interest was quite consistent, even between cancer patients and healthy individuals. At first, I thought this was counter-intuitive as one would have expected that the nucleosome organisation would be perturbed in cancerous patients to the extent that patterns that gave rise to particular states would not be consistent with that of healthy individuals. On reflection, this result seems realistic as it should be the case that the majority of loci in the cancer genome are unperturbed while only a minority of dysregulated candidate regions have disrupted NPSs? This would explain why learning is transferrable between cancer and non-cancer conditions.

3.4.3.3 Multi-label ML classification of chromatin states

To test if the performance consistency improved in a dataset with multiple lables, a dataset was prepared with multiple different chromatin states. Multi-label classification was performed using the regions from a 20 state ChromHMM dataset (mESC) from (Juan *et al.* 2016) (Figure 24). To identify the set of model parameters (i.e. batch size, convolutional window etc.) that would maximize the prediction accuracy, a grid search was performed. 60 different sets of parameters were used to train and test models on the dataset. The set of parameters that performed the best resulted in the below precision-recall curves in Figure 24.

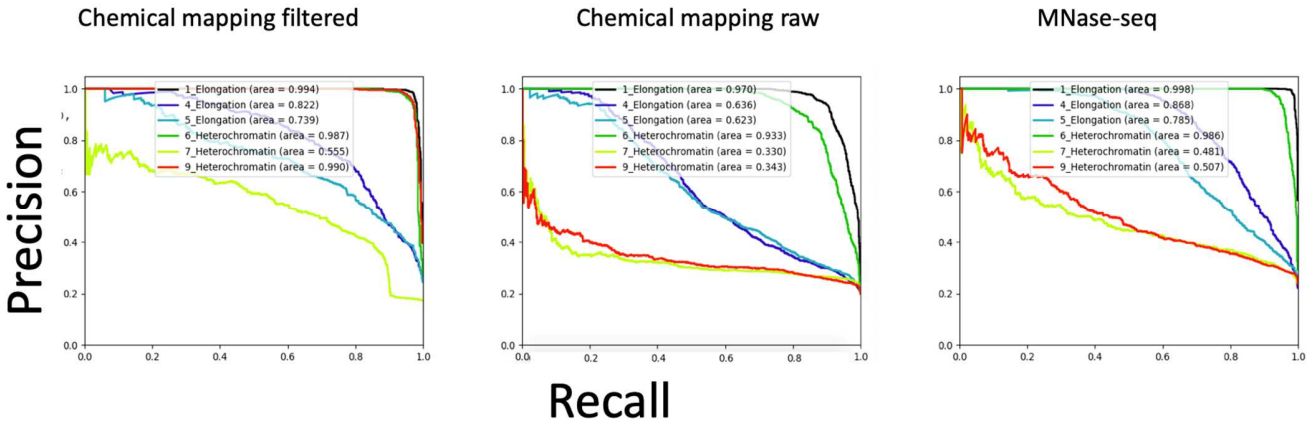


Figure 24. Precision-recall curves for the top performing multi-label classifier across different experimental inputs. The experimental data is based on the filtered, non-overlapping nucleosome dataset derived using chemical mapping by Voong et al (2016). The used ChromHMM dataset is from mESC (Juan *et al.* 2016).

Upon switching to a multi-label system, the performance consistency across the different input data greatly improved in comparison with the binary classification performed for the same dataset (Figure 22). A possible reason for this improvement could be that in the case of binary classification, the 0 labels were randomly selected and may have been more difficult to distinguish from the real ones whereas the complexity of patterns was more contained in the case of a multi-labelled system. Furthermore, the active states generally stood out as having a consistent and regularly identifiable pattern. This is in line with the experimental results in section 3.4.1, where the NRLs in active regions were shown to be smaller than in the case of heterochromatin and hence, perhaps, more regular phasing of nucleosomes was present in these regions.

3.4.4 NPS-based ML method for cancer diagnostics

After having shown that we could distinguish different chromatin states, we decided to do apply the technique to MNase-assisted histone H3 ChIP-seq data from patients with chronic lymphocytic leukemia (CLL) and healthy individuals – to see if nucleosome positioning patterns could be inferred to distinguish the cancer samples from the non-cancerous ones. Here we wanted to test if the nucleosome organisation would be perturbed in cancerous patients to the extent that they could be distinguished from that of healthy individuals. To assess this possibility we performed binary classification while labelling the data according to whether they came from a cancerous (rows labelled '1') or non-cancerous (rows labelled '0') (Figure 25).

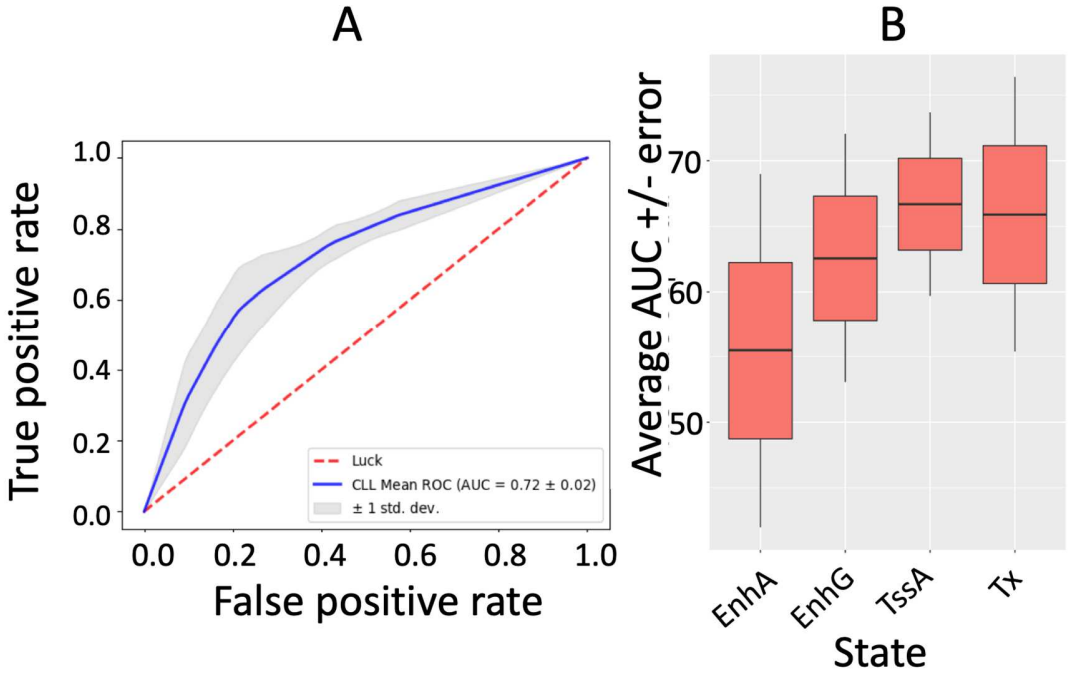


Figure 25. Prediction accuracy of discriminating CLL vs healthy B-cells from peripheral blood, based on MNase-assisted H3 ChIP-seq nucleosome positioning patterns. (A) ROC curve for identifying cancer vs healthy with all chromatin states included. (B) Performance in identifying cancer vs healthy across chromatin states.

Figure 25 shows that this model successfully identified B-cells from CLL patients with 72% AUC (when not separating by the chromatin state state). This suggests that such information could potentially be used in future diagnostic tests if predictive power is improved.

Despite the success of the above experiment, it is not expected that one would typically analyse the NPS of any single locus in the genome to predict the presence of cancer. Instead nucleosome patterns may be perturbed at many specific loci. Therefore, we tested to see if there might be a confounding factor in the data that could be inflating the classification performance (Figure 26).

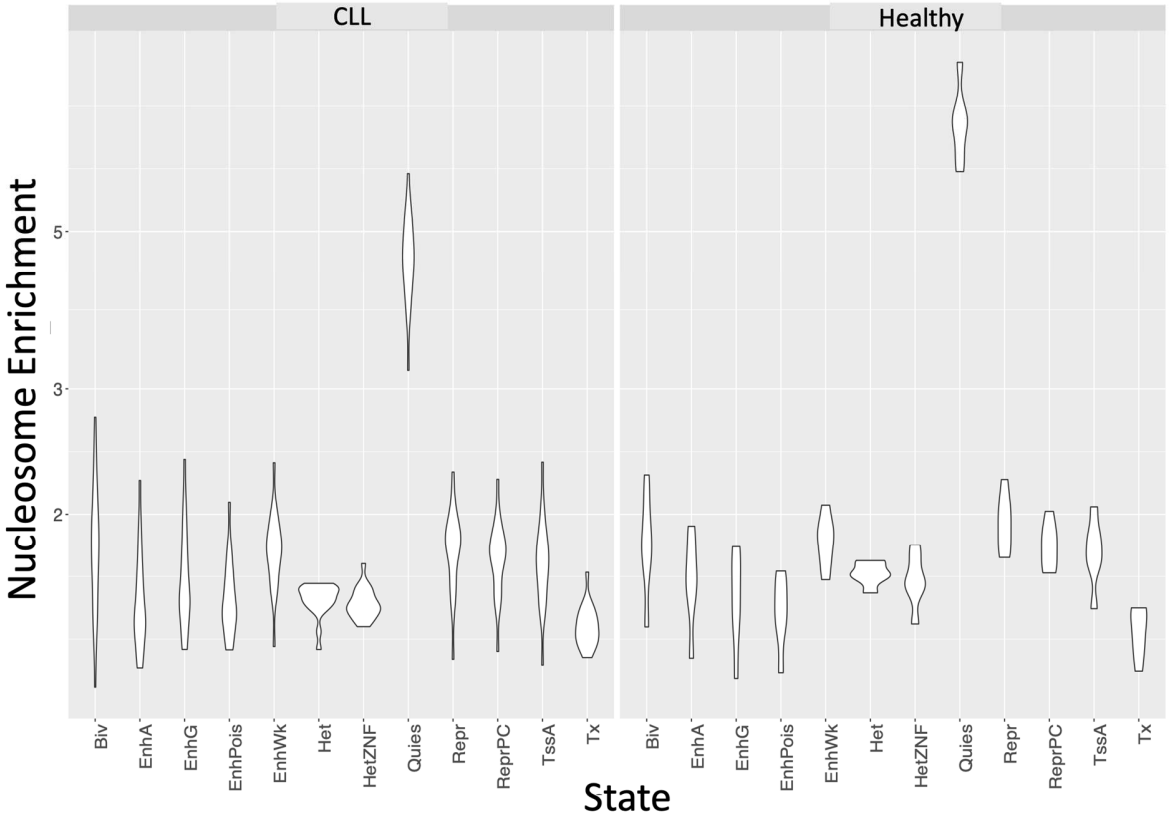


Figure 26. Distribution of the enrichment of nucleosomes in different chromatin states states determined by ChromHMM (Mallm et al., 2019). The enrichment was defined as the density of nucleosomes in the state of interest divided by the genome-average density, calculated separately for B-cells coming from CLL patients and helthy individuals.

Despite the success in distinguishing cancer samples from non-cancerous ones, some problems have been identified in the H3 experimental data as illustrated in Figure 26. It appears that cancers cells have significantly smaller nucleosome density than healthy cells for the quiescent chromatin state (which accounts for the majority of the genome in this case). To calculate the enrichment of nucleosomes in a given chromatin state, I divided the density of nucleosomes in the real locations within this chromatin state by the simulated genome-average nucleosome density. This analysis showed that there are differences in nucleosome enrichment in cancer patients and healthy individuals (Figure 26). Hence the input numbers of nucleosomes in row data will be generally quite different between cancerous and healthy samples, which could confound the ML model performance.

3.4.5 Identification of the NPSs characteristics for different chromatin states/samples

I was then interested to see if the CNN could report the patterns that were recognized when deciding whether the input was from a particular state from a cancerous or non-cancerous sample. In attempt to extract biologically meaningful patterns we used 3 techniques:

- Explainable AI
- Generative adversarial networks (GANs)
- Separately studying the patterns of high and low scoring inputs.

These analyses are outlined below.

3.4.5.1 Finding characteristic NPSs using explainable AI

There have been several recent efforts to increase the interpretability of machine learning models. An example is the software "SHapley Additive exPlanations" (SHAP). This software performs the following: a row with nucleosome positions corresponding to an individual genomic region is taken as input to the trained model and SHAP outputs a vector indicating which parts of the input the model is most reacting to (Lundberg and Lee 2017). This strategy was adapted for our input 1D vectors with nucleosome positions – see below in Figure 27 how example input vectors were studied in contrast to the corresponding SHAP values.

Note: The chromatin state '9_heterochromatin' and the filtered chemical mapping nucleosome input data were chosen as a use case to demonstrate this method because the trained model performed relatively well in the binary classification task of distinguishing profiles from "9_heterochromatin" regions vs regions that were not in that state (89% AUC).

In Figure 27 exemplary input sequences are compared to the corresponding SHAP values. A SHAP value is an indication of whether or not the CNN 'likes' the input. Hence, at each position where a nucleosome center is marked, the corresponding SHAP value will either be positive or negative and indicate whether the nucleosome being there makes the nucleosome pattern more or less likely to be classified as cancerous. If the output SHAP values have a net positive the input will be classified as '1'. While the output of SHAP has presented a viable option for studying nucleosome patterns, we could not find a discernible pattern that consistently emerged between the high and low scoring inputs.

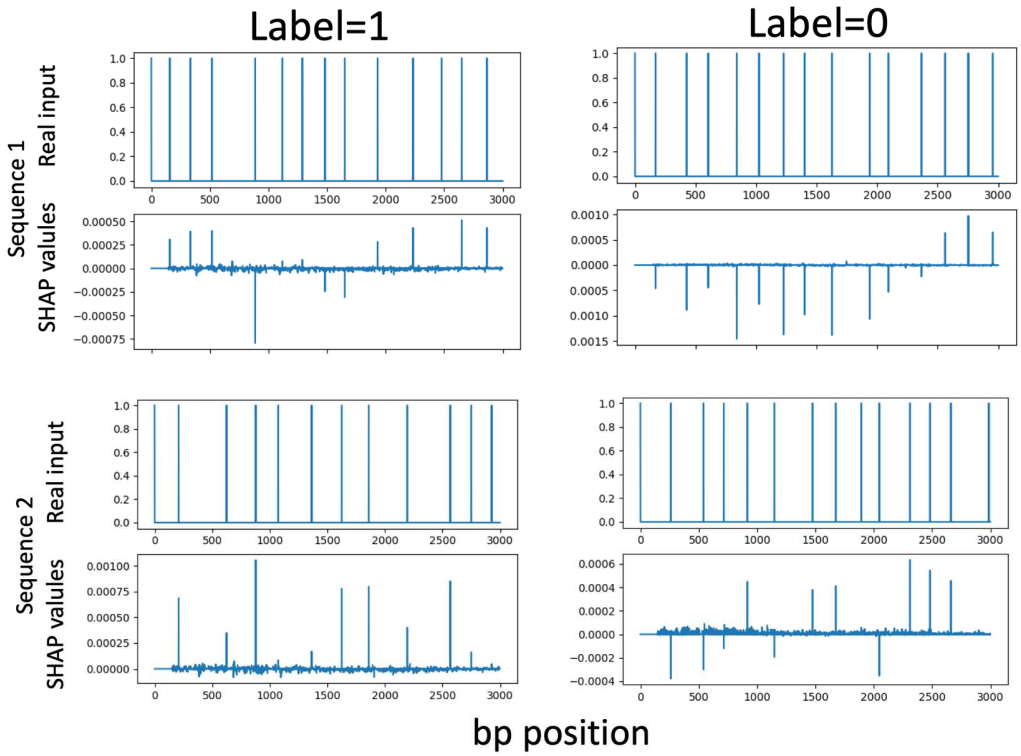


Figure 27. SHAP output visualizing the way that the CNN network is reacting to example input sequences from loci belonging to the state '9_Heterochromatin' (Label = 1) and randomly selected regions (Label = 0). The plots in the left column are exemplary inputs where the label was 1 and those in the right are exemplary inputs where the label was 0. In each panel (upper left, upper right, lower left and lower right), the upper plot shows the real input sequence (where each nucleosome center is marked by a '1') and the lower plot shows the SHAP reaction values to the input.

3.4.5.2 Finding characteristic NPSs using GANs

Another possibility to estimate which part of the nucleosome positioning input is important to a trained neural network is to use a generative adversarial network (GAN). Herein, the trained neural network (referred to as the discriminator) is exposed to fake nucleosome positioning patterns

output from another neural network (referred to as a generator). The weights of the discriminator are 'frozen' so that it cannot learn whereas the generator will be allowed keep producing fake images and learning from the feedback of the discriminator until it begins to 'fool' the discriminator. This can produce sequences that are emblematic of the learned features/ patterns of the original trained model (Goodfellow *et al.* 2014). The goal is to observe the 'loss' of the generator and discriminator across each round that fake sequences are produced. The loss can be simplified as the amount that a model is 'learning' in order to beat the other model (see Methods for details). When the loss of each model reaches an equilibrium one can then observe the produced fake sequences to see if there is a pattern that is a typical representation of the state of interest. Figure 28 shows the results of using a GAN to generate sequences that are representative for the state '9 heterochromatin'.

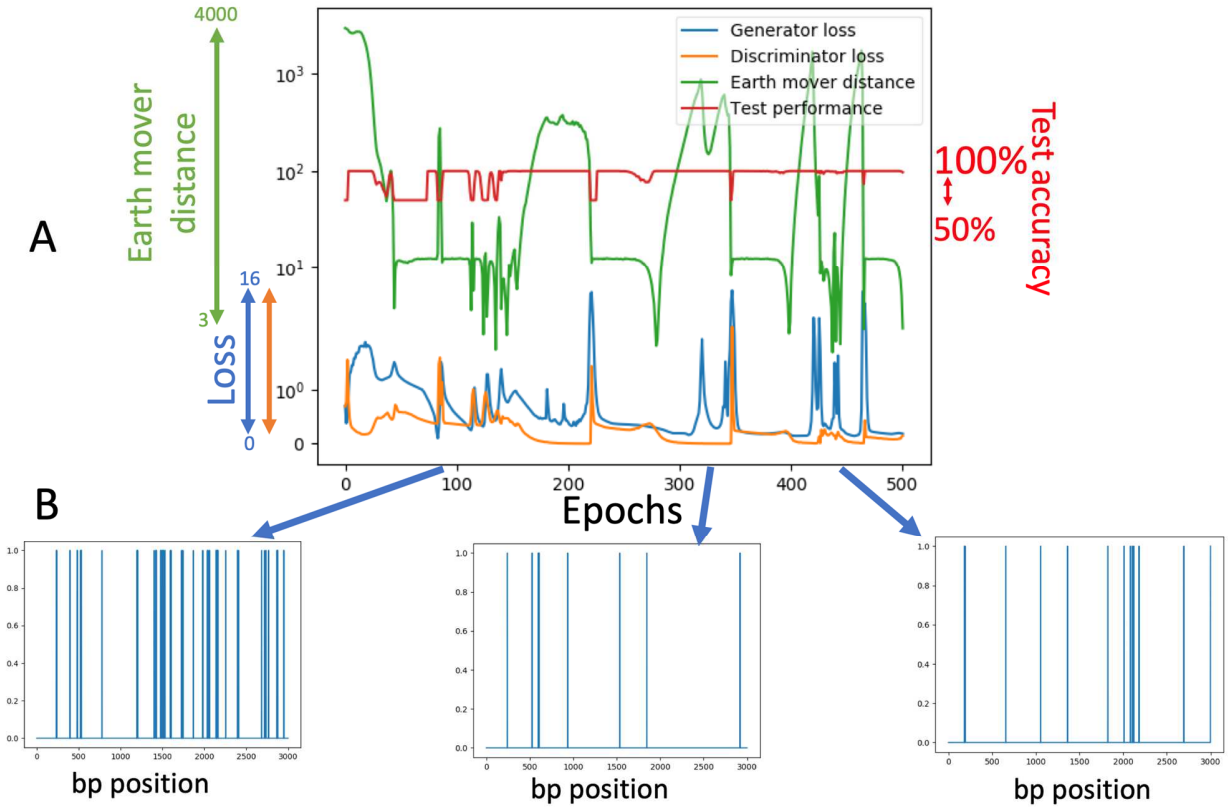


Figure 28. Generative adversarial network (GAN) used to generate sequences that are representative of the patterns recognized to distinguish the chromatin state '9_heterochromatin' from the rest of the genome in mESCs based on the filtered unique nucleosome dataset obtained with chemical mapping. (A) The loss profiles of the discriminator and generator across 500 epochs (rounds). The x-axis represents the number of epochs/ rounds that the generator was run in making fake input sequences. The blue and orange lines, respectively, represent the loss of the generator and discriminator in each round. The green line represents the distance of the generated inputs to the real inputs calculated, in terms of the number of nucleosomes in each sequence, using the earth mover distance algorithm. The red line represents the percentage accuracy with which the discriminator distinguished real from input sequences. (B) The output fake sequences from various

stages of training. The blue arrows extending from panel (A) indicate the stage of training that produced the fake output displayed in panel (B).

The goal of the above plot is to observe how the loss profiles of the generator and discriminator change with each round (epoch) of the generator attempting to fool the discriminator. The loss profiles of the discriminator and generator spike frequently, which indicates that the discriminator is being fooled. The test accuracy occasionally goes down to 50%, which is further verification that the generator is "fooling" the discriminator. However, on viewing the generated sequences output from different epochs, no one pattern consistently emerges. Furthermore, if the GAN had really achieved the goal of generating emblematic representations of the real input, the loss of the generator and discriminator would have reached equilibrium and the test accuracy would remain at 50%. Instead they remain in disequilibrium, so one can never be sure if the generated output is viable. The solution to this problem may be to increase the number of epochs. However, this is a very computationally demanding task.

