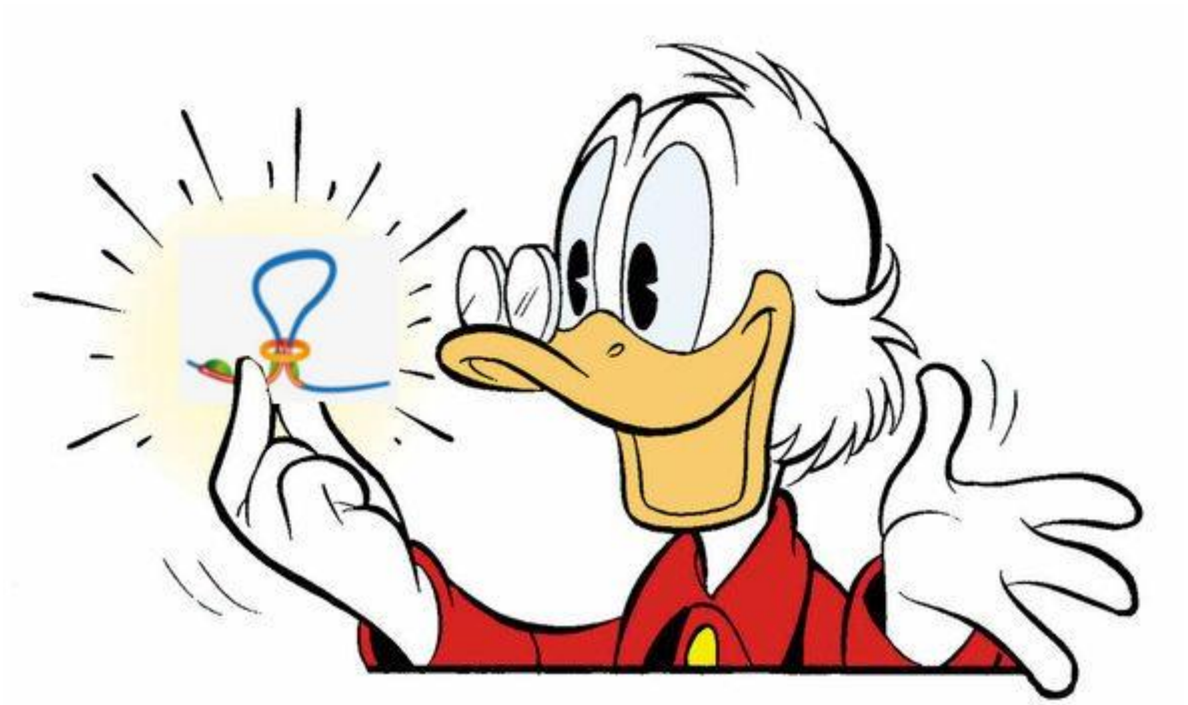
Large-scale discovery of epigenomic factors important for chromatin looping using Random Forests



By: Dieter Gerrit Gijsbert Stoker
Student number: 4159853
Supervisor: Dr. Jeroen de Ridder
Daily supervisor: Dr. Amin Allahyar
Second reviewer: Prof. Dr. Wouter de Laat

Abstract

The genome is folded into a functional 3D structure that influences transcription of genes and accessibility of DNA for modifications. This 3D structure encompasses many levels, from chromosome territories to topologically associated domains (TADs) to chromatin loops. Sequencing-based technologies to investigate 3D genome architecture have proliferated in recent years, allowing ever-deeper insight into the mechanics of this regulatory layer. Cohesin and CTCF are chiefly responsible for chromatin loop formation and maintenance. These factors, however, do not exist in an epigenomic vacuum, but rather in a complex landscape of epigenomic modifications and transcription factor binding events. Small-scale studies have highlighted the effects of some histone modifications for loop formations. However, a large-scale unbiased investigation into the importance of epigenomic modifications for chromatin loop formation is lacking. Here, we tackle this question using a machine-learning approach. We investigate what epigenomic signatures differentiates loop anchors from other genomic areas, pairs of anchors that loop from those that do not, and looping anchors in one cell type from those in another. We show that H3K36me3, H3K4me1, and H3K4me2 are important signals of chromatin loop anchors. Additionally, we show that epigenomic modifications in a cell type are highly predictive of looping in that cell type, with features between loop anchors being the best predictors for loop provenance. Lastly, we identify ATF-2 and HDGF as novel targets of interest for chromatin loop formation regulation. Our approach constitutes the first complete and unbiased look at epigenomic modification importance and can serve as a baseline of factor importance for more intricate classification schemes.

Layman’s summary

Every cell in your body has the same DNA, yet a liver cell has to do very different things than an immune cell. How is this done? To make sure that different cells do different things, there are many so-called regulatory layers that influence which cell does what. To understand how that works, let’s first take a look at the central dogma. This states that DNA is read by RNA polymerase, which makes a molecule of messenger RNA (mRNA). This is called transcription. This mRNA can travel outside the nucleus (the heart of the cell where DNA resides in eukaryotes, such as animals, plants, and fungi) and dock at a ribosome. A ribosome is a protein factory that reads off the mRNA and links amino acids together in a sequence stated in the code of the mRNA. The end result is a protein with a certain function, for example a sensor that senses bacteria for an immune cell, or an enzyme that can break down alcohol for liver cells. Thus, DNA is made into (m)RNA which is translated into protein. We said that RNA polymerase makes mRNA in this process. To do that, it needs to bind to a specific site in the DNA. This is one regulatory layer. Proteins called transcription factors bind to certain parts of the DNA in certain cell types, and make sure that RNA polymerase binds there to transcribe that gene. Different cells make different transcription factors. Another regulatory layer is the mRNA itself. Different modifications to the mRNA can change its stability. Less stable mRNA might never make it to a ribosome, while very stable mRNA could be read by a ribosome multiple times, leading to multiple copies of a protein. Here, we are interested in another regulatory layer: the 3D folding of the genome. The DNA is not a long flat stretch in your cells, but rather is tightly coiled up, and wound around proteins called histones. Modifications to these histones cause the DNA to be more open or closed, and therefore easier or harder to access for transcription factors, respectively. That is not all, however. On a somewhat larger scale there are chromatin loops, stretches of DNA that are extruded and held together by two anchors that act as clamps. Such a loop can therefore bring regions of the DNA that are far away close together. This can lead to interactions, allowing, for example, different transcription factors to bind than would otherwise, and hence leading to different proteins. In this project, we investigate what modifications to DNA (binding of transcription factors and/or histone modifications) influence the formation of loops. We do this by giving examples of regions that loop and that do not and the DNA modifications there, and letting a machine-learning algorithm figure out what the differences between the two are. By doing this on a large scale, we gain new insights into what causes loops. Understanding how they form is the first step to fixing them when things go awry in, for example, cancer cells.

Table of Contents

Introduction

The 3D structure of the genome controls the transcriptional output of genes[1]. At the turn of the millennium, this regulatory layer was perhaps the least understood of all[1]. Since then, the power of (next-generation) sequencing has been brought to bear upon this problem, resulting first in Chromosome Conformation Capture (3C), and recently in the genome-wide technique HiC[2–4]. Both techniques use the same core principles: DNA is cross-linked, cut, and ligated, and the resultant products are sequenced. DNA sequence that was close together in 3D but that is linearly far apart on the genome can thereby be identified, shedding light on the 3D structure of the genome[5]. Variations exist: cross-linking can be omitted[6,7], all interactions mediated by a specific protein can be selected for (e.g. ChIA-PET)[8,9], and long-read sequencing in MC4C allows identification of many interacting regions at once[10]. Large-scale studies of the human genome have uncovered hallmarks of genome organisation: chromosome territories[1], areas with active and inactive chromatin (A/B compartments)[11,12], topologically associated domains (TADs, larger genomic areas that have more genomic interactions within them than with DNA outside of them)[13,14], chromatin loops[15], and finally nucleosome clusters and nucleosomes[16] (**Figure 1**).

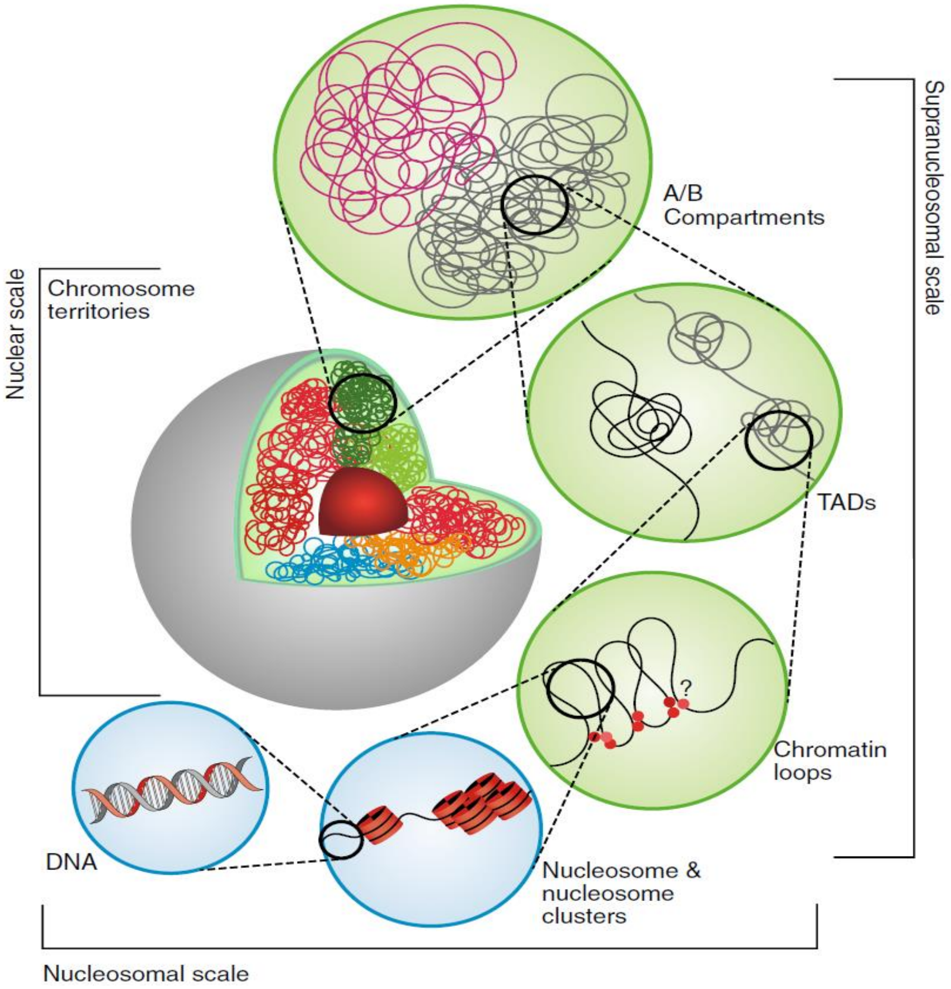


Figure 1. 3D organisation of DNA in the nucleus. *Chromosome territories, A/B compartments, TADs, chromatin loops and nucleosomes all play a role in regulating the accessibility of and contacts within DNA. Figure taken from:* [16].

From these large-scale studies, it has become clear that CTCF and cohesin are the chief regulators of chromatin loop formation[15]: only 18% of loops are (partially) resistant to ablation of CTCF[17], and cohesin loss removes all loop domains over time[18]. Current understanding is that cohesin complexes extrude DNA loops until they encounter CTCF bound at convergently-oriented motifs and then stop[17–19]. How exactly cohesin is able to rapidly extrude DNA yet sense small differences in orientation of CTCF proteins remains to be determined[17]. While cohesin and CTCF are thus highly important, less is known about other factors influencing chromatin loop formation. Recent machine-learning approaches that try to predict loops, such as Lollipop and CTCF-MP, have included both sequence-based features and a few epigenomic factors to make their predictions[20,21]. Fewer studies focus solely on epigenomic factor importance. Nine histone modifications and CTCF have been shown to have some predictive power in delineating loop anchors that loop with many different genomic locations from those that loop with few genomic locations[22].

Another study used seven histone marks in murine cells and found that DNA between anchors that form a loop was enriched in H3K4me1, H3K36me3 and H3K27me3 marks, and was able to identify five categories of loops with differing levels of enrichment and depletion of the marks studied[23]. Most recently, 28 epigenomic factors (CTCF, histone marks, and transcription factors) were found to be predictive for microTAD boundaries[24]. These efforts show that specific epigenomic marks are associated with loop (or microTAD) formation, but a systematic inquiry into the importance of all epigenomic marks, including transcription factor binding, for loop formation is lacking. Given the current deluge of data on genome interactions, including data with up to 10 regions contacting each other at the same time[10], it is becoming both more feasible and more important to understand how loops are formed. This, combined with the important regulatory function of chromatin loops, and their recurrent disruption in cancer[25], means that more fundamental knowledge on their formation and regulation is urgently needed.

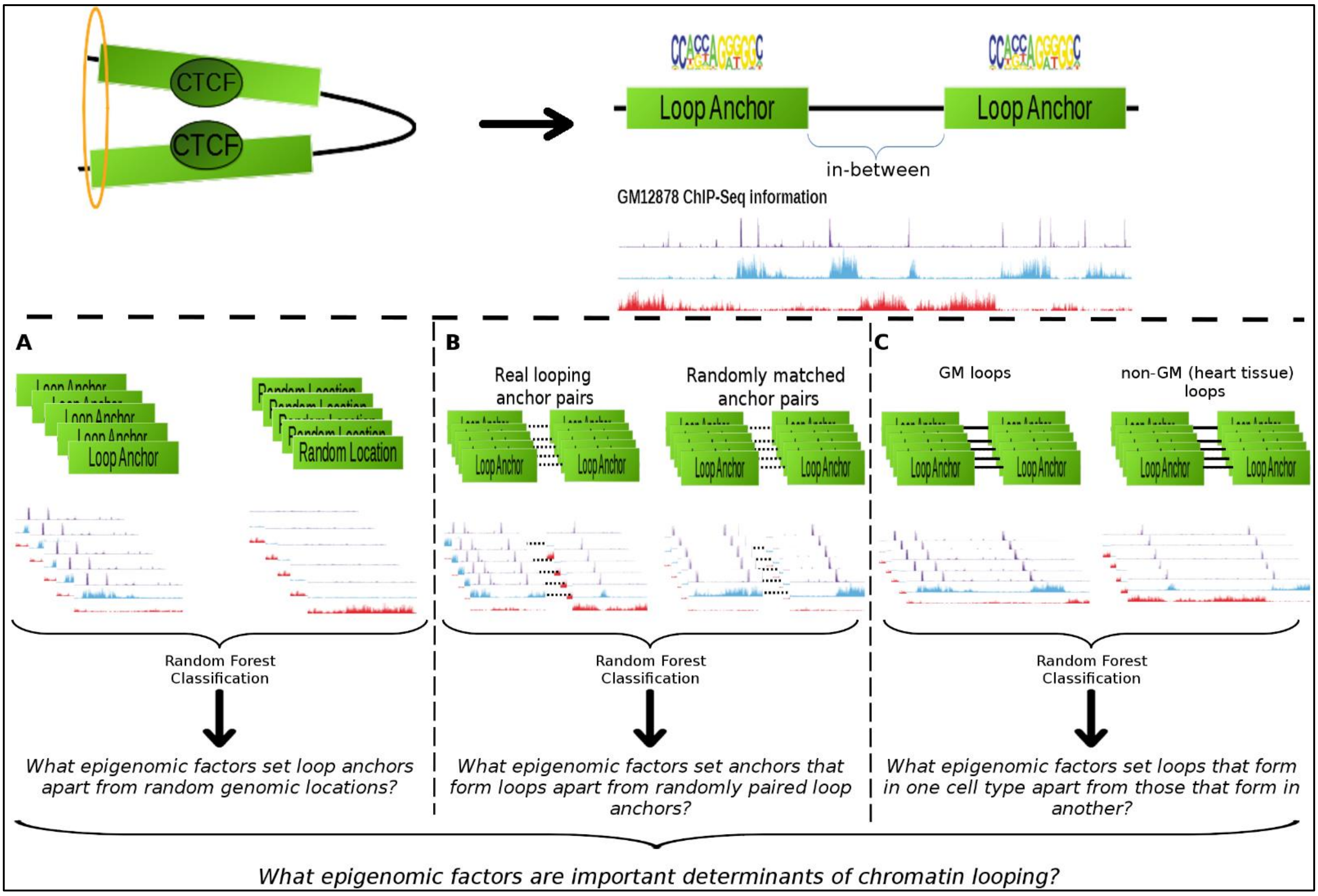

CTCF
CTCF
Loop Anchor
Loop Anchor
in-between
GM12878 ChIP-Seq information
A
Loop Anchor
Random Location
Random Forest Classification
What epigenomic factors set loop anchors apart from random genomic locations?
B
Real looping anchor pairs
Randomly matched anchor pairs
Loop Anchor
Random Forest Classification
What epigenomic factors set anchors that form loops apart from randomly paired loop anchors?
C
GM loops
non-GM (heart tissue) loops
Loop Anchor
Random Forest Classification
What epigenomic factors set loops that form in one cell type apart from those that form in another?
What epigenomic factors are important determinants of chromatin looping?

Figure 2. Using machine-learning to uncover epigenomic factors important for chromatin looping*. Top left: schematic of a chromatin loop, formed between two CTCF-bound sites, held together by cohesin (yellow ring). Top right: loop data consists of loop anchors (with CTCF motifs) and the areas in between for GM12878 (GM) and Heart (called loops obtained from de Laat et al.). Features are calculated from ChIP-Seq data for 153 chromatin marks in GM cells downloaded from Encode[26]. A: classification of single loop anchors and a negative set of random genomic locations. B: classification of pairs of looping anchors and a negative set of randomly matched anchor pairs. C: classification of chromatin loops as GM or non-GM using ChIP-Seq data from GM. Using this three-pronged approach, we uncover what epigenomic factors are important determinants of chromatin looping.*

In this work, we perform a large-scale study on the importance of epigenomic factors for loop formation using a machine-learning approach. We obtained a dataset with 259,936 loops in heart tissue and an immune cell line and downloaded data on 153 ChIP-Seq marks. We then set out to answer three fundamental questions (**Figure 2**): a) Are there combinations of epigenomic marks that differentiate chromatin loop anchor sites from non-loop anchor sites?, b) Are there combinations of epigenomic marks that differentiate pairs of chromatin loop anchor sites that form loops together from those that do not?, and c) Are there combinations of epigenomic marks that differentiate loops that occur in one cell type from those that do not? We trained Random Forest (RF) classifiers in each case and determined what epigenomic marks were most informative for the classification.

We show that anchor sites are only moderately distinguishable from non-anchor sites in their epigenomic signature, that pairs of looping anchor sites are not different from pairs of non-looping anchor sites in their epigenomic signature, and that one can distinguish loops that occur in a cell type from those that do not based solely on epigenomic marks with high accuracy. Additionally, we find that epigenomic modifications between two loop anchors are most informative for accurate prediction of loop provenance, retrieve histone marks known to be important for loop formation, and identify ATF-2 and HDGF as possible regulators of looping. This work advances the field by giving an unbiased view of the importance of all measured epigenome factors to chromatin looping, and provides a baseline classification performance so that improvements via more complex methodologies (such as deep neural nets) can be compared for their added value.

Results

Real single loop anchors are weakly distinct from randomly sampled anchor sites

We first wanted to know what epigenomic factors might differentiatie between loop anchors and other genomic locations. To examine this, we used 20,000 high-confidence chromatin loops called in GM12878 HiC data (**see methods**). Every loop has two anchor positions (the focal residues that form the loop, **see methods**) and an anchor size. The latter is an interval in base pairs (bp) around the anchor sites for which we incorporate available ChIP-Seq information in the features for that loop anchor. We did not know a priori how much of the epigenomic marks in the surrounding area to include. Earlier work suggested that important factors had noticeable peaks right in the middle of loop anchors, and were higher to 5 Mb outwards on both sides (although this is an average signal)[22]. This is a stretch longer than many of the loops in our data and therefore unusable for calculating an epigenomic signature for our loop anchors. A recent classification approach used a window of 4 Kb around the CTCF peak of an anchor[20]. We chose to calculate features for anchors of two sizes: 2,500 bp and 10,000 bp surrounding the principal anchor site, and compare their results. Additionally, we subdivided each anchor into a left and a right part. For each of the 20,000 GM loops, we took its two anchor positions as single anchors, yielding 40,000 of them. As a negative set, we sampled random anchors according to the real distribution of anchor positions per chromosome in the data, to control for the influence of genomic background. We obtained the overlap of these sets of single anchors with significant peaks of 153 chromatin marks and TFs from ChIP-Seq data in GM12878 using BEDintersect[27] and calculated features such as coverage fraction and average signal value for them. We then performed Random Forest (RF) classification[28] with the anchor type (real or fake) as the label (**see methods**). In short, we trained 50 classifiers per condition (5 repeats of 10 cross-validation folds with 5-fold internal cross-validations to determine optimal hyperparameters) for three conditions: features calculated for a) a 2,500 bp anchor size; b) a 10,000 bp anchor size; c) both. Results shown are median data of 5 repeats per condition, unless otherwise indicated.

We find that real loop anchors are only weakly distinct from fake loop anchors (**Figure 3, Supplementary Figure 1-3**). Median ROC AUC score over folds is around 0.6 for both cases where we include only features calculated for a certain anchor size, while it is appreciably less when we supply both (**Figure 3B**). This might be because the doubling of the number of features, while many features encode more or less the same information (i.e. they are collinear), makes it more difficult to assess what features are important for classification[29]. Additionally, we see that a larger anchor size performs better, presumably because there is simply more information to classify on (i.e. the ChIP-Seq feature data is less sparse) (**Figure 3A and B**). If we perform the same classification with uniformly sampled random anchors, the results are very similar (**Supplementary Figure 1-3**).

Histone marks H3K36me3, H3K4me1, and H3K4me2 are important epigenomic marks for single anchor classification

Though classification accuracy is low, looking into what features are consistently used by the classifiers can still tell us something about marks that might be important for distinguishing between anchors and non-anchors. To look into this, we took the top 15 feature importances over all classifiers for each condition. These feature or variable importances are the summed improvements in the split criterion for every tree in the random forest[30,31]. They therefore show importance of variables for the correct classification of samples[30,31]. In our Random Forest implementation these variable importances are normalised such that they sum to 1 per classifier. Our approach results in 45 (3 sets of 15) top feature importances for the classifiers: 15 for every condition (2,500 bp anchor size; 10,000 bp anchor size; both combined). This data was interrogated in different ways. We first asked what feature types were most important by counting the occurrence of each in the top 45 features. No single feature type is most important or occurs more often than the other: they all have relatively low feature importance values and similar counts (**Supplementary Figure 4**). Similarly, for combinations of the side of the anchor (left or right part of each single anchor) with the feature types, there is not a specific anchor side which is more important (**Supplementary Figure 5**). Looking at a combination of anchor side and chromatin mark name, H3K36me3 is important over the whole anchor, as are H3K4me1 and H3K4me2 (all are important in both the right and left part of the anchor; **Supplementary Figure 6**). Indeed, in total, H3K36me3, H3K4me1, and H3K4me2 appear in 25, 10, and 9 of the top 45 features, respectively (**Figure 3C**). H3K36me3 is a histone modification that was initially known to mark active exons. Recently, however, it has also been implicated in regulating facultative and constitutive heterochromatin and pre-mRNA splicing[32–34]. H3K4me1 is known to be enriched at enhancers, and together with H3K36me3 depletion, it can distinguish between enhancers and proximal promoters[35,36]. H3K4me2 reliably marks transcription factor binding regions in at least three different cell types, including GM12878, and is a marker of active promoters[37,38]. H3K4me1 and H3K36me3 were also found to be enriched between anchors in loops in murine ES cells[23]. An earlier paper assigned loop anchors into groups based on the amount of other anchors they loop with, and found that out of 10 marks (9 histone marks and CTCF), H3K4me1, H3K4me2, H3K4me3 and H3K36me3 had higher normalised ChIP-Seq coverage in anchors that form (many) loops[22]. This agrees with our data (although we excluded H3K36me3 because there was no replicate data). However, they also found H3K27ac, H3K9ac, H4K20me1, and CTCF to be enriched, while H3K9me3 was depleted in anchors with a high or median contact amount[22]. We do not find these in our top features. This might be because we used IDR-thresholded significant peaks, rather than continuous ChIP-Seq signal over control signal (see discussion). Interestingly, H3K36me3 was also given the highest feature weight when predicting HiC matrix contact frequencies from chromatin marks in drosophila data using a dense neural network with a 1D convolutional layer, further corroborating its importance for determining DNA contacts[39] (**Supplementary Data Figure 4**).

