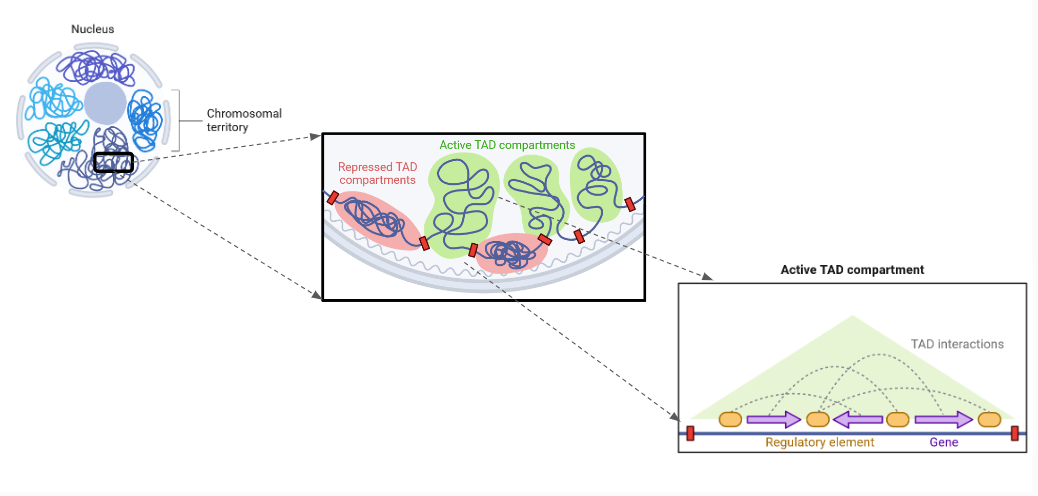
**Identifying Local TAD Differences Across Hi-C Data**

Sumitra Lele, Gabriel Fonseca, Simon Chow

1. **Introduction**

Epigenetic diseases are a cornerstone of novel biological research. Studying different open histone chromatin regions in cancer cells can allow for identification of implicated chromosomal regions. Topologically Associated Domains (TADs) group regions of the chromosome by chromatin interaction, a key epigenetic process that regulates gene expression. Using this idea, comparing TADs across different experimental conditions (namely, nasopharyngeal cancer vs healthy cells) can suggest specific regions of DNA implicated in disease (cancer). The goal of the project is therefore to study chromosomal similarity via a variety of computational methods. We attempted to implement a method from Natalie Sauerwald and Carl Kingsford’s paper titled - “Quantifying the similarity of topological domains across normal and cancer human cell types”. The authors introduce a novel computational approach for comparing TADs across experimental conditions.

**Biological Background: Hi-C Analysis & TADs**

**

*Figure 1a - Overview of TADs (Courtesy: Biorender)*

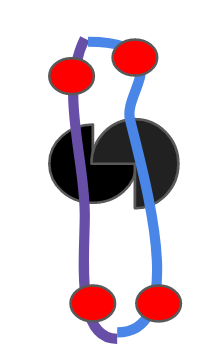
Significance

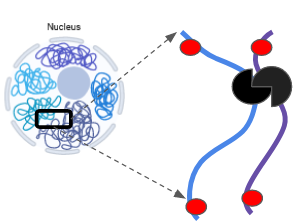
Through various modes of biological data, one can find a lot of answers regarding the biological state of a given specimen. Transcriptomics data tells us which gene is being misregulated. Epigenomic data might give us an idea about how the misregulation is happening. However, there is no modality that helps one posit a causal relationship between the sub-cellular processes and the biological state. This is where understanding the chromatin structure becomes important - it tells us why a gene is being expressed the way it is, or why the non-sequence modifications on the chromatin are the way they are. Hi-C data tells us about the ultimate governing forces in the nucleus: the chromatin interactions. Learning about 3D genomics and performing Hi-C analysis will help us better understand the 3D chromosome structure, allowing us to analyze and compare across different samples. This shines light upon the intricacies of genetic processes and helps with the understanding and prediction of the relationships between specific diseases and DNA itself.