3.4.5.3 Finding characteristic NPSs by comparing ML-derived genomic regions with high vs low prediction confidence

As another alternative to make the ML model explainable, we decided to study the characteristics of low vs high scoring inputs separately (i.e. input sequences that were predicted to have a high chance of being labelled '1' vs sequences that were predicted to have a low chance of being labelled '1'). Figure 29 shows an example study of how the different classes of ouput from the model predictions could be investigated.

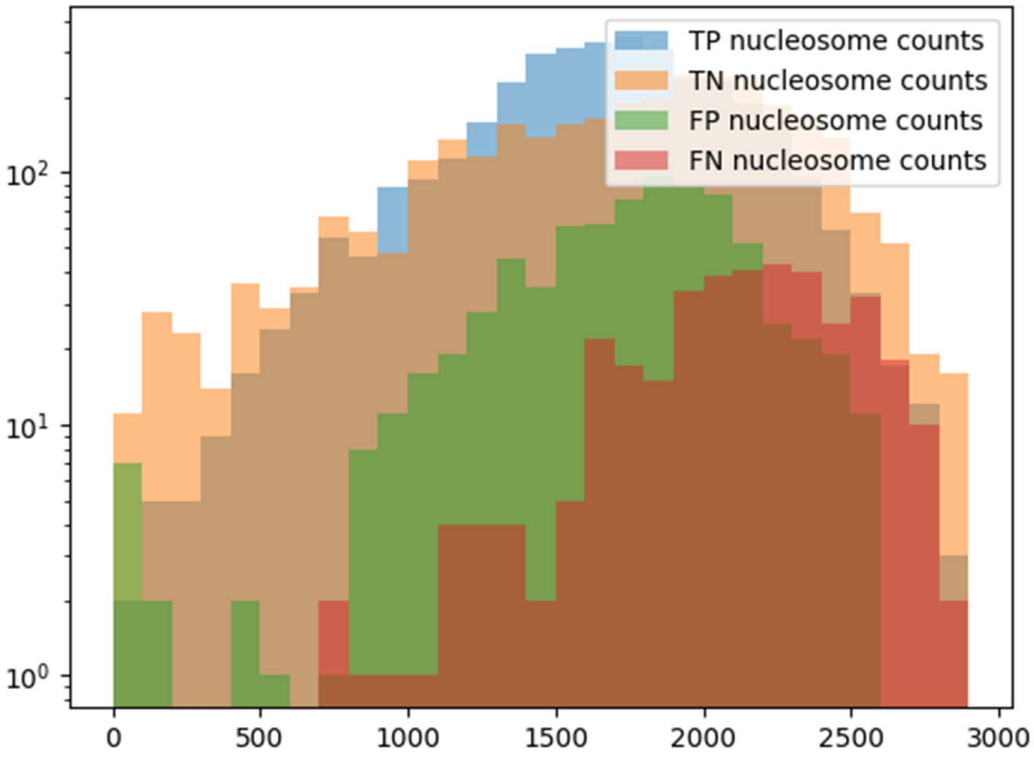


Figure 29. Distribution of the number of nucleosomes in each genomic region in mESCs, obtained from the dataset derived from MNase data while recognising '12_heterochromatin'. The X-axis documents the number of nucleosomes within the individual genomic region). The Y-axis shows the frequency of such occurences for the genomic regions which are classified by the model as true positives (TP), blue; true negatives (TN), orange; false positives (FP), green; false negatives (FN), red.

The distribution of nucleosome counts on Figure 29 was plotted for the correctly predicted genomic regions and the incorrectly predicted ones (separated according to whether the region was a TP, TN, FP, FN). While there is some separation between the categories, it is not enough to say confidently whether the model is learning solely based on the number of non-zeros in a sequence, and hence there must be some recurrent pattern(s) that the model is selecting in its classification decisions.

As a separate attempt at making sense of the rules learned by the model which predicted whether a row of data came from a CLL patient or a healthy individual, the rows were separated into different states and the distribution of scores was plotted (Figure 30).

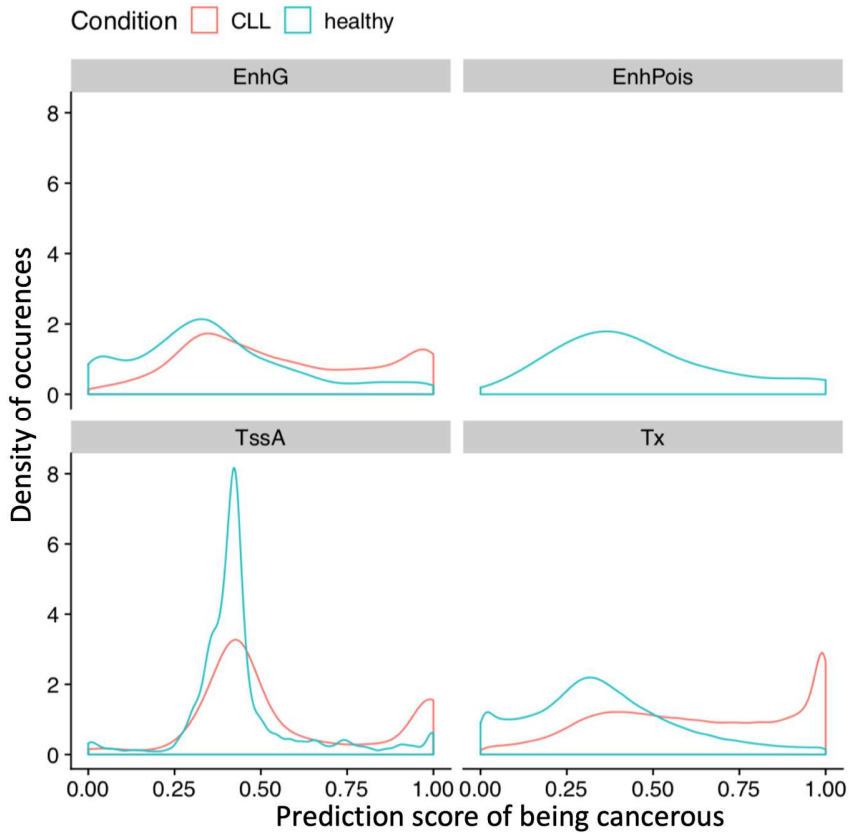


Figure 30. The density of the scores predicting the probability that a given input sequence belongs to a cancer sample, using samples from CLL patients and healthy individuals, calculated separately for genomic regions belonging to different chromatin states. The ChromHMM annotation of CLL patients is from (Mallm et al, 2019). The aim of this calculation is to see if prediction was particularly good in any of the individual chromatin states. NOTE: the regions of EnhancerPois were not abundant enough in the CLL data to provide a sufficient number of rows and be included in the ML analysis.

Figure 30 shows the distribution of scores that predict for an individual region from a given chromatin states the chance of being cancerous, based on its NPS. One can see that the majority of rejected regions (i.e. those with scores below 50%) are given a score between 30 and 45%, while the majority of accepted ones (those with a score above 50%) have scores > 87.5%. This suggests the possibility that the trained CNN model had settled on a very specific set of patterns to classify as 'definitely cancerous' and/or 'not sure', and hence with a score below 50%. Since the main part of the peak for the "not sure" class was around 32-40%, we decided to perform an additional analysis specifically for the regions which fall in to this category, as opposed to the category of "definitely cancerous" regions which were defined as score >95%. We used these two classes of regions to calculate the average nucleosome occupancy around the TSSs nearest to the sites of interest (Figure 31).

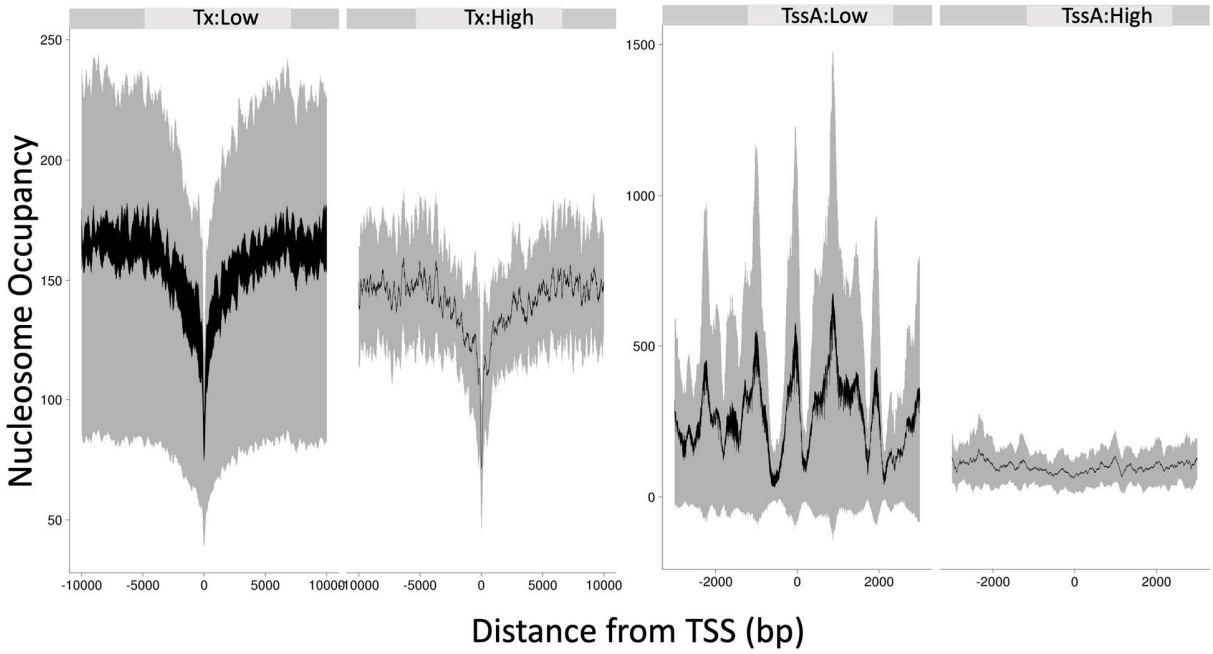


Figure 31. Average nucleosome occupancy around TSSs, calculated separately for the regions which were scored by the model as high- or low-probability of being cancerous. Two chromatin states are used: Tx (two left panels) and TssA (two right panels). In each of these categories, the 'high'(above 95% score) and 'low' (32% to 40%) refer to the assigned confidence score prediction of whether or not cancer is present. The shaded areas represent the standard error.

Figure 31 shows that for the Tx state, the high scoring regions have a lower baseline of nucleosome occupancy than the low scoring ones. However, their central nucleosome-depleted regions at TSS are less pronounced. In the case of TssA there is even more dramatic difference: the high-scoring regions do not have the oscillation of nucleosome occupancy around TSS, unlike the low scoring ones. This means that the high-scoring regions, both based on the Tx and TssA chromatin state, correspond to the genes which are active in healthy cells but shut down in CLL. Both the high-

scoring and low-scoring regions belonging to Tx chromatin state were enriched for ATP-binding (P = 5.5e-17 and P = 4.5e-33 correspondingly), mRNA processing (P = 2.1e-17 and P = 1.3e-17 correspondingly), DNA repair (P = 3.8e-10 and P = 1e-13 correspondingly), helicase (P = 1.9e-10 and P = 1.4e-18 correspondingly) and transcription (P = 8.8e-10 and P = 9.2e-16 correspondingly). The same situation was observed the high-scoring TssA regions, while for low-scoring TssA we had only 15 regions which was not enough for GO analysis. Thus, these results indicate that B-cells in CLL patients have a distinct chromatin structure at a number of functionally relevant genes. Importantly these genes could not be identified using a classical gene expression analysis (Mallm *et al.* 2019). Thus, the nucleosome signature-based analysis offers a unique, complementary way for cancer diagnostics.

3.4.6 Using NPS-based machine learning for patient stratification

Next we tested the viability of this ML model as a diagnostic test. I have plotted the percentage of correctly identified patients (having vs not having cancer) across a range of different scores used as thresholds of cancer detection (Figure 32). This analysis shows that by increasing the score threshold, the percentage of correctly identified cases decreases. For example, if a medical doctor would use the current method, setting the threshold to 90% probability that a given person has cancer, then about 50% of patients would be correctly diagnosed. Thus, this method still needs to be improved before it can be implemented clinically.

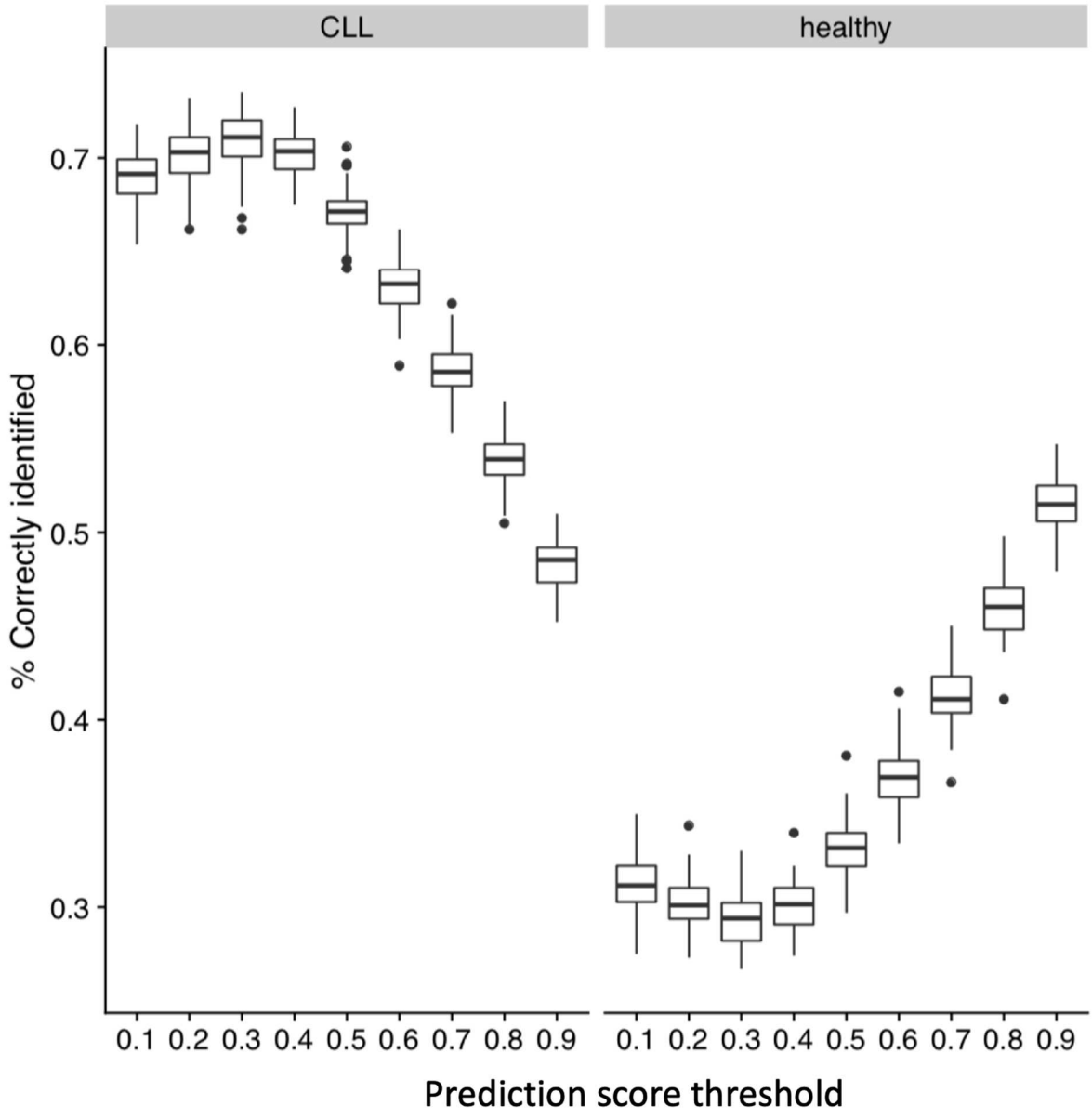


Figure 32. The percentage of correctly diagnosed individuals as a function of the prediction score threshold used as a criteria to classify an individual as having cancer (CLL) or not. The ML-based prediction based on nucleosome positioning signatures was performed for all available B-cell samples from CLL patients and healthy individuals (see Methods).

3.4.7 Comparing CNNs to other models

Our collaborators, Ravikiran Chimatapu and Hani Hagras from the School of Computer Sciences and Electronic Engineering at the University of Essex used a host of other models on our data to compare with ours CNN model. These included: SVMs, logistic regression and others that involved fuzzy logic.

In the case of the chromatin state detection as in Figures 22-25, the CNN model outperformed all other investigated models. In the case of cancer identification, the alternative models achieved accuracy levels that were consistently just below the CNN, but still comparable. In their analysis they also encountered the problem that the nucleosome density in the healthy individuals were generally higher than in CLL samples, similar to my Figure 26, which confounds the model decisions. On the other hand, the nucleosome density differences per could be used as part of the "classical" analysis of nucleosome positioning, Thus, the future modification of the diagnostic pipeline based on nucleosome positioning could use a combination of the classic nucleosome density/occupancy analysis with machine learning. Additional work will be required to clearly demonstrate the applicability of this pipeline in the clinic.

3.5 Discussion

In this chapter, we used machine learning (predominantly in the form of CNNs) to discriminate between different types of chromatin state solely from the nucleosome positioning patterns. The input data for this analysis included nucleosome positioning in mouse embryonic stem cells and in B-cells from healthy people and CLL patients. In the case of classifying the chromatin state, different types of chromatin showed different levels of distinguishability and hence were generally more regularly identifiable than others. It is noteworthy that the chromatin states that were active

were more readily identifiable than heterochromatin states. This result concurs with my finding the NRL of active regions is more distinguishable from genome-average than the NRL of heterochromatin regions (Figure 21). A number of other studies also showed that active states have a generally much tighter NRL than inactive states (Teif and Clarkson 2019). One can imagine that if the active chromatin states were comprised of shorter loci with denser packing of nucleosomes, the resulting NPS would be much more regular than that of inactive states. We started this project with the aim of being able to identify chromatin states for individual small genomic regions (as opposed to the need to use large regions or a large number of regions for the classical NRL analysis). We have indeed succeeded to classify individual genomic regions as small as 3,000 bp to different chromatin states purely based on their NPS using machine learning.

In the case of classifying cancerous and non-cancerous chromatin, a lot of work still needs to be done before we can recommend the use of this method in the clinic. This mostly stems from the fact that the CNN is a 'black-box' model and hence it is difficult if not impossible to know if it was classifying based on the NPS alone or on other confounding information (Figure 26).

The major challenge to make the CNN model explainable is to know what patterns are being identified as important when classifying. We attempted to solve this in various different ways with varying degrees of success (section 3.4.5). Regardless of what algorithms are used to classify input data, this will continue to be a major issue to be solved in the future. It is vital to know the different patterns that are being recognised by the model of choice.

The use of models besides CNNs was briefly discussed in section 3.4. Other models as simple as logistic regression classified cancerous/non-cancerous input sequences with accuracy levels that were similar to that of the CNN. While the simpler models were not similarly successful in

classifying the chromatin state data, it may be advantageous to be able to consistently use a simpler model as this would allow for more explainability and subsequently for the technique to give rise to more biologically useful information.

Another important extension of this approach in the future may involve the use of cell-free DNA instead of the genomic DNA (so-called "liquid biopsy". The cell-free DNA comes from many different cells, including cancer cells if the person has cancer. Cell-free DNA is naturally cut by apoptotic nucleases between nucleosomes. Thus, most pieces of cell-free DNA represent the nucleosomal DNA, allowing to reconstruct the original nucleosome landscape in the tissues. Therefore, in future the techniques outlined in this section can be used to analyse cell-free DNA to diagnose and stratify cancer patients. Overall the work done here could potentially prove useful both for applied cancer research and the fundamental study of chromatin structure (Snyder *et al.* 2016).

4 Conclusions

In my PhD I worked on several different aspects of chromatin structure. Vladimir gave me free reign to work on subjects that interested me most. About one year into the PhD I began the machine learning project and 6 months later, I began to simultaneously work on the CTCF project (which eventually turned into a publishable body of work).

As detailed in Chapter 2, the first paper from this thesis (Clarkson et al., 2019) revealed the intrincate interplay between CTCF and nucleosomes and suggested how this can affect chromatin boundaries. The novel findings in this study can be summarised as follows:

- We reported a new effect: CTCF binding affinity is inversely proportional to the NRL near its binding site.
- We found that in mouse embryonic stem cells the nucleosome array organisation is asymmetric near CTCF binding sites. This is due to the presence of an AT-rich region that sits behind the forward facing motif. The degree of asymmetry is dependent on the strength/conservation of the CTCF binding motif.
- We discovered that a subset of CTCF binding sites that remain bound during the differentiation of mouse embryonic stem cells maintain short distances between the neighbouring nucleosomes. This could be important for maintaining genome integrity at key loci throughout differentiation.
- Furthermore we observed that CTCF binding sites occur in clusters at TAD boundaries, and proposed a new model of chromatin boundary formation through ordered, asymmetric nucleosome arrays.