The importance of these features points to some role of recognising (active) enhancers and promoters in distinguishing real single loop anchor locations from fake single loop anchor locations. While the prominence of these features across repeats and cross-validations shows them to be important, we can deduce from the low absolute values that the classification still leans heavily on combinations of many ChIP-Seq features (**Figure 3C**). Hence, no simple combination of features can accurately distinguish between real and fake single loop anchors. Surprisingly, neither CTCF nor cohesin is found in the top features separating real from fake loop anchors. This is surprising, since most chromatin loops are held together by dynamic Loop Maintenance Complexes (LMC) encompassing these two factors[17]. Only 18% of loops are (partially) resistant to ablation of CTCF[17]. One would therefore expect CTCF to be a major determinant of real loops. One point of note is that other factors can be more upregulated than CTCF at loop anchors (H3K4me1 and H3K4me1 in the referenced publication), which might cause the classifier to not use it as a top feature[22]. We looked into this further by tallying the amount of CTCF and cohesin (RAD21 subunit) peaks in real loop anchors and randomly sampled loop anchors. There are more CTCF and cohesin peaks in real than in random single anchors (**Table 1**). We also see that most CTCF peaks co-occur with cohesin peaks. However, only 8,505 out of 40,000 real anchors have a significant CTCF peak (out of 43,865 CTCF peaks total in the ChIP-Seq data). It is therefore most likely that this small difference between classes is not enough for the classifier to include CTCF and cohesin in the top-scoring features. Given this, a more relaxed threshold for peak calling (or the use of normalised signal rather than pre-called peaks) is an important avenue for improvement (**see discussion**).

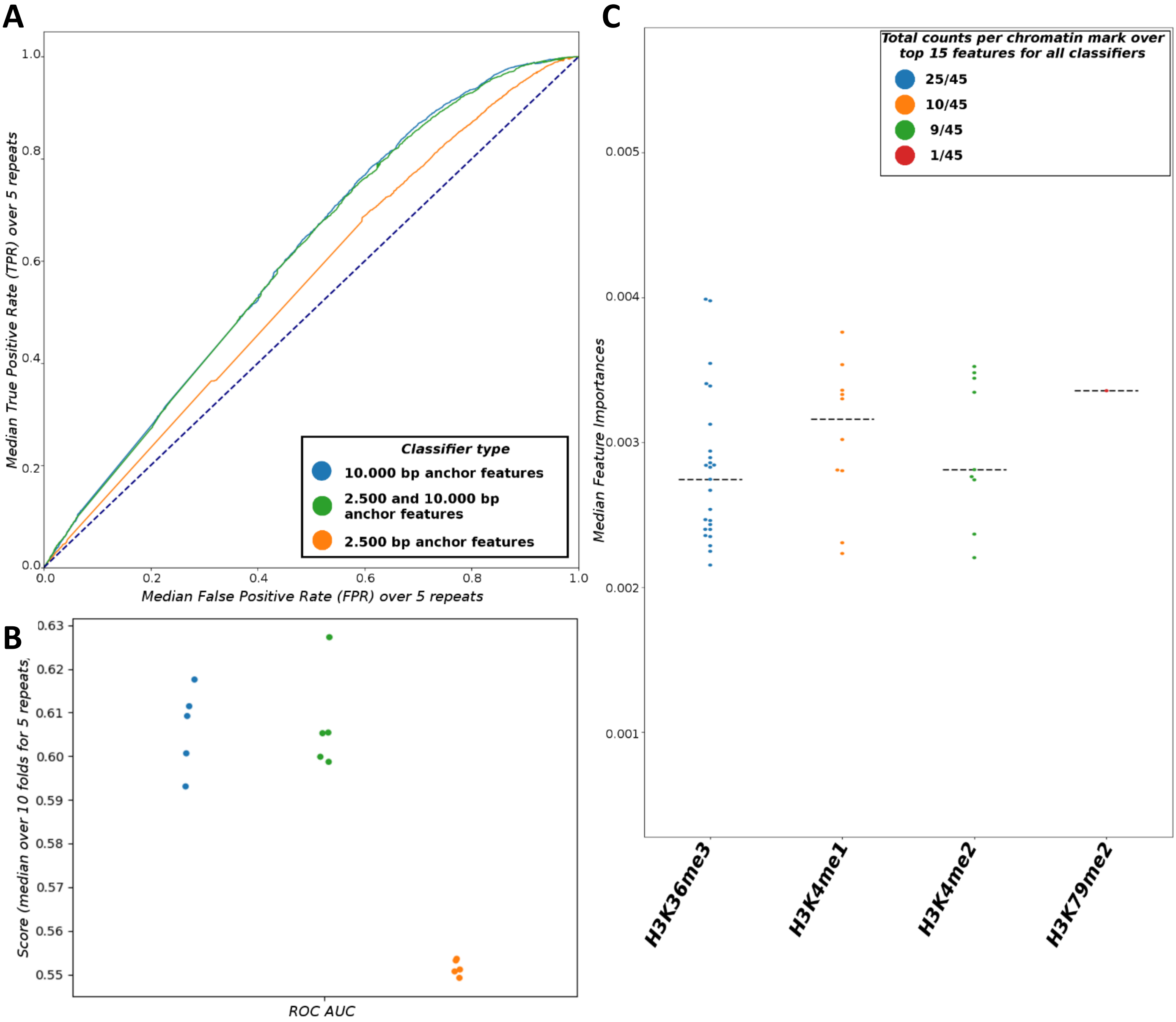


Figure 3. Classification of real and randomly sampled single anchors. *Randomly sampled anchors were sampled according to distributions of real anchors over chromosomes (mimick). A: Median receiver-operating characteristic (ROC) curve for three different feature sets. B: Median area under the ROC curve (ROC AUC) three different feature sets. Calculated over 5 repeats of 10-fold cross-validated RF classifiers per set. Classifiers trained on features calculated for an anchor size of 2.500 bp (orange), 10.000 bp (blue) or with features calculated for both combined (green). C: Count and median feature importance of specific chromatin marks occurring in the top 15 feature importances over classifiers. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest. Dotted lines indicate the median feature importance of all features calculated from a specific chromatin mark.*

Table 1. Occurrences of CTCF and cohesin peaks in IDR-thresholded single-anchor data. *Real anchors have more CTCF and cohesin peaks than fake anchors sampled according to real single anchor distribution along chromosomes (mimick). Mimick anchors, in turn, have more than uniformly sampled random single anchors (uniform).*

Data set	Anchors with CTCF peaks	Anchors with cohesin peaks	Anchors with both peaks	Total anchors
Real GM loop anchors (2,500 bp)	2776 (6.94 %)	2718 (6.80 %)	2171 (5.43 %)	40,000
Real GM loop anchors (10,000 bp)	8505 (21.26 %)	8450 (21.13 %)	7045 (17.61 %)	40,000
Mimick loop anchors (2,500 bp)	2146 (5.37 %)	2030 (5.08 %)	1663 (4.16 %)	40,000
Mimick loop anchors (10,000 bp)	7103 (17.76 %)	6828 (17.07%)	5785 (14.46 %)	40,000
Uniform loop anchors (2,500 bp)	1632 (4.08 %)	1574 (3.94 %)	1265 (3.16 %)	40,000
Uniform loop anchors (10,000 bp)	5473 (13.68 %)	5329 (13.32 %)	4482 (11.21 %)	40,000

Anchor pairs that form loops have no within-anchor epigenetic profile that sets them apart from anchor pairs that do not

Given the moderate success of classifying single anchors, we wondered whether certain epigenomic marks or TFs influence which pairs of anchors form loops. To investigate this question, we used 20,000 real loops from GM12878, and generated a dataset of mismatched anchor pairs by selecting, for every anchor, the 5 anchor positions closest to its real cognate anchor, and designating these as loops. From the resultant set of 200,000 fake loops, we sampled 20,000 to act as the negative set. Feature calculation was performed for both anchors, but not for the area in-between the two anchors. Thus, only influences of within-anchor epigenomic state on looping propensity were studied. Due to time constraints, we only trained classifiers for the combined case (which had ChIP-Seq-based features for both 2,500 and 10,000 bp intervals around the middle anchor site). There is a very weak signal, almost equivalent to guessing (**Figure 4A and B**). The feature importances echo this as they have very low absolute values (**Figure 4C**). It is thus impossible to decide whether two loop anchors will or will not loop based solely on within-anchor features. This is not unexpected. Indeed, if we look at the errors the classifier makes, we see that it predicts that pairs of anchors are fake the majority of times, no matter if they are a real pair in truth (**Supplementary Figure 7**). We know that loop formation is a dynamic and complex process that depends on many factors[17,40]. We therefore do not expect that there is an easy encoding within the anchors themselves that predestines anchors to interact. This is borne out by the data.

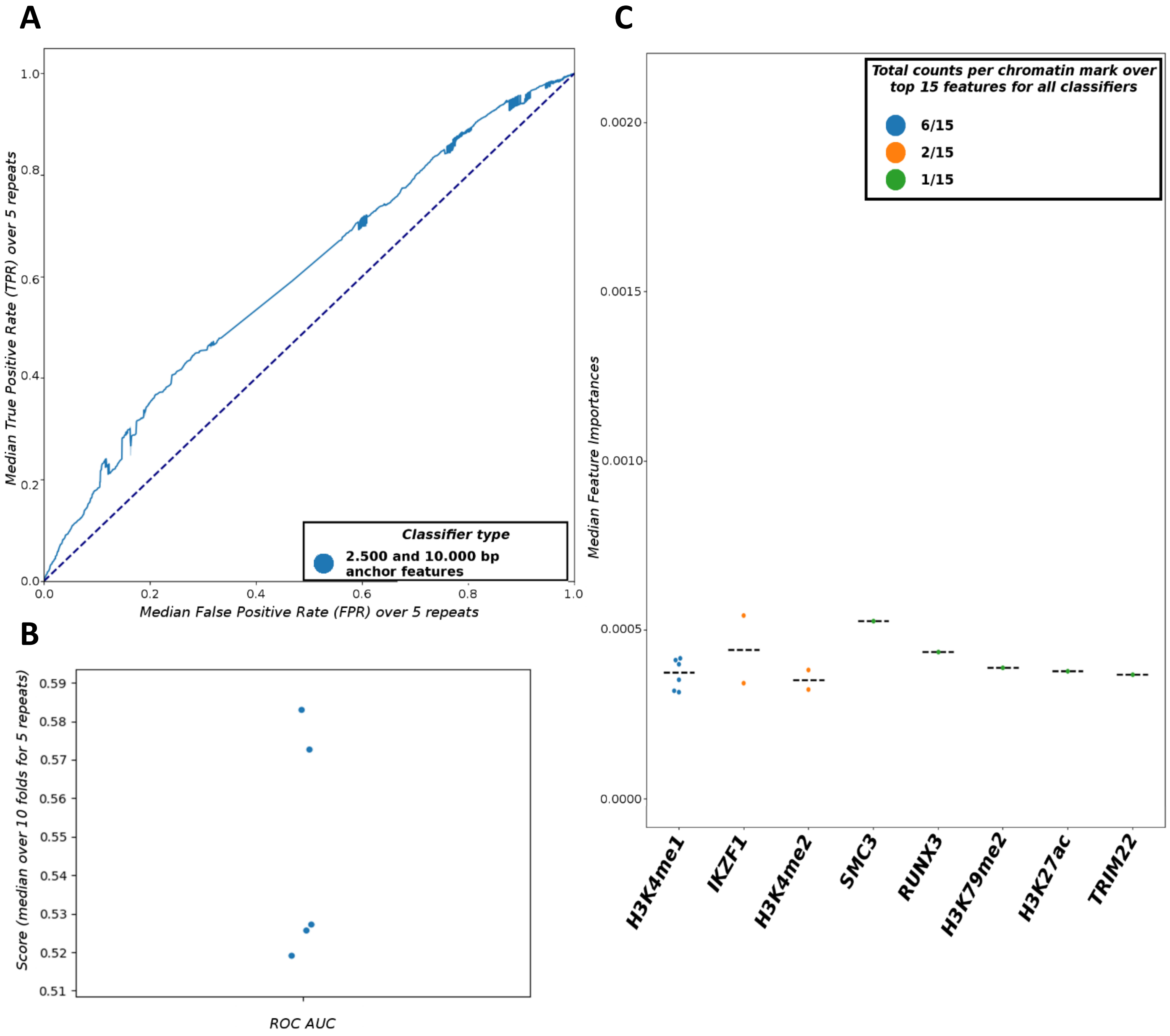


***Figure 4. Classification of paired loop anchors from real loops and mismatched loop anchors from real loops**. A: Median receiver-operating characteristic (ROC) curve for classifier. B: Median area under the ROC curve (ROC AUC) for classifier. Calculated over 5 repeats of 10-fold cross-validated RF classifiers. Classifier trained on features calculated for both anchor size of 2.500 bp and 10.000 bp only, due to time constraints. C: Count and median feature importance of specific chromatin marks occurring in the top 15 feature importances for the classifier. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest. Dotted lines indicate the median feature importance of all features calculated from a specific chromatin mark.*

Classification of GM and non-GM loops using only within anchor-features is somewhat successful and again identifies H3K36me3, H3K4me1 and H3K4me2 as important features.

We now know that single anchors (can) have some epigenomic marks that set them apart from non-loop anchors. We also know that this does not hold for anchor pairs without in-between data. However, our interest lies chiefly in determining the epigenomic features that cause or signal looping. To find whether and, if so, which factors do this, we use two sets of 20,000 loops. The first are the 20,000 GM12878 loops introduced above (GM loops), the second are real loops that are (almost) not observed in GM12878 (non-GM loops, they are observed chiefly in heart tissue; see methods). In short, the two sets are defined by relative difference in coverage between loop callings in GM12878 and heart tissue HiC experiments (**Supplementary Figure 8**). We calculate features for both sets of loops based on GM12878 ChIP-Seq data. The result is one set of observed loops where epigenomic features match up with the loops, and one set of observed loops where they do not. If epigenomic data is indeed informative for loop formation, we should be able to confidently classify loops as either GM or non-GM based on these features. We can then find out which epigenomic marks are most important for classification. First, we checked whether such a classification might be possible without in-between features. Using the same approach as before, we trained RF classifiers to classify loops as either GM or non-GM in 3 conditions, based solely on within-anchor features. Classification is possible but not very accurate (**Supplementary Figure 9A and B)**. Median ROC AUC fluctuates around 0.7 for the two conditions with features calculated for only one anchor size (**Supplementary Figure 9B**). This result clearly mirrors our classification of single anchors (**Figure 3**), especially if we look at feature importances, where H3K36me3, H3K4me1, and H3K4me2 are again found to be important (**Supplementary Figure 9C**). This underlines that these factors somehow distinguish loop anchors from non-loop anchors.

There is some disparity in loop size distributions between the GM and non-GM loops (**Supplementary Figure 10A**). In fact, there is a difference on a per-chromosome basis (**Supplementary Figure 11**). We corrected for this overall disparity via selective downsampling (**Supplementary Figure 10B**; see methods) and reran this classification. Results were unchanged, showing that this is not an artefact of disparate loop distributions (**Supplementary Figure 12-14**). Interestingly, if we look at whether features at the inside of the loop (i.e. the rightmost part of the left anchor and the leftmost part of the right anchor) or at the outside of the loops are most important, we see that H3K36me3 is important in both locations, whilst H3K4me1 and H3K4me2 seem to be found at the inside of the loops (**Supplementary Figure 15**). This hints at the importance of the epigenomic state of the DNA between the two loop anchors for correct loop prediction, which we explored further.

In-between epigenomic features are most important for predicting whether two anchors loop or not, ATF-2 and HDGF chief among them

To identify how and what in-between epigenomic modifications influence chromatin looping, we next performed classifications where we included features calculated for this in-between area in the classification. The resultant classification has an excellent ROC AUC of ~0.91 (**Figure 5A and B**). Apparently, in-between information gives the classifier a much more detailed picture of loop formation conditions than anchor sites alone. An anchor size of 2,500 bp yields the best results. Given these results, we wished to see what factors the classifiers were using. The main factors used across classifiers were ATF-2, HDGF, and CTCF (**Figure 5C**).

ATF-2 is a transcription factor (TF) in the nucleus that can form heterodimers with many other TFs, thereby regulating a wide range of targets[41]. It is a moonlighting protein with many different functions[42]. Indeed, besides its role as a TF, it can also shuttle out of the nucleus and influence cell death by engaging in processes at the mitochondrial membrane[43]. Interestingly, on its own, ATF-2 acetylates histone H2B and H4 *in vitro*[44]. Its yeast homologues are involved in histone deacetylation and correct heterochromatin formation[41]. To our knowledge, no literature explicitly linking ATF-2 to chromatin loops exists, so its prominence here is of note.

HDGF stands for Hepatoma-Derived Growth Factor. It is a nuclear-targeted mitogen that is upregulated in many cancers[45]. HDGF and its related factors have a PWWP domain, a 70-amino acid domain that is weakly conserved and found in more than 60 eukaryotic proteins. Many proteins containing this domain also contain chromatin remodelling domains[46]. In a large-scale study of protein-protein interactors of HDGF, it was found to interact with 19 proteins involved in chromatin remodelling, although it was also found to function in ribosome biogenesis, translation, RNA processing and splicing, and DNA repair[47]. The authors propose that some of its transcriptional regulating activity might hinge on its cooperation with chromatin remodellers. These tangential links notwithstanding, to the best of our knowledge, no specific link between HDGF and loop formation has been mentioned in the literature. It would thus be interesting to investigate this connection.

Finally, CTCF is in 7/45 top features for these classifiers. It is in third place, whereas CTCF and cohesin are known as integral to loop formation, again probably because our peak calls are too strict[19,48]. Notice, though, that the actual scores for feature importance are again very low; while it is interesting that all three classifiers use the same features, accurately deciding whether there is looping or not requires data on many features (**Figure 5C**).

Given the high performance, we wondered whether this was truly solely or mostly accomplished by features derived from the in-between area. This is indeed the case, with all top features in the three classifiers being in-between features (**Supplementary Figure 16; data not shown)**. This is logical for both technical and biological reasons. On the technical side, the in-between area spans many more base pairs, which reduces sparsity in the features, and gives the classifier more possible splits for arriving at an accurate classification than anchor-based features can provide. On the biological side, interactions are more frequent within TADs than from TAD to TAD[14,40], and loops form in the presence of a menagerie of epigenomic conditions that govern whether an area is transcriptionally active or inactive, amenable to binding events or not, etcetera[12,14,40]. Given this information, it is clear that information on conditions between the anchors, that can signal whether a genomic area is amenable to looping or not, might be very informative for predicting loop formation. However, two recent approaches, Lollipop and CTCF-MP, focused chiefly on features immediately surrounding CTCF-site anchors[20,21]. In contrast, we show here that the epigenomic state of the genome within the loop, between the two loop anchors, is most important for determining loop provenance.

This result prompts further investigation into the relative merits of anchor-based and in-between loop prediction. To rule out loop size distribution disparities as a cause for the classification, we again corrected for this and reran the classification. The results were unchanged (**Supplementary Figure 17-18**). Thus, we can conclude that epigenomic information alone holds enough information to reliably differentiate between loops found in one cell type and loops found in another cell type, that ATF-2, HDGF and CTCF have some importance for this classification, and that it is the information about epigenomic modifications between loop anchors that is most important to correctly classifying these loops.

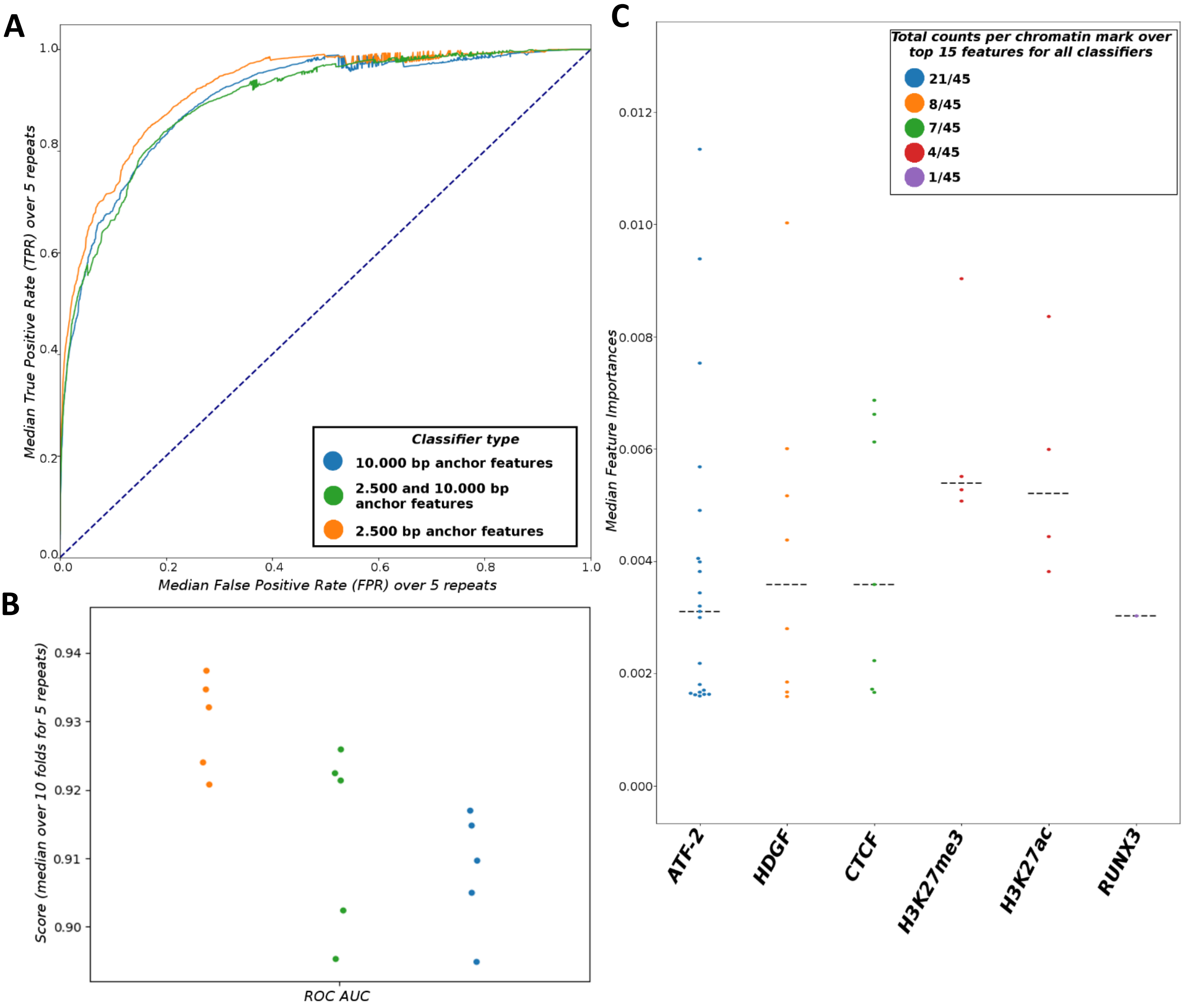


Figure 5. ***Classification of GM and non-GM loops obtained from the de Laat lab including feature data in-between loop anchors.*** *A: Median receiver-operating characteristic (ROC) curve for three different feature sets. B: Median area under the ROC curve (ROC AUC) for three different feature sets. Calculated over 5 repeats of 10-fold cross-validated RF classifiers per set. Classifiers trained on features calculated for an anchor size of 2.500 bp (orange), 10.000 bp (blue) or with features calculated for both combined (green). C: Count and median feature importance of specific chromatin marks occurring in the top 15 feature importances over classifiers. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest. Dotted lines indicate the median feature importance of all features calculated from a specific chromatin mark.*

Discussion

In this project, we sought to distinguish which epigenomic factors govern chromatin loop formation by using a machine learning approach. Interest in the 3D conformation of the genome and its effects on regulation has soared in the last fifteen years, as sequencing-based technologies have superseded more low-throughput methods[3–5,15]. With the advent of native HiC technologies and the development of novel loop identification methods, data on 3D organisation are becoming ever more precise and numerous[6,7,49]. This, combined with the recent innovation of multi-contact methods (MC4C)[10], which allows one to study higher-level organisation, means there is an urgent need now to leverage this data into better understanding of fundamental principles governing 3D organisation, such as chromatin loop formation. Here, we focussed on the fundamental question of which epigenomic features distinguish a) chromatin loop anchors from genomic sites that are not chromatin loop anchors, b) pairs of looping anchors from pairs of non-looping anchors, and c) real loops in a cell type from real loops in another tissue (**Figure 2**). To this end, we obtained high-confidence loops in GM12878 cells and heart tissue, calculated ChIP-Seq-based features, trained Random Forest classifiers, and investigated what features were found to be important for loop classification.