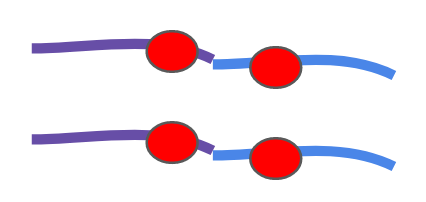
Hi-C Matrices

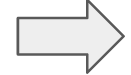
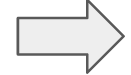
Hi-C matrices are two-dimensional matrices, where each row represents the extent of interaction between the corresponding x and y coordinates on the chromosome. Hi-C matrices contain data that reveal which regions of the chromosome exhibit relatively higher interaction. This is useful for understanding where chromosome regions interact with each other. These Hi-C datasets can be compared across different samples to ultimately provide specific insights into differences in gene regulation and expression across the samples. These matrices are visually presented as a heatmap: a two-by-two matrix where each square is a value from a x-y pair in the Hi-C matrix. Heatmaps show clear patterns of interaction along the diagonal, suggesting most chromatin interactions are local. These interactions can represent various sub-compartments of the chromosome - TADs, A/B compartments, loops etc. However, like most other sequencing techniques, this is also known to have a lot of false positive interactions. TADs help to resolve this issue by reducing the precision of the analysis, such that only clear divergence in chromosome structure is significant [1].

Hi-C Data Collection

Hi-C data collection is a process that can broadly be divided into two parts - isolating the interacting regions and then performing paired-end sequencing. The interacting regions are isolated by first crosslinking the interacting regions, usually using formaldehyde, then adding biotin beads along the rest of the chromosome. These cross-linked chromosomes are then cut around the biotin beads using restriction enzymes.







*Fig 1b: Schematic representation of isolated cross-linked chromosomes with the red circles as the biotin beads being digested, ligated and sequenced*

Then, these regions are ligated on either end, the cross-linking is removed and these paired ends are sequenced. There are many variants of Hi-C which make changes at different stages of this process.

Topologically Associating Domains (TADs)

Topologically associating domains (TADs) are a crucial part of Hi-C analysis. These TADs are specific intervals in the chromosome that interact more with other chromosomal regions compared to the non-TAD intervals. TADs also follow a hierarchical structure, meaning that TAD intervals may contain nested TADs (sub-intervals) within them. These groupings are formed based on the TAD intervals’ specific chromatin interaction patterns. On the Hi-C heatmap, we can see TADs as squares along the diagonal.

1. **Data**

The dataset we used came from a study of Epstein-Barr virus (EPV) related nasopharyngeal carcinoma [2]. They used Hi-C sequencing, RNA Sequencing and Chip-Seq to measure chromatin structure with the introduction of the EPV virus. They wanted to validate and explore their hypothesis - Epigenomic rewiring and enhancer dysregulation were driving the cancer progression. They found that there were apparent TAD boundary disruptions between the two datasets, and that the virus genome was contributing to pathogenesis through enhanced infestation, i.e., by interacting with inactive B compartments of the genome and activating the enhancers within them randomly. The datasets used for this project were therefore two Hi-C matrices: one from a control cell line, NP69T\_Hi-C (GSM7042526), and one derived from a EPV-induced nasopharyngeal carcinoma cell line, C666-1\_Hi-C (GSM7042527).

Initial Challenges with Hi-C Data Reading

One of the first challenges encountered during the project was reading in our downloaded Hi-C matrices. Considering that Hi-C matrices cover the interaction levels between regions across the entire genome, these are large, encrypted files that cannot be directly read and processed without specific decompression algorithms. As such, it was necessary to utilize specialized packages that enable the reading of a specific chromosome as a sparse matrix (leaving out the data where no interactivity was observed) to make the Hi-C data small enough to read.