Further work can be done on this subject. After my PhD I intend to do some investigation on the spreading of repressive histone marks (H3K27me3 and H3K9me3) in the presence/absence of CTCF binding. One of my favourite papers in relation to this is that of (Nora *et al.* 2016) where it was shown that CTCF removal does not lead to the spreading of H3K27me3 to invade neighbouring insulated chromatin loci. I think it would be interesting to further investigate this – only this time consider the number of CTCF motifs in the vicinity of the lost CTCF binding site and then observe how H3K27me3 spreads as a function of this. I suspect that the mark would spread at loci where there is only one CTCF motif whereas it would not at loci where there are clusters of binding motifs. I also suspect that, as well as CTCF binding, the DNA sequence (i.e. AT-rich DNA) plays a role in barrier formation and that the presence of multiple CTCF motifs,

with adjacent AT rich DNA sites, could enhance this effect. If this hypothesis is true, it would demonstrate an interesting new paradigm for the interplay between epigenetics and the cellular development instructions hardwired in the DNA sequence.

Chapter 3 describes a project where I used machine learning to attempt to derive a relationship between small-scale chromatin structure (individual nucleosome level) and higher order chromatin architecture (chromatin state). I then adapted this machine learning pipeline to be tested as a diagnostic tool in distinguishing data coming from CLL patients vs healthy individuals. This project showed that:

- Nucleosome positioning alone can be used to predict chromatin state
- This could potentially be used as clinically actionable data when diagnosing cancer- if the prediction accuracy can be enhanced.

While it is interesting and novel that chromatin state can be predicted from nucleosome positioning patterns (depending on the chromatin state in question), one would need to derive the actual patterns that are being recognized by the model to provide biological context. Furthermore the rate at which the model distinguishes cancerous from non-cancerous data could be improved to outperform any already existing methods.

In the future, if time permits I will take this developed technique and apply it to a new task, namely the prediction of the cell type from which the cell-free DNA originates, based on nucleosome positioning. This is currently an unaccomplished task, and it is an important challenge to solve in order to be able to use liquid biopsy for diagnostics.

References

Abadi M, Agarwal A, Barham P *et al. TensorFlow : Large-Scale Machine Learning on Heterogeneous Distributed Systems*., 2015.

Afek A, Schipper JL, Horton J *et al.* Protein-DNA binding in the absence of specific base-pair recognition. *Proc Natl Acad Sci U S A* 2014, DOI: 10.1073/pnas.1410569111.

Allis CD, Jenuwein T. The molecular hallmarks of epigenetic control. *Nat Rev Genet* 2016;**17**:487–500.

De Ambrosis A, Ferrari N, Bonassi S *et al.* Nucleosomal repeat length in active and inactive genes. *FEBS Lett* 1987, DOI: 10.1016/0014-5793(87)81142-0.

Avsec Ž, Weilert M, Shrikumar A *et al. Deep Learning at Base-Resolution Reveals Motif Syntax of the Cis-Regulatory Code*., 2019.

Baldi S. Nucleosome positioning and spacing: From genome-wide maps to single arrays. *Essays Biochem* 2019, DOI: 10.1042/EBC20180058.

Baldi S, Krebs S, Blum H *et al.* Genome-wide measurement of local nucleosome array regularity and spacing by nanopore sequencing. *Nat Struct Mol Biol* 2018, DOI: 10.1038/s41594-018-0110-0.

Barisic D, Stadler MB, Iurlaro M *et al.* Mammalian ISWI and SWI/SNF selectively mediate binding of distinct transcription factors. *Nature* 2019, DOI: 10.1038/s41586-019-1115-5.

Barrington C, Georgopoulou D, Pezic D *et al.* Enhancer accessibility and CTCF occupancy underlie asymmetric TAD architecture and cell type specific genome topology. *Nat Commun* 2019, DOI: 10.1038/s41467-019-10725-9.

Barutcu AR, Maass PG, Lewandowski JP *et al.* A TAD boundary is preserved upon deletion of the CTCF-rich Firre locus. *Nat Commun* 2018, DOI: 10.1038/s41467-018-03614-0.

Bascom GD, Kim T, Schlick T. Kilobase Pair Chromatin Fiber Contacts Promoted by Living-System-Like DNA Linker Length Distributions and Nucleosome Depletion. *J Phys Chem B* 2017, DOI: 10.1021/acs.jpcb.7b00998.

Bass M V., Nikitina T, Norouzi D *et al.* Nucleosome spacing periodically modulates nucleosome chain folding and DNA topology in circular nucleosome arrays. *J Biol Chem* 2019, DOI: 10.1074/jbc.RA118.006412.

Beshnova DA, Cherstvy AG, Vainshtein Y *et al.* Regulation of the Nucleosome Repeat Length In Vivo by the DNA Sequence, Protein Concentrations and Long-Range Interactions. *PLOS Comput Biol* 2014;**10**:e1003698.

Bogu GK, Vizán P, Stanton LW *et al.* Chromatin and RNA Maps Reveal Regulatory Long Noncoding RNAs in Mouse. *Mol Cell Biol* 2016;**36**:809 LP – 819.

Bonev B, Mendelson Cohen N, Szabo Q *et al.* Multiscale 3D Genome Rewiring during Mouse Neural Development. *Cell* 2017, DOI: 10.1016/j.cell.2017.09.043.

Bornelöv S, Reynolds N, Xenophontos M *et al.* The Nucleosome Remodeling and Deacetylation Complex Modulates Chromatin Structure at Sites of Active Transcription to Fine-Tune Gene Expression. *Mol Cell* 2018, DOI: 10.1016/j.molcel.2018.06.003.

Bourque G, Leong B, Vega VB *et al.* Evolution of the mammalian transcription factor binding repertoire via transposable elements. *Genome Res* 2008, DOI: 10.1101/gr.080663.108.

Bradbury EM. K. E. Van Holde. Chromatin. Series in molecular biology. Springer-Verlag, New York. 1988. 530 pp. $98.00. *J Mol Recognit* 1989, DOI: 10.1002/jmr.300020308.

Buscema PM, Massini G, Breda M *et al.* Artificial neural networks. *Studies in Systems, Decision and Control*. Vol 131. Springer International Publishing, 2018, 11–35.

Carrire L, Graziani S, Alibert O *et al.* Genomic binding of Pol III transcription machinery and

relationship with TFIIS transcription factor distribution in mouse embryonic stem cells. *Nucleic Acids Res* 2012, DOI: 10.1093/nar/gkr737.

Castro-Mondragon JA, Jaeger S, Thieffry D *et al.* RSAT matrix-clustering: Dynamic exploration and redundancy reduction of transcription factor binding motif collections. *Nucleic Acids Res* 2017, DOI: 10.1093/nar/gkx314.

Cavalli G, Heard E. Advances in epigenetics link genetics to the environment and disease. *Nature* 2019;**571**:489–99.

Chen H, Tian Y, Shu W *et al.* Comprehensive identification and annotation of cell type-specific and ubiquitous CTCF-binding sites in the human genome. *PLoS One* 2012a, DOI: 10.1371/journal.pone.0041374.

Chen K, Wang L, Yang M *et al.* Sequence signatures of nucleosome positioning in caenorhabditis elegans. *Genomics, Proteomics Bioinforma* 2010;**8**:92–102.

Chen W, Lin H, Feng P-M *et al.* iNuc-PhysChem: A Sequence-Based Predictor for Identifying Nucleosomes via Physicochemical Properties. *PLoS One* 2012b;**7**:e47843.

Chen Z, Gabizon R, Brown AI *et al.* High-resolution and high-accuracy topographic and transcriptional maps of the nucleosome barrier. *Elife* 2019, DOI: 10.7554/elife.48281.

Chereji R V., Ramachandran S, Bryson TD *et al.* Precise genome-wide mapping of single nucleosomes and linkers in vivo. *Genome Biol* 2018, DOI: 10.1186/s13059-018-1398-0.

Choudhary MN, Friedman RZ, Wang JT *et al.* Co-opted transposons help perpetuate conserved higher-order chromosomal structures. *bioRxiv* 2018, DOI: 10.1101/485342.

Clarkson C, Deeks E, Samarista R *et al.* CTCF-dependent chromatin boundaries formed by asymmetric nucleosome arrays with decreased linker length. *Nucleic Acids Res* 2019, DOI: 10.1093/nar/gkz908.

Cremer T, Cremer M. Chromosome Territories. *Cold Spring Harb Perspect Biol* 2010;**2**:1–23.

Cuartero S, Weiss FD, Dharmalingam G *et al.* Control of inducible gene expression links cohesin to hematopoietic progenitor self-renewal and differentiation. *Nat Immunol* 2018, DOI: 10.1038/s41590-018-0184-1.

Dekker J, Mirny L. The 3D Genome as Moderator of Chromosomal Communication. *Cell* 2016;**164**:1110–21.

Dekker J, Rippe K, Dekker M *et al.* Capturing Chromosome Conformation. *Science (80-)* 2002;**295**:1306 LP – 1311.

Deniz Ö, Flores O, Battistini F *et al.* Physical properties of naked DNA influence nucleosome positioning and correlate with transcription start and termination sites in yeast. *BMC Genomics* 2011;**12**:489.

De Dieuleveult M, Yen K, Hmitou I *et al.* Genome-wide nucleosome specificity and function of chromatin remodellers in ES cells. *Nature* 2016, DOI: 10.1038/nature16505.

Dunham I, Kundaje A, Aldred SF *et al.* An Integrated Encyclopedia of DNA Elements in the Human Genome. *Nature* 2012;**489**:57–74.

Ernst J, Kellis M. ChromHMM: automating chromatin-state discovery and characterization. *Nat Meth* 2012;**9**:215–6.

Fang R, Wang C, Skogerbo G *et al.* Functional diversity of CTCFs is encoded in their binding motifs. *BMC Genomics* 2015, DOI: 10.1186/s12864-015-1824-6.

Filion GJ, van Bemmel JG, Braunschweig U *et al.* Systematic protein location mapping reveals five principal chromatin types in Drosophila cells. *Cell* 2010;**143**:212–24.

Flaus A. Principles and practice of nucleosome positioning in vitro. *Front Life Sci* 2011;**5**:5–27.

Fu Y, Sinha M, Peterson CL *et al.* The insulator binding protein CTCF positions 20 nucleosomes

around its binding sites across the human genome. *PLoS Genet* 2008, DOI: 10.1371/journal.pgen.1000138.

Fudenberg G, Imakaev M, Lu C *et al.* Formation of Chromosomal Domains by Loop Extrusion. *Cell Rep* 2016, DOI: 10.1016/j.celrep.2016.04.085.

Gaffney DJ, McVicker G, Pai AA *et al.* Controls of Nucleosome Positioning in the Human Genome. *PLOS Genet* 2012;**8**:e1003036.

Di Gangi M, Lo Bosco G, Rizzo R. Deep learning architectures for prediction of nucleosome positioning from sequences data. *BMC Bioinformatics* 2018;**19**:418.

Géron A. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*., 2019.

Ghavi-Helm Y, Jankowski A, Meiers S *et al.* Highly rearranged chromosomes reveal uncoupling between genome topology and gene expression. *Nat Genet* 2019, DOI: 10.1038/s41588-019-0462-3.

Ghirlando R, Felsenfeld G. CTCF: Making the right connections. *Genes Dev* 2016, DOI: 10.1101/gad.277863.116.

Gibson BA, Doolittle LK, Jensen LE *et al.* Organization and Regulation of Chromatin by Liquid-Liquid Phase Separation. *bioRxiv* 2019, DOI: 10.1101/523662.

Giles KA, Gould CM, Du Q *et al.* Integrated epigenomic analysis stratifies chromatin remodellers into distinct functional groups. *Epigenetics and Chromatin* 2019, DOI: 10.1186/s13072-019-0258-9.

Goodfellow IJ, Pouget-Abadie J, Mirza M *et al.* Generative Adversarial Networks. 2014:1–9.

Gottesfeld JM, Melton DA. The length of nucleosome-associated DNA is the same in both transcribed and nontranscribed regions of chromatin. *Nature* 1978, DOI: 10.1038/273317a0.

Hansen AS, Pustova I, Cattoglio C *et al.* CTCF and cohesin regulate chromatin loop stability with distinct dynamics. *Elife* 2017, DOI: 10.7554/eLife.25776.

Hathaway NA, Bell O, Hodges C *et al.* Dynamics and memory of heterochromatin in living cells. *Cell* 2012;**149**:1447–60.

van der Heijden T, van Vugt JJFA, Logie C *et al.* Sequence-based prediction of single nucleosome positioning and genome-wide nucleosome occupancy. *Proc Natl Acad Sci U S A* 2012;**109**:E2514–22.

Heinz S, Benner C, Spann N *et al.* Simple Combinations of Lineage-Determining Transcription Factors Prime cis-Regulatory Elements Required for Macrophage and B Cell Identities. *Mol Cell* 2010, DOI: 10.1016/j.molcel.2010.05.004.

Hennig BP, Bendrin K, Zhou Y *et al.* Chd1 chromatin remodelers maintain nucleosome organization and repress cryptic transcription. *EMBO Rep* 2012, DOI: 10.1038/embor.2012.146.

Herold M, Bartkuhn M, Renkawitz R. CTCF: Insights into insulator function during development. *Development* 2012, DOI: 10.1242/dev.065268.

Hnisz D, Day DS, Young RA. Insulated Neighborhoods: Structural and Functional Units of Mammalian Gene Control. *Cell* 2016;**167**:1188–200.

Ho L, Jothi R, Ronan JL *et al.* An embryonic stem cell chromatin remodeling complex, esBAF, is an essential component of the core pluripotency transcriptional network. *Proc Natl Acad Sci U S A* 2009, DOI: 10.1073/pnas.0812888106.

Van Holde K, Zlatanova J. What determines the folding of the chromatin fiber? *Proc Natl Acad Sci U S A* 1996;**93**:10548–55.

Hore TA, Deakin JE, Marshall Graves JA. The evolution of epigenetic regulators CTCF and

BORIS/CTCFL in amniotes. *PLoS Genet* 2008, DOI: 10.1371/journal.pgen.1000169.

Hsieh T-HS, Slobodyanyuk E, Hansen AS *et al.* Resolving the 3D landscape of transcription-linked mammalian chromatin folding. *bioRxiv* 2019:638775.

Ishihara K, Oshimura M, Nakao M. CTCF-Dependent Chromatin Insulator Is Linked to Epigenetic Remodeling. *Mol Cell* 2006, DOI: 10.1016/j.molcel.2006.08.008.

Jenkinson G, Pujadas E, Goutsias J *et al.* Potential energy landscapes identify the information-theoretic nature of the epigenome. *Nat Genet* 2017, DOI: 10.1038/ng.3811.

Juan D, Perner J, Carrillo de Santa Pau E *et al.* Epigenomic Co-localization and Co-evolution Reveal a Key Role for 5hmC as a Communication Hub in the Chromatin Network of ESCs. *Cell Rep* 2016;**14**:1246–57.

Kaplan T, Li X-Y, Sabo PJ *et al.* Quantitative Models of the Mechanisms That Control Genome-Wide Patterns of Transcription Factor Binding during Early Drosophila Development. *PLOS Genet* 2011;**7**:e1001290.

Karreth FA, Tay Y, Pandolfi PP. Target competition: transcription factors enter the limelight. *Genome Biol* 2014;**15**:114.

Kentepozidou E, Aitken SJ, Feig C *et al.* Clustered CTCF binding is an evolutionary mechanism to maintain topologically associating domains. *bioRxiv* 2019, DOI: 10.1101/668855.

Khan A, Fornes O, Stigliani A *et al.* JASPAR 2018: Update of the open-access database of transcription factor binding profiles and its web framework. *Nucleic Acids Res* 2018, DOI: 10.1093/nar/gkx1126.

Kornberg RD. Chromatin structure: A repeating unit of histones and DNA. *Science (80-)* 1974, DOI: 10.1126/science.184.4139.868.

Kresge N, Simoni RD, Hill RL. Chromatin Structure and the Nucleosome: the Work of Kensal

E. van Holde. *J Biol Chem* 2010;**285**:e5–6.

Kubik S, Bruzzone MJ, Challal D *et al.* Opposing chromatin remodelers control transcription initiation frequency and start site selection. *Nat Struct Mol Biol* 2019, DOI: 10.1038/s41594-019-0273-3.

Kubo N, Ishii H, Gorkin D *et al.* Preservation of Chromatin Organization after Acute Loss of CTCF in Mouse Embryonic Stem Cells. *bioRxiv* 2017.

Kuhn M, Johnson K. *Applied Predictive Modeling*., 2013.

Kundaje A, Kyriazopoulou-Panagiotopoulou S, Libbrecht M *et al.* Ubiquitous heterogeneity and asymmetry of the chromatin environment at regulatory elements. *Genome Res* 2012, DOI: 10.1101/gr.136366.111.

Lai B, Gao W, Cui K *et al.* Principles of nucleosome organization revealed by single-cell micrococcal nuclease sequencing. *Nature* 2018, DOI: 10.1038/s41586-018-0567-3.

Lai WKM, Pugh BF. Understanding nucleosome dynamics and their links to gene expression and DNA replication. *Nat Rev Mol Cell Biol* 2017, DOI: 10.1038/nrm.2017.47.

Langmead B, Wilks C, Antonescu V *et al.* Scaling read aligners to hundreds of threads on general-purpose processors. *Bioinformatics* 2019, DOI: 10.1093/bioinformatics/bty648.

Längst G, Becker PB. Nucleosome remodeling: One mechanism, many phenomena? *Biochim Biophys Acta - Gene Struct Expr* 2004;**1677**:58–63.

Lankaš F, Šponer J, Langowski J *et al.* DNA Basepair Step Deformability Inferred from Molecular Dynamics Simulations. *Biophys J* 2003;**85**:2872–83.

Lantermann AB, Straub T, Stralfors A *et al.* Schizosaccharomyces pombe genome-wide nucleosome mapping reveals positioning mechanisms distinct from those of Saccharomyces cerevisiae. *Nat Struct Mol Biol* 2010;**17**:251–7.

Larson AG, Elnatan D, Keenen MM *et al.* Liquid droplet formation by HP1α suggests a role for phase separation in heterochromatin. *Nature* 2017, DOI: 10.1038/nature22822.

Lawrence M, Huber W, Pagès H *et al.* Software for Computing and Annotating Genomic Ranges. *PLoS Comput Biol* 2013, DOI: 10.1371/journal.pcbi.1003118.

Lieberman-Aiden E, van Berkum NL, Williams L *et al.* Comprehensive Mapping of Long-Range Interactions Reveals Folding Principles of the Human Genome. *Science (80-)* 2009;**326**:289 LP – 293.

Liu M, Seddon AE, Tsai ZT *et al.* Determinants of nucleosome positioning and their influence on plant gene expression. *Genome Res* 2015:1–14.

Liu S, Zhang L, Quan H *et al.* From 1D sequence to 3D chromatin dynamics and cellular functions: A phase separation perspective. *Nucleic Acids Res* 2018, DOI: 10.1093/nar/gky633.

Lobanenkov V V., Zentner GE. Discovering a binary ctcf code with a little help from boris. *Nucleus* 2018, DOI: 10.1080/19491034.2017.1394536.

Locke G, Tolkunov D, Moqtaderi Z *et al.* High-throughput sequencing reveals a simple model of nucleosome energetics. *Proc Natl Acad Sci U S A* 2010;**107**:20998–1003.

Lohr D, Tatchell K, Van Holde KE. On the occurrence of nucleosome phasing in chromatin. *Cell* 1977, DOI: 10.1016/0092-8674(77)90281-1.

Lowary PT, Widom J, Angel A *et al.* New DNA sequence rules for high affinity binding to histone octamer and sequence-directed nucleosome positioning. *J Mol Biol* 1998;**276**:19–42.

Luger K, Mäder a W, Richmond RK *et al.* Crystal structure of the nucleosome core particle at 2.8 A resolution. *Nature* 1997;**389**:251–60.

Lundberg SM, Lee SI. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*. 2017.

Lunyak V V., Prefontaine GG, Núñez E *et al.* Developmentally regulated activation of a SINE B2 repeat as a domain boundary in organogenesis. *Science (80-)* 2007, DOI: 10.1126/science.1140871.

Maeshima K, Ide S, Babokhov M. Dynamic chromatin organization without the 30-nm fiber. *Curr Opin Cell Biol* 2019, DOI: 10.1016/j.ceb.2019.02.003.

Mallm J, Iskar M, Ishaque N *et al.* Linking aberrant chromatin features in chronic lymphocytic leukemia to transcription factor networks. *Mol Syst Biol* 2019, DOI: 10.15252/msb.20188339.

Martin D, Pantoja C, Fernández-Miñán A *et al.* Genome-wide CTCF distribution in vertebrates defines equivalent sites that aid the identification of disease-associated genes. *Nat Struct Mol Biol* 2011, DOI: 10.1038/nsmb.2059.

McKinney W, Team PD. Pandas - Powerful Python Data Analysis Toolkit. *Pandas - Powerful Python Data Anal Toolkit* 2015.

Meers MP, Janssens DH, Henikoff S. Pioneer Factor-Nucleosome Binding Events during Differentiation Are Motif Encoded. *Mol Cell* 2019, DOI: 10.1016/j.molcel.2019.05.025.

Merkenschlager M, Nora EP. CTCF and Cohesin in Genome Folding and Transcriptional Gene Regulation. *Annu Rev Genomics Hum Genet* 2016, DOI: 10.1146/annurev-genom-083115-022339.

Miller JA, Widom J. Collaborative Competition Mechanism for Gene Activation In Vivo. *Mol Cell Biol* 2003;**23**:1623–32.

Mir M, Bickmore W, Furlong EEM *et al.* Chromatin topology, condensates and gene regulation:

shifting paradigms or just a phase? *Development* 2019;**146**:1–6.