Predicting which genomic locations are loop anchors

We first asked what epigenomic features set loop anchors apart from non-loop anchors. Our results suggest that, in GM loops, the epigenomic neighbourhood of loop anchors is not drastically different from non-loop anchors. Indeed, the best ROC AUC is about 0.63 (**Figure 3B**), which is better than chance, but only slightly so. Nevertheless, for the signal that is there, H3K36me3, H3K4me1, and H3K4me2 are consistently found to be important across classifiers (**Figure 3C**). These factors are known to mark exons, promoters, enhancers, and function in regulation of heterochromatin[32–38]. An earlier paper assigned loop anchors into groups based on the amount of other anchors they loop with, and found differential enrichment of factors between the groups[22]. This shows an important improvement: we performed this classification as a naïve method of identifying any marks that set apart loop anchors from non-loop anchors. However, this paper clearly shows that there are important differences in anchors based on the amount of loops they can form. Thus, a classification of groups of single anchors, split by the amount of interactions they engage in, might yield better performances and specific factors important in each case. This paper used windows of 300 Kb around interaction hubs, much larger than ours, yet we do find some of the same features. This strengthens the case for their involvement in designating loop anchors. Indeed, when we classify on GM and non-GM loops without in-between features, we again find that H3K36me3, H3K4me1 and H3K4me2 signal what an actual loop anchor is in a given cell type (**Supplementary Figure 9**). Further showcasing their involvement is the importance of H3K36me3 and H3K4me1 for predicting HiC matrices (**Supplementary Data Figure 4**).

Predicting which anchors will loop using within-anchor epigenomic features

We then attempted to predict which pairs of anchors will loop or not. We took a set of anchors that form loops in GM12878, and a set of mismatched loop anchors as fake loops. Classification results were worse than in the single anchor case (**Figure 4A and B**). This was mirrored in the feature importances, which were very low (**Figure 4C**). Evidently, there is no within-anchor epigenomic profile that predicts whether two anchors truly interact or not. It is known that topologically associated domains (TADs) govern interactions within them, and that chromatin loops happen in the wider context of A/B compartments[12,14,40]. Therefore, it is unsurprising that attempting to classify which loop anchors (which are all real loop anchors known to form loops with at least one other anchor) will pair without any information on surrounding genomic state is impossible. The in-between information is crucial, as we then showed (though see **Limitations of feature selection and in-between feature importance**).

Predicting which loops form in a specific cell type

Our results suggest that predicting whether a loop is present in a certain cell type can be done to a very high degree of accuracy using solely epigenomic information (**Figure 5**). ROC AUC scores fluctuate around the 0.9 mark, with the classifier that used smaller windows around the anchor sites (2,500 bp) performing best (**Figure 5A and B**). Interestingly, we found that features pertaining to the transcription factors ATF-2 and HDGF were important, which is unexpected (**Figure 5C**). Though both proteins have tangential relations to chromatin and DNA-binding activity[41–47], we did not find any literature readily linking them to loops. This is thus a fruitful avenue for further study.

We found that the classifier assigned by far most importance to features calculated for the genomic area in-between the two looping anchors (**Supplementary Figure 16**). This is a novel and important result. Current efforts focus mostly on (pairs of) anchor locations[20,21], whereas our method reveals that genomic state in-between anchors is by far the most important determinant of chromatin looping. Strangely, this even held for CTCF, whereas we know its function is to hold together looping anchors and therefore expect it to be important at anchor sites[17,19]. One explanation might be that high signals of in-between CTCF show that this is an area that harbours or can harbour multiple loops, i.e. a genomic area that is amenable to looping, making it more likely that two specific anchors there might loop. In the single anchor case, we do see that random genomic stretches sampled in areas where real loop anchors are situated have more CTCF peaks than genome-wide uniformly sampled sites (**Table 1**). Additionally, the Lollipop CTCF loop classifier had in-between CTCF signal as its 6th most important feature in classifications trained on GM12878 data[20]. Taken together, this shows high CTCF signal can indicate a loop-dense area. We find only little enrichment of CTCF and cohesion at anchor sites (**Table 1**). This might be due to differential treatment of ChIP-Seq data (Lollipop used SICER[50], whereas we used the IDR-thresholded peaks[51]). Nevertheless, our approach allows us to accurately distinguish between loops that occur in GM12878 from those that do not occur (or occur much less frequently) in GM12878 but do occur in heart tissue, based solely on epigenomic state. Moreover, we identify two interesting candidates for further research (ATF-2 and HDGF), although it should be noted that the classification is still highly multifactorial (no single feature dominates in feature importance) (**Figure 5C**).

Limitations of loops and their selection for inclusion

We selected only the strongest loops occurring between 10 Kb bins in the genome for further processing. This might bias our results. However, the choice was necessary, given that we need some interval around the loop anchor for which to calculate features. Indeed, we calculate these also for a 10 Kb window surrounding the anchors, and if we would allow multiple anchors in one bin, surrounding features would become extremely similar, hindering classification. We therefore feel this limitation is justified. Besides our selection of only the strongest loops, the provenance of the loops also introduces uncertainty. The non-GM loops are called in heart tissue, whose signal differs markedly from the single cell type GM12878 data. This is expressed in the size of loops called (**Supplementary Figure 10**). Heart tissue has more large loops. We have corrected for this on a global scale by selective downsampling, where we simply bin the loops into bins of equal size, and remove loops until the amount of non-GM and GM loops is equal in each bin. This procedure could be improved by performing it on a per-chromosome basis, since the loop size disparity differs per chromosome, although this might result in large swathes of training data being discarded. Chromosome 10, for instance, has many more loops in heart tissue than in GM12878 cells, most of which would need to be removed (**Supplementary Figure 11**). A final point of note is that we are more certain that our GM-specific loops are truly GM-only than we are sure that heart tissue loops truly don't occur in GM (**Supplementary Figure 8**). This introduces some noise into the classification, although judging by the results the effect is minor, as we can still classify with high accuracy (**Figure 5**). Ideally, however, we would like to be as sure of our non-GM loops as of the GM loops to make the best possible distinction and thereby identify the most salient epigenomic marks.

Limitations of feature selection and in-between feature importance

In our current approach, we solely calculated features in a relatively small window around the middle sites of anchors that loop, and for the entire stretch in-between these two anchors for some classifications. This presents some problems. Firstly, we conclude now that in-between features are very important. However, we can only conclude that larger stretches of ChIP-Seq signals are important. There are two interrelated problems: we do not know the importance of location, and we do not know the effect of feature sparsity. Regarding location, we only included a large amount of epigenomic information in-between anchors in our current approach. Without also testing the effect of inclusion of features for longer stretches of genome outside loop anchors, or enlarging the anchor size, it is not clear that in-between features are important, but only that features aggregated over a longer stretch of DNA fare better in classification. This is related to the second problem of sparsity: it could simply be that the anchor sites have data that is too sparse for accurate classification. For instance, many calculated features for anchor sites are simply 0 in the current set-up, because most anchors are not covered by peaks in many ChIP-Seq marks. The exception are the long stretches of in-between features, where more signals will be found, allowing for more and better splits in the RF classifier. Therefore, making loop anchors (much) larger might also influence this, as the anchor sizes we use now could simply be too small for proper differentiation on the basis of epigenomic state. To solve the issue of location importance, we need to calculate features for large stretches bordering on the loop anchors (outside the loops) and for much larger regions with the loop anchors at their center, and perform separate classifications with these. To control for sparsity, we could down-sample ChIP-Seq peaks in in-between areas, calculate features, and observe classification performance.

Despite these problems, we do find histone marks known to be important[22]. Note that the Lollipop CTCF interaction classifier also used small anchor sites of only 4,000 bp[20]. Ultimately, it would be best if we need not manually define the areas of importance for classification (see future work). Another issue is that we calculate the median and average q value, and the median and average signal value, and include both of these in all classifications. However, these features are highly correlated. Tree-based classifiers are somewhat resistant to collinearity in features negatively affecting their performance[29]. They are still affected, however, especially where defining feature importance is concerned (if two variables encode more or less the same information, then you could use either to split on)[29]. We could use a PCA or similar technique to reduce collinearity and classify afterwards, or run separate trials where we use only one of the features that encodes the same information and see which performs best. We can then use only in in final classification for optimal results and interpretability.

Future work

One avenue for further research is a deeper look into the importance of ATF-2 and HDGF for loop formation, preferably via an experimental (knockout) screening. We also have not yet queried the directionality of genomic mark importance (is it important for H3K36me3 to be enriched or depleted for a site to be designated as a possible anchor sit?). This would be a simple extension to the current work. Another simple extension would be to look at normalised ChIP-Seq signal rather than the seemingly quite stringent IDR-thresholded peaks. We find CTCF peaks in only 7% of the GM12878 loop anchors, which is markedly less than the ~80% we would expect from literature[15,17]. Using normalised signal could solve this problem, although it could also partially be due to the much larger volume of loop calls we use, which might include relatively more loops without CTCF marks (259,963 loops total). Another avenue would be to cluster loops into categories based on combinations of a few histone marks known to be important, or to cluster loop anchors based on the number of interactions they engage in. Earlier work shows that both affect the histone mark profile[22,23]. Classifying these subclasses of loops might yield a more fine-grained look at epigenomic marks affecting them.

An important improvement to the current method would be automatic selection of genomic regions whose epigenomic marks are of import to correct classification, i.e. a classification where the encoding from raw data into optimal suitable features for classifications is also learned. This is possible by using convolutional neural networks with many layers (deep learning)[52,53]. These techniques have recently precipitated breakthrough successes in classification of biological datasets[54–57], and could do so here as well. We can see the immense potential in this technology and have therefore performed a proof-of-principle of using deep learning for these types of classifications by replicating a recent application of deep learning to predict HiC matrices[39] (**Supplementary Data 1: prediction of HiC matrix using a dense neural network**). Given the importance of the in-between area, classification with deep learning should at least include some high-level convolutions of an entire genomic region, besides many training samples of paired anchors that form loops and fake anchors that do not. While deep neural networks offer challenges in understanding what they have learned, these can be overcome[58–60], allowing us to use their power to better understand basic mechanisms of chromatin looping. This is an important part of future work. Finally, while we wish to find epigenomic factors of importance for chromatin loop formation, we have no knowledge of the causal nature of these factors (besides CTCF and cohesin). Are, for example, ATF-2 and HDGF deposited, which stimulates loop formation, or does loop formation precipitate the binding of ATF-2 and HDGF within the loop? We know that loops form and become undone during the cell cycle[17]. As HiC experiments become cheaper and commonplace, time-series HiC data within specific cell types combined with time-series ChIP-Seq data might begin to uncover what is cause and what is effect by studying the sequence of steps that leads to loops.

Methods

Processing of chromatin loop data

Loop data for 259,963 loops were obtained from the de Laat lab (unpublished data). Loops were called between 10 kb bins of the genome in HiC performed on GM12878 (GM) cells and heart tissue using the novel peakHiC method, which models background contact chances using monotonic regression, and identifies loops based on reciprocal peak callings in virtual 4C assays[61]. To obtain heart- and GM12878-specific loops, we first removed the top and bottom 2.5% of data based on the maximum virtual 4C score to remove outliers. We then sorted on this score from high to low, and filtered out duplicate loops (i.e. loops forming between identical bin pairs) based on their identifiers. We thus obtained the highest-scoring interaction (loop) between any two bins in the genome. We then calculated the difference in normalised coverage quantity between heart and GM12878 cells (delta; covQGM – covQHeart; **Supplementary Figure 8**). The highest-scoring loops are therefore highly GM12878-specific, the lowest-scoring heart tissue-specific. We took the top 20,000 most cell-type specific loops from each group to obtain a total of 40,000 loops for training of Random Forest classifiers.

Generation of data for single anchor and mismatched anchor classification from loop data

To test for baseline performances on single anchors and mismatched anchor pairs (i.e. fake loops) in GM12878, we constructed two separate datasets. The first of these was made for classifying real single loop anchors and random single loop anchors. We took the 40,000 single anchors of the 20,000 GM loops as a set of real loop anchors. We then constructed a negative set by either uniformly sampling 40,000 random locations across the genome (uniform), or by binning the 40,000 real single anchors into 30 bins along each chromosome and uniformly sampling as many random anchors as real anchors within each of these bins (mimick). The former is a more naïve negative set, whereas the latter should more closely resemble the general genomic conditions around a loop anchor.

For mismatched anchor classification, we took the 20,000 real GM loops as a positive set. Then, for each anchor of each loop, we selected the five anchors that were closest to it on the genome and designated those anchors as forming a loop with the original loop anchor. In this way, we obtained a set of 200,000 mismatched loop anchors (i.e. fake loops). We did not filter these loops for ones that do, in fact, occur, since this is likely to be a small fraction and the inclusion of few such cases makes the classification more challenging. We sampled 20,000 of these loops, and classified fake versus real loops based on ChIP-Seq data, in this case without features calculated in the area between the two anchors.

Processing of ChIP-Seq data

ChIP-Seq data was downloaded from the ENCODE portal (https://www.encodeproject.org/[26,62]). We filtered on GRCh38, isogenic replicates, and ChIP-Seq data, and downloaded 200 files. Several measured factors had multiple datasets associated with them; there were only 157 unique factors. We resolved duplicate datasets by choosing the newest entries. This solved the problem for all but two cases. For these, we performed manual selection. For the factor ETV6, we used the data from Richard Myers' lab (ENCSR626VUC**)**, as a comment on the other entry mentions possible sample contamination. For PAX5, the date and protocol were the same, but the antibody used differed (SC-1975 versus SC-1964; targeting either the C-terminal or N-terminal part of the protein). We used ENCSR000BHD, because the antibody used (SC-1974) is still listed on the manufacturer's website, whereas SC-1975 is not. Six factors were only mapped to hg19. We used CrossMap[63] (http://crossmap.sourceforge.net/) to switch these coordinates to GRCh38. For the total of 143 factors now mentioned, we used the optimal IDR-thresholded peaks, where threshold for peak signals was determined by reproducibility based on the irreducible discovery rate[51] (IDR). The other 14 factors were not processed in this way, as they have broad peaks. These factors include histone subunits and chromatin methylation/acetylation (H3K27ac, H3K27me3, etc.). For these marks we used the replicated peak calls. Factors with only one file (not replicated) were not used (EGFR1; SREBF1; SREBF2; H3k9me3). In total, we thus have peaks for 153 unique factors in BED broadPeak format (**Supplementary Table 2**).

Construction of loop features

To construct features for the loops, we first defined the three areas of interest (the left loop anchor, the in-between area, and the right loop anchor). Each anchor is centered on a certain residue, and we calculate intervals from it (for example, 500 bp to either side for an interval of 1,000 bp). We split each anchor into two parts, yielding an area inside the loop and outside the loop, following the Lollipop paper[20]. This yields five areas of interest: leftAnchorLeft, leftAnchorRight, in-between, rightAnchorLeft, and rightAnchorRight. For every chromatin mark, we calculate ten features. We first use BEDTools[27] (https://bedtools.readthedocs.io/en/latest/) through pybedtools 0.8 with the -wao flag to intersect the coordinates for the different areas with the chromatin mark coordinates. This outputs intersects and base pair overlaps. We calculate ten different features, per factor, per area, based on this data. IntersectsPerFactor is the number of peaks an area overlaps with. TotalOverlap is the total base pairs of overlap, and totalOverlapFraction is a fraction of the base pairs in the area that are covered by the chromatin mark. AvgSignalValue is the mean enrichment for the area, medianSignalValue is the median of this enrichment. AvgPValueNegativeLog10 is the mean -log10(p-value), and medianPValueNegativeLog10 the median value. AvgQValueNegativeLog10 is the mean -log10(Benjamini-Hochberg multiple testing-corrected[64] p-value), and medianQValueNegativeLog10 the median value. Lastly, sumPointPeaksInInterval shows how many point peaks of each factor are in the areas of interest. That is to say, an anchor area can overlap with multiple enriched areas (peaks) of a factor of interest. Each such area also has a point peak, where enrichment is highest. This point peak needn't be overlapping with the area, even though part of the enriched area is. This feature thus captures how many focal points of a factor lie in an area, rather than simple partial overlap. Since many of the ChIP-Seq files lacked a p-value, we removed it from final classification, leaving 153 * 8 = 1.224 features per area, for a total of 6.120 features per loop if all areas (leftAnchorLeft, leftAnchorRight, in-between, rightAnchorLeft and rightAnchorRight) are included.

Training of Random Forest classifiers

Training of classifiers was performed on the High Performance Computing facility of the UU. We used the Scikit-learn python library to partition the data into 10 cross-validation folds, training on 9 folds and testing on 1 fold each time[65]. Within each fold, we trained a Random Forest classifier[28], using the RandomSearchCV function (with n_iter = 3 and number of cross-folds (cv) set to 5; **Supplementary Table 1**) to select optimal hyperparameters for that fold, such as the number of estimators (trees), and the number of samples required for a split. This random search is a faster alternative to grid-based searches that can yield better results[66], though we picked it chiefly for its speed. It chooses the combination of hyperparameters which yield the highest score. In our case, the score to be optimised is the ROC AUC. For each classifier, we performed 5 repeats, to make sure that classifier performance does not hinge on seed of the pseudorandom number generator. For every fold (50 total folds for 5 repeats), we calculated the receiver-operator characteristic curve (ROC curve)[67], the area under the ROC curve (ROC AUC)[68], as well as precision and recall[69]. We also generated (normalised) confusion matrices and output the feature importances (which are relative feature importances in the Scikit-learn implementation, so they sum to 1 for each classifier[31]). Lastly, we output the hyperparameters chosen per fold (data not shown in this report).

Interpretation of Random Forest classification results

We first took the median of all metrics over the 10 folds per repeat per condition. Conditions are classification with features calculated for: a) anchors of size 2,500 bp, b) anchors of size 10,000 bp, and c) both. We looked at within-repeat variability (by plotting, and by assessing the standard deviations), but do not show this data here. We calculated a median ROC curve per classification case by taking the median of the values calculated for the ROC curve for each repeat per condition. For feature importances, we calculated the median and mean feature importance of all features across repeats (**Supplementary Figure 20**). We used the more conservative median scores, and then took the 15 features with highest scores per condition. We sliced this data in multiple ways. A full feature is, for example, the coverage fraction of H3K36me3 in the left part of the left anchor. One slice consisted of simply looking at what feature types (coverage fraction, full coverage in base pairs, median or average Q value) were most prevalent in the most important features, disregarding what chromatin mark or location they were for. Another slice looks at chromatin marks only, revealing which marks are most informative for classification. We additionally looked at the combination of both (irrespective of location), and at which location was most important (left anchor, right anchor, or in-between, if present). To extend this, we also made a combined grouping of leftAnchorRight and rightAnchorLeft features as the in-between side (i.e. bordering the DNA region that loops) and the other two parts as the outside (i.e. bordering DNA not involved in the loop), and looked what location was preferred. For all slices, we ordered by counts and then by median feature importance. As such, features that are chosen across conditions in many classifiers are shown first (see, for example, **Figure 3C**).

Computational processing details

Loop and ChIP-Seq processing were accomplished with custom scripts written in Python 3.6.8. Loop feature calculation was performed using custom scripts and the BEDTools suite[27]. Specifically, calculations were done in parallel for each loop separately, and the resulting features were combined into the final feature table for machine learning. Random Forest classification was performed in parallel for different folds on the HPC using custom scripts. All scripts are available upon request.

References

1. Cremer, T. & Cremer, C. Chromosome territories, architecture and gene regulation in mammalian cells. **2**, 292–301 (2001).

2. Ferraiuolo, M. A., Sanyal, A., Naumova, N., Dekker, J. & Dostie, J. From cells to chromatin: Capturing snapshots of genome organization with 5C technology. *Methods* **58**, 255–267 (2012).

3. Dekker, J., Rippe, K., Dekker, M. & Kleckner, N. Capturing Chromosome Conformation. *Science (80-.).* **295**, 1306–1311 (2002).

4. Lieberman-aiden, E. *et al.* Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science (80-.).* **33292**, 289–293 (2009).

5. Grob, S. & Cavalli, G. Technical Review: A Hitchhiker's Guide to Chromosome Conformation Capture. in *Plant Chromatin Dynamics: Methods and Protocols* (2018). doi:10.1007/978-1-4939-7318-7

6. Brant, L. *et al.* Exploiting native forces to capture chromosome conformation in mammalian cell nuclei. *Mol. Syst. Biol.* **12**, 891 (2016).

7. Rowley, M. J. & Corces, V. G. Capturing native interactions: intrinsic methods to study chromatin conformation. *Mol. Syst. Biol.* **12**, 897 (2016).

8. Barutcu, A. R. *et al.* C-ing the Genome: A Compendium of Chromosome Conformation Capture Methods to Study Higher-Order Chromatin Organization. *J. Cell. Physiol.* **231**, 31–35 (2016).

9. Li, G. *et al.* Chromatin interaction analysis with paired-end tag (ChIA-PET) sequencing technology and application. *BMC Genomics* **15**, 1–10 (2014).