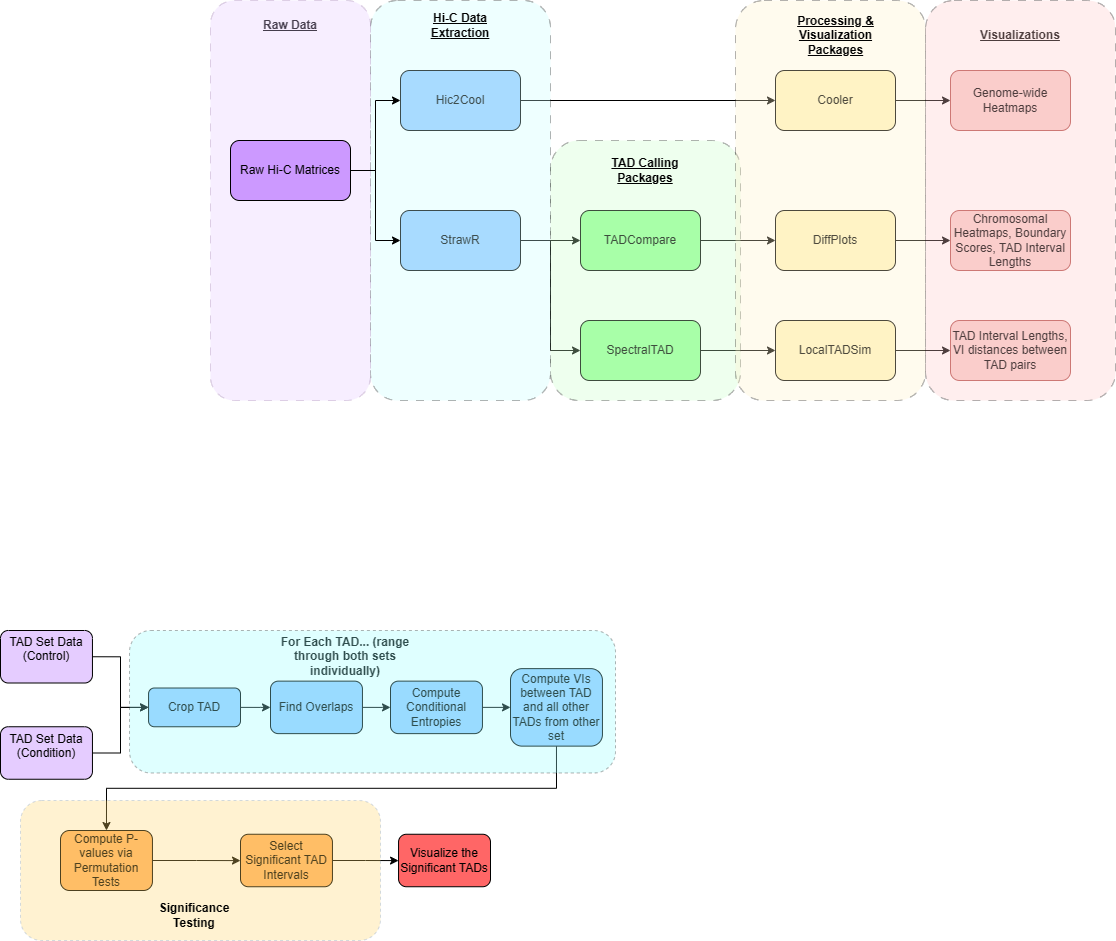
Solution: StrawR package

To perform this task, experiments with installing and using multiple packages, such as Juicer and HiCExplorer were done. We found success in this regard by using the StrawR package. It was able to successfully output the HiC data for a specific chromosome, with a specified bin-size parameter (for the size of an interval of base pairs). For our analyses, the chosen chromosomal resolution was 10000bp. That is, each datapoint in the HiC matrix is 10000bp apart on the chromosome. We also used HiC2Cool to directly read Hi-C matrices directly in Python, where we implemented the paper’s TAD similarity algorithm.