Mirny L a. Nucleosome-mediated cooperativity between transcription factors. *Proc Natl Acad Sci U S A* 2010;**107**:22534–9.

Mitchell TM. The Discipline of Machine Learning. *Mach Learn* 2006, DOI: 10.1080/026404199365326.

Mivelaz M, Cao A-M, Kubik S *et al.* The mechanistic basis for chromatin invasion and remodeling by the yeast pioneer transcription factor Rap1. *bioRxiv* 2019, DOI: 10.1101/541284.

Möbius W, Osberg B, Tsankov AM *et al.* Toward a unified physical model of nucleosome patterns flanking transcription start sites. *Proc Natl Acad Sci U S A* 2013, DOI: 10.1073/pnas.1214048110.

Molitor J, Mallm J-P, Rippe K *et al.* Retrieving Chromatin Patterns from Deep Sequencing Data Using Correlation Functions. *Biophys J* 2017;**112**:473–90.

Nanni L, Wang C, Manders F *et al.* The CTCF Anatomy of Topologically Associating Domains. *bioRxiv* 2019, DOI: 10.1101/746610.

Neph S, Reynolds AP, Kuehn MS *et al.* Operating on genomic ranges using BEDOPS. *Methods in Molecular Biology*. 2016.

Nichols MH, Corces VG. A CTCF Code for 3D Genome Architecture. *Cell* 2015, DOI: 10.1016/j.cell.2015.07.053.

Nikitina T, Norouzi D, Grigoryev SA *et al.* DNA topology in chromatin is defined by nucleosome spacing. *Sci Adv* 2017, DOI: 10.1126/sciadv.1700957.

Nora EP, Goloborodko A, Valton A-L *et al.* Targeted degradation of CTCF decouples local insulation of chromosome domains from higher-order genomic compartmentalization.

bioRxiv 2016.

Ocampo J, Chereji R V., Eriksson PR *et al.* The ISW1 and CHD1 ATP-dependent chromatin remodelers compete to set nucleosome spacing in vivo. *Nucleic Acids Res* 2016, DOI: 10.1093/nar/gkw068.

Olins AL, Olins DE. Spheroid Chromatin Units (ν Bodies). *Science (80-)* 1974;**183**:330 LP – 332.

Oliphant T, Millma J k. A guide to NumPy. *Trelgol Publ* 2006, DOI: DOI:10.1109/MCSE.2007.58.

Owens N, Papadopoulou T, Festuccia N *et al.* CTCF confers local nucleosome resiliency after dna replication and during mitosis. *Elife* 2019, DOI: 10.7554/eLife.47898.

Ozonov EA, van Nimwegen E. Nucleosome Free Regions in Yeast Promoters Result from Competitive Binding of Transcription Factors That Interact with Chromatin Modifiers. *PLOS Comput Biol* 2013;**9**:e1003181.

Padinhateeri R, Marko JF. Nucleosome positioning in a model of active chromatin remodeling enzymes. *Proc Natl Acad Sci U S A* 2011;**108**:7799–803.

Pavlaki I, Docquier F, Chernukhin I *et al.* Poly(ADP-ribosyl)ation associated changes in CTCF-chromatin binding and gene expression in breast cells. *bioRxiv* 2018.

Pedregosa F, Varoquaux G, Gramfort A *et al.* Scikit-learn: Machine learning in Python. *J Mach Learn Res* 2011.

Perkel DH. Logical neurons: the enigmatic legacy of Warren McCulloch. *Trends Neurosci* 1988, DOI: 10.1016/0166-2236(88)90041-0.

Phillips JE, Corces VG. CTCF: Master Weaver of the Genome. *Cell* 2009, DOI: 10.1016/j.cell.2009.06.001.

Di Pierro M, Cheng RR, Lieberman Aiden E *et al.* De novo prediction of human chromosome structures: Epigenetic marking patterns encode genome architecture. *Proc Natl Acad Sci* 2017;**114**:12126–31.

Quinlan AR. BEDTools: The Swiss-Army tool for genome feature analysis. *Curr Protoc Bioinforma* 2014, DOI: 10.1002/0471250953.bi1112s47.

Rampasek L, Goldenberg A. TensorFlow: Biology's Gateway to Deep Learning? *Cell Syst* 2016, DOI: 10.1016/j.cels.2016.01.009.

Rao SSP, Huang SC, Glenn St Hilaire B *et al.* Cohesin Loss Eliminates All Loop Domains. *Cell* 2017, DOI: 10.1016/j.cell.2017.09.026.

Rasim Barutcu A, Lian JB, Stein JL *et al.* The connection between BRG1, CTCF and topoisomerases at TAD boundaries. *Nucleus* 2017, DOI: 10.1080/19491034.2016.1276145.

Ricci MA, Manzo C, García-Parajo MF *et al.* Chromatin fibers are formed by heterogeneous groups of nucleosomes in vivo. *Cell* 2015, DOI: 10.1016/j.cell.2015.01.054.

Rippe VBT and K. Statistical–mechanical lattice models for protein–DNA binding in chromatin. *J Phys Condens Matter* 2010;**22**:414105.

Risca VI, Denny SK, Straight AF *et al.* Variable chromatin structure revealed by in situ spatially correlated DNA cleavage mapping. *Nature* 2017, DOI: 10.1038/nature20781.

Roadmap Epigenomics Consortium, Kundaje A, Meuleman W *et al.* Integrative analysis of 111 reference human epigenomes. *Nature* 2015, DOI: 10.1038/nature14248.

Robertson KD. DNA methylation and human disease. *Nat Rev Genet* 2005, DOI: 10.1038/nrg1655.

Roider HG, Kanhere A, Manke T *et al.* Predicting transcription factor affinities to DNA from a biophysical model. *Bioinformatics* 2007;**23**:134–41.

Routh A, Sandin S, Rhodes D. Nucleosome repeat length and linker histone stoichiometry determine chromatin fiber structure. *Proc Natl Acad Sci U S A* 2008, DOI: 10.1073/pnas.0802336105.

Salih BF, Teif VB, Tripathi V *et al.* Strong nucleosomes of mouse genome including recovered centromeric sequences. *J Biomol Struct Dyn* 2015;**33**:1164–75.

Satchwell SC, Drew HR, Travers AA. Sequence periodicities in chicken nucleosome core DNA. *J Mol Biol* 1986;**191**:659–75.

Sawyer IA, Sturgill D, Dundr M. Membraneless nuclear organelles and the search for phases within phases. *Wiley Interdiscip Rev RNA* 2019, DOI: 10.1002/wrna.1514.

Schiessel H, Widom J, Bruinsma RF *et al.* Polymer Reptation and Nucleosome Repositioning. *Phys Rev Lett* 2001;**86**:4414–7.

Schmidt D, Schwalie PC, Wilson MD *et al.* Waves of retrotransposon expansion remodel genome organization and CTCF binding in multiple mammalian lineages. *Cell* 2012, DOI: 10.1016/j.cell.2011.11.058.

Schwarzer W, Abdennur N, Goloborodko A *et al.* Two independent modes of chromosome organization are revealed by cohesin removal. *bioRxiv* 2016, DOI: 10.1101/094185.

Segal E, Fondufe-Mittendorf Y, Chen L *et al.* A genomic code for nucleosome positioning. *Nature* 2006;**442**:772–8.

Segal E, Widom J. Poly(dA:dT) Tracts: Major Determinants of Nucleosome Organization. *Curr Opin Struct Biol* 2009;**19**:65–71.

Shrikumar A, Greenside P, Kundaje A. Learning important features through propagating activation differences. *34th International Conference on Machine Learning, ICML 2017*. 2017.

Simms TA, Dugas SL, Gremillion JC *et al.* TFIIIC binding sites function as both heterochromatin barriers and chromatin insulators in Saccharomyces cerevisiae. *Eukaryot Cell* 2008, DOI: 10.1128/EC.00128-08.

Snyder MW, Kircher M, Hill AJ *et al.* Cell-free DNA Comprises an in Vivo Nucleosome Footprint that Informs Its Tissues-Of-Origin. *Cell* 2016, DOI: 10.1016/j.cell.2015.11.050.

Stadhouders R, Vidal E, Serra F *et al.* Transcription factors orchestrate dynamic interplay between genome topology and gene regulation during cell reprogramming. *Nat Genet* 2018, DOI: 10.1038/s41588-017-0030-7.

Stolz RC, Bishop TC. ICM Web: the interactive chromatin modeling web server. *Nucleic Acids Res* 2010;**38**:W254–61.

Struhl K, Segal E. Determinants of nucleosome positioning. *Nat Struct &Amp; Mol Biol* 2013;**20**:267.

Sun F-L, Cuaycong MH, Elgin SCR. Long-Range Nucleosome Ordering Is Associated with Gene Silencing in Drosophila melanogaster Pericentric Heterochromatin. *Mol Cell Biol* 2001, DOI: 10.1128/mcb.21.8.2867-2879.2001.

Sun F, Chronis C, Kronenberg M *et al.* Promoter-Enhancer Communication Occurs Primarily within Insulated Neighborhoods. *Mol Cell* 2019, DOI: 10.1016/j.molcel.2018.10.039.

Tan G, Lenhard B. TFBSTools: An R/bioconductor package for transcription factor binding site analysis. *Bioinformatics* 2016, DOI: 10.1093/bioinformatics/btw024.

Teif VB, Beshnova DA, Vainshtein Y *et al.* Nucleosome repositioning links DNA (de)methylation and differential CTCF binding during stem cell development. *Genome Res* 2014;**24**:1285–95.

Teif VB, Clarkson CT. Nucleosome Positioning. *Encyclopedia of Bioinformatics and*

Computational Biology. 2019.

Teif VB, Erdel F, Beshnova DA *et al.* Taking into account nucleosomes for predicting gene expression. *Methods* 2013;**62**:26–38.

Teif VB, Rippe K. Predicting nucleosome positions on the DNA: combining intrinsic sequence preferences and remodeler activities. *Nucleic Acids Res* 2009;**37**:5641–55.

Teif VB, Vainshtein Y, Caudron-Herger M *et al.* Genome-wide nucleosome positioning during embryonic stem cell development. *Nat Struct {&} Mol Biol* 2012;**19**:1185–92.

Todolli S, Perez PJ, Clauvelin N *et al.* Contributions of Sequence to the Higher-Order Structures of DNA. *Biophys J* 2016;**112**:1–11.

Tolstorukov MY, Choudhary V, Olson WK *et al.* nuScore: a web-interface for nucleosome positioning predictions. *Bioinformatics* 2008;**24**:1456–8.

Trapnell C, Roberts A, Goff L *et al.* Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat Protoc* 2012, DOI: 10.1038/nprot.2012.016.

Trifonov EN, Nibhani R. Review fifteen years of search for strong nucleosomes. *Biopolymers* 2015, DOI: 10.1002/bip.22604.

Trifonov EN, Sussman JL. The pitch of chromatin DNA is reflected in its nucleotide sequence. *Proc Natl Acad Sci* 1980;**77**:3816–20.

Uhler C, Shivashankar G V. Chromosome Intermingling: Mechanical Hotspots for Genome Regulation. *Trends Cell Biol* 2017, DOI: 10.1016/j.tcb.2017.06.005.

Vainshtein Y, Rippe K, Teif VB. NucTools: analysis of chromatin feature occupancy profiles from high-throughput sequencing data. *BMC Genomics* 2017;**18**:158.

Valouev A, Johnson SM, Boyd SD *et al.* Determinants of nucleosome organization in primary

human cells. *Nature* 2011;**474**:516–20.

VanderPlas J. *Python Data Science Handbook*., 2016.

Vilarrasa-Blasi R, Soler-Vila P, Verdaguer-Dot N *et al.* Dynamics of genome architecture and chromatin function during human B cell differentiation and neoplastic transformation. *bioRxiv* 2019, DOI: 10.1101/764910.

Voong LN, Xi L, Sebeson AC *et al.* Insights into Nucleosome Organization in Mouse Embryonic Stem Cells through Chemical Mapping. *Cell* 2016, DOI: 10.1016/j.cell.2016.10.049.

Voong LN, Xi L, Wang JP *et al.* Genome-wide Mapping of the Nucleosome Landscape by Micrococcal Nuclease and Chemical Mapping. *Trends Genet* 2017;**33**:495–507.

Wang H, Maurano MT, Qu H *et al.* Widespread plasticity in CTCF occupancy linked to DNA methylation. *Genome Res* 2012, DOI: 10.1101/gr.136101.111.

Wang J, Lawry ST, Cohen AL *et al.* Chromosome boundary elements and regulation of heterochromatin spreading. *Cell Mol Life Sci* 2014;**71**:4841–52.

Wang L, Gao Y, Zheng X *et al.* Histone Modifications Regulate Chromatin Compartmentalization by Contributing to a Phase Separation Mechanism. *Mol Cell* 2019, DOI: 10.1016/j.molcel.2019.08.019.

Weintraub H. The nucleosome repeat length increases during erythropoiesis in the chick. *Nucleic Acids Res* 1978, DOI: 10.1093/nar/5.4.1179.

Wiechens N, Singh V, Gkikopoulos T *et al.* The Chromatin Remodelling Enzymes SNF2H and SNF2L Position Nucleosomes adjacent to CTCF and Other Transcription Factors. *PLoS Genet* 2016, DOI: 10.1371/journal.pgen.1005940.

Wiehle L, Thorn GJ, Raddatz G *et al.* DNA (de)methylation in embryonic stem cells controls

CTCF-dependent chromatin boundaries. *Genome Res* 2019;**29**:750–61.

Xi L, Brogaard K, Zhang Q *et al.* A Locally Convoluted Cluster Model for Nucleosome Positioning Signals in Chemical Maps. *J Am Stat Assoc* 2014, DOI: 10.1080/01621459.2013.862169.

Xi L, Fondufe-Mittendorf Y, Xia L *et al.* Predicting nucleosome positioning using a duration Hidden Markov Model. *BMC Bioinformatics* 2010;**11**:346.

Xu F, Olson WK. DNA Architecture, Deformability, and Nucleosome Positioning. *J Biomol Struct Dyn* 2010;**27**:725–39.

Yue F, Cheng Y, Breschi A *et al.* A comparative encyclopedia of DNA elements in the mouse genome. *Nature* 2014, DOI: 10.1038/nature13992.

Zaret KS, Carroll JS. Pioneer transcription factors: Establishing competence for gene expression. *Genes Dev* 2011, DOI: 10.1101/gad.176826.111.

Zentner GE, Henikoff S. Surveying the epigenomic landscape, one base at a time. *Genome Biol* 2012;**13**:250.

Zhang Y, Li T, Preissl S *et al.* 3D Chromatin Architecture Remodeling during Human Cardiomyocyte Differentiation Reveals A Role Of HERV-H In Demarcating Chromatin Domains. *bioRxiv* 2018, DOI: 10.1101/485961.

Zhang Y, Li T, Preissl S *et al.* Transcriptionally active HERV-H retrotransposons demarcate topologically associating domains in human pluripotent stem cells. *Nat Genet* 2019, DOI: 10.1038/s41588-019-0479-7.

Zhang Z, Wippo CJ, Wal M *et al.* A packing mechanism for nucleosome organization reconstituted across a eukaryotic genome. *Science (80-)* 2011, DOI: 10.1126/science.1200508.

5 APPENDIX

5.1 Supplementary figures

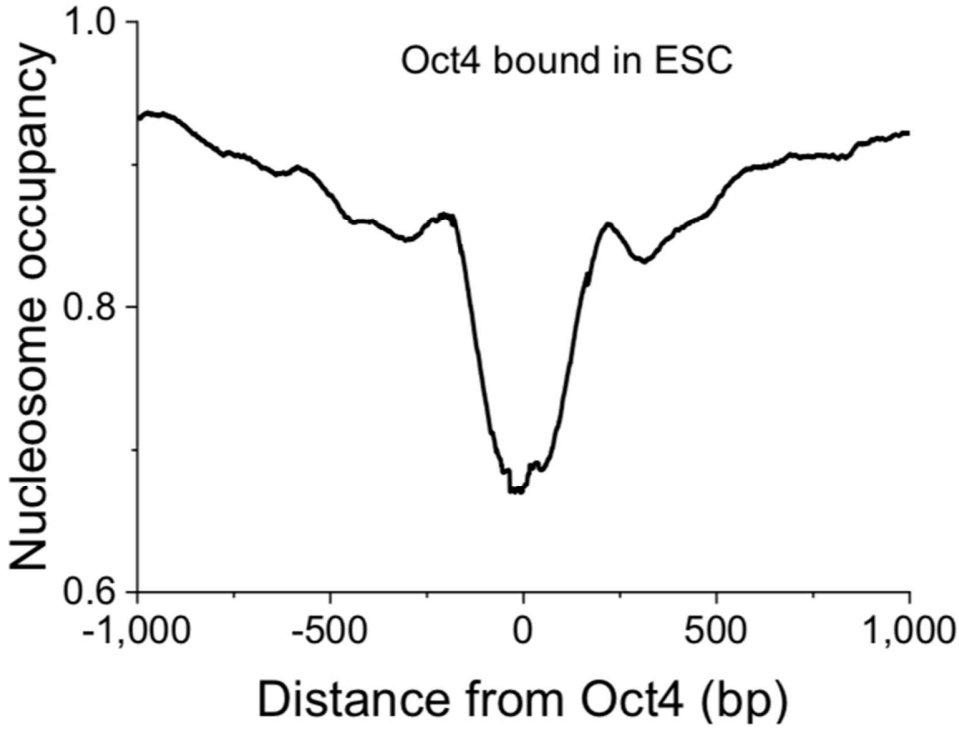


Figure S1. Average nucleosome occupancy profiles in ESC around bound Oct4. ChIP-seq data for Oct4 is from (Teif and Clarkson 2019) and MNase-seq data is from (Baldi 2019).

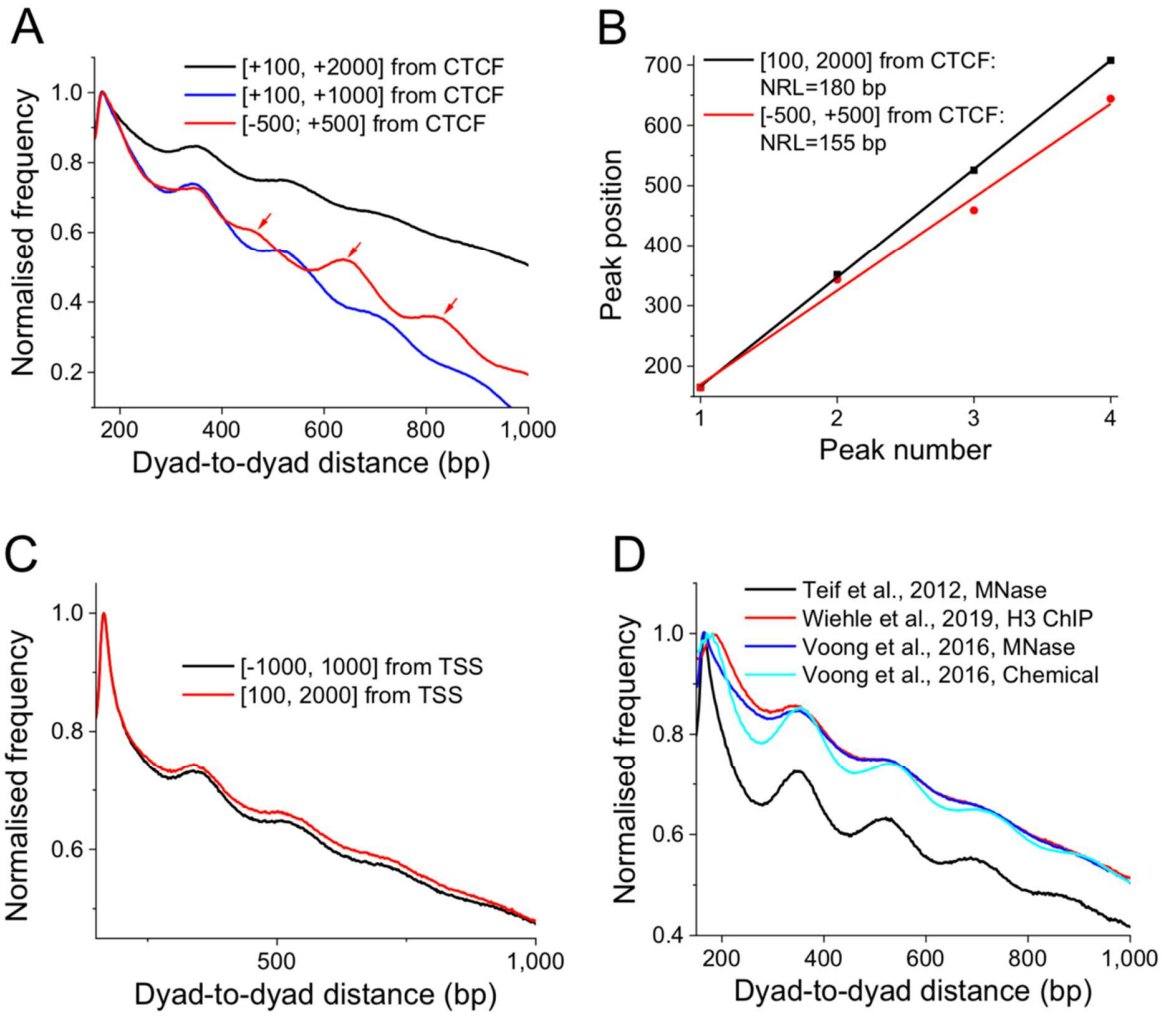


Figure S2. The effect of region location near CTCF on the apparent NRL value. A) Phasograms depicting the normalised frequency of nucleosome dyad-to-dyad distances calculated using NucTools for three different regions near CTCF sites: [100, 2000], [100, 1000] and [-500, 500]. Both [100, 2000] and [100, 1000] patterns oscillate with NRL=174. In the case of the [-500, 500] phasogram additional peaks (indicated by red arrows) appear which correspond to distances between nucleosomes on different sides of CTCF, thus resulting in an apparent NRL