10. Allahyar, A. *et al.* Enhancer hubs and loop collisions identified from single-allele topologies. *Nat. Genet.* **50**, 1151–1160 (2018).

11. Simonis, M. *et al.* Nuclear organization of active and inactive chromatin domains uncovered by chromosome conformation capture-on-chip (4C). *Nat. Genet.* **38**, 1348–1354 (2006).

12. Fortin, J. P. & Hansen, K. D. Reconstructing A/B compartments as revealed by Hi-C using long-range correlations in epigenetic data. *Genome Biol.* **16**, 1–23 (2015).

13. Dixon, J. R. *et al.* Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature* **485**, 376–380 (2012).

14. Dekker, J. & Heard, E. Structural and functional diversity of Topologically Associating Domains. *FEBS Lett.* **589**, 2877–2884 (2015).

15. Rao, S. S. P. *et al.* A 3D map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell* **159**, 1665–1680 (2014).

16. Doğan, E. S. & Liu, C. Three-dimensional chromatin packing and positioning of plant genomes. *Nat. Plants* **4**, 521–529 (2018).

17. Hansen, A. S., Cattoglio, C., Darzacq, X. & Tjian, R. Recent evidence that TADs and chromatin loops are dynamic structures. *Nucleus* **9**, 20–32 (2018).

18. Rao, S. S. P. *et al.* Cohesin Loss Eliminates All Loop Domains. *Cell* **171**, 305-320.e24 (2017).

19. Hansen, A. S., Pustova, I., Cattoglio, C., Tjian, R. & Darzacq, X. CTCF and cohesin regulate chromatin loop stability with distinct dynamics. *Elife* **6**, 1–33 (2017).

20. Peng, W. *et al.* Predicting CTCF-mediated chromatin interactions by integrating genomic and epigenomic features. *Nat. Commun.* **9**, (2018).

21. Zhang, R., Wang, Y., Yang, Y., Zhang, Y. & Ma, J. Predicting CTCF-mediated chromatin loops using CTCF-MP. *Bioinformatics* **34**, i133–i141 (2018).

22. Huang, J., Marco, E., Pinello, L. & Yuan, G. C. Predicting chromatin organization using histone marks. *Genome Biol.* **16**, 1–11 (2015).

23. Handoko, L. *et al.* CTCF-mediated functional chromatin interactome in pluripotent cells. *Nat. Genet.* **43**, 630–8 (2011).

24. Hsieh, T.-H. S. *et al.* Resolving the 3D landscape of transcription-linked mammalian chromatin folding. *bioRxiv* 638775 (2019). doi:10.1101/638775

25. Kaiser, V. B. & Semple, C. A. Chromatin loop anchors are associated with genome instability in cancer and recombination hotspots in the germline. *Genome Biol.* **19**, 1–14 (2018).

26. Hitz, B. C. *et al.* The Encyclopedia of DNA elements (ENCODE): data portal update. *Nucleic Acids Res.* **46**, D794–D801 (2017).

27. Quinlan, A. R. & Hall, I. M. BEDTools: A flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841–842 (2010).

28. Breiman, L. Random forests. *Mach. Learn.* **45**, 5–32 (2001).

29. Dormann, C. F. *et al.* Collinearity: A review of methods to deal with it and a simulation study evaluating their performance. *Ecography (Cop.).* **36**, 027–046 (2013).

30. Hastie, T., Tibshirani, R. & Friedman, J. *Elements Of Statistical Learning*. (Springer, 2017). doi:10.1007/b94608

31. Breiman, L. *Classification and regression trees*. (Routledge, 2017).

32. Thuret, J.-Y. *et al.* Histone H3 trimethylation at lysine 36 is associated with constitutive and facultative heterochromatin. *Genome Res.* **21**, 1426–1437 (2011).

33. de Almeida, S. F. & Carmo-Fonseca, M. Design principles of interconnections between chromatin and pre-mRNA splicing. *Trends Biochem. Sci.* **37**, 248–253 (2012).

34. Wagner, E. J. & Carpenter, P. B. Understanding the language of Lys36 methylation at histone H3. *Nat Rev Mol Cell Biol* **13**, 115–126 (2012).

35. Rada-Iglesias, A. Is H3K4me1 at enhancers correlative or causative? *Nat. Genet.* **50**, 4–5 (2018).

36. Heintzman, N. D. *et al.* Histone modifications at human enhancers reflect global cell-type-specific gene expression. *Nature* **459**, 108–112 (2009).

37. Wang, Y., Li, X. & Hu, H. H3K4me2 reliably defines transcription factor binding regions in different cells. *Genomics* **103**, 222–228 (2014).

38. Zhou, V. W., Goren, A. & Bernstein, B. E. Charting histone modifications and the functional organization of mammalian genomes. *Nat. Rev. Genet.* **12**, 7–18 (2011).

39. Farré, P., Heurteau, A., Cuvier, O. & Emberly, E. Dense neural networks for predicting chromatin conformation. 1–12 (2018). doi:10.1186/s12859-018-2286-z

40. Rada-Iglesias, A., Grosveld, F. G. & Papantonis, A. Forces driving the three-dimensional folding of eukaryotic genomes. *Mol. Syst. Biol.* **14**, e8214 (2018).

41. Lau, E. & Ronai, Z. A. ATF2 – at the crossroad of nuclear and cytosolic functions. *J. Cell Sci.* **125**, 2815–2824 (2012).

42. Jeffery, C. J. An introduction to protein moonlighting. *Biochem. Soc. Trans.* **42**, 1679–1683 (2014).

43. Watson, G., Ronai, Z. & Lau, E. ATF2, a paradigm of the multifaceted regulation of transcription factors in biology and disease. *Pharmacol. Res.* **119**, 347–357 (2017).

44. Kawasaki, H. *et al.* ATF-2 has intrinsic histone acetyltransferase activity which is modulated by phosphorylation. *Nature* **405**, 195–200 (2000).

45. Thakar, K. *et al.* Interaction of HRP-2 isoforms with HDGF. Chromatin binding of a specific heteromer. *FEBS J.* **279**, 737–751 (2012).

46. Lukasik, S. M. High resolution structure of the HDGF PWWP domain: A potential DNA binding domain. *Protein Sci.* **15**, 314–323 (2005).

47. Zhao, J. *et al.* Interactome study suggests multiple cellular functions of hepatoma-derived growth factor (HDGF). *J. Proteomics* **75**, 588–602 (2011).

48. Ong, C.-T. & Corces, V. G. CTCF: An Architectural Protein Bridging Genome Topology and Function. *Nat. Rev. Genet.* **15**, 234–246 (2014).

49. Geeven, G., Teunissen, H., de Laat, W. & de Wit, E. peakC: a flexible, non-parametric peak calling package for 4C and Capture-C data. *Nucleic Acids Res.* **46**, e91–e91 (2018).

50. Zang, C. *et al.* A clustering approach for identification of enriched domains from histone modification ChIP-Seq data. *Bioinformatics* **25**, 1952–1958 (2009).

51. Li, Q., Brown, J. B., Huang, H. & Bickel, P. J. Measuring reproducibility of high-throughput experiments. *Ann. Appl. Stat.* **5**, 1752–1779 (2011).

52. Lecun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).

53. Rawat, W. & Wang, Z. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Comput.* **29**, 2352–2449 (2017).

54. Zhou, J. & Troyanskaya, O. G. Predicting effects of noncoding variants with deep learning-based sequence model. *Nat. Methods* **12**, 931–934 (2015).

55. Li, Z., Nguyen, S. P., Xu, D. & Shang, Y. Protein loop modeling using deep generative adversarial network. *Proc. - Int. Conf. Tools with Artif. Intell. ICTAI* **2017**-**Novem**, 1085–1091 (2018).

56. Xiao, Y., Wu, J., Lin, Z. & Zhao, X. A deep learning-based multi-model ensemble method for cancer prediction. *Comput. Methods Programs Biomed.* **153**, 1–9 (2018).

57. Li, W., Wong, W. H. & Jiang, R. DeepTACT: predicting 3D chromatin contacts via bootstrapping deep learning. *Nucleic Acids Res.* **47**, 1–14 (2019).

58. Binder, A., Lapuschkin, S., Muller, K.-R., Montavon, G. & Samek, W. Evaluating the Visualization of What a Deep Neural Network Has Learned. *IEEE Trans. Neural Networks Learn. Syst.* **28**, 2660–2673 (2016).

59. Mahendran, A. & Vedaldi, A. Understanding deep image representations by inverting them. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* **07**-**12**-**June**, 5188–5196 (2015).

60. Erhan, D., Bengio, Y., Courville, A. & Vincent, P. Visualizing higher-layer features of a deep network. *Univ. Montréal, Tech. Rep* 1–13 (2009). doi:10.2464/jilm.23.425

61. Bianchi, V. *et al.* Detailed Regulatory Interaction Map of the Human Heart Facilitates Gene Discovery for Cardiovascular Disease. *BioRX* (2019).

62. Dunham, I. *et al.* An integrated encyclopedia of DNA elements in the human genome. *Nature* **489**, 57–74 (2012).

63. Zhao, H. *et al.* CrossMap: A versatile tool for coordinate conversion between genome assemblies. *Bioinformatics* **30**, 1006–1007 (2014).

64. Benjamini, Y., Drai, D., Elmer, G., Kafkafi, N. & Golani, I. Controlling the false discovery rate in behavior genetics research. *Behav. Brain Res.* **125**, 279–284 (2001).

65. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

66. Bergstra, J. & Bengio, Y. Random Search for Hyper-Parameter Optimization. **13**, 1–25 (2012).

67. Hajian-Tilaki, K. Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation. *Casp. J. Intern. Med.* **4**, 627–635 (2013).

68. Bradley, A. P. THE USE OF THE AREA UNDER THE ROC CURVE IN THE EVALUATION OF MACHINE LEARNING ALGORITHMS. *Pattern Recognit.* **30**, 1145–1159 (1997).

69. Buckland, M. & Gey, F. The relationship between Recall and Precision. *J. Am. Soc. Inf. Sci.* **45**, 12–19 (1994).

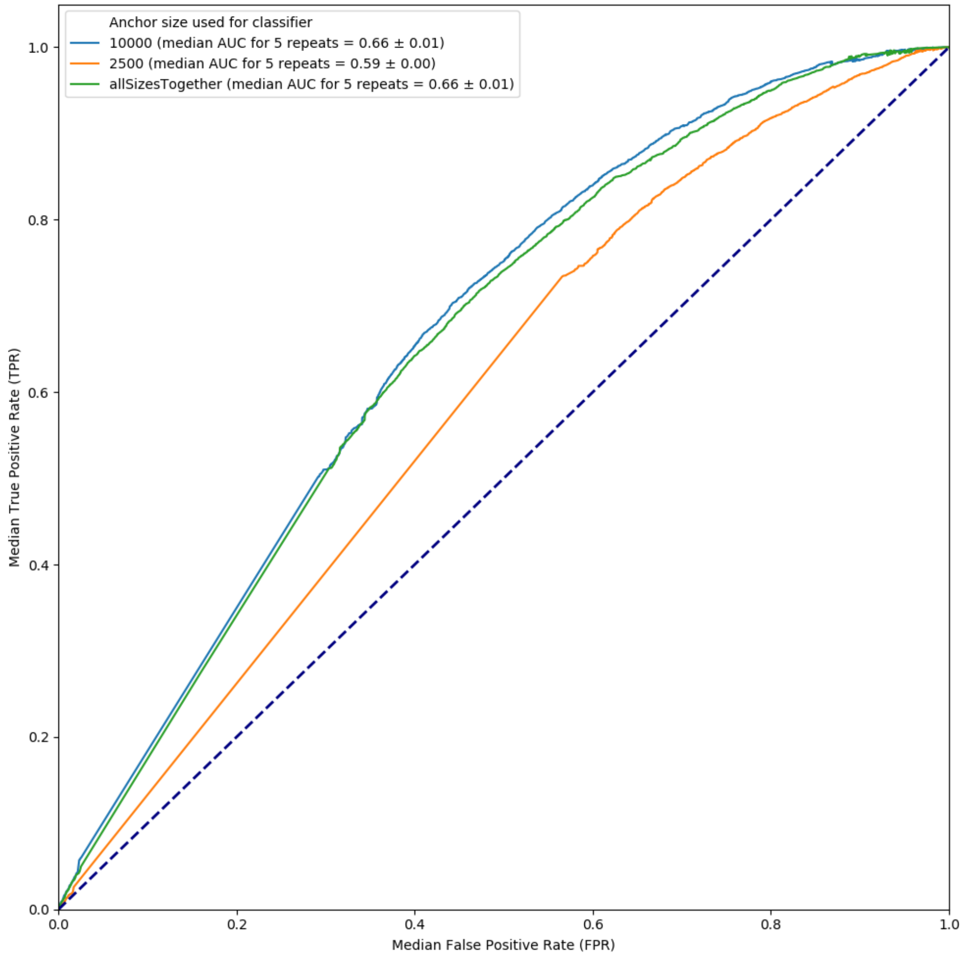
70. Cavalli, G. *et al.* Cooperativity, Specificity, and Evolutionary Stability of Polycomb Targeting in Drosophila. *Cell Rep.* **9**, 219–233 (2014).

71. Snyder, M. *et al.* Unlocking the secrets of the genome. *Nature* **459**, 927–930 (2009).

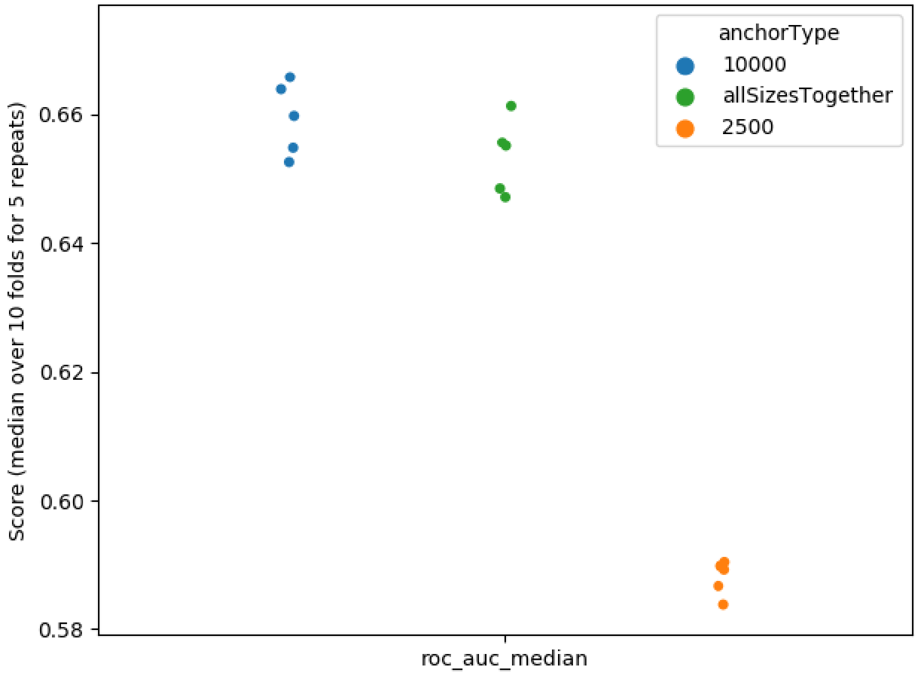
72. Kumar, R., Sobhy, H., Stenberg, P. & Lizana, L. Genome contact map explorer: A platform for the comparison, interactive visualization and analysis of genome contact maps. *Nucleic Acids Res.* **45**, 1–8 (2017).

73. Chollet, F. & others. Keras. (2015).

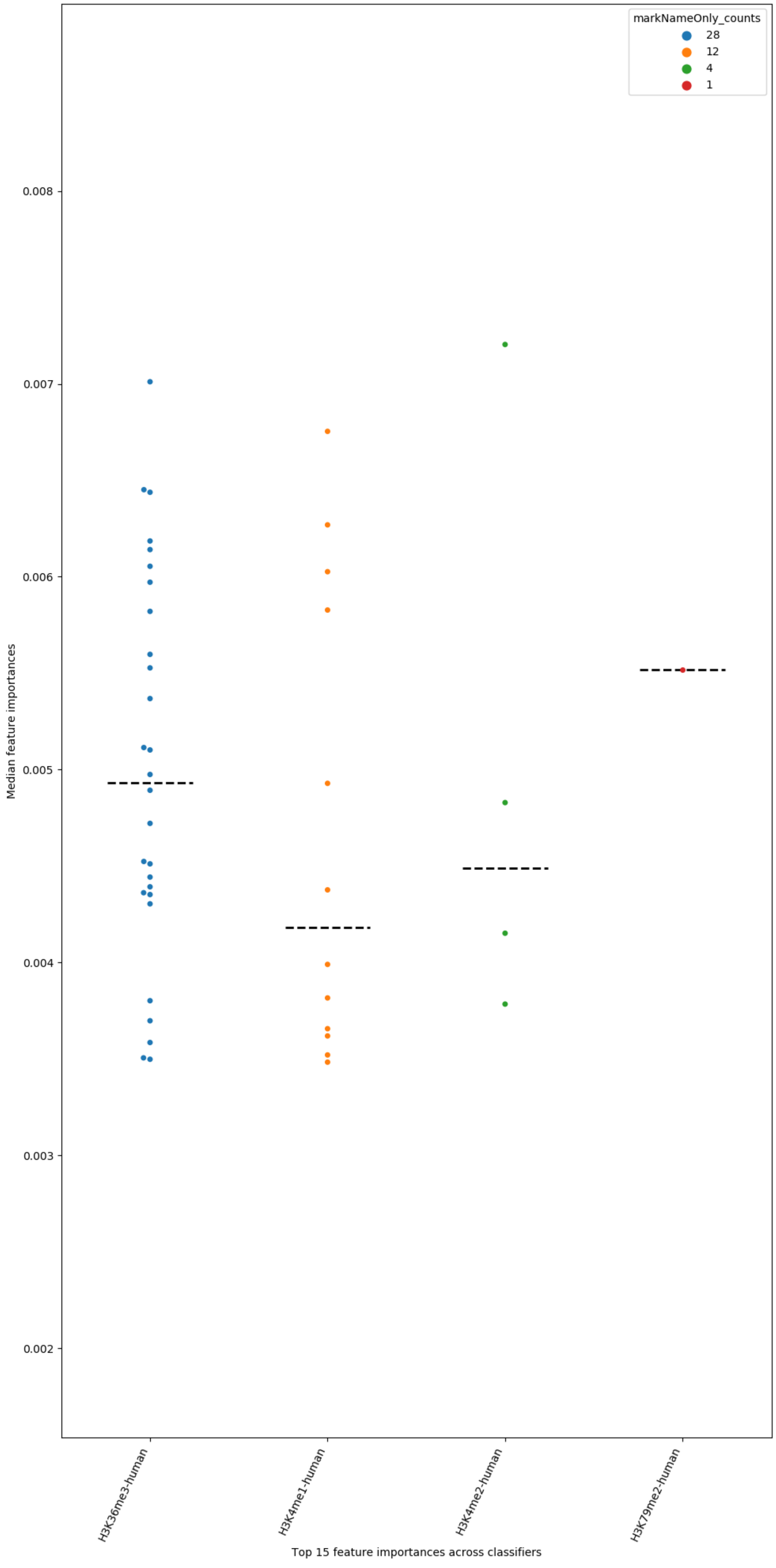
Supplementary Figures



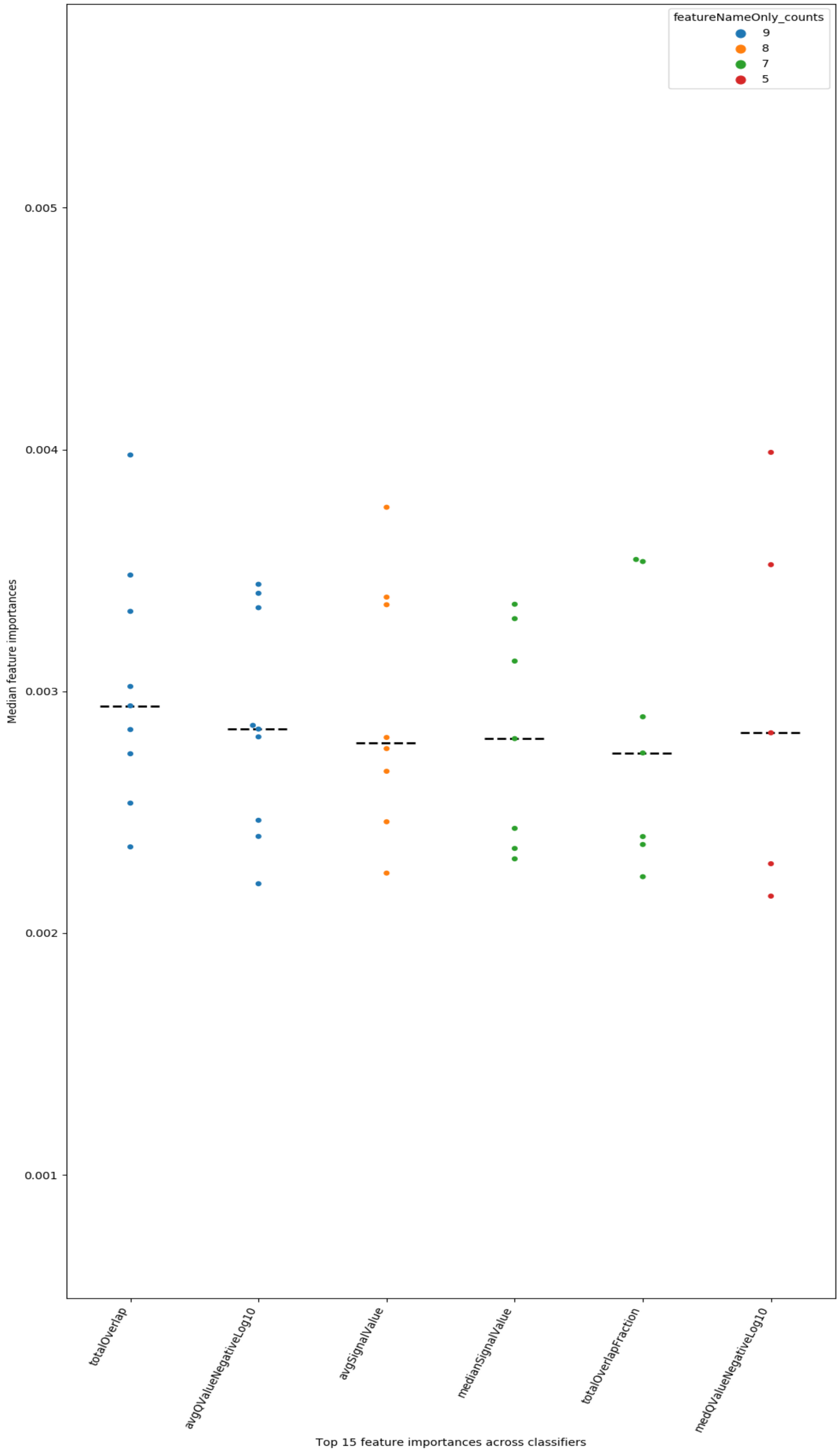
Supplementary Figure 1. Median ROC curve for classification of real and randomly sampled single anchors over 5 repeats of 10-fold cross-validated RF classifiers*. Randomly sampled anchors were sampled uniformly across the genome (uniform).*