1. **Methods**

Methods Overview

After extracting the Hi-C data using StrawR and HiC2Cool, the next step is TAD calling, which identifies the TAD intervals along the chromosome(s) of interest.

****

*Figure 2 - Main High-Level Workflow: Loading and processing of two Hi-C data samples, followed by multiple methods of TAD calling and visualizations of comparing sets of TADs between the two samples*

1. TAD Calling

For TAD calling, the computational problem is as follows:

| **TAD Calling Computational Problem**  Input: a contact matrix *ContactMatrix*, where *ContactMatrix[i][j]* is a float representing the amount of interaction between genomic positions i and j  Output: a list of integers TAD Boundaries representing the genomic positions where the amount of contact is locally minimized |
| --- |

For TAD calling, we utilized the SpectralTAD R package. At a high level, the SpectralTAD software utilizes a Laplacian matrix to locate chromosomal regions of significant difference for each sliding window in the chromosome. Spectral decomposition, or eigenvalue decomposition is then used to determine which windows have significantly more interactivity, leading to the output of TAD intervals. These resulting TADs are then outputted as coordinates in a BED file. After obtaining the TAD intervals, the next step would be to compare the set of TADs across different samples, which is the primary goal of this project.

1. TAD Comparison

| **TAD Comparison Computational Problem**  Input: a list of integers *TADBoundariesControl* and *TADBoundariesExperimental* representing the genomic locations separating TADS from the control and experimental samples  Output: a list of floats *DifferentialContact* where *DifferentialContact*[i] represents the similarity between TAD boundaries from *TADBoundariesControl* and *TADBoundariesExperimental* at genomic position *i*. |
| --- |

TAD comparison can be done in various ways, depending on the desired similarity metrics. The choice of the method of comparison depends entirely on the representation of the Hi-C matrix. One can either use the two-dimensional matrix as is and perform a “pixel-by-pixel” comparison, or compare the partitions of the Hi-C matrix by a variety of different transformations.

The two methods that we have used to compare the TAD similarity quantification use the second comparison category. TADCompare[3] converts the Hi-C matrix to an adjacency list, gets the Eigenvector gaps and compares that value between the two datasets to find boundary disruptions. LocalTADSim[1], calculates the variance of information based on the cluster sizes and intersections, and finds the significant hits using a permutation test.

There are many other ways of comparing TAD boundaries as well. A very widely used method is - Stratum-Adjusted Correlation Coefficient. In this method, the Hi-C matrix is stratified according to the genomic bins and then these stratified subparts are compared using some similarity metric, mainly Pearson’s correlation [4]. Another method of comparison first pre-processes the matrices, using a mean filter, to handle the false positives, and then takes the difference of the logarithm of both the matrices. This is done based on the fact that the decay of the contacts between the clusters is exponential with respect to the genomic bins [5].

Here, the main objective is to utilize multiple different TAD comparison methods and evaluate their results against one another.

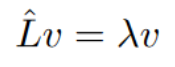
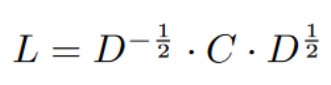
The method implementations we focus on here are:

* TADCompare (R package)
* LocalTADSim (our implementation of Sauerwald and Kingsford’s paper)

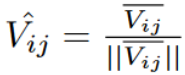
TADCompare method overview

The TADCompare package produces a graph representation of the Hi-C interaction matrix, which shows us the quantified contact level between region ‘i’ and region ‘j’. Essentially, this is a graph adjacency matrix where the vertices are the genomic regions, edges represent contacts, and the edge weights represent contact amount.