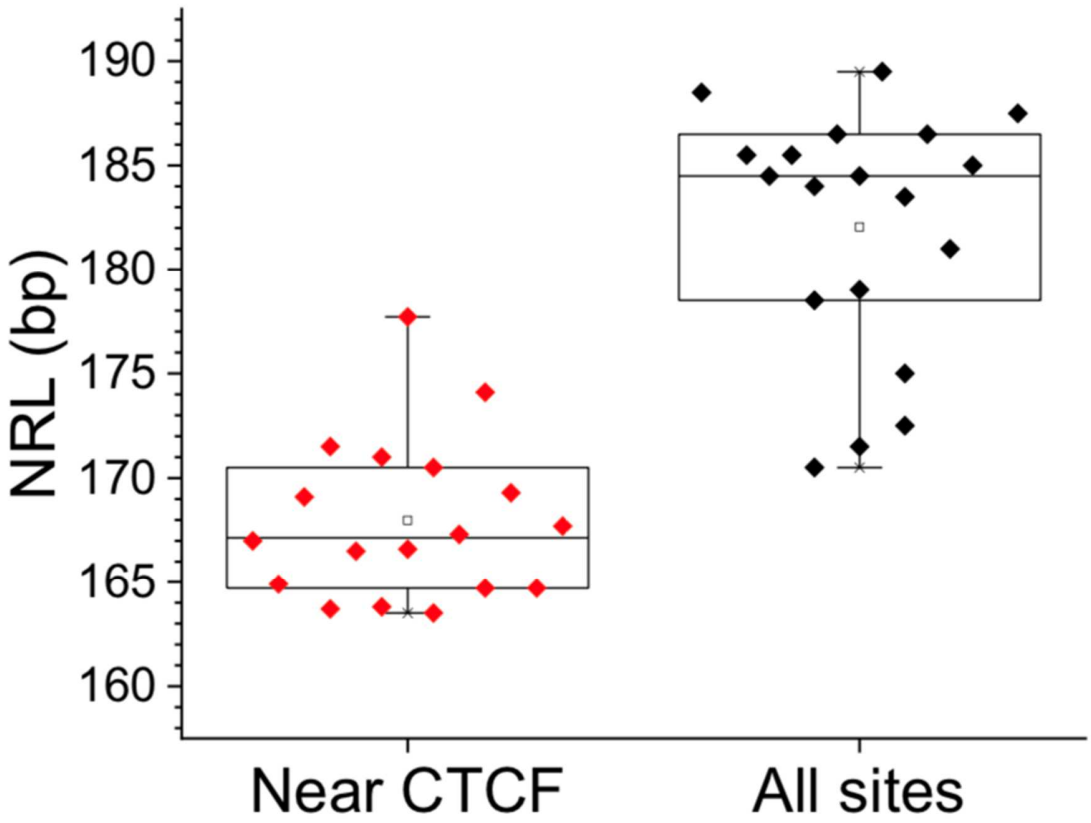


Figure S3. NRLs calculated near binding sites of 18 stemness-related chromatin proteins in ESCs in the region [250, 1000] from the TF sites. The same nucleosome positioning and TF-binding datasets as in Figure 1C are used. Left: TF binding sites in the vicinity of CTCF; right: all TF binding sites irrespective of their distance from CTCF

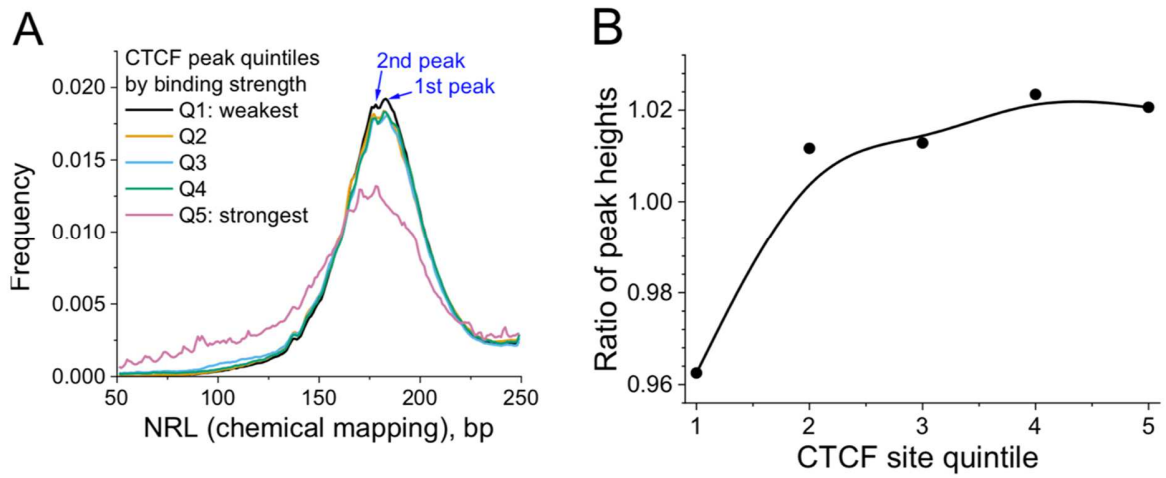


Figure S4. A) A histogram of nucleosome dyad-to-dyad distances determined using chemical mapping for different CTCF quintiles defined based on CTCF binding strength determined by the height of ChIP-seq CTCF peaks. Note that the dyad-todyad distances determined using chemical mapping cannot be directly compared to MNase-seq based NRLs due to an inherent bias of the chemical mapping experimental setup that we discussed previously (Merkenschlager and Nora 2016). B) The ratio between heights of 2 nd peak and 1st peak of the distribution of lengths of chemical mapping-based dyadto-dyad distances shown in panel (A) as a function of the CTCF site quintile based on the heights of CTCF ChIP-seq peaks.

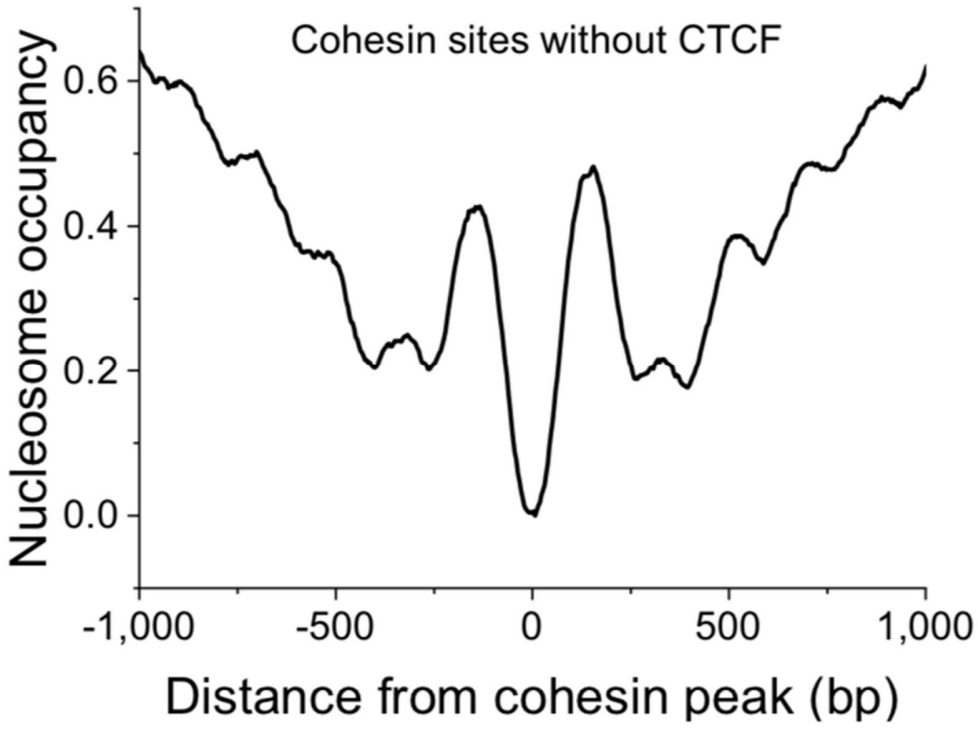


Figure S5. Average nucleosome occupancy profiles around sites bound by cohesin in ESC, taking into account only sites that do not contain CTCF motifs.

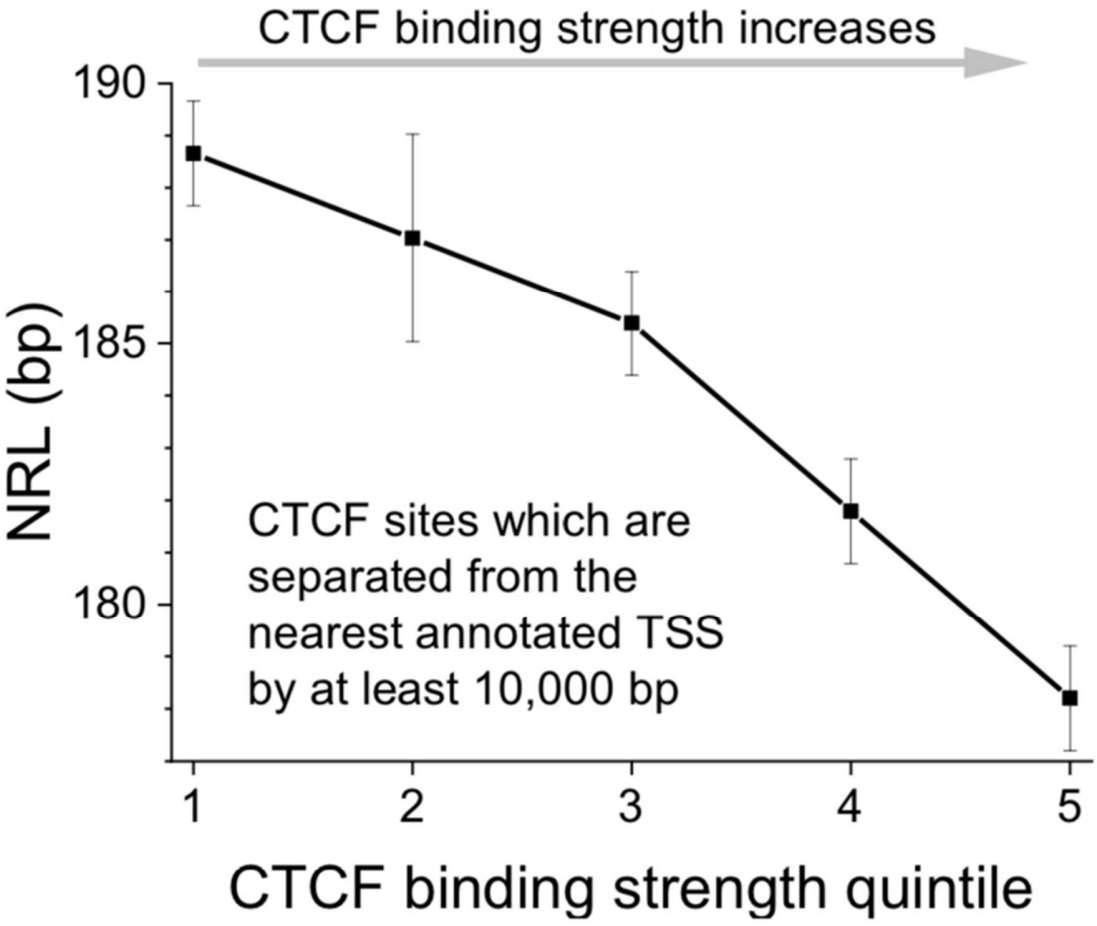


Figure S6. Dependence of NRL in the region [100, 2000] near CTCF on the strength of CTCF binding, excluding the effect of promoters. This calculation used only CTCF sites separated from the nearest TSS by at least 10,000 bp. The error bars show standard deviation.

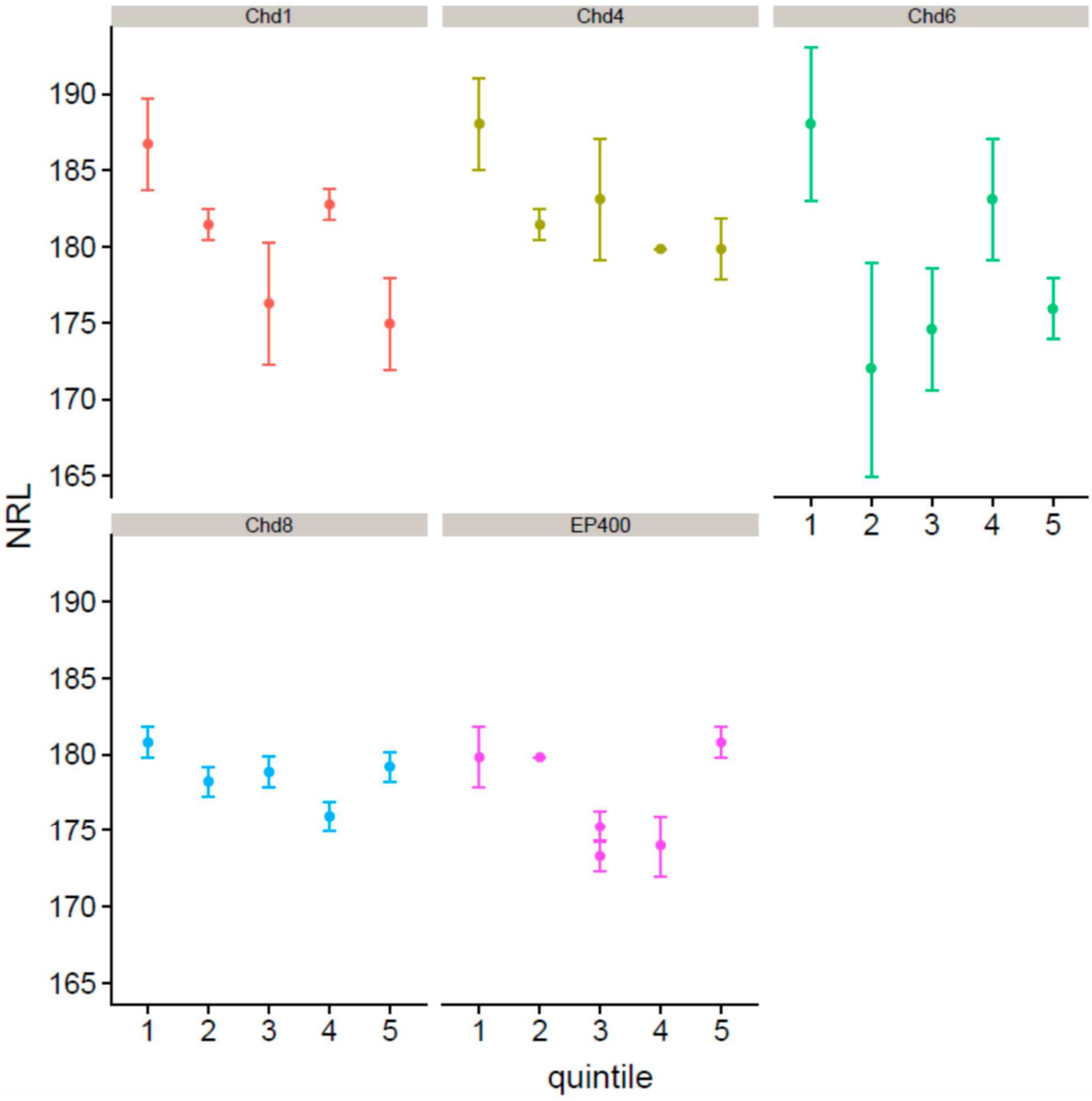


Figure S7. NRL calculated in the region [-1000, 1000] from the summits of ChIP-seq peaks of chromatin remodellers Chd1, Chd4, Chd6, Chd8, EP400 in ESCs.

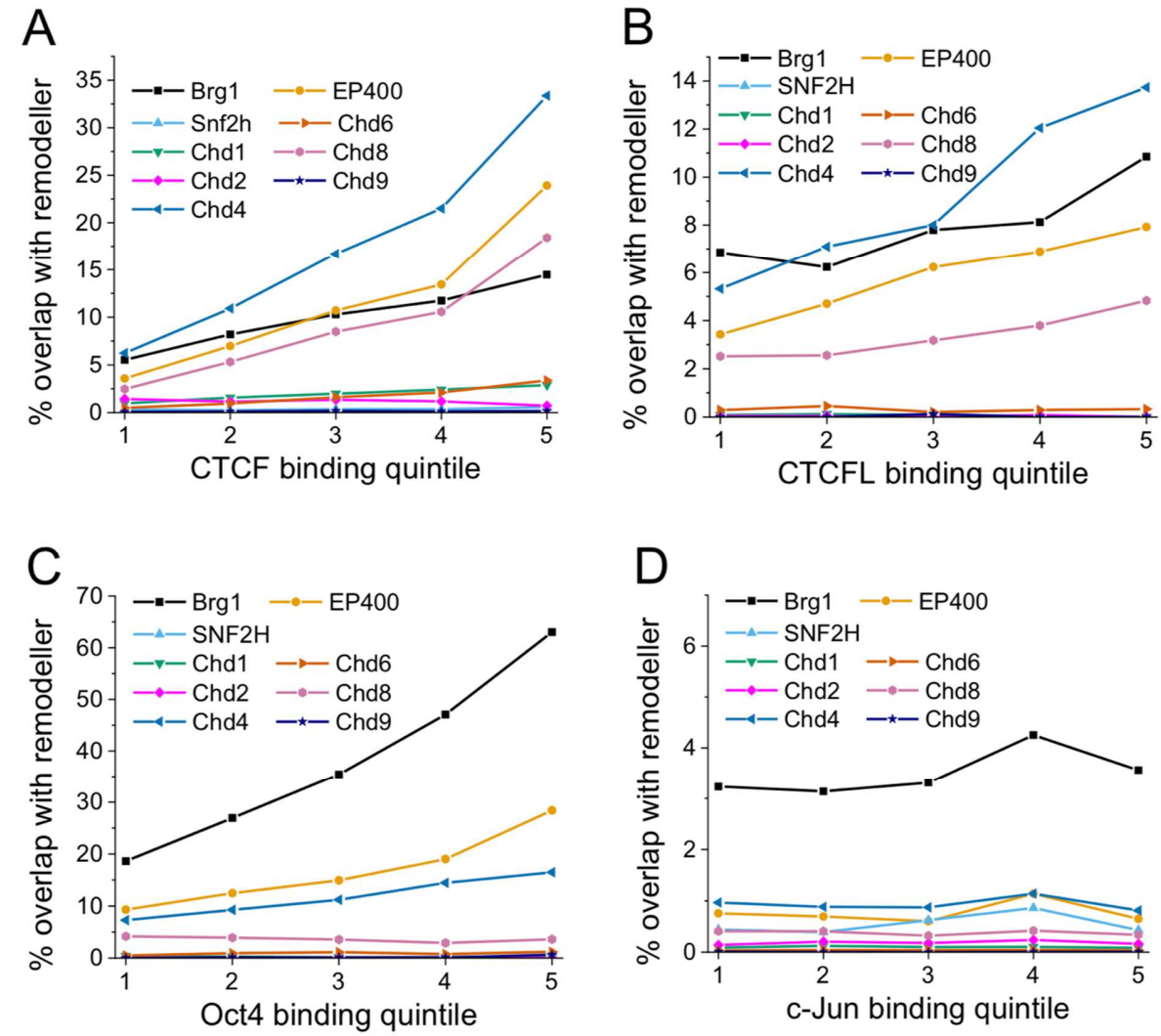


Figure S8. Overlapping of chromatin remodellers with different TFs. The percentage of overlap of a given TF with a given remodeller was defined as the ratio of TF sites overlapping with ChIP-seq peaks of a given remodeller to the total number of CTCF sites in a given quintile. Binding sites of each TF were split into quintiles according to their predicted binding strengths. A) CTCF, B) CTCFL (BORIS), C) Oct4, D) c-Jun.