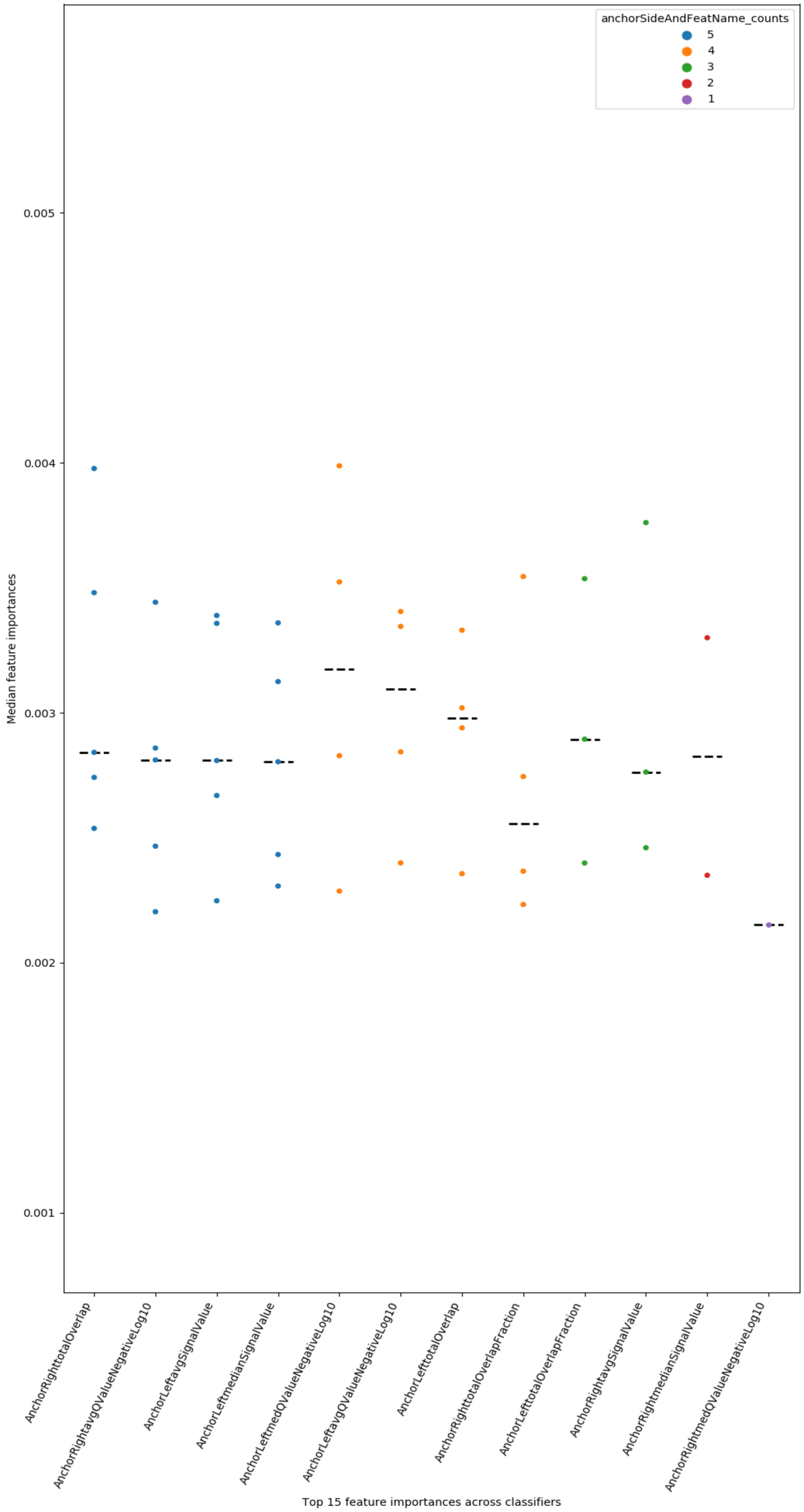
Supplementary Figure 2. Median ROC AUC for classification of real and randomly sampled (uniform) single anchors over 5 repeats of 10-fold cross-validated RF classifiers*.*


markNameOnly_counts
28
12
4
1
0.008
0.007
0.006
0.005
0.004
0.003
0.002
Median feature importances
H3K36me3-human
H3K4me1-human
H3K4me2-human
H3K79me2-human
Top 15 feature importances across classifiers

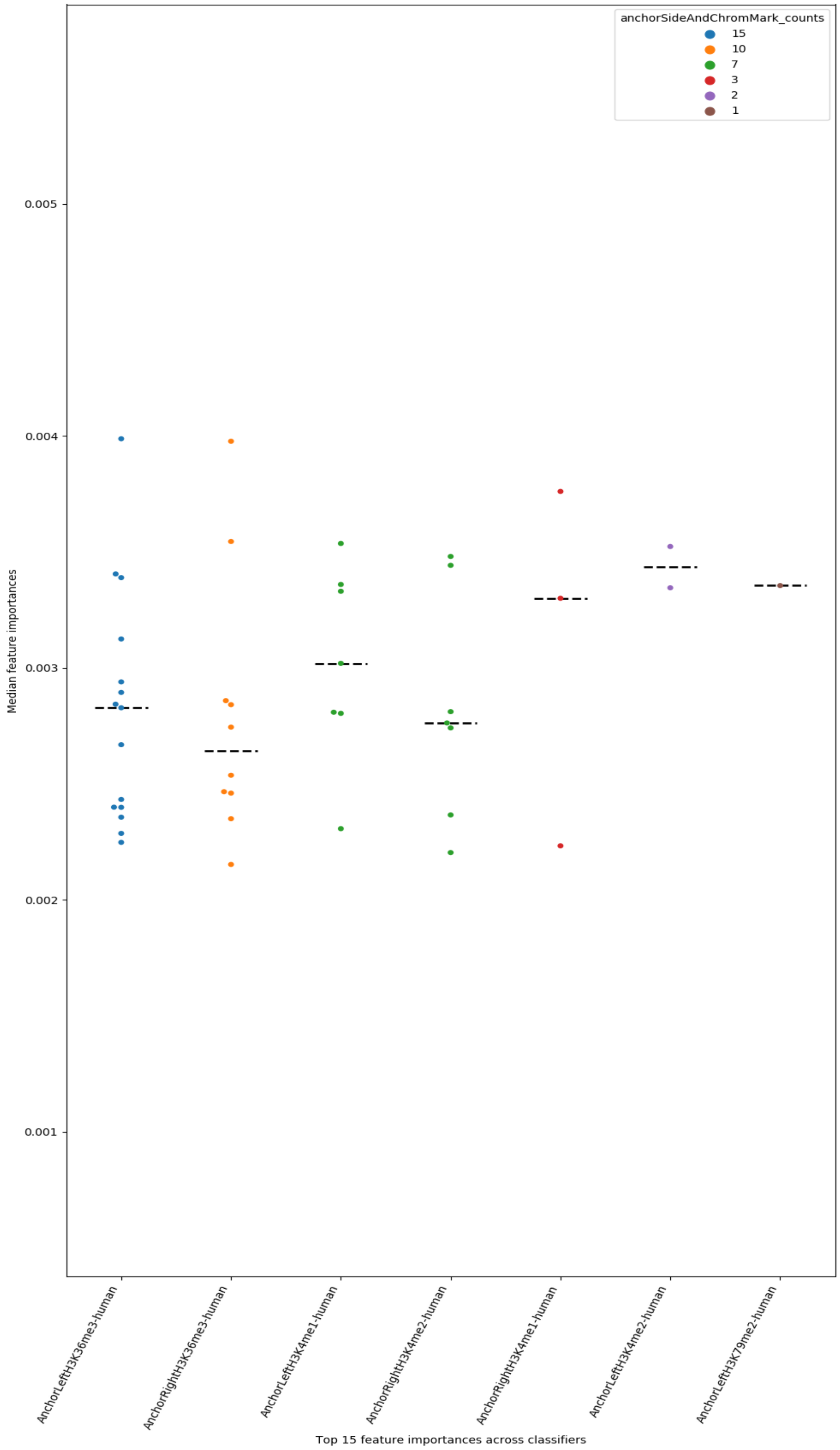
Supplementary Figure 3. Count and median feature importance of specific chromatin marks occurring in the top 15 feature importances for classification of real and randomly sampled (uniform) single anchors. *Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.*



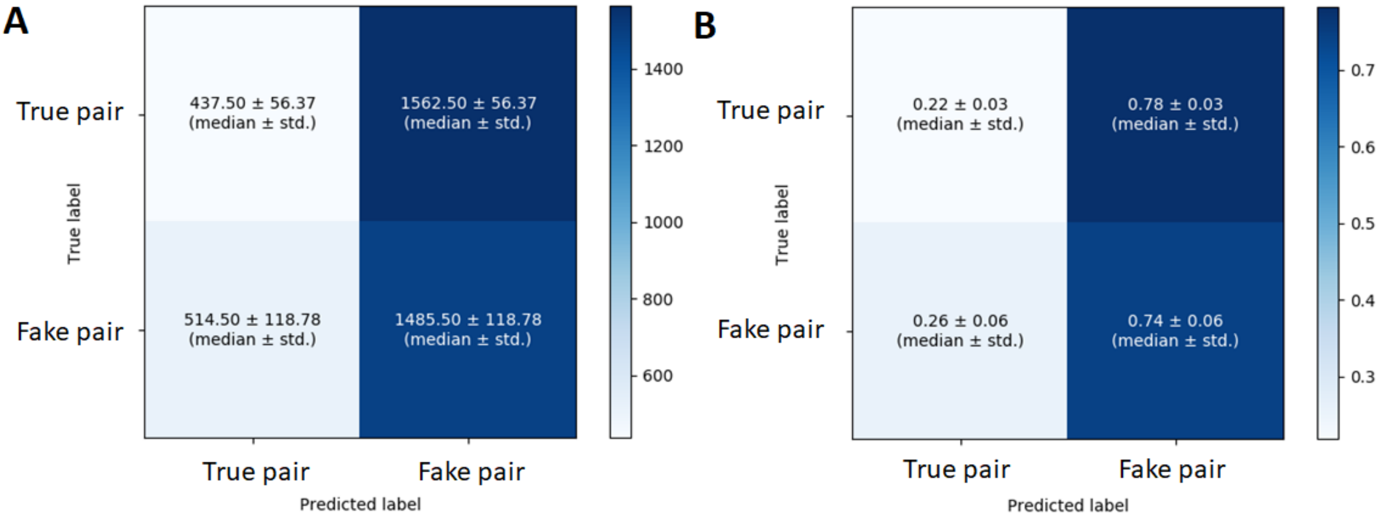
Supplementary Figure 4. Count and median feature importance of specific feature types occurring in the top 15 feature importances for classification of real and randomly sampled (mimick) single anchors. *Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.*



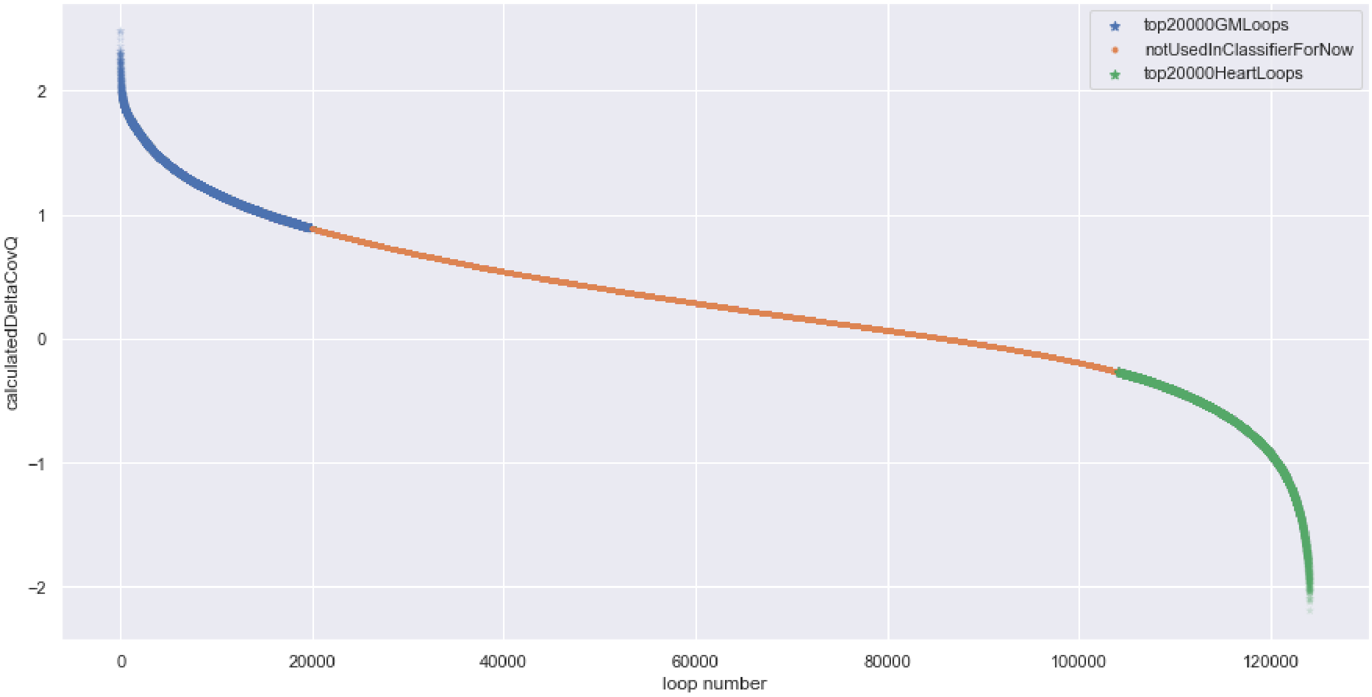
Supplementary Figure 5. Count and median feature importance of specific feature types and anchor sides (left or right) occurring in the top 15 feature importances for classification of real and randomly sampled (mimick) single anchors. *Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.*



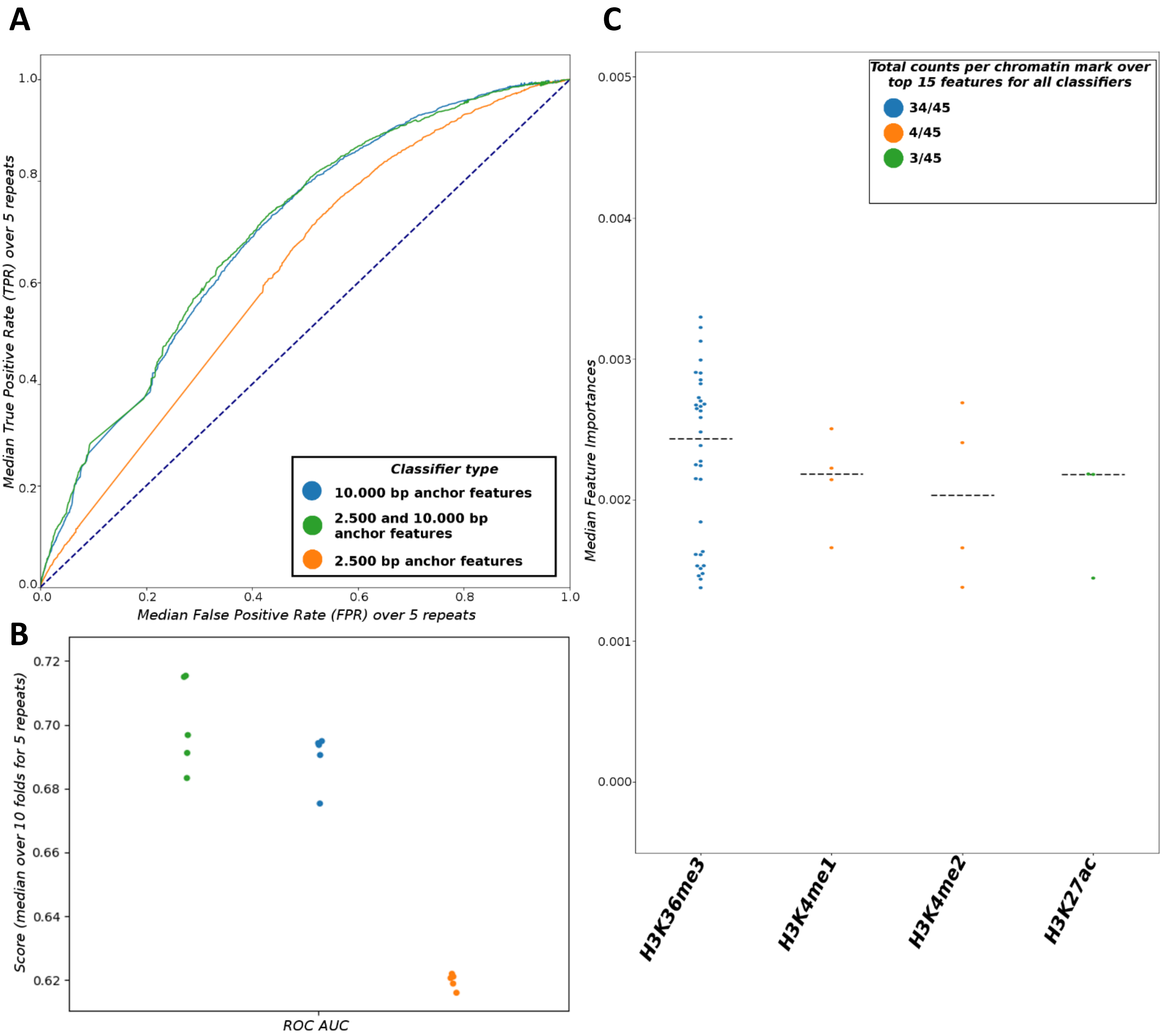
***Supplementary Figure 6. Count and median feature importance of specific epigenomic marks and anchor sides (left or right) occurring in the top 15 feature importances for classification of real and randomly sampled (mimick) single anchors**. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.*



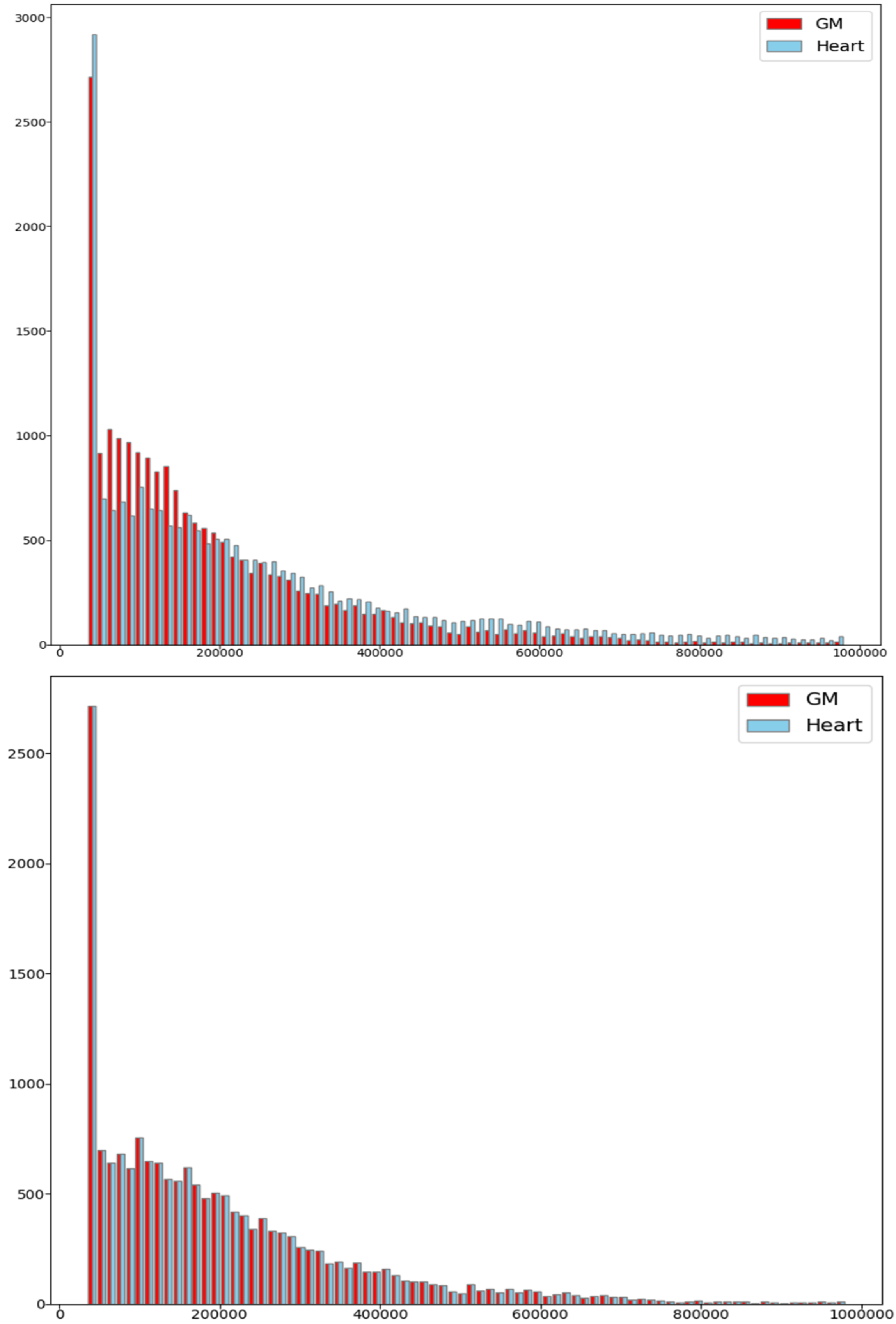
***Supplementary Figure 7. Confusion matrices for classification of real loop anchors and mismatched real loop anchors**. A: not normalised. B: normalised to 1 per class label. The classifier predicts both fake and true pairs overwhelmingly as fake anchor pairs.*