Once TADCompare has the adjacency list, it gets the Laplacian *L*, to get regions of high gradient. Once it gets the Laplacian, it performs Eigen decomposition, producing . After this, it gets the Eigenvalue gap, which can be thought of as a normalized difference between the eigenvectors, where each eigenvector represents the variance in the contact map. After this step, the eigenvector gaps are used to identify the regions containing TAD boundaries. One can visualize this as a graph of spikes, where a TAD boundary is represented by a large spike. Smaller spikes of intensity are named the eigenvector gaps , where we keep the top two eigenvectors for each. These gaps represent boundaries between TADs. On the basis of the eigenvector gap values, it calculates the boundary scores, B, using the formula below, where D is the Euclidean distance between the corresponding eigenvectors of adjacent rows. The boundary score is calculated by getting a distribution of the values of D. For comparing two samples, the boundary scores between the two samples are compared.



*Formula to get the Laplacian and then perform the eigenvalue decomposition*

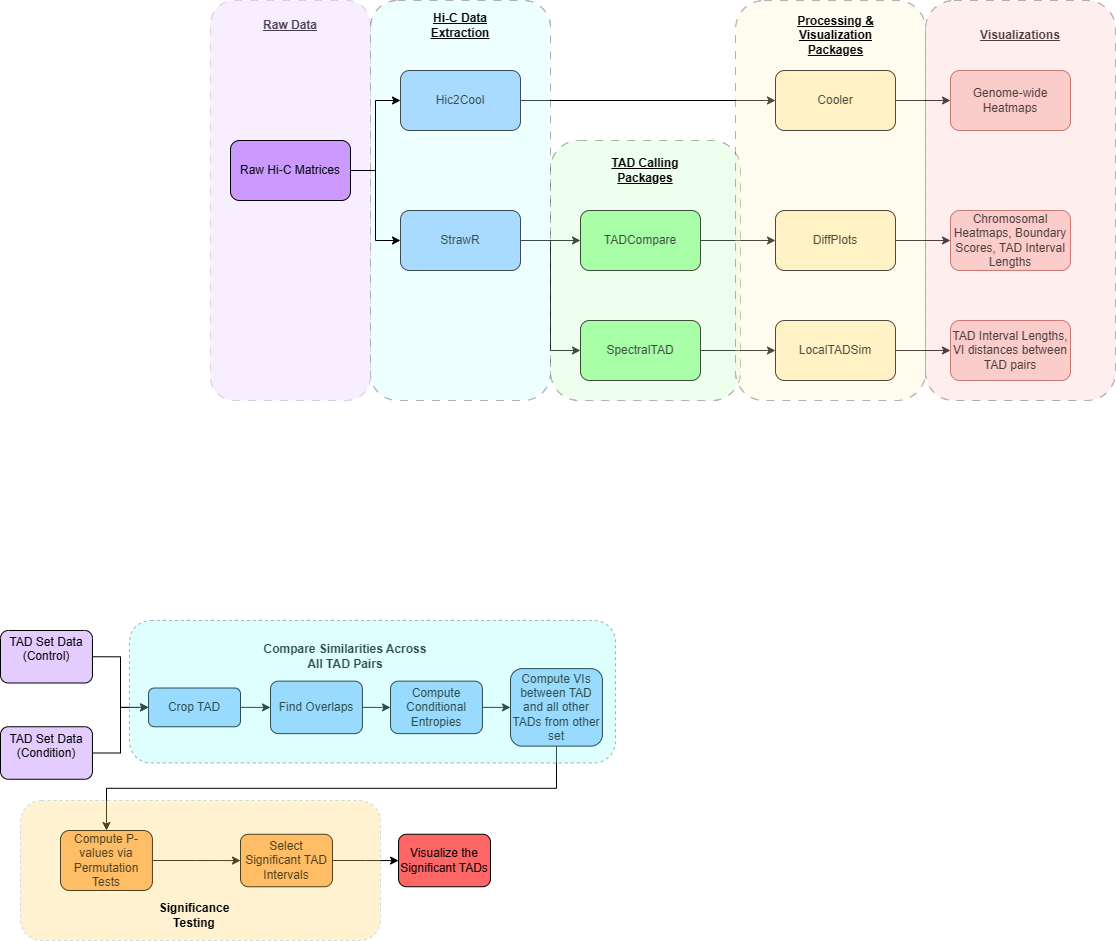


*Formula to get the eigenvector gaps*  ( *) and boundary scores (B)*

With many tests performed to compute the p-value, correcting for our p-value is important, regardless of the TAD comparison method. Without p-value correction, TAD intervals may be falsely deemed significant. The Benjamini-Hochberg correction adjusts p-values, improving the confidence of the significance analysis. After correcting p-values, the standard 5% significance threshold is used to determine whether the TAD is actually significant.

LocalTADSim method overview