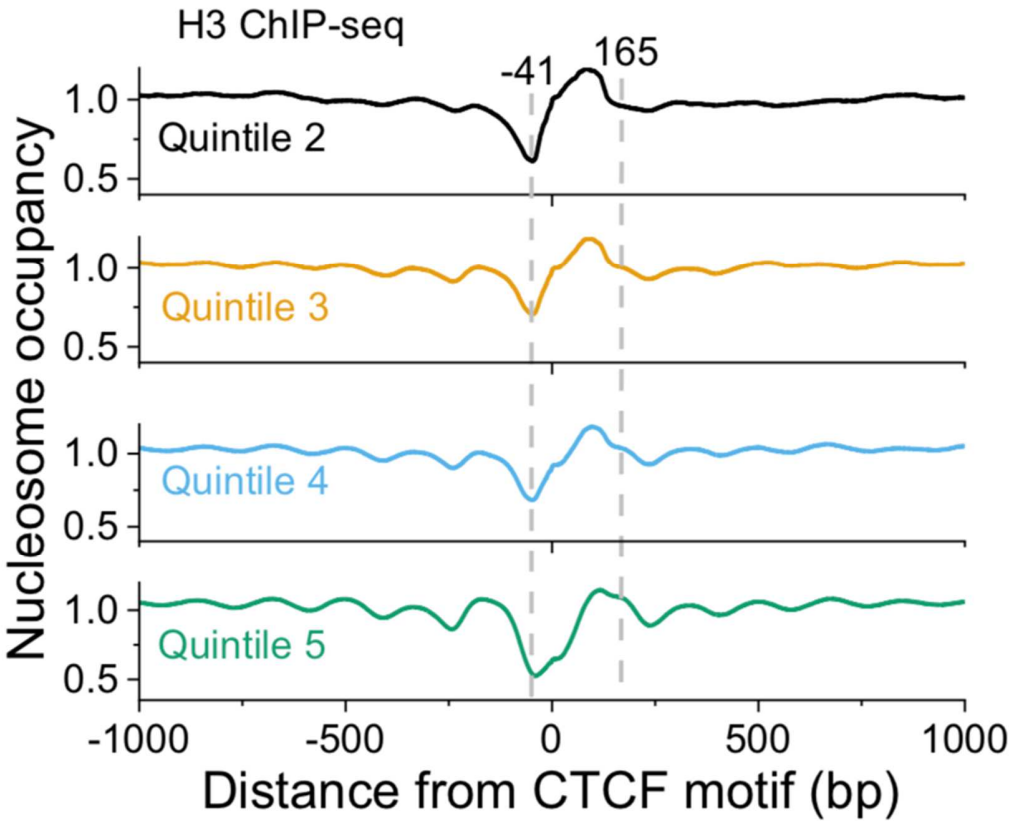
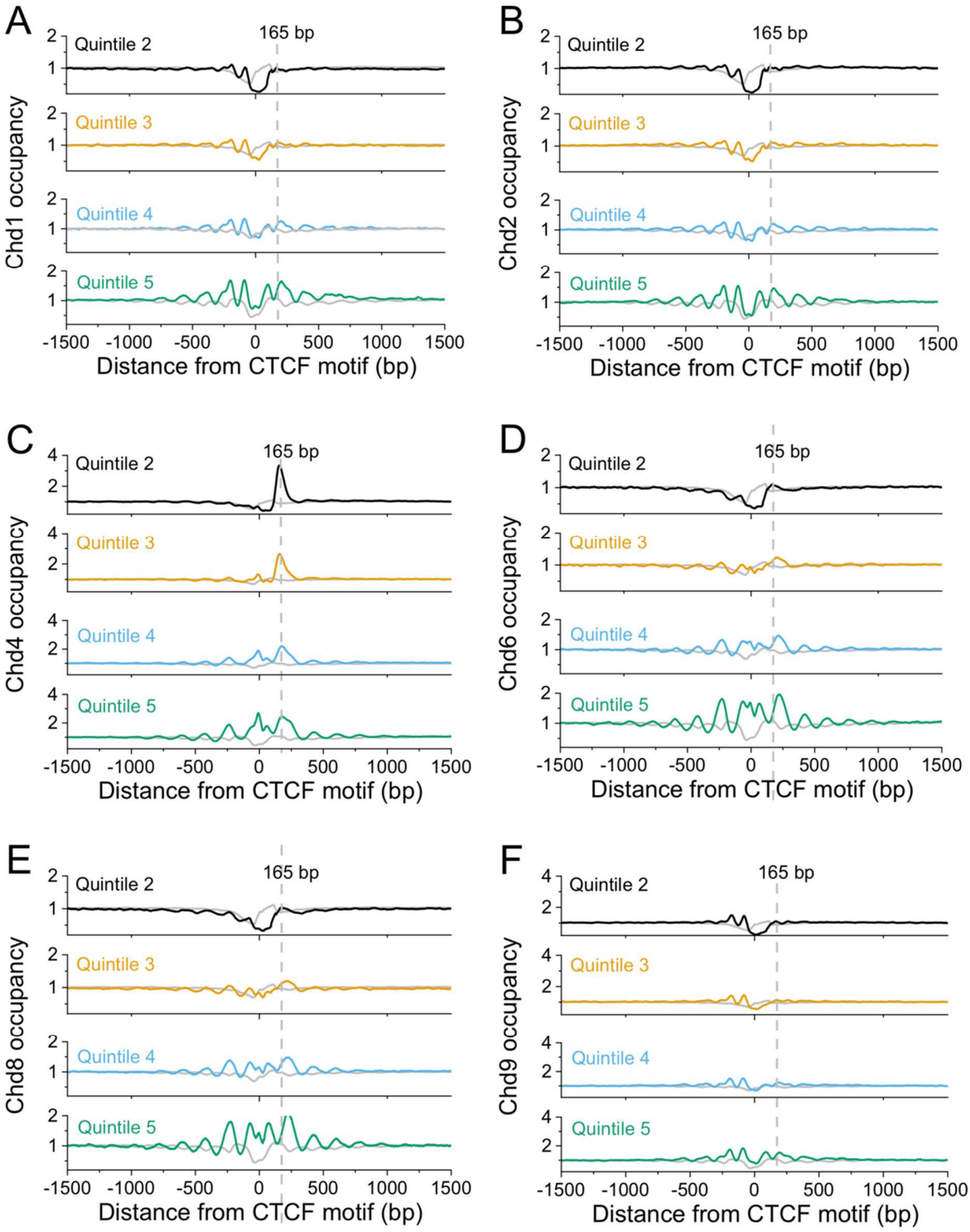


Figure S9. Aggregate nucleosome occupancy profiles around directional CTCF as in Figure 5, calculated using MNase-assisted histone H3 ChIP-seq signal from (Wiehle et al., 2019).


A
Chd1 occupancy
Quintile 2
165 bp
Quintile 3
Quintile 4
Quintile 5
-1500 -1000 -500 0 500 1000 1500
Distance from CTCF motif (bp)
B
Chd2 occupancy
Quintile 2
165 bp
Quintile 3
Quintile 4
Quintile 5
-1500 -1000 -500 0 500 1000 1500
Distance from CTCF motif (bp)
C
Chd4 occupancy
Quintile 2
165 bp
Quintile 3
Quintile 4
Quintile 5
-1500 -1000 -500 0 500 1000 1500
Distance from CTCF motif (bp)
D
Chd6 occupancy
Quintile 2
165 bp
Quintile 3
Quintile 4
Quintile 5
-1500 -1000 -500 0 500 1000 1500
Distance from CTCF motif (bp)
E
Chd8 occupancy
Quintile 2
165 bp
Quintile 3
Quintile 4
Quintile 5
-1500 -1000 -500 0 500 1000 1500
Distance from CTCF motif (bp)
F
Chd9 occupancy
Quintile 2
165 bp
Quintile 3
Quintile 4
Quintile 5
-1500 -1000 -500 0 500 1000 1500
Distance from CTCF motif (bp)

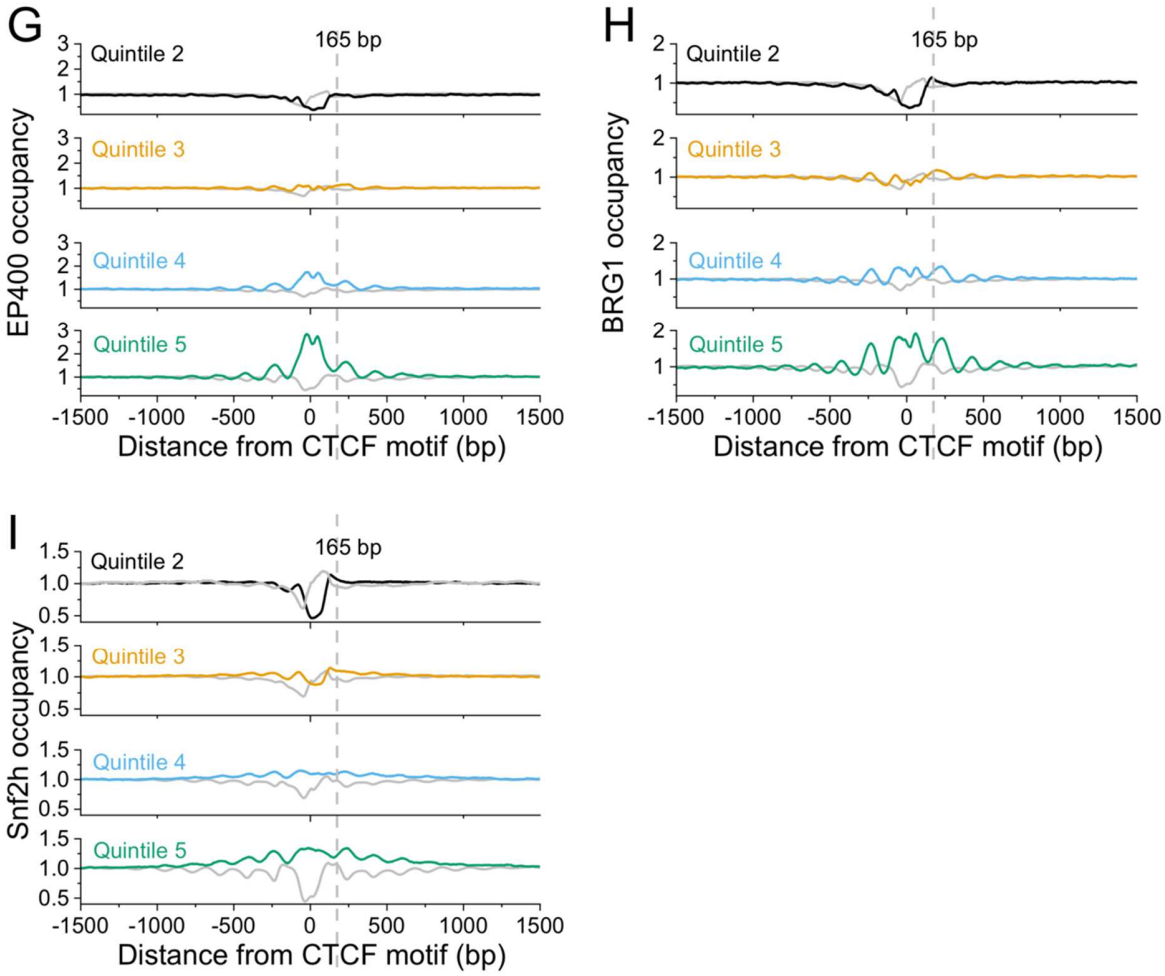


Figure S10. Aggregate profiles of the occupancy of chromatin remodellers near directional CTCF motifs in ESCs. Aggregate occupancy profiles for nine chromatin remodellers around all predicted CTCF motifs in the mouse genome show that only Chd4 has a CTCF-dependent peak at 165 bp (note a different scale for Chd4). Remodeller profiles are aligned around CTCF motifs split into quintiles with increasing CTCF binding strength as follows: black – 2 nd quintile; orange – 3 rd quintile; blue – 4 th quintile; green – 5 th quintile. Grey solid lines show the corresponding nucleosome occupancy based on MNase-seq as in Figure 5.

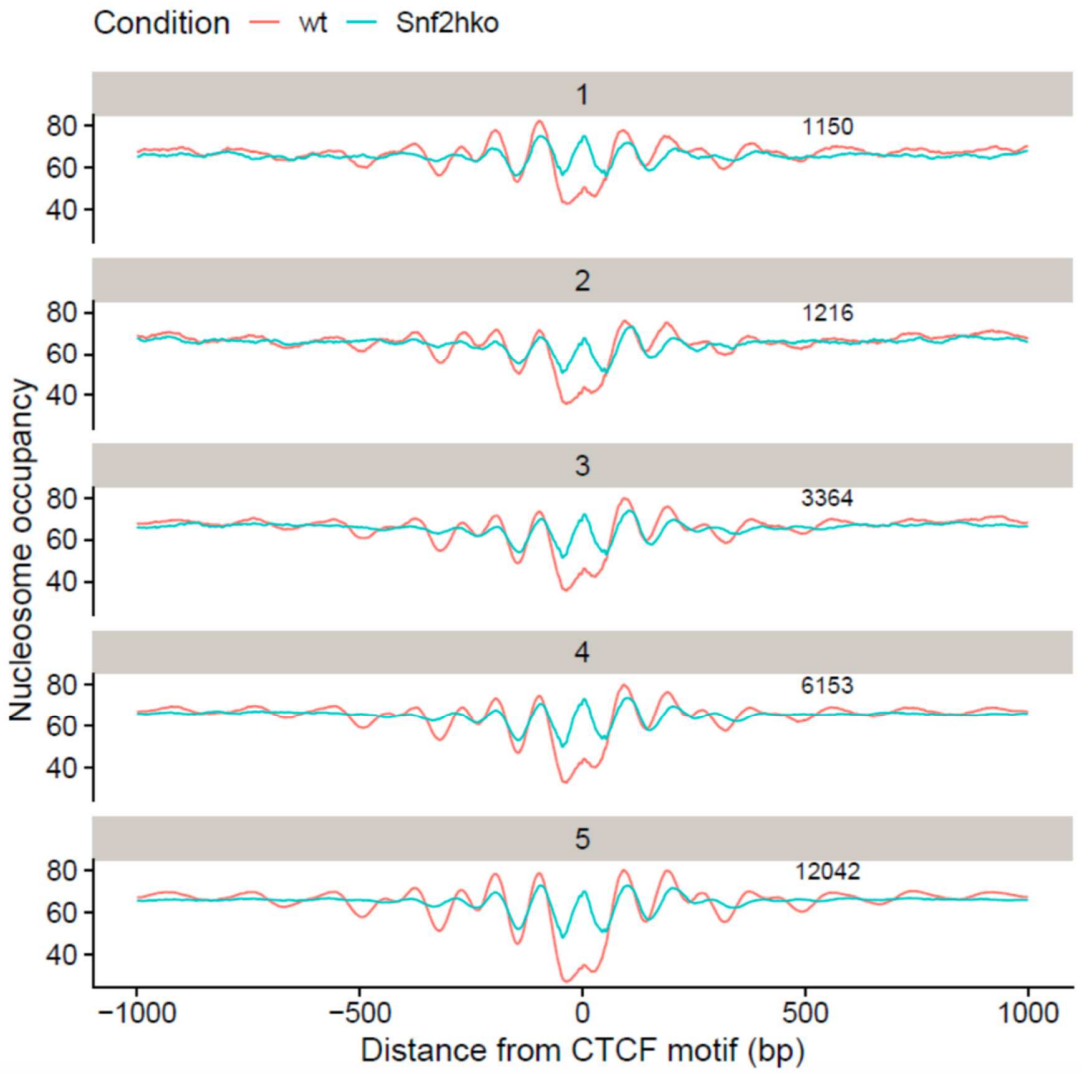


Figure S11. Aggregate profiles of nucleosome occupancy around CTCF sites lost upon Snf2h knockout. Red lines – wild type ESCs, blue lines – Snf2h knockout. The calculations are performed based on MNase-seq and CTCF ChIP-seq data from (Rao *et al.* 2017). The CTCF motifs predicted in the mouse genome with up to 80% similarity score were split into 5 quintiles as described in Methods, and intersected with CTCF sites lost upon Snf2h knockout (Barisic *et al.* 2019). The numbers of CTCF motifs remaining in each quintile upon this intersection are indicated on the figure.

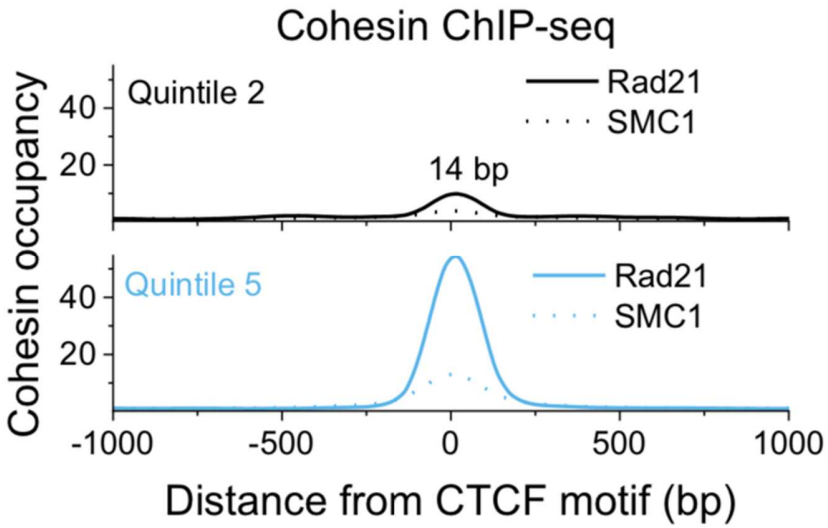


Figure S12. Aggregate profiles of cohesin subunits SMC1 and Rad21 measured by ChIP-seq in ESC, around directional CTCF motifs. The peak of Rad21 is shifted 14 bp from the center of CTCF motif while the peak of SMC1 coincides with the center of CTCF motif.

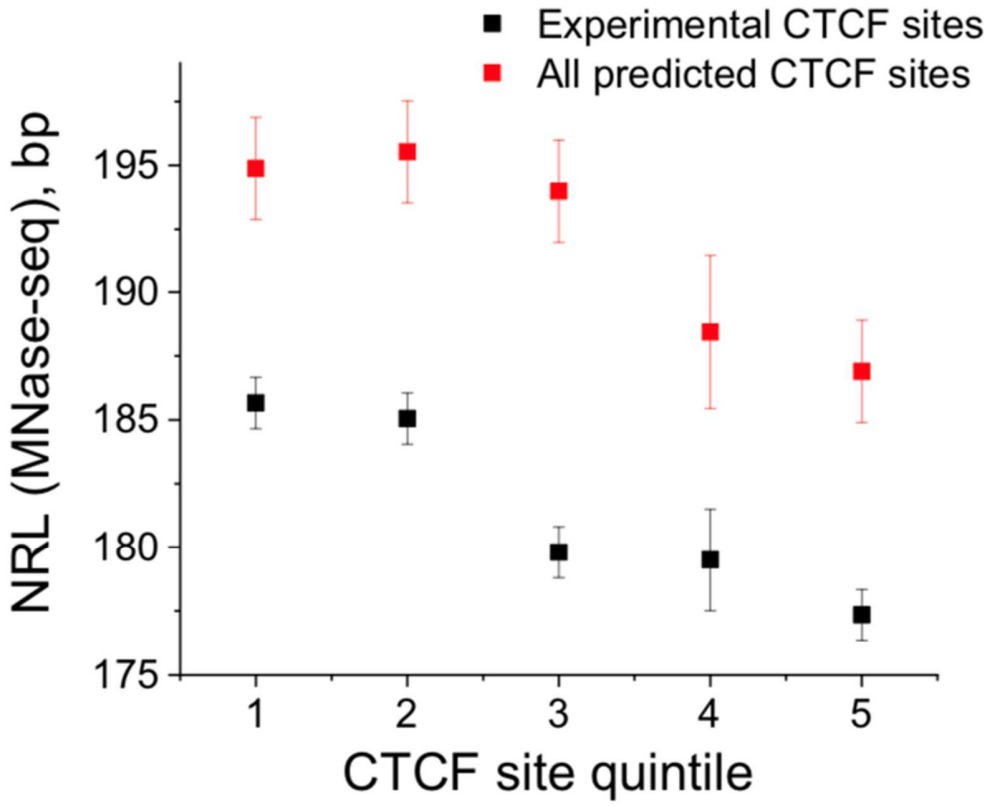


Figure S13. NRL in the region [300, 2000] near CTCF as a function of CTCF binding strength. The effect of NRL decrease with increase of CTCF binding strength remains even after excluding the CTCF-dependent nucleosome at +165 bp and performing NRL calculation for the region [300, 2000] downstream of CTCF sites (to the right from CTCF using plus strand coordinates).

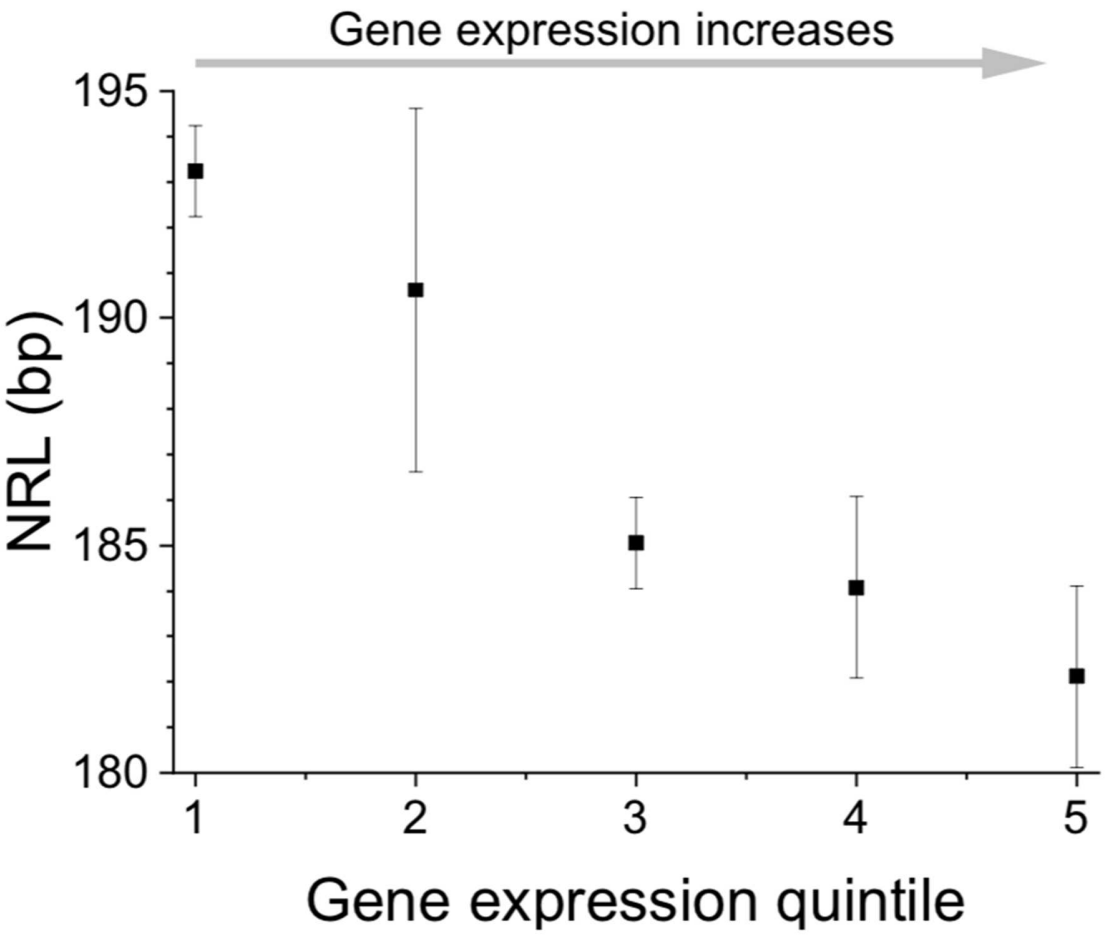


Figure S14. NRL in the region [-1000, 1000] near TSS as a function of gene expression. Genes have been split into 5 quintiles according to their normalised expression levels reported in (4).