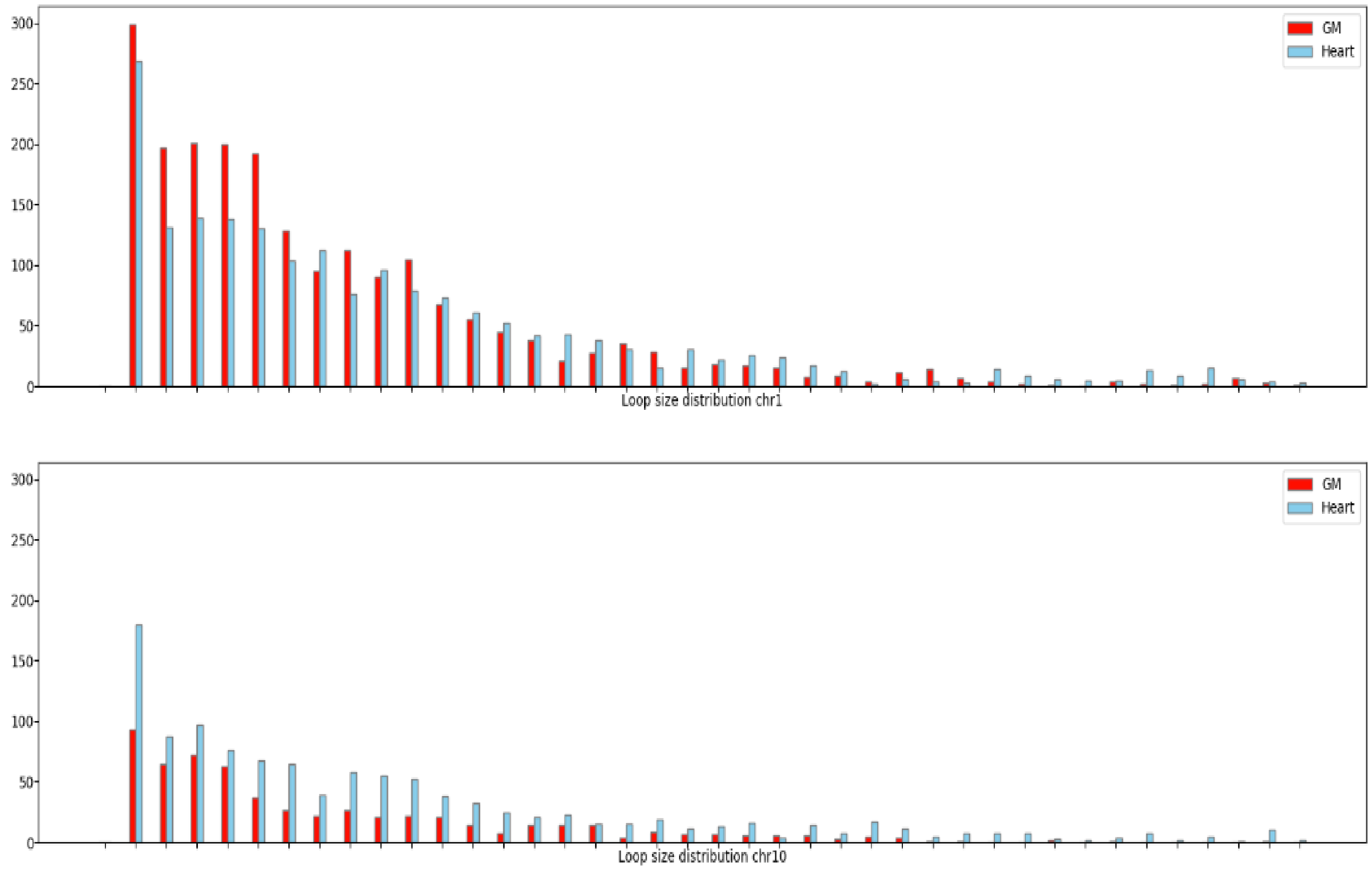
***Supplementary Figure 8. Difference in relative coverage of GM12878 loops and heart tissue loops in GM12878 and heart tissue for the strongest loops between 10 Kb bins in the genome (delta).** Note that the top 20,000 GM loops (blue) have higher absolute delta values than the top 20,000 heart loops (green): we are more sure that our GM loops occur almost solely in GM12878 than that heart tissue (non-GM) loops occur solely in heart tissue.*



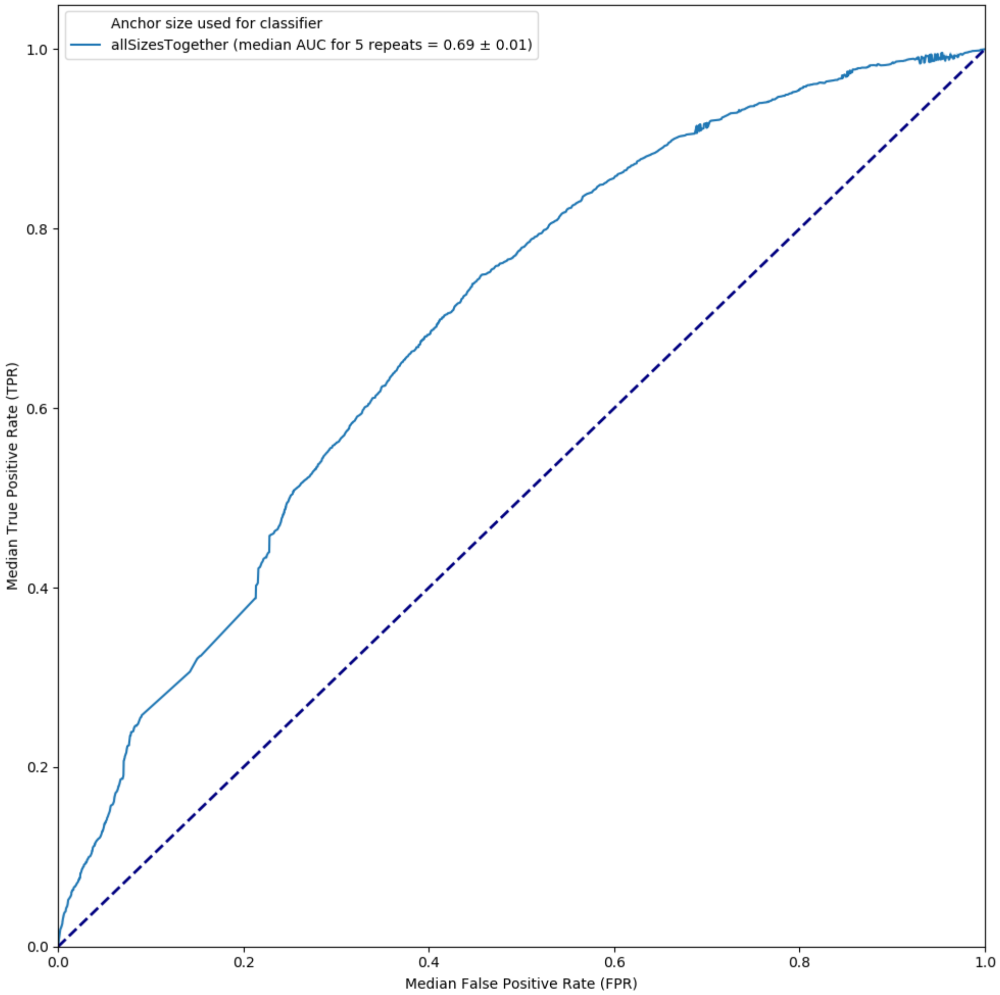
Supplementary Figure 9. Classification of 20.000 GM and non-GM (heart tissue) chromatin loops based on within-anchor features*. Loop callings obtained from de Laat group (see methods). A: Median receiver-operating characteristic (ROC) curve over 5 repeats of 10-fold cross-validated RF classifiers. B: Median area under the ROC curve (ROC AUC) for over 5 repeats of 10-fold cross-validated RF classifiers. Classifiers trained on features calculated for an anchor size of 2.500 bp (orange), 10.000 bp (blue) or with features calculated for both combined (green). C: Count and median feature importance of specific chromatin marks occurring in the top 15 feature importances over classifiers. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest. Dotted lines indicate the median feature importance of all features calculated from a specific chromatin mark. H3K36me3, H3K4me1, and H3K4me2 again turn out to be important for discerning anchors from non-anchors (in a specific cell type) based solely on within-anchor features.*



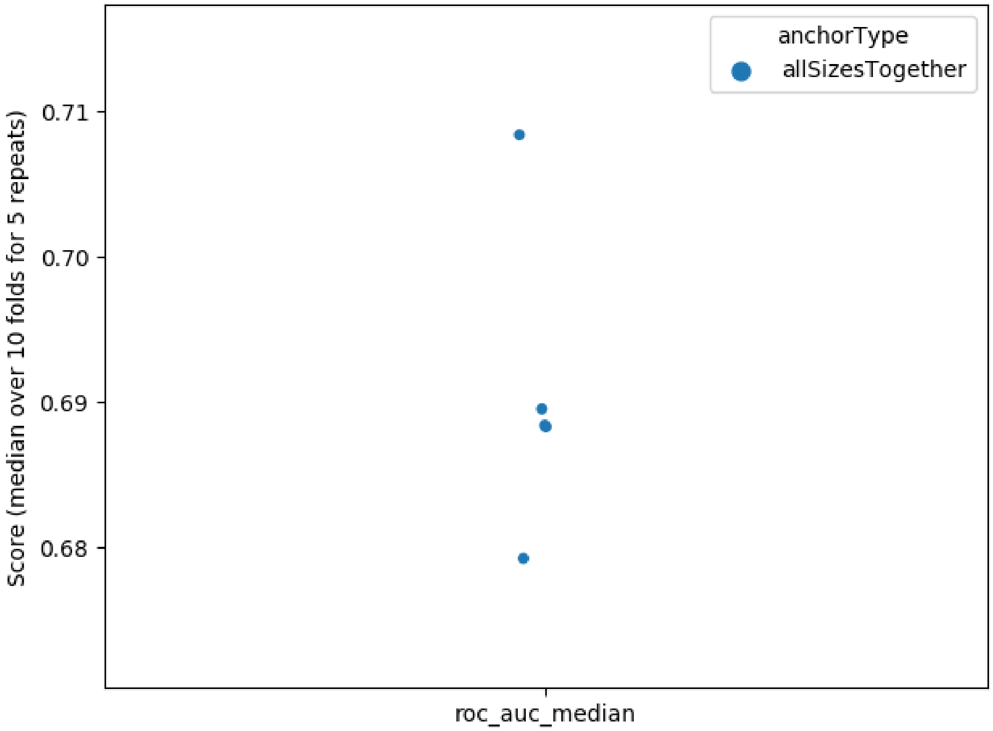
Supplementary Figure 10. Histogram of loop counts per loop size for GM and non-GM (heart tissue) loops, aggregated over all chromosomes, corrected (B) and uncorrected (A) for loop size disparity. *A: Histogram of loop sizes before selective downsampling. GM12878 clearly has more short loops (<200.000 bp), whereas larger loop sizes are overrepresented in heart loops. Since heart loops are called on tissue HiC-data, which is known to be more heterogenous (Amin Allahyar, personal communication), the loops called are less certain. Since longer loops are captured less often, one would expect heart loops to be shorter, since it is easier to get support from many reads for shorter loops. We observe the opposite, which is unexpected. B: histogram of loop sizes after selective downsampling. For every bin with more heart tissue loops, we randomly removed a loop, and then removed a GM loop from a bin with more GM loops.*



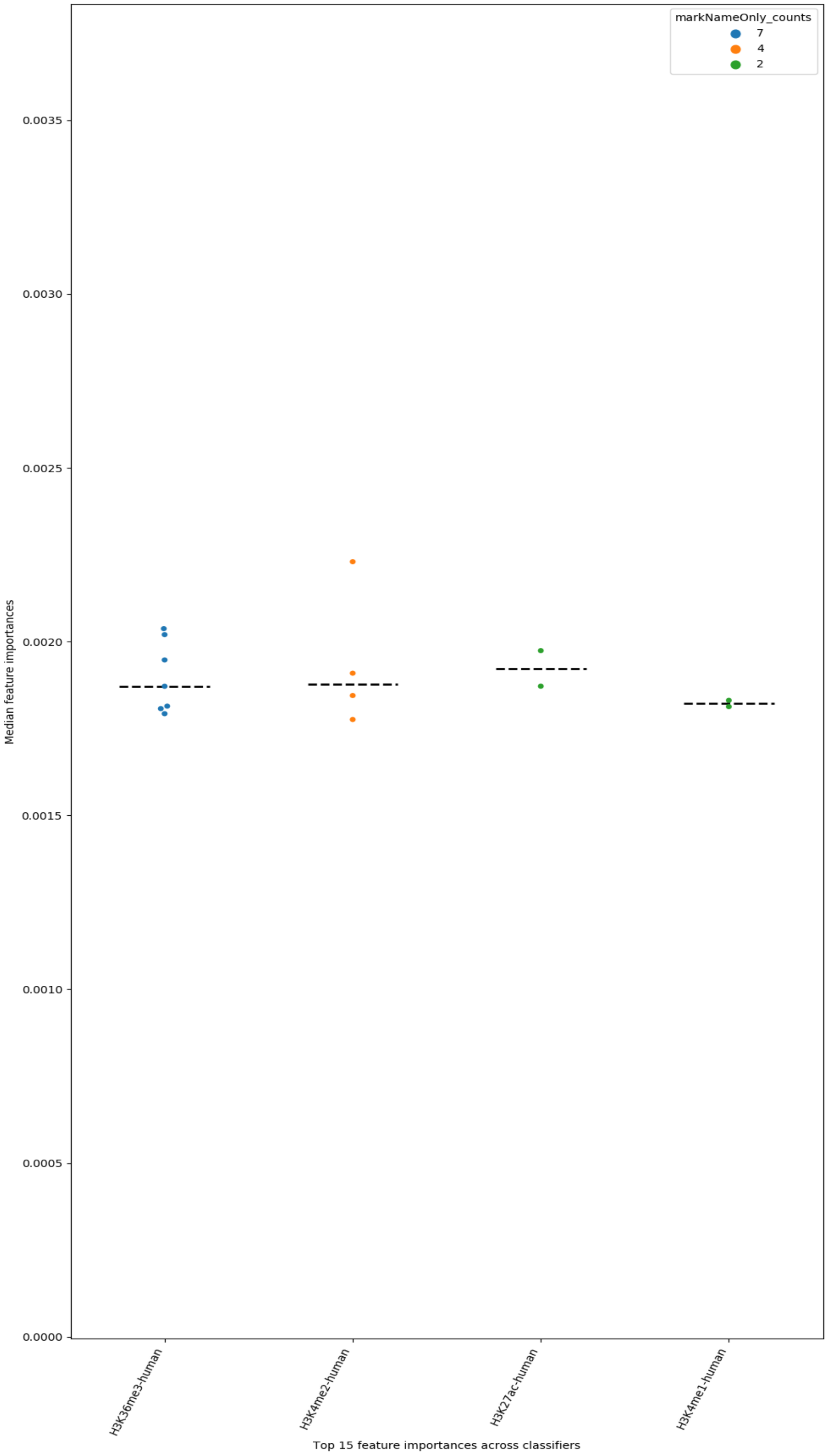
***Supplementary Figure 11. Histogram of loop size for GM and non-GM (heart tissue) loops on chromosome 1 and 10**. Although chromosome 1 nicely reflects the aggregate trend (Error! Reference source not found.), with GM predominating for short loops and heart tissue for longer oops, chromosome 10 has an overall overabundance of loops called in heart. While we performed correction on the overall loop size disparity (see below), an optimal strategy would be to do this on a per-chromosome basis (see discussion).*



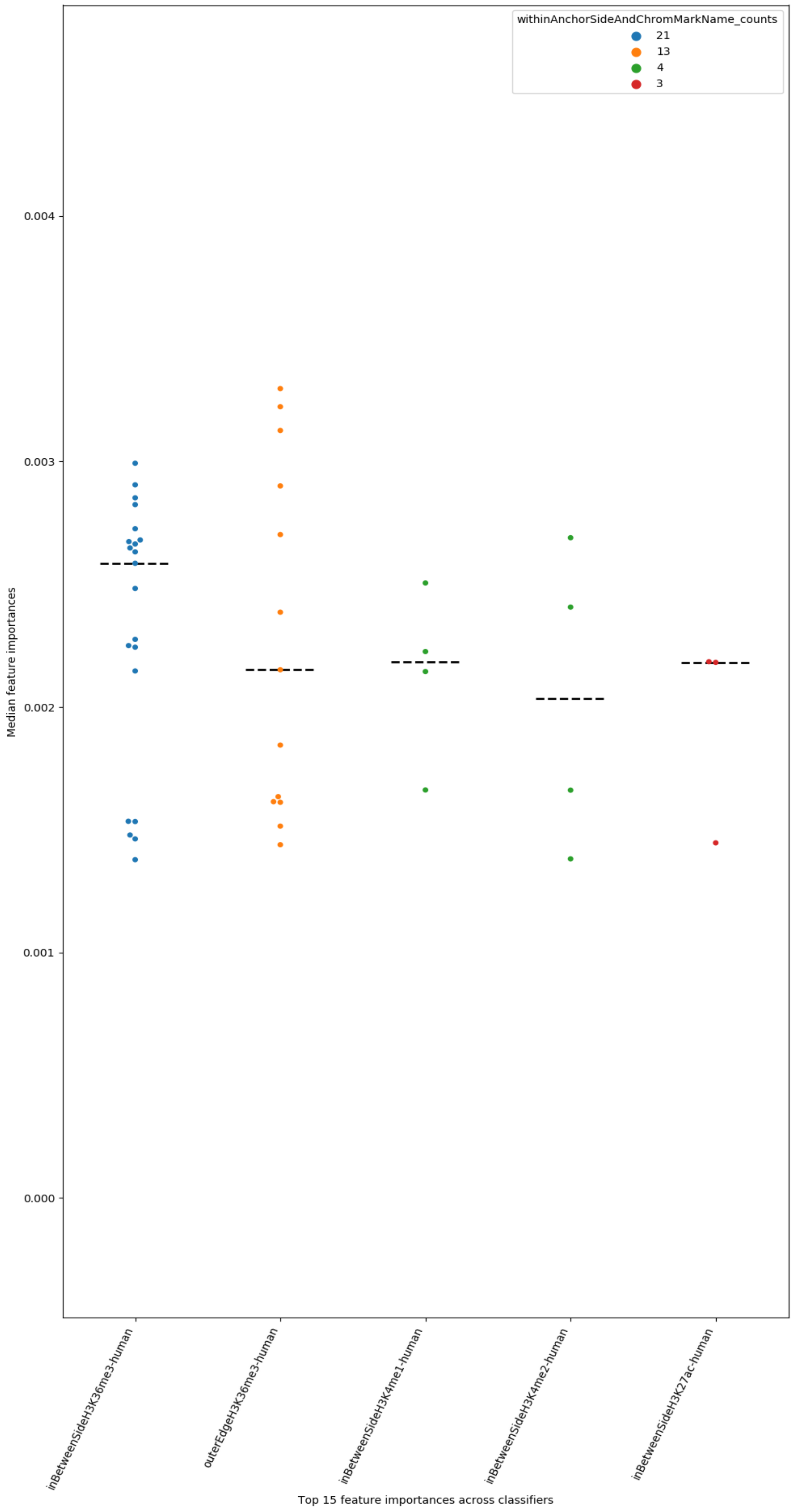
Supplementary Figure 12. ROC curve for classification of selectively downsampled GM and non-GM loops. *Combined feature set only.*



Supplementary Figure 13. Median ROC AUC values of 5 repeats for classification of selectively downsampled GM and non-GM loops. *Combined feature set only.*

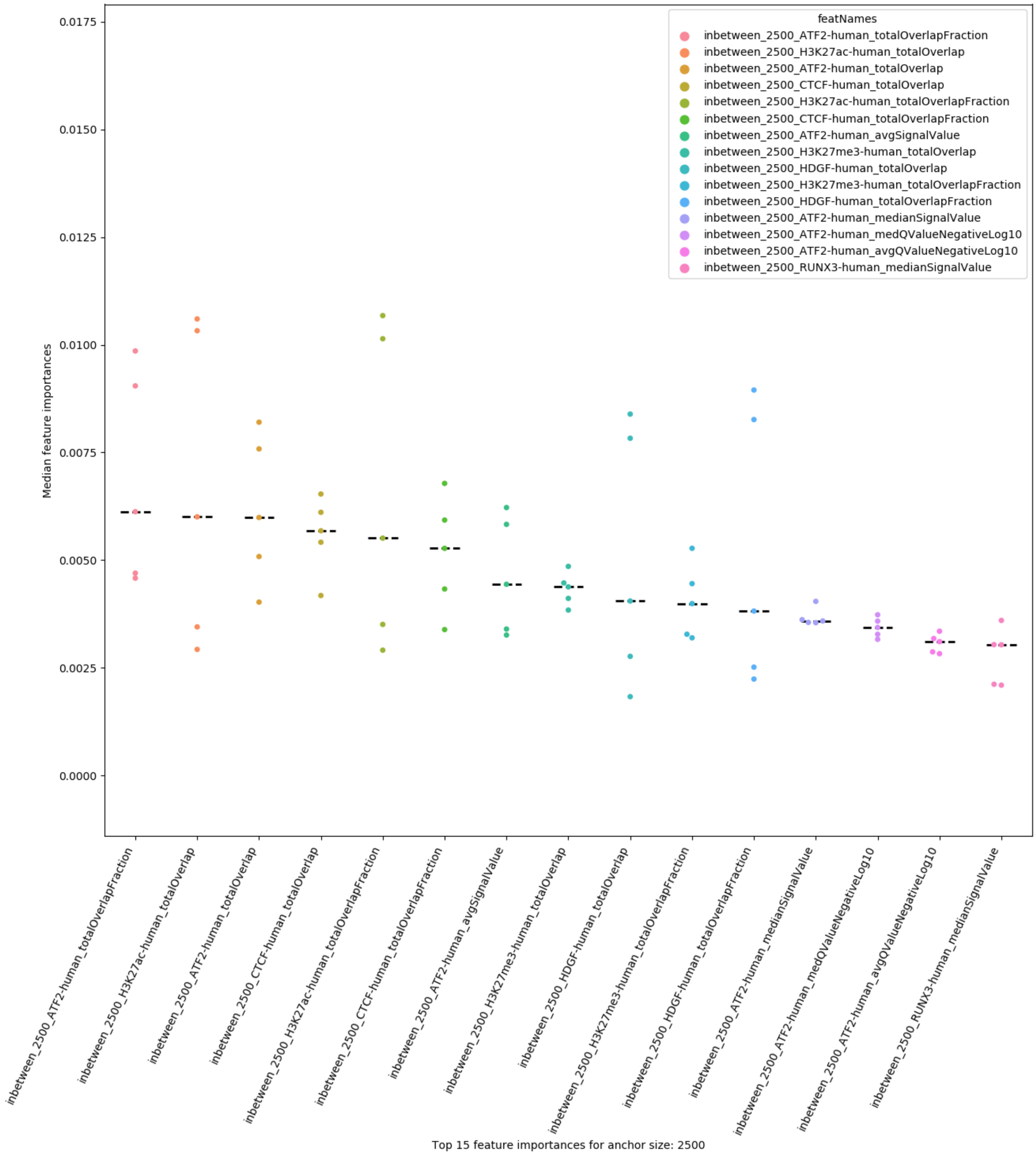


***Supplementary Figure 14. Count and median feature importance of specific chromatin marks occurring in the top 15 feature importances for classification of selectively downsampled GM and non-GM (heart tissue) loops.** Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.*