The TADCompare implementation serves as comparators for evaluating our LocalTADSim implementation and its TAD results.

****

*Figure 3 - LocalTADSim Workflow:* *Compute VIs between TAD pairs, followed by significance testing to determine the significant TAD intervals between control and condition samples*

The LocalTADSim implementation starts off with the resulting data files produced by SpectralTAD[7]: the TAD intervals for the control and condition samples (healthy vs cancer cells). The first stage of this implementation is to go through the TAD intervals between both sets (control and condition), and compute how ‘similar’ each pair of TADs across both sets are.

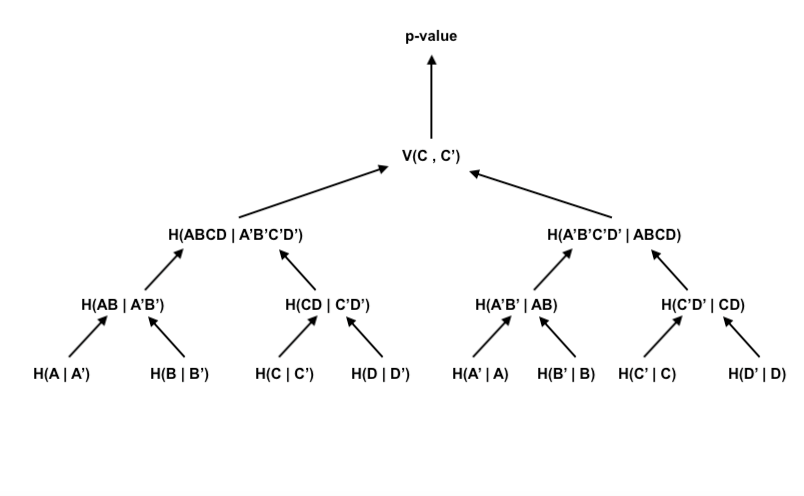
*Compute ‘Distances’ Between TAD Pairs*

To establish how ‘similar’ each pair of TADs are, the distance metric --variation of information (VI) -- is calculated for each of these pairings. The VI metric is based off of a normalized sum of the conditional entropies regarding each of the TADs in the current TAD pair being evaluated. Obtaining the overlapping base pair regions between the TAD pairs is key to computing the conditional entropies for each of our TAD sets. The TAD set is a partition of the chromosome. For example, if the chromosome is C, then the partition is where each is an interval in the chromosome (i.e. the end of the tad). Then the VI metric is defined:

Where:

Note that the conditional entropy between two clusters depends only on the size of each cluster. Furthermore, if the intersection of the two TADs does not exist, their VI distance is essentially infinite (the denominator in the log term is zero, if we define the size of the intersection as 0). To circumvent this problem, we only compared TADs which have an overlap, to compute the overall VI of each chromosome subset.