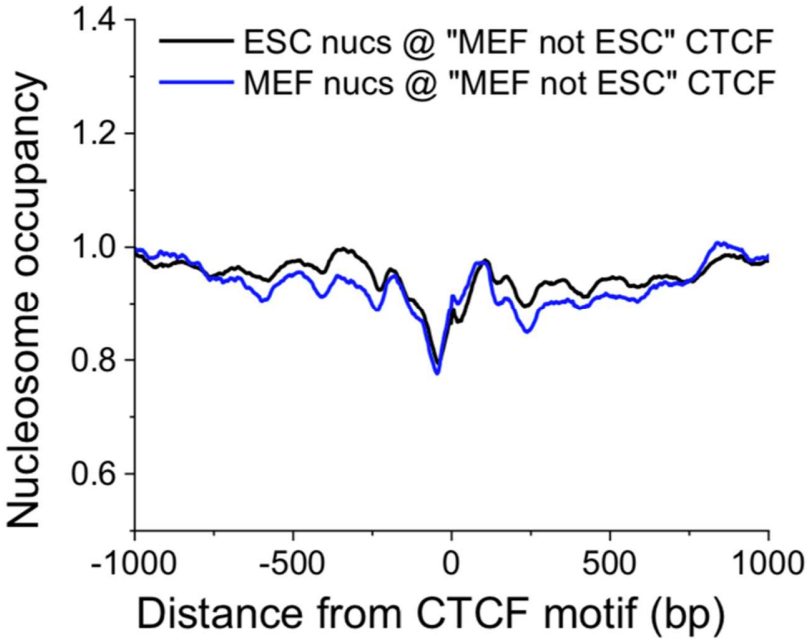


Figure S15. Effects of CTCF-dependent boundary directionality in stem cell differentiation. Nucleosome occupancy in ESCs (black) and MEFs (blue) around CTCF sites "MEF not ESC" that are present in MEFs but not in ESCs, calculated taking into account CTCF motif directionality.

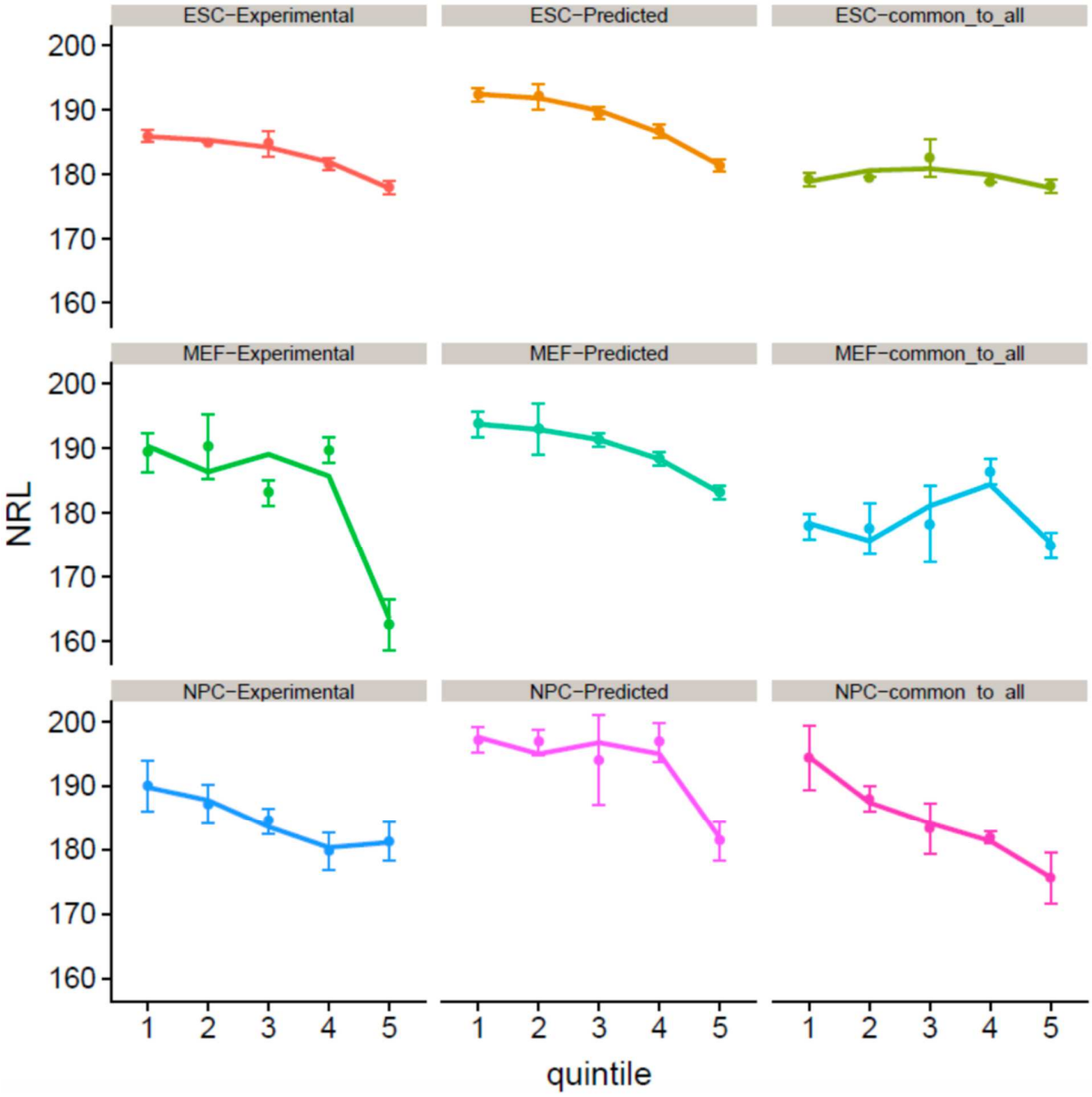


Figure S16. Effect of ESC differentiation on NRL. NRL values are calculated as a function of the CTCF site quintile. Top row: NRLs calculated based on the MNaseseq dataset in ESCs from Teif et al., 2012 for all experimental CTCF sites in ESCs determined in Shen et al, 2012 (left), all computationally predicted CTCF sites (middle) and common CTCF sites that have been determined experimentally in each of ESCs, NPCs and MEFs (right). Middle row: NRLs calculated based on the MNase-seq dataset in MEFs from Teif et al., 2012 for all experimental

CTCF sites in MEF determined in Shen et al, 2012 (left), all computationally predicted CTCF sites (middle) and common CTCF sites that have been determined experimentally in each of ESCs, NPCs and MEFs (right). Bottom row: NRLs calculated based on the MNase-seq dataset in NPCs from Teif et al., 2012 for all experimental CTCF sites in NPCs determined in Bonev et al., 2017 (left), all computationally predicted CTCF sites (middle) and common CTCF sites that have been determined experimentally in each of ESCs, NPCs and MEFs (right).

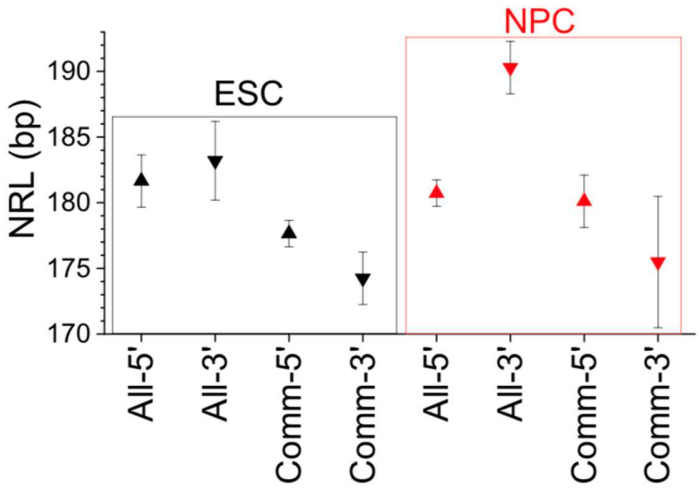


Figure S17. Effect of CTCF directionality on NRL preservation in ESC differentiation. NRLs in region [100, 2000] from CTCF's binding motifs overlapping with experimentally confirmed CTCF binding sites were calculated separately 5'-upstream and 3'-downstream of CTCF motifs in ESCs and NPCs. The major NRL change during differentiation is in the region 3'-downstream of CTCF motifs.

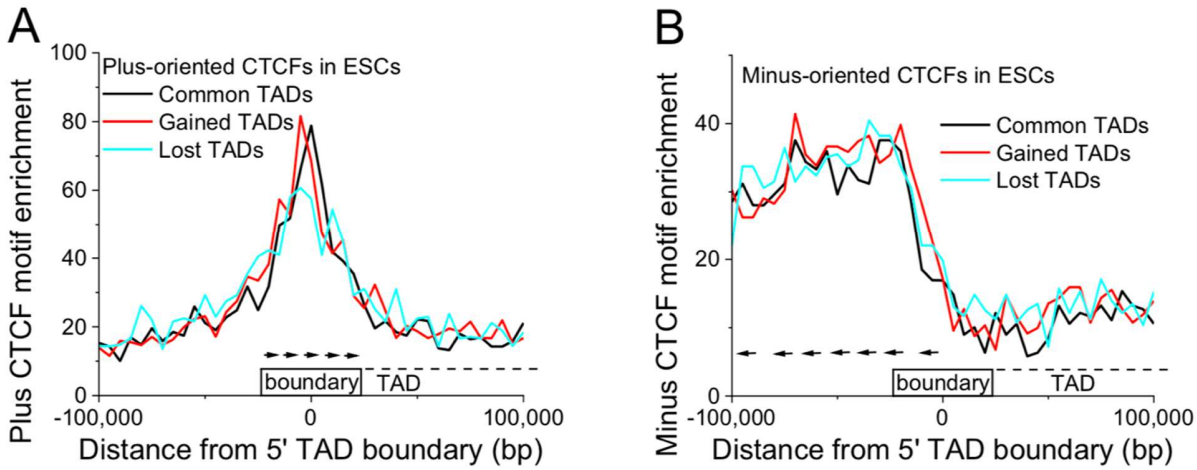


Figure S18. Enrichment of CTCF motifs bound by CTCF in ESCs near TAD boundaries common to ESC and NPC (black line), TAD boundaries not present in ESC but gained in NPCs (red) and TAD boundaries present in ESC but lost in NPC (light blue line). The arrows demonstrate directions of CTCF motifs.

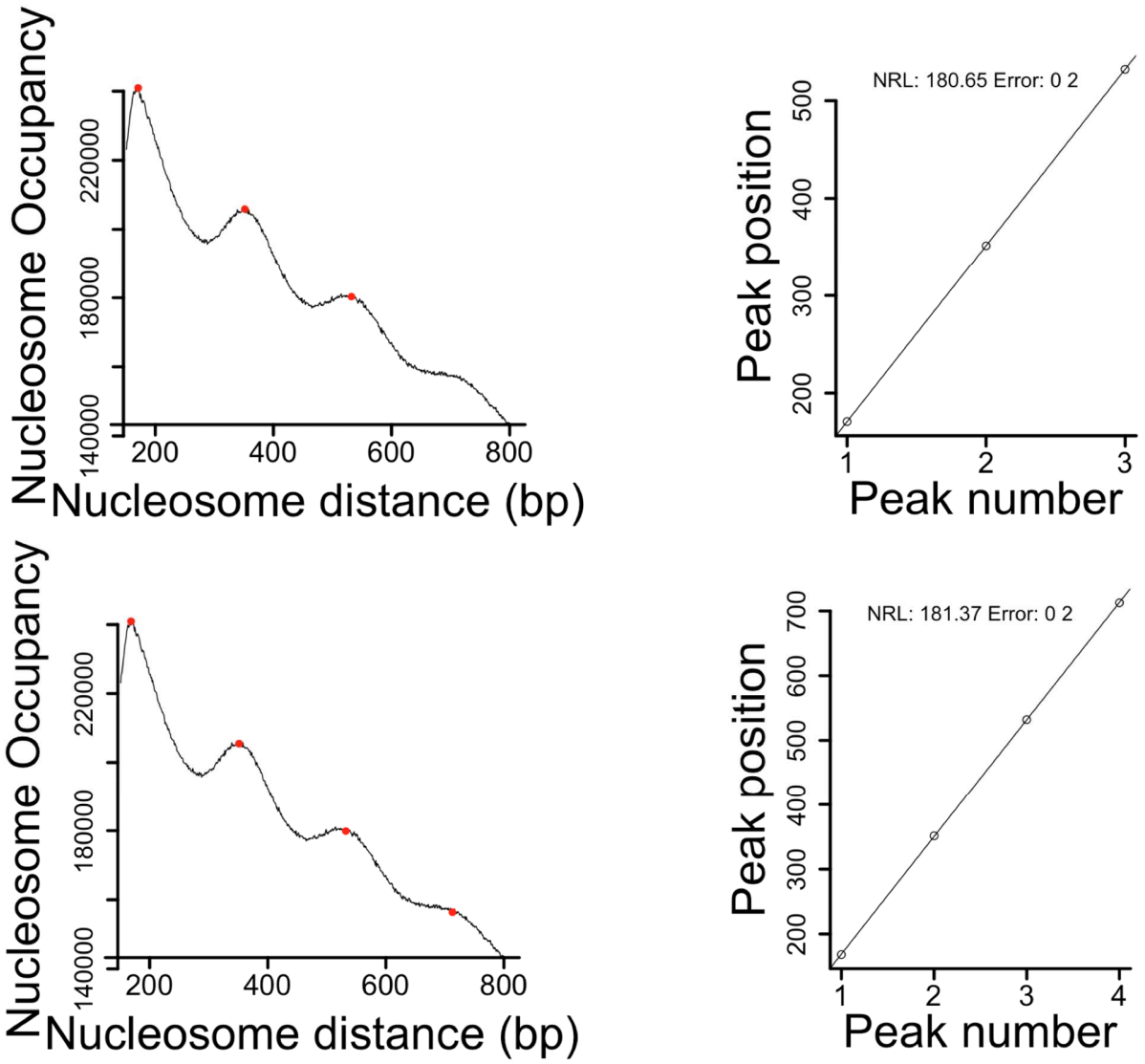


Figure S19. Comparing the resulting NRL from the selection of 3 points vs 4 points.

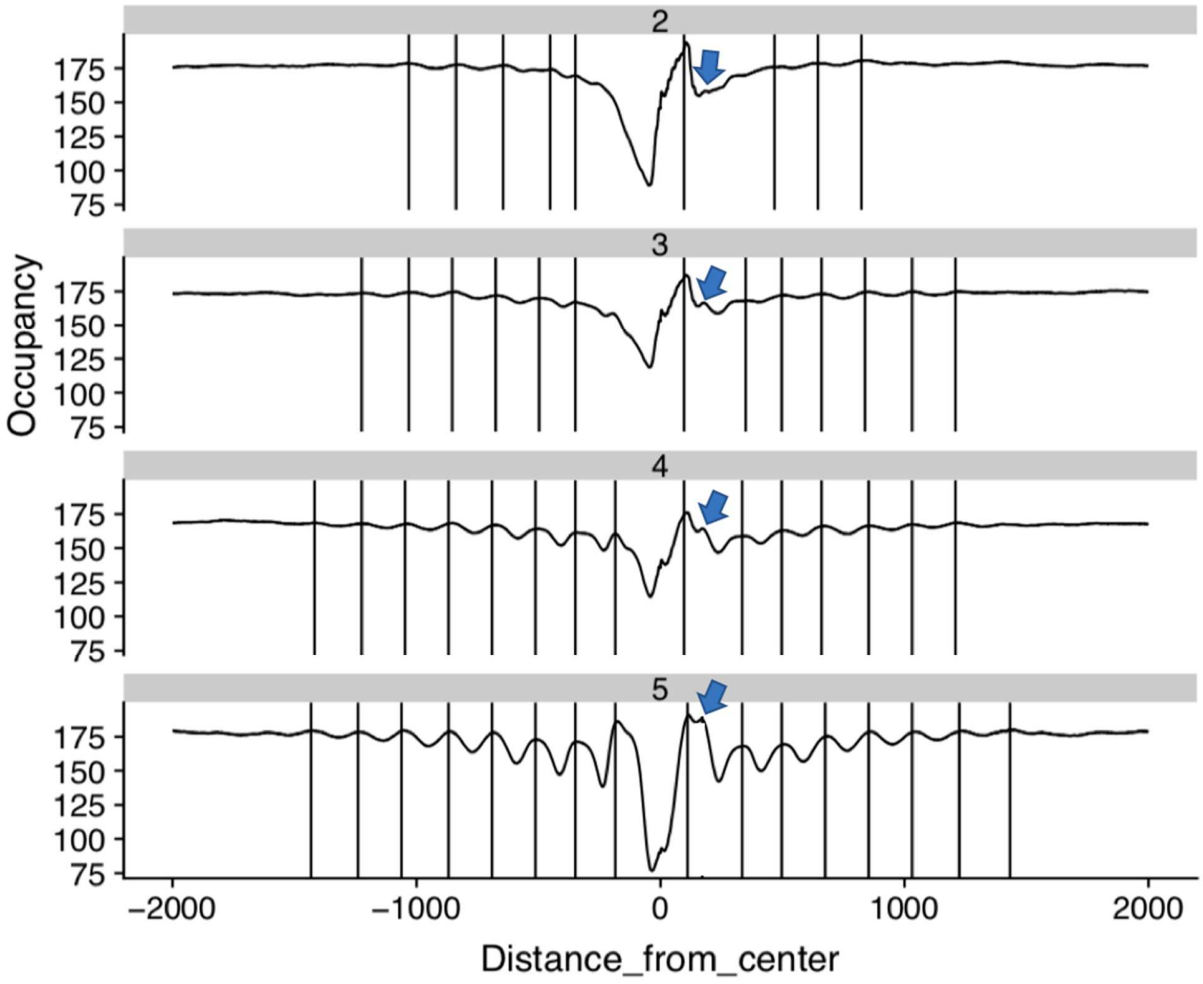


Figure S20. The chance of having a nucleosome at position 165 increases proportionally with binding strength.

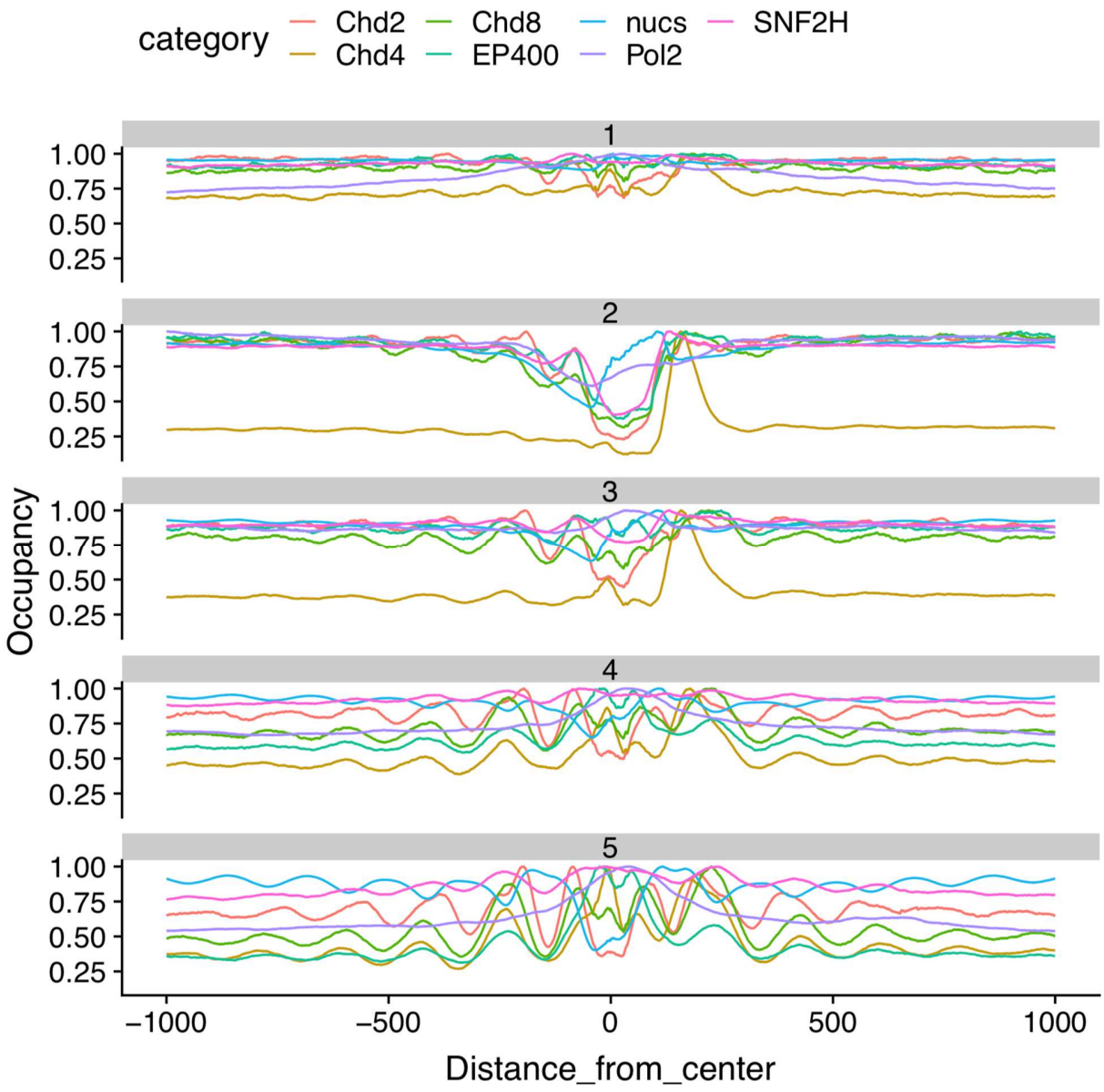


Figure S21. Oscillations of remodellers around CTCF.