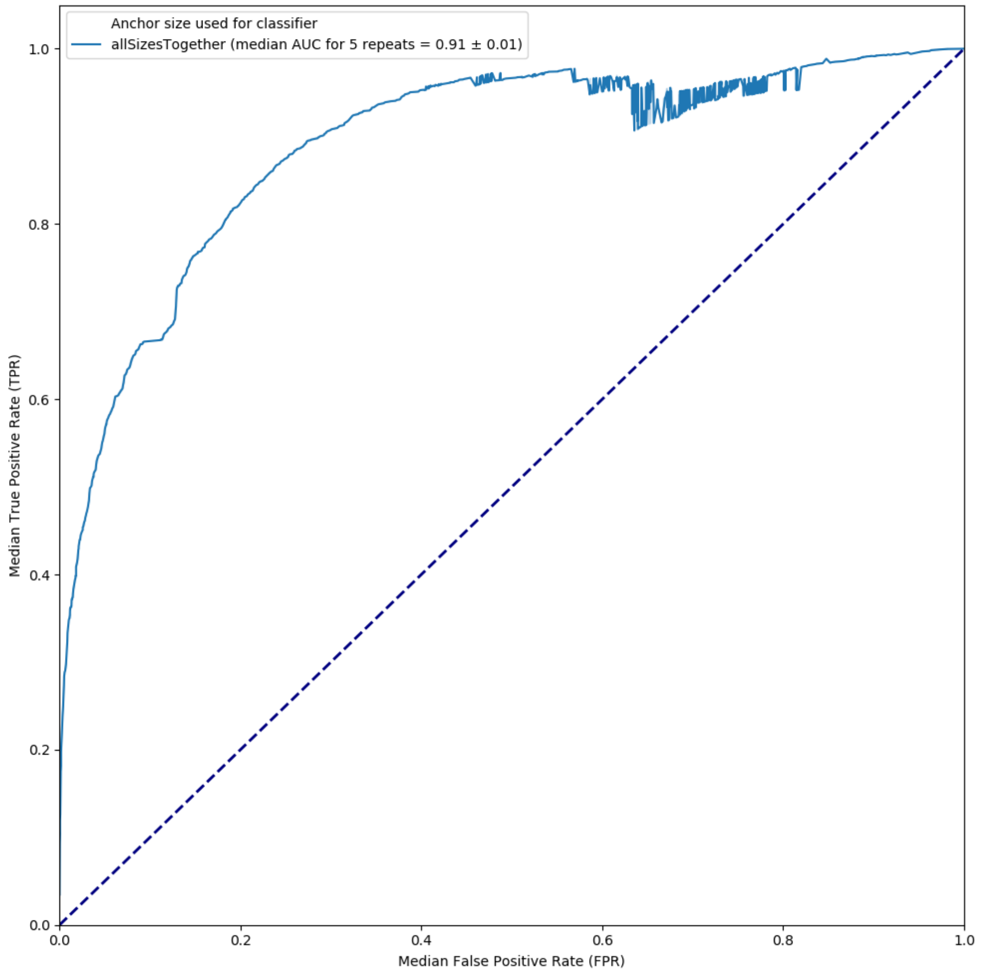


Supplementary Figure 15. Occurrence counts and feature importance values for chromatin marks on the in-between side or the outer edge of the loop anchors across classifiers. *H3K36me3 is assigned relatively high importance on both the in-between side and outer edge, whereas*

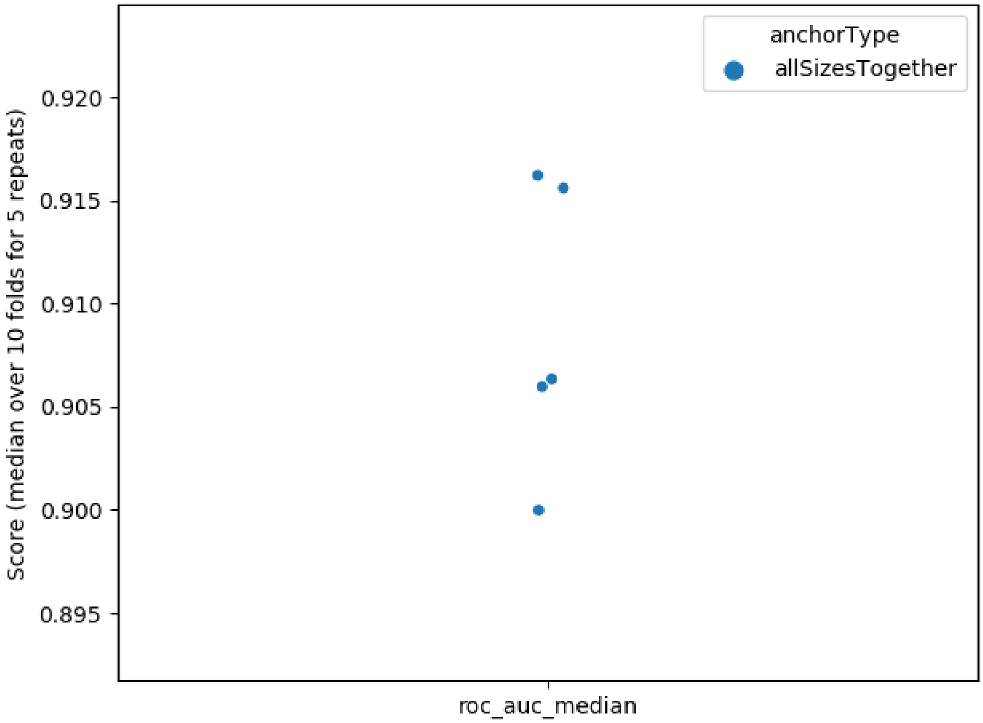
H3K4me1 and H3K4me2 are important solely on the inside of the loop. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.



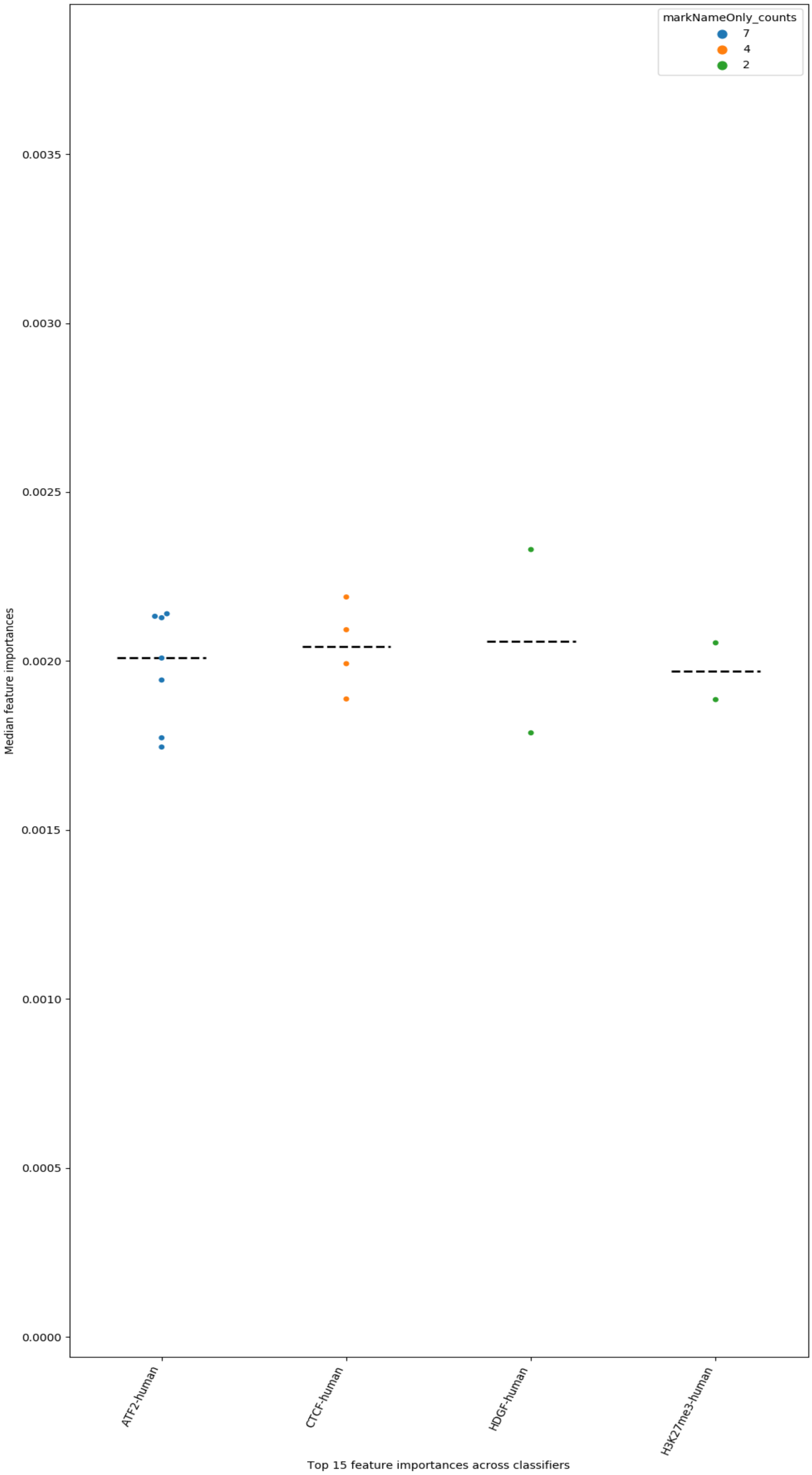
Supplementary Figure 16. Top 15 most important features in for RF classifiers trained on feature set with an anchor size of 2.500 bp. *All top-scoring features are those calculated on the in-between area of the loop, rather than any within-anchor features. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.*



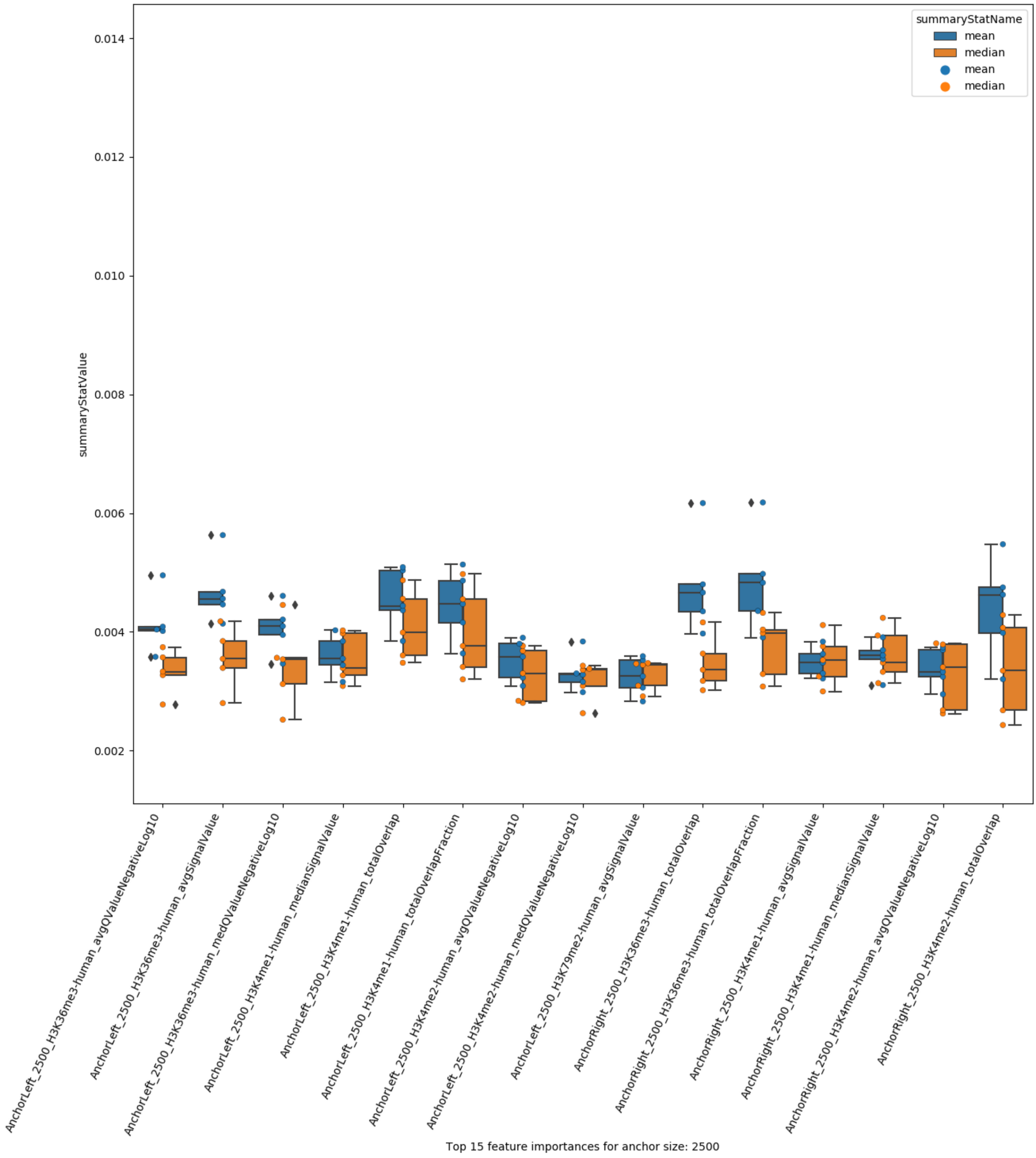
Supplementary Figure 17. Median ROC curve for classification of selectively downsampled GM and non-GM loops, with features calculated for area in-between the loop anchors. *Combined feature set only.*



Supplementary Figure 18. Median ROC AUC values of 5 repeats for classification of selectively downsampled GM and non-GM loops, with features calculated for area in-between the loop anchors. *Combined feature set only.*



Supplementary Figure 19. Count and median feature importance of specific chromatin marks occurring in the top 15 feature importances for classification of selectively downsampled GM and non-GM (heart tissue) loops, with features calculated for area in-between loop anchors. *Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats was highest.*



Supplementary Figure 20. Boxplots of median (orange) or mean (blue) top 15 feature importances over repeats overlaid with the actual data points for single anchor classification (mimick). *Median importances seem more conservative and are less sensitive to outliers, so we used that as our summary statistic across folds and repeats. Top feature importances were defined as the 15 features whose median feature importance over 5 classifier repeats (i.e. 5 repeats of classifier training on 10 cross-folds) was highest.*

Supplementary Tables

Supplementary Table 1. Parameter values allowed for sampling in the random search cross-validation performed per classifier. *None for max_features indicates that the maximal amount of features to consider when looking for the best split are all features. "sqrt" and "log2" indicate to consider a maximum of sqrt(n_features) or log2(n_features) for the best split, respectively. None for max_depth indicates that nodes are expanded until all leaves are pure or contain less than min_samples_split samples. Random search means that a random value is chosen for each parameter per iteration.*

N_estimators	Max_depth	Max_features	Min_samples_split	Min_samples_leaf
50	1	"sqrt"	2	1
150	10	"log2"	5	5
300	20	None	10	10
600	60	-	25	-
-	100	-	75	-
-	None	-	-	-

Supplementary Table 2. Accessions used for the 153 ChIP-Seq marks for GM12878 downloaded from Ensembl.

Factor name	Accession used
E2F4-human	ENCFF687SFB
TBP-human	ENCFF896UZB
ARID3A-human	ENCFF003VDB
HDAC2-human	ENCFF299UPZ
RB1-human	ENCFF034OSV
CHD4-human	ENCFF249SIN
RUNX3-human	ENCFF677QUK
ZNF217-human	ENCFF200SLC
ZNF687-human	ENCFF137BRA
CBX5-human	ENCFF417SVR
NFYA-human	ENCFF278GJK
KDM1A-human	ENCFF799KZP
MEF2A-human	ENCFF958GXF
MYB-human	ENCFF402TSJ
KLF5-human	ENCFF417WPC
MAFK-human	ENCFF186AWV
PBX3-human	ENCFF926LHG
SMAD5-human	ENCFF855SJG
ZBTB40-human	ENCFF084IUW
NRF1-human	ENCFF652BRY
LARP7-human	ENCFF305SLO
ZBED1-human	ENCFF630FLK
CBX3-human	ENCFF552QOA
ZSCAN29-human	ENCFF214NJL
USF1-human	ENCFF701QXK
NFXL1-human	ENCFF860IXB
E2F8-human	ENCFF412GFI
E4F1-human	ENCFF035GFS
MAX-human	ENCFF270NAL
ZFP36-human	ENCFF224WII
SIX5-human	ENCFF864TFH
JUNB-human	ENCFF478XNA

GABPA-human	ENCFF946ACA
FOXK2-human	ENCFF990MTR
TCF7-human	ENCFF152RNE
NFATC3-human	ENCFF704PDA
RAD51-human	ENCFF996NBR
RFX5-human	ENCFF259LNG
ZNF24-human	ENCFF313HBL
ATF7-human	ENCFF495PWL
ZNF207-human	ENCFF676BIG
CREM-human	ENCFF091YID
BCL3-human	ENCFF247MHT
CHD1-human	ENCFF863CTN
KAT2A-human	ENCFF710ROZ
NFATC1-human	ENCFF138ZBJ
TBX21-human	ENCFF971VHK
BACH1-human	ENCFF725YZH
MLLT1-human	ENCFF125MEN
USF2-human	ENCFF514SWA
NKRF-human	ENCFF084NXU
CBFB-human	ENCFF070SOX
MEF2C-human	ENCFF830BRO
NR2F1-human	ENCFF531KOV
EED-human	ENCFF023ALY
SUPT20H-human	ENCFF069YVD
ELF1-human	ENCFF948CPI
HDGF-human	ENCFF442WRJ
RBBP5-human	ENCFF687SSY
ZNF592-human	ENCFF615DTQ
NR2C1-human	ENCFF462AKP
CEBPZ-human	ENCFF243GOG
ELK1-human	ENCFF432AQP
RELB-human	ENCFF105YDI
ASH2L-human	ENCFF096XRG
MTA2-human	ENCFF587POH
PAX8-human	ENCFF992JWY
MEF2B-human	ENCFF623FAW
YY1-human	ENCFF223MUF
GATAD2B-human	ENCFF298AIX
TBL1XR1-human	ENCFF392JWA
SMAD1-human	ENCFF987PGY
NR2C2-human	ENCFF434HVY
DPF2-human	ENCFF771IAW
CUX1-human	ENCFF567NFS
POLR2AphosphoS2-human	ENCFF847DXY
HCFC1-human	ENCFF722QBB
MXI1-human	ENCFF199HGX
EZH2-human	ENCFF615NYO
ARNT-human	ENCFF758RQJ
IRF5-human	ENCFF843HDK

ESRRA-human	ENCFF722LJP
NBN-human	ENCFF811VEN
IRF4-human	ENCFF720YMW
ETS1-human	ENCFF980VOD
YBX1-human	ENCFF500RBO
ZNF622-human	ENCFF777DVJ
ZZZ3-human	ENCFF260NAX
NFIC-human	ENCFF480WDX
RAD21-human	ENCFF654EGO
ZEB1-human	ENCFF204LCG
WRNIP1-human	ENCFF514DDI
CEBPB-human	ENCFF786YYI
BMI1-human	ENCFF592LPO
STAT1-human	ENCFF323QQU
HSF1-human	ENCFF603BID
RXRA-human	ENCFF313BDA
MAZ-human	ENCFF348STZ
SMARCA5-human	ENCFF052STI
PKNOX1-human	ENCFF335ADU
SKIL-human	ENCFF903KEI
ZNF384-human	ENCFF942MDT
SMC3-human	ENCFF572RPI
TAF1-human	ENCFF540AAP
JUND-human	ENCFF873DJD
NFE2-human	ENCFF743UMZ
UBTF-human	ENCFF295ZLM
BRCA1-human	ENCFF005JKU
BATF-human	ENCFF832YIE
STAT3-human	ENCFF923CHO
NFYB-human	ENCFF510NDO
BCL11A-human	ENCFF383HAY
CHD2-human	ENCFF546AYN
STAT5A-human	ENCFF383YEA
POLR2A-human	ENCFF455ZLJ
SIN3A-human	ENCFF050CYK
POLR2AphosphoS5-human	ENCFF600GQL
MTA3-human	ENCFF661FMB
RCOR1-human	ENCFF470ZMK
EBF1-human	ENCFF249SVT
IKZF1-human	ENCFF018NNF
ZBTB33-human	ENCFF475DID
TRIM22-human	ENCFF552WAH
REST-human	ENCFF313CII
ATF2-human	ENCFF210HTZ
BHLHE40-human	ENCFF370ZNL
ZNF143-human	ENCFF153TQR
TCF12-human	ENCFF897RYA
BCLAF1-human	ENCFF381ZDU
IKZF2-human	ENCFF088OLI

TARDBP-human	ENCFF871LZM
IRF3-human	ENCFF604AZX
SRF-human	ENCFF766WWB
EP300-human	ENCFF865UDD
CTCF-human	ENCFF960ZGP
ETV6-human	ENCFF745ANU
PAX5-human	ENCFF196JGP
H3K4me2-human	ENCFF108JKZ
H3K79me2-human	ENCFF213GPU
H3K9ac-human	ENCFF069KAG
H2AFZ-human	ENCFF512VHW
H3K4me1-human	ENCFF453PEP
H4K20me1-human	ENCFF025VJJ
H3K36me3-human	ENCFF268HMO
H3K27ac-human	ENCFF367KIF
H3K27me3-human	ENCFF035PQG
H3K4me3-human	ENCFF188SZS
TCF3-human	ENCFF002CIA
ATF3-human	ENCFF002CGP
PML-human	ENCFF002CHM
POU2F2-human	ENCFF002CHP
FOXM1-human	ENCFF002CGZ
ZNF274-human	ENCFF002CPX

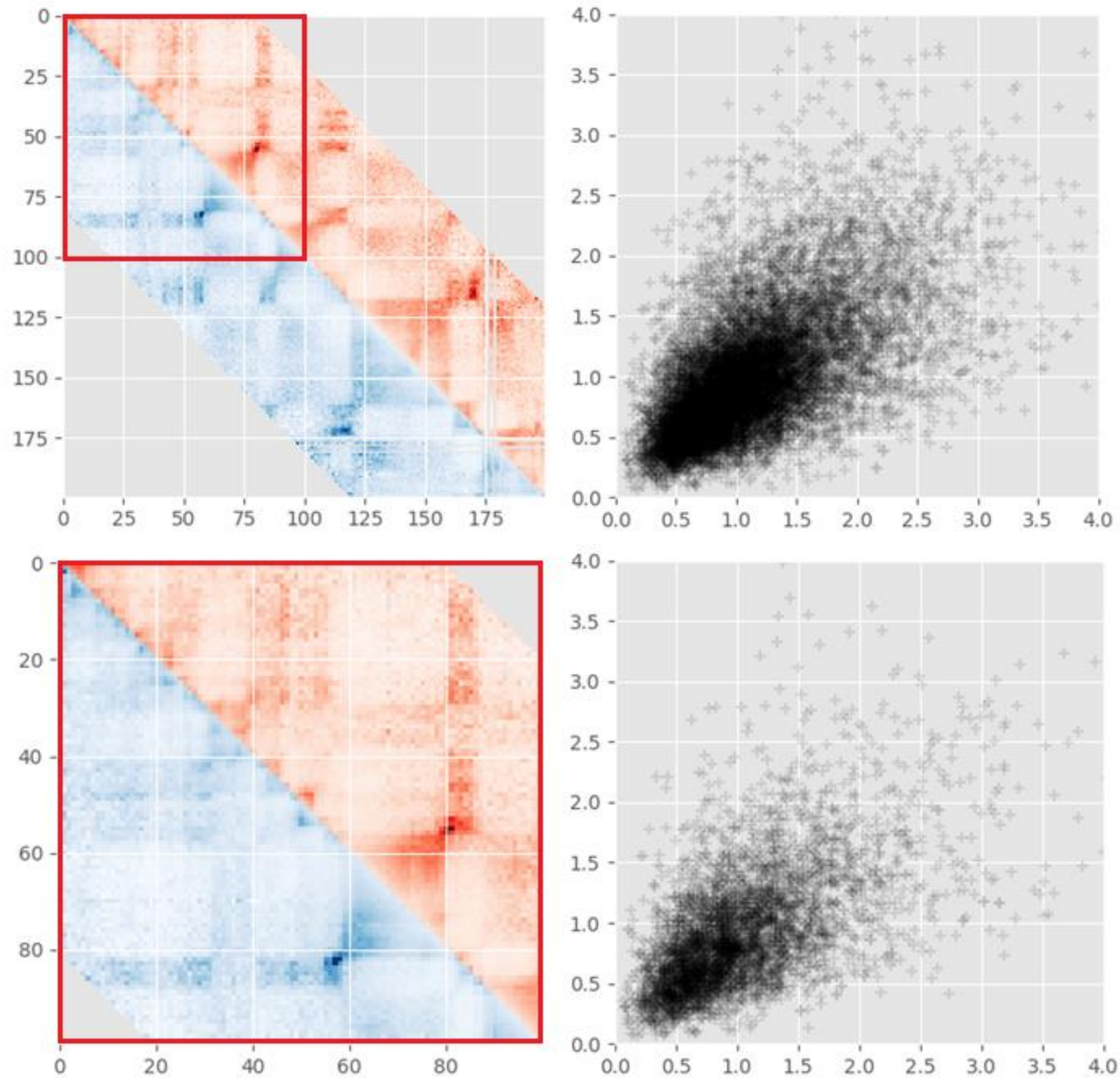
Supplementary Data 1: prediction of HiC matrix using a dense neural network