5.2 Computer codes

5.2.1 Code used in “CTCF-dependent chromatin boundaries formed by asymmetric nucleosome arrays with decreased linker length”

The code used for the paper (Clarkson et al. 2019) is available at https://github.com/chrisclarkson/NRLcalc.

5.2.2 Code used in “Nucleosome positioning is predictive of chromatin state”

The combinations.txt file records all loci coordinates, state, patient number and other necessary data

job_no	chrom	start	end	patient	state	cancer
1	chr1	535800	566600	CES_CX0577_PE	TssA	CLL
2	chr1	570000	714800	CES_CX0577_PE	Tx	CLL
.						
.						
.						
.						
.						
n	chr1	5970700	5970800	CES_BK0916_PE	Tx	healthy

Python wrapper to extract data from ‘combinations.txt’ submit jobs in an array with cap on number of jobs submitted simultaneously

```
import pandas as pd
import subprocess
combs=pd.read_csv('combinations.txt',sep='\t',header=None).values
counter=1
first_submission='qsub -t 1-{} -N {} -tc 200 array.sh {}_combinations.txt'.format()
submission='qsub -t 1-{} -N {} -hold_jid {} -tc 200 array.sh {}_combinations.txt'.format()
for we in range(0,combs.shape[0]-70000,70000):
    start=i
    end=i+70000
    array_length=70000
    if end > combs.shape[0]:
        end=combs.shape[0]
        array_length=70000-end
    sub=combs[start:end,:]
    with open(str(counter)+'_combinations.txt', 'a') as f:
        pd.DataFrame(sub).to_csv(f, header=False)
    if counter > 1:
        subprocess.call(submission.format(str(array_length),str(counter),str(counter-
1),str(counter)).split())
    else:
        subprocess.call(first_submission.format(str(array_length),str(counter),str(counter)).
split())
    counter+=1
```

Bash script to extract nucleosomes into temporary BED file

```
#!/bin/bash

J=${JOB_NAME}

N=${SGE_TASK_ID}

info=`head /storage/projects/teif/CLL/_CancerEpiSys_new/ChromHMM/${J}_combinations.txt -
n ${N} | tail -n -1`

pair=($info)

c=${pair[1]}

echo ${c}

chr=${pair[2]}

start=${pair[3]}

end=${pair[4]}

pat=${pair[5]}

state=${pair[6]}

h=${pair[7]}

echo $state


cd ${h}_${state}_${pat}_mnase_csvs_bedtools/

echo $PWD     # create a working directory for this job if not already existing

        jobdir="${health}_${state}_${patient}_mnase_csvs_bedtools/"

        if [ ! -d $jobdir ];

        then

                echo "Jobdir    : created $jobdir"
```

```
            mkdir $jobdir
        fi
        cd $jobdir
        nucleosomefile="$nucleosomecore/${patient}/${patient}_nucleosomes_sorted.bed"
        awkoutputfile="${state}_${start}_${end}.bed"
        numpyscript="$coredir/numpy_rolling_window.py"
        echo "nuclmefile: $nucleosomefile"
        echo "awkoutput : $awkoutputfile"
        echo "numptyscrt: $numpyscript"
        echo "creating awk output file....."
        awk -v chrom="$chromosome" -v startval="$start" -v endval="$end" '{ if ($1 == chrom
&& $3 <= endval && $2 >= startval) print $0 }' $nucleosomefile > $awkoutputfile
        echo "DONE."
        if [ "$teststatus" == "real" ];
        then
            echo "CASE IS REAL. RUNNING numptyscript....."
            python $numpyscript ${state}_${health}_csvs $awkoutputfile ${start}_${end}_in
$start $end $chromosome
        fi
fi
```

Python script to write rows of tiled data to CSV files

```
import sys
import subprocess
import random
import pandas as pd
import numpy as np
from collections import Counter
import random

f=sys.argv[2]
file=pd.read_csv(str(f), sep='\t',header=None)

def sliding_window_center_on_first_nuc(start,sequence,winSize,step=1):
        # Verify the inputs
        if not ((type(winSize) == type(0)) and (type(step) == type(0))):
                raise Exception("**ERROR** type(winSize) and type(step) must be int.")
        if step > winSize:
                raise Exception("**ERROR** step must not be larger than winSize.")
        if winSize > len(sequence):
                raise Exception("**ERROR** winSize must not be larger than sequence length.")
        # Pre-compute number of chunks to emit
        numOfChunks = ((len(sequence)-winSize)/step)+1
        # Do the work
```

```
        nucpositions=np.where(sequence)[0]
        takeClosest = lambda num,collection:min(collection,key=lambda x:abs(x-num))
        for we in range(nucpositions[0],numOfChunks*step,step):
                i=takeClosest(i,nucpositions)
                if sequence[i:i+winSize].shape[0] == winSize:
                        yield sequence[i:i+winSize], start+i,start+i+winSize
                else:
                        break


job_id=str(sys.argv[3])
differences=[]
step=300
window=3000
starting=int(sys.argv[4])
end=int(sys.argv[5])
print(end-starting)
if end-starting>window:
        print(end-starting)
        values=np.zeros(int(end-starting))
        centers=(file[2]-file[1])/2+file[1]
        centers=centers.apply(lambda x: int(round(x)))
        unique_elements, counts_elements = np.unique(centers, return_counts=True)
        for n in xrange(0,len(unique_elements)):
```

```
            if starting <= unique_elements[n] <= end:
                values[unique_elements[n]-starting-1]=counts_elements[n]
        #result=strided_app(values, window, step)
        result=np.ones(window)
        starts=[]
        ends=[]
        for seq,start,end in sliding_window_center_on_first_nuc(starting,values,window,step):
            result=np.vstack([result,seq])
            starts.append(start)
            ends.append(end)
        result=result[1:]
        print(result)
        if 'in' in job_id:
            with open(str(job_id)+'_one_hot_encoded.csv', 'a') as f:
        pd.concat([pd.DataFrame(starts),pd.DataFrame(ends),pd.DataFrame(np.ones(result.sha
pe[0]).astype(int)), pd.DataFrame(result)],axis=1,ignore_index=True).to_csv(f, header=False)
        else:
            with open(str(job_id)+'_one_hot_encoded.csv', 'a') as f:
        pd.concat([pd.DataFrame(starts),pd.DataFrame(ends),pd.DataFrame(np.zeros(result.sh
ape[0]).astype(int)), pd.DataFrame(result)],axis=1,ignore_index=True).to_csv(f,
header=False)
```

5.3.3 Data Preparation

```
import pandas as pd
import numpy as np
dataframe = pd.read_csv('cancer_data/all_'+str(h)+'_'+str(state)+'.csv',header=None)
dataframe = dataframe.reindex(np.random.permutation(dataframe.index))
df=dataframe.values
X = df[0:df.shape[0],label_col+1:df.shape[1]].astype(int)
X = X / X.mean(axis=1,keepdims=True)
X=pad_sequences(X)
X = X.reshape(X.shape[0], X.shape[1],1)
y=df[:,label_col]
```

5.3.4 Classification

Binary classification

```
import tensorflow as tf
from keras.models import
from keras.layers import Sequential, Dense, Activation, Dropout, Conv1D, MaxPooling1D,
Flatten
from keras.utils import np_utils
def create_model(shape,repeat_length,stride):
        model = Sequential()
        model.add(Conv1D(75,repeat_length,strides=stride,padding='same',
input_shape=shape, activation='relu'))
```

model.add(MaxPooling1D(repeat_length))

model.add(Flatten())

model.add(Dense(1, activation='sigmoid'))

model.compile(loss='binary_crossentropy', optimizer='rmsprop', metrics=['accuracy'])

return model

Multi-label classification

```
def create_shallow_model_categorical(shape,repeat_length,stride,no_cats):
    model = Sequential()
    model.add(Conv1D(75,repeat_length,strides=stride, input_shape=shape,
activation='relu'))
    model.add(MaxPooling1D(repeat_length))
    model.add(Flatten())
    model.add(Dense(no_cats+1, activation='softmax'))
    model.compile(loss='categorical_crossentropy', optimizer='rmsprop',
metrics=['accuracy'])
    return model
```

Handling of imbalanced datasets

```
def handle_imbalance(…):
……
if np.count_nonzero(y) < np.count_nonzero(y==0):
        y_train=y[train]
        y_0_train=y_train[y_train==0]
        y_1_train=y_train[y_train==1]
        train_idx=np.random.randint(0,len(y_0_train),len(y_1_train))
        y_0_sub_train=np.array(y_0_train)[train_idx]
        train_y_2=np.concatenate((y_0_sub_train, y_1_train),axis=0)
        train_reshuffle=np.arange(train_y_2.shape[0])
        np.random.shuffle(train_reshuffle)
        train_y_2=train_y_2[train_reshuffle]
        ……. #same done for 'y_test', 'X_train' and 'X_test'
else:
        y_train=y[train]
        y_0_train=y_train[y_train==0]
        y_1_train=y_train[y_train==1]
        train_idx=np.random.randint(0,len(y_1_train),len(y_0_train))
        y_1_sub_train=np.array(y_1_train)[train_idx]
        train_y_2=np.concatenate((y_1_sub_train, y_0_train),axis=0)
        train_reshuffle=np.arange(train_y_2.shape[0])
        np.random.shuffle(train_reshuffle)
```

```
        train_y_2=train_y_2[train_reshuffle]
        …... #same done for 'y_test', 'X_train' and 'X_test'
return train_y_2, train_X_2, test_y_2, test_X_2
```

10-fold cross-validation

```
from sklearn.model_selection import StratifiedKFold
cv = StratifiedKFold(n_splits=10)
counter=1
for train, test in cv.split(X, y):
        train_y,train_X,test_y,test_X=sample_judge(X,y,train,test)
        print(counter)
        if model_type=='shallow':
                model=create_shallow_model(X.shape[1:],window,1)
        else:
                model=cnn_lstm(X.shape[1:],window,1)
        tprs,aucs=calculate_roc(model,3,100,train_X,train_y,test_X,test_y,tprs,aucs)
```

Grid search

```
parameters = {'batch_size': [100],
        'epochs': [3],
        'optimizer': ['adam', 'rmsprop'],
        'activation_last' : ['softmax'],
        'filters1' : [60,75,100],
```

```
        'kernel_sizes' : [2000,1000,3000,500]
}
def search_grid(parameters):
        flat = [[(k, v) for v in vs] for k, vs in parameters.items()]
        from itertools import product
        return [dict(items) for items in product(*flat)]
for p in grid:
        print(p)
        print(counter2)
        model=create_model(optimizer=p['optimizer'],activation_last =
p['activation_last'],dropout=0.2,kernel_sizes=p['kernel_sizes'],filte
        rs1=p['filters1'],no_cats=len(categories))
        Y_test,y_score=prepare_precision_recall_curve_with_cv(X, y,model,cv=10)
        from sklearn.metrics import precision_recall_curve
        from sklearn.metrics import average_precision_score
        precision = dict()
        recall = dict()
        average_precision = dict()
        n_classes=len(categories)
        for we in range(n_classes):
                precision[i], recall[i], _ = precision_recall_curve(Y_test[:, i],y_score[:, i])
                average_precision[i] = average_precision_score(Y_test[:, i], y_score[:, i])
        cmap='nipy_spectral'
```

```
        fig, ax = plt.subplots(1, 1)
        for we in range(0,n_classes):
                color = plt.cm.get_cmap(cmap)(float(i) / n_classes)
                ax.plot(recall[i], precision[i], lw=2,
                                label='{0} '
                                        '(area =
{1:0.3f})'.format(categories[i],average_precision[i]),color=color)
        ax.set_xlim([0.0, 1.0])
        ax.set_ylim([0.0, 1.05])
        ax.set_xlabel('Recall')
        ax.set_ylabel('Precision')
        #ax.tick_params(labelsize=text_fontsize)
        ax.legend(loc='upper center')
        plt.savefig(str(counter2)+'_'+typ+'_multi-class_included.png')
        plt.close()
        K.clear_session()
        mean_precision=np.mean(np.array(list(average_precision.values())))
        mean_precision=' '.join(str(x) for x in list(average_precision.values()))
        df_[counter2][typ]=str(mean_precision)
```

Performance assessment

Binary classification- ROC plots:

```
def roc():
        …..
        probas = model.predict_classes(test_X)
        fpr, tpr, thresholds = roc_curve(test_y, probas)
        tprs.append(interp(mean_fpr, fpr, tpr))
        tprs[-1][0] = 0.0
        roc_auc = auc(fpr, tpr)
        aucs.append(roc_auc)
        ……
        mean_fpr = np.linspace(0, 1, 100)
        if not i:
                plt.plot([0, 1], [0, 1], linestyle='--', lw=2, color='r',label='Luck', alpha=.8)
        mean_tpr = np.mean(tprs, axis=0)
        #mean_tpr[-1] = 1.0
        mean_auc = auc(mean_fpr, mean_tpr)
        std_auc = np.std(aucs)
        col=['b','g','k']
        plt.plot(mean_fpr, mean_tpr, color=col[i],label=str(types)+r' Mean ROC (AUC = %0.2f
$\pm$ %0.2f)' % (mean_auc, std_auc),lw=2, alpha=.8)
        std_tpr = np.std(tprs, axis=0)
        tprs_upper = np.minimum(mean_tpr + std_tpr, 1)
```

```
        tprs_lower = np.maximum(mean_tpr - std_tpr, 0)
        plt.fill_between(mean_fpr, tprs_lower, tprs_upper, color='grey',
alpha=.2,label=r'$\pm$ 1 std. dev.')
        plt.xlim([-0.05, 1.05])
        plt.ylim([-0.05, 1.05])
        plt.xlabel('False Positive Rate')
        plt.ylabel('True Positive Rate')
        plt.legend(loc="lower right")
        return plt
```

Multi class labelled systems- precision recall curves

```
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import average_precision_score
precision = dict()
recall = dict()
average_precision = dict()
n_classes=len(categories)
for we in range(n_classes):
        precision[i], recall[i], _ = precision_recall_curve(Y_test[:, i],y_score[:, i])
        average_precision[i] = average_precision_score(Y_test[:, i], y_score[:, i])
cmap='nipy_spectral'
fig, ax = plt.subplots(1, 1)
for we in range(0,n_classes):
```

```
        color = plt.cm.get_cmap(cmap)(float(i) / n_classes)
        ax.plot(recall[i], precision[i], lw=2,
                        label='{0} '
                                '(area =
{1:0.3f})'.format(categories[i],average_precision[i]),color=color)
ax.set_xlim([0.0, 1.0])
ax.set_ylim([0.0, 1.05])
ax.set_xlabel('Recall')
ax.set_ylabel('Precision')
```

Training the model

```
model.fit(train_X, train_y,epochs=e,batch_size=b)
```

5.3.5 Explainable AI

SHAP

```
import shap

def vis(seq_to_explain,train_X):

        background = train_X[np.random.choice(train_X.shape[0], 100, replace=False)]

        e = shap.DeepExplainer(keras_model, background)

        shap_values = e.shap_values(seq_to_explain)

        shap_value_reshape=shap_values[0].reshape((1,3000))[0]

        plt.plot(shap_value_reshape)
```

GAN

```
#build generator

def build_generator(shape):

        model = Sequential()

        model.add(Conv1D(128, kernel_size=3, padding="same",input_shape=shape))

        model.add(BatchNormalization(momentum=0.8))

        model.add(LeakyReLU(alpha=0.2))

        model.add(UpSampling1D())

        model.add(Conv1D(64, kernel_size=3, padding="same"))

        model.add(BatchNormalization(momentum=0.8))

        model.add(LeakyReLU(alpha=0.2))

        model.add(UpSampling1D())

        model.add(Conv1D(64, kernel_size=3, padding="same"))
```

```
        model.add(BatchNormalization(momentum=0.8))
        model.add(Activation("sigmoid"))
        model.add(Conv1D(1, kernel_size=3, padding="same",activation='sigmoid')) # ensure
fake output is binary to be consistent with real input
        #model.add(Activation("tanh"))
        model.summary()
        return model

noise=np.random.normal(0, 1, (1, int(train_X.shape[1]/4))) #start with noise input ¼ the width
of X
noise=noise.reshape(noise.shape[0], noise.shape[1],1)
generator = build_generator(noise.shape[1:])
first_half_length=len(generator.layers)

# For the combined model we will only train the generator
img_shape = (1, train_X.shape[1], 1)
latent_dim = train_X.shape[1]
optimizer = Adam(0.0002, 0.5)

def add_layers(generator,discriminator,number): #add the generator and discriminator to make
a new model
        for l in discriminator.layers[0:number]:
                generator.add(l)
```

```
    return generator

combined=add_layers(generator,discriminator,len(discriminator.layers))

def save_imgs(epoch,gen_loss,dis_loss,perf,state,typ,wassers): # record sample fake outputs at
regular intervals of GAN training
    noise = np.random.normal(0, 1, (1, int(train_X.shape[1]/4)))
    noise=noise.reshape(noise.shape[0], noise.shape[1],1)
    gen_imgs = generator.predict_classes(noise)
    gen_imgs=gen_imgs.reshape(gen_imgs.shape[1])
    plt.plot(gen_imgs)
    plt.savefig(str(state)+"_epoch_%d.png" % epoch)
    plt.close()
    plt.plot(gen_loss)
    plt.plot(dis_loss)
    plt.plot(wassers)
    plt.plot(perf)
    plt.yscale('symlog')
    leg = plt.legend(('Generator loss','Discriminator loss','Earth mover distance', 'Test
performance'),loc='best')
    leg.get_frame().set_alpha(0.5)
    plt.savefig(str(state)+'_'+str(epoch)+'_fully_latest_gan.png')
    plt.close()
```

```
generator=add_layers(Sequential(),combined,first_half_length)
gen_loss=[]
dis_loss=[]
perf=[]
wassers=[]
epochs=500
batch_size=500
save_interval=20
valid = np.ones((batch_size, 1))
fake = np.zeros((batch_size, 1))
for epoch in range(epochs+1): #loop through 500 epochs recreating the combined generator
discriminator each time
        idx = np.random.randint(0, train_X.shape[0], batch_size)
        imgs = train_X[idx]
        # Sample noise and generate a batch of new images
        noise = np.random.normal(0, 1, (batch_size, int(train_X.shape[1]/4)))
        noise=noise.reshape(noise.shape[0], noise.shape[1],1)
        gen_imgs = generator.predict_classes(noise)
        # Train the discriminator (real classified as ones and generated as zeros)
        d_loss_real = discriminator.train_on_batch(imgs, valid)
        d_loss_fake = discriminator.train_on_batch(gen_imgs, fake)
        d_loss = 0.5 * np.add(d_loss_real, d_loss_fake)
        # Train the generator (wants discriminator to mistake images as real)
```

```
        g_loss = combined.train_on_batch(noise, valid)
        generator=add_layers(Sequential(),combined,first_half_length)
        gen_loss.append(g_loss)
        dis_loss.append(d_loss[0])
        # Plot the progress
        idx_te = np.random.randint(0, test_X.shape[0], batch_size)
        imgs = test_X[idx_te]
        # Sample noise and generate a batch of new images
        noise = np.random.normal(0, 1, (batch_size, int(test_X.shape[1]/4)))
        noise=noise.reshape(noise.shape[0], noise.shape[1],1)
        gen_imgs = generator.predict_classes(noise)
        # Train the discriminator (real classified as ones and generated as zeros)
        ims=np.concatenate((gen_imgs,imgs),axis=0)
        ys=np.concatenate((fake,valid),axis=0)
        performance = discriminator.test_on_batch(ims, ys)
        perf.append(performance[1]*100)
        print ("%d [D loss: %f, acc.: %.2f%%] [G loss: %f] [Performance: %.2f%%]" %
(epoch, d_loss[0], 100*d_loss[1], g_loss,100*performance[1]))
        wasser_v=wasser_loss(gen_imgs,imgs)
        wassers.append(wasser_v)
        if epoch % save_interval == 0:
                save_imgs(epoch,gen_loss,dis_loss,perf,state,typ,wassers)
        K.clear_session()
```