Introduction

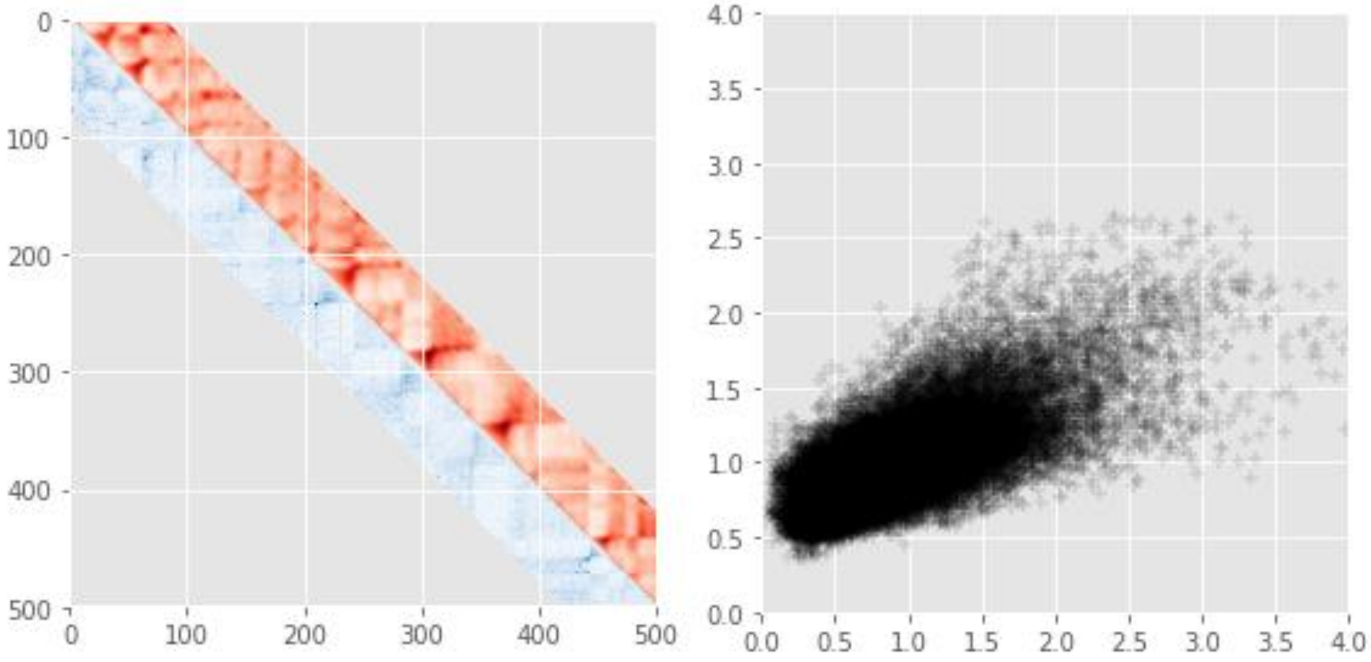
In this work, we have used Random Forest classifiers. Tree-based classifiers have a number of attractive properties. They are resistant to collinearity, meaning you can give many features that are highly similar and still get good results[29,30]. Additionally, and importantly for the biological sciences, tree-based classifiers assign easily-interpretable feature importances, allowing one to find factors that are important in making predictions[28,30]. We used this characteristic to identify ATF-2 and HDGF as possible factors in chromatin looping. The downside of this approach is that we need to manually encode features. Specifically, there are many variations of the stretches of DNA between loop anchors that we could include, and the anchor sizes we select are also arbitrary. Ideally, one would try all possible permutations of anchor sizes and in-between stretches for which to include features, but that was computationally intractable within the confines of this project. The recent success of deep learning hinges on the fact that the network learns both the optimal representation of raw data and the best decision function for classification (along with the recent increase of data availability for select problems, increase in computing power, and development of better algorithms)[52,53]. We were particularly inspired by recent work in which a dense neural network with a simple 1D convolutional layer was used with ChIP-Seq data on 50 factors to predict entire HiC interaction matrices in a part of the Drosophila genome[39]. Though our focus is on chromatin loops rather than the entire HiC matrix, we here repeated this previous work to show the feasibility of using such an approach. We encountered some strange behaviour that requires follow-up.

Results

We wanted to see whether we could reproduce the findings in the paper. To that end, we downloaded the scripts from their Git repository and the same data from the sources indicated in the paper. We slightly modified the approach by only including chromatin marks that have a ‘repset’ file, i.e. for which the ChIP-Seq peaks are confirmed in at least two separate experiments. We then made matching windows of ChIP-Seq and HiC data to serve as a training set. Briefly, we took HiC and ChIP-Seq data for chromosome 2R, 2L, half of 3R, and 3L for training, and tested on the other half of chromosome 3R. We differed in the buffer we kept at the edge of the chromosomes where we made no bins (since no complete windows are available there). This means that our bins do not line up completely with the bins used in the paper, but their agreement is high (**Supplementary Data Figure 1**). Otherwise, we used exactly the same training and testing regimen. The results are the same as in the paper: a moderately good agreement with the actual data, with an average correlation of predicted and real HiC matrix values of 0.69 (**Supplementary Data Figure 2**).

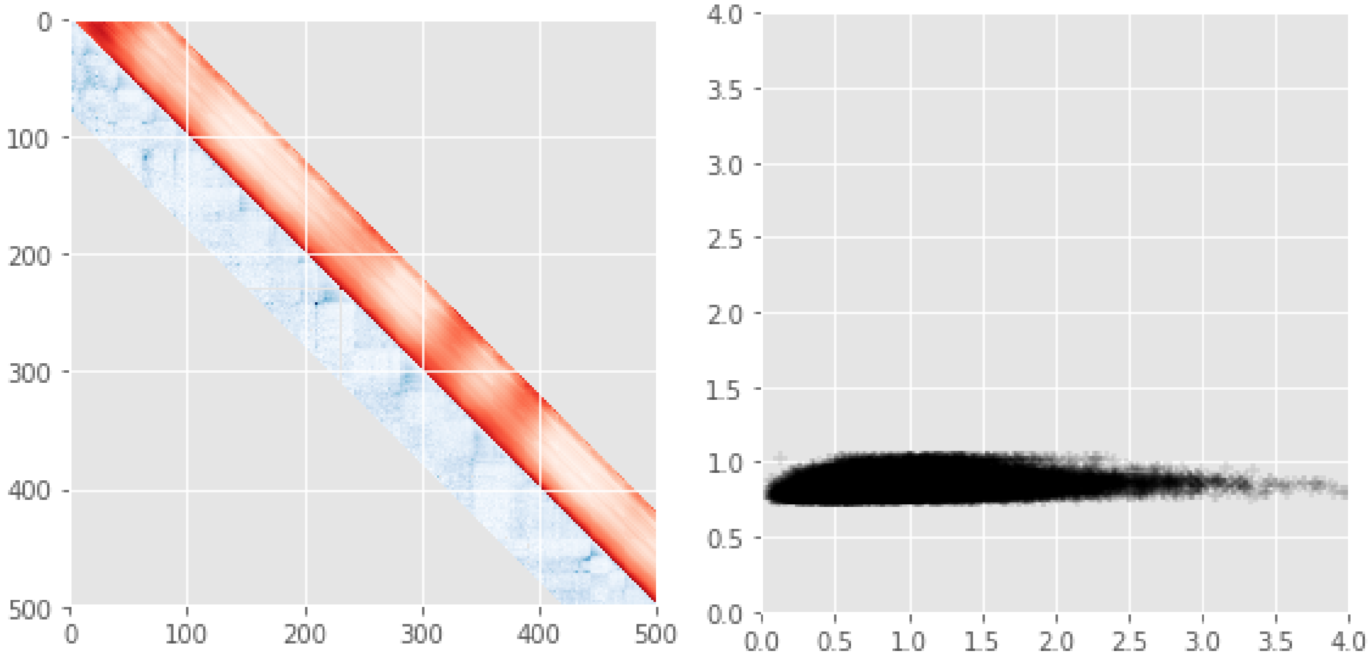


Supplementary Data Figure 1. Agreement of our HiC matrix bins with their HiC matrix bins for chromosome 2L. *Left: side-by-side comparison of HiC signal in a heatmap of paper data (blue) and our data (red). Right: correlation of bins. Bins are slightly shifted due to us keeping a larger buffer zone at the end of chromosomes. Bottom is a zoom-in of top. Average correlation across bins is 0.85 despite shifting.*



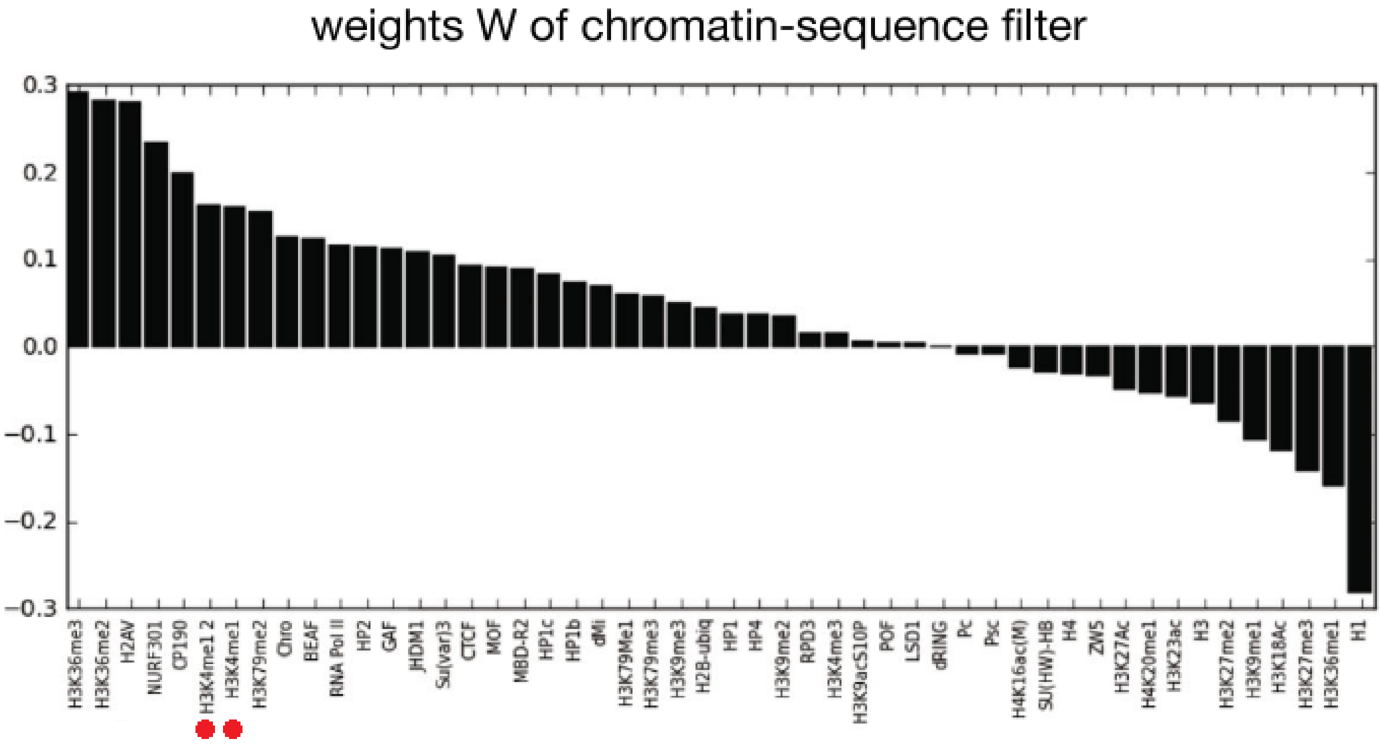
Supplementary Data Figure 2. Real (blue) and predicted (red) HiC matrix interactions (left) and predicted normalised interaction frequency versus real normalised interaction frequency (right). *49 chromatin marks were used for classification (two datasets for H3K4me1). Average Pearson correlation of real data with predicted data was 0.69.*

Strangely, if we removed one of the marks that was duplicated in the data (H3K4me1) and reran the classification, performance was completely removed, with an abysmal average correlation of 0.24 (**Supplementary Data Figure 3**).



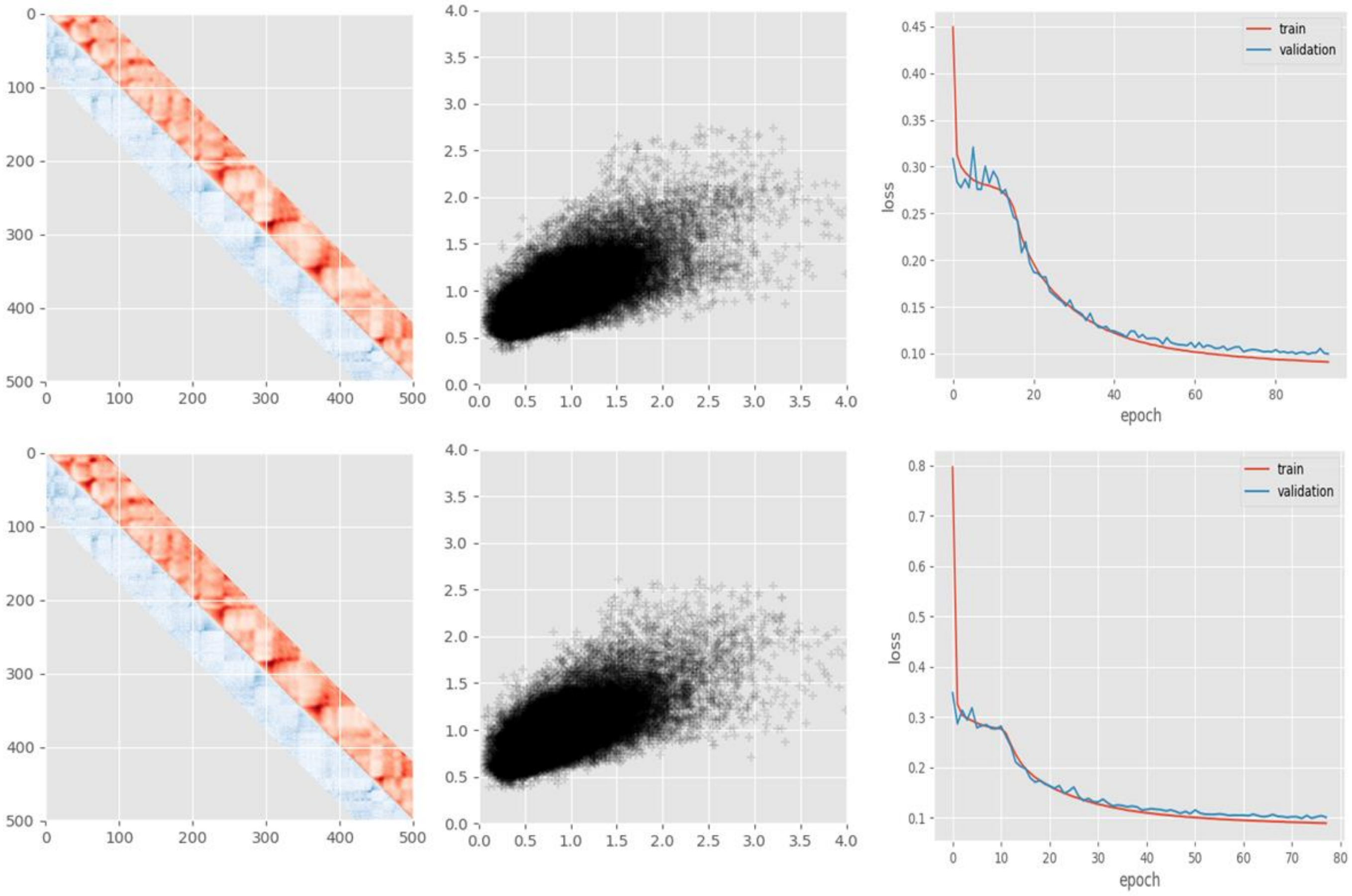
Supplementary Data Figure 3. Real (blue) and predicted (red) HiC matrix interactions (left) and predicted normalised interaction frequency versus real normalised interaction frequency (right). *48 chromatin marks were used for classification (one dataset for H3K4me1). Average Pearson correlation of real data with predicted data was only 0.24.*

This is even stranger given that this mark is not given most importance in the 1D convolutional filter, and that the weights given to each of the datasets supplying information of that mark is about equal (**Supplementary Data Figure 4**). Therefore, removing one such mark should not have the destructive impact on performance that it has.



Supplementary Data Figure 4. Weights of chromatin-sequence filter as reported in Emberly et al. *Red dots denote the weights given to the two datasets for H3K4me1 coverage. Adapted from:* [39]*.*

Given this fact, we refactored the paper code for our purposes and followed their training regimen exactly, populating the bins with chromatin mark coverages using their code, rather than our own. Now, the difference between inclusion and exclusion of the chromatin mark was negligible (**Supplementary Data Figure 5**).



Supplementary Data Figure 5. Prediction performance using paper code for assigning ChIP-Seq values to chromosome bins. *Top: Prediction performance with duplicate H3K4me1. Bottom: prediction performance with one signal for H3K4me1. Left: agreement of real data (blue) and predicted data (red). Middle: Real and predicted interaction frequency. Right: loss over training epochs.*

Given that our genome bin – ChIP-Seq signal is highly similar to that of the paper (**Supplementary Data Figure 1**), it is strange that the problem disappears completely when the paper code is modified and used to train either with or without the duplicate mark (**Supplementary Data Figure 5**). We did not manage to find the problem that causes this yet.

Discussion

We trained a dense neural network with a 1D convolutional layer to predict contact frequencies in a HiC matrix. Strangely, we found that including only one dataset for H3K4me1 completely removed correlation with the actual matrix (**Supplementary Data Figure 3**), whereas including two as in the original paper produced the same accuracy as reported there (**Supplementary Data Figure 2**). This should not happen, and the original authors suggested we must have made an error in data handling (Pau Farré, personal communication). This remains the most parsimonious explanation, especially given the fact that switching completely to the paper's method of generating genome bin-ChIP-Seq coverage pairs removed any such errors (**Supplementary Data Figure 5**). Due to time constraints, we did not pursue this further and instead turned our focus to RF classification.

Future work

The first step should be to investigate the strange behaviour of the classification as mentioned. Ultimately, this classification is simply a first step. An immediate improvement on the methods employed here is to use more intricate combinations of convolution: a 1D convolutional filter makes little use of surrounding information for each window. There is some translation of nearby chromatin state into the final prediction because predictions for overlapping windows of ChIP-Seq-Hic pairs are averaged, but more convolutions could integrate information on surrounding marks into higher-level features and leverage these for the prediction for each position of the HiC matrix[53]. This would come at the expense of interpretability (the current method has an immediate biological read-out of what marks are conducive or restrictive for DNA contacts; **Supplementary Data Figure 4**). However, it would much more effectively leverage the power of automatic feature encoding, in a manner similar to what we envision for predicting chromatin loops. The issue of interpretability could be overcome with novel methods that are coming out to question what neural networks have learned[58–60]. As we mentioned in the discussion in the main text, a deep learning approach for identifying epigenomic marks important to loop formation would need anchor pairs that form loops and anchor pairs that do not as training data, supplemented with information on the in-between area (since it is so important for classification performance) that could benefit greatly from many convolutional layers. Our view is that this could increase performance and knowledge discovery greatly. However, it would require a more substantial investment of time in understanding the learned features and neural network performance. Given this, our Random Forest approach can serve as a baseline of chromatin loop prediction and epigenomic mark importance, which can be improved by using a convolutional neural network architecture.

Methods

Processing of Drosophila DNN data

To repeat the work done by Emberly et al. (2018), we first downloaded the data they used for their report (HiC data for *Drosophila melanogaster* embryos performed at 16-18 hours post egg-laying (GSE61471)[70], 50 chromatin factors measured with ChIP-Seq in *D. melanogaster* embryos 14-16 hours post egg-laying[71]). We then downloaded their code from Github (https://github.com/pau557/HiC-DNN; though note that it is offline at the time of writing pending a rewrite to make it more easily usable). The HiC data was ICE-normalised using gcMapExplorer[72], and further handled by a custom Python script (*HiCProcessingUsinggcMapExploreFiles2.py*) that split the data in the manner employed in[39]. This resulted in 5 HiC matrices (chr2R, chr2L, chr3RFirstHalf, chr3RSecondHalf, chr3L). For chromatin mark data, data were unzipped, and we calculated the fraction of each 10 kb bin that was covered by areas enriched for each chromatin mark using a custom Python script (*ChIP-SeqProcessing3.py*). We found that one of the chromatin marks had no “repset” file, which combines results from two separate peak calling experiments. We disregarded this mark. We further found that H3K4me1 was duplicated in the data (as evident from **Supplementary Data Figure 4**). We decided to take only one of these marks (based on newest antibody lot, GSE47281, modENCODE ID 5092). Thus, 48 chromatin marks were left. This resulted in a 240 by 48 chromatin mark matrix. For comparison with their data, we also performed a run where we left this mark in the data (creating a 240 by 49 chromatin mark matrix for each HiC data matrix slice).

Training data generation and neural network training

To generate training data, we made matched slices of 80 10 kb bins of data of both the HiC and ChIP-Seq data matrices, following the paper. We did not include the edges of chromosomes (where 80 bins were not available). To be safe, we kept a buffer of 120 bins at the edges of chromosomes, whereas the paper used only 100 bins. We used the same neural network architecture as used in the source paper, with a 1D convolutional filter (with the same weights for each bin). The network was made, trained, and evaluated using Keras[73], exactly as in the paper (*MachineLearningOnDrosophilaData6.py*).

Acknowledgements

This work could not have been carried out without a generous amount of support from my daily supervisor Amin. Amin wowed me with his constant positive attitude, skill at making schematic drawings of even intricate machine-learning schemes, and humility. It is quite an education to work together with someone with such a different educational background on the same project. Amin, I have enjoyed working with you, even if I darned the project to heck for significant stretches of time.

I also wish to thank Jeroen de Ridder for giving me a place in his lab and for fruitful discussions about the project. I think you have great values and love the openness you aspire to in the lab. Even when I myself felt like I didn't quite belong, you made sure to include me in activities such as the group outing. Also, I finally have a correct book, so now I won't be needing the incorrect ones. Thanks!

As a final remark, the jargon-filled twin-like communication synergy that the two gentlemen named above can exhibit in conversations is remarkable. In 50 words and two minutes they can navigate more intricacies of the machine-learning approach at hand than a mere mortal can in thrice the time. Inspiring to watch and learn from for sure.

Thanks, too, to Geert Geeven and Valerio Bianchi for making available their chromatin loop data for use in this project. I couldn't have done any of it without them. On that note, I wish to thank Prof. Dr. de Laat for agreeing to be my second reviewer and for a fruitful discussion on HiC data classification and deep learning at the beginning of this internship.