This implies a relatively simple procedure: compute the VI for each possible overlapping region in the chromosome - That is, regions with the same ending location for both condition and control (i.e. the clustering C is control and the clustering C’ is the condition). In the paper, a dynamic programming algorithm was suggested that uses partial VI computations within a chromosome subset, to allow for less redundant computation (recall that the current definition of VI requires computing pairwise distances of each TAD). At a high level, the computation graph is as follows:



*Figure 5 - Dynamic Programming Calculations for the Conditional Entropies to Compute the VIs*

The detailed equations can be found in section 2.3 of the paper [1]. The implementation of this part of the code is not necessary for the correctness of the algorithm, but it saves computation time. Essentially, computing VIs for larger subintervals depends on the VI of subintervals, so it saves the VIs of subintervals in memory to speed up computation. It also takes into account if the two subintervals overlap or not, adjusting the size of each TAD to account for this. For our implementation, we used dynamic programming to get the initial VIs, but switched to the brute force approach to perform the permutation tests, since these lead to issues with making multiple group members’ code compatible.

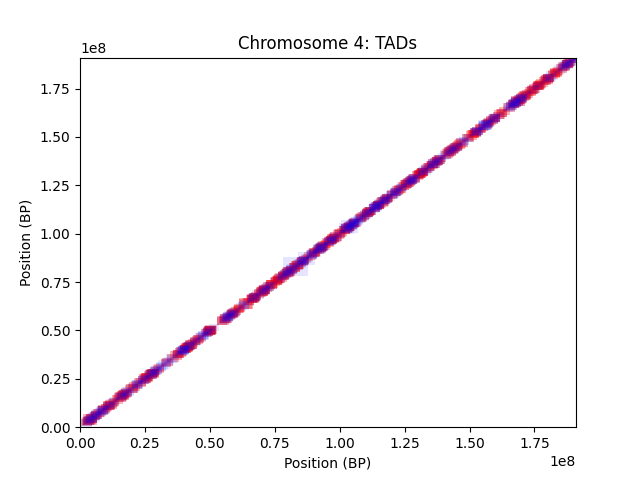
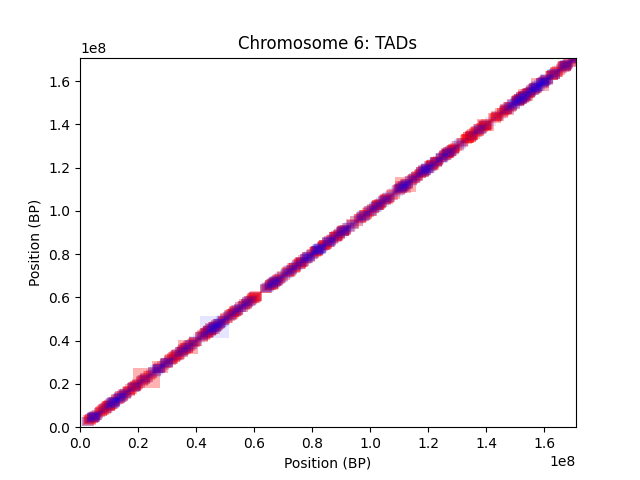
*Significance Testing*

Once the VI values have been computed for the TAD pairs, the next stage is to determine which of these intervals are actually significantly different between the samples.

To determine the statistical significance, a permutation test is deployed regarding the current TAD pair. First, one TAD is kept constant while the other TAD’s sub intervals are randomly shuffled around for a thousand times (leading to one thousand permutations). The VI is then recalculated between the constant TAD and each permutation of the other TAD. This forms a list of one thousand new VI scores. Next, the fraction of VI scores that is greater than the original VI score (between the current TAD pair) is computed and becomes one of the two initial p-values for the current TAD pair. Next, the other TAD of the pair is kept constant while permutations are run through the first TAD. Then, the final p-value for the current TAD pair is the average of the two initial p-values. Considering the p-value represents the probability of a scenario occurring by random chance, the random permutations represent the random chance scenarios and the count of VI scores greater than the original exemplify cases where that VI score is surpassed by chance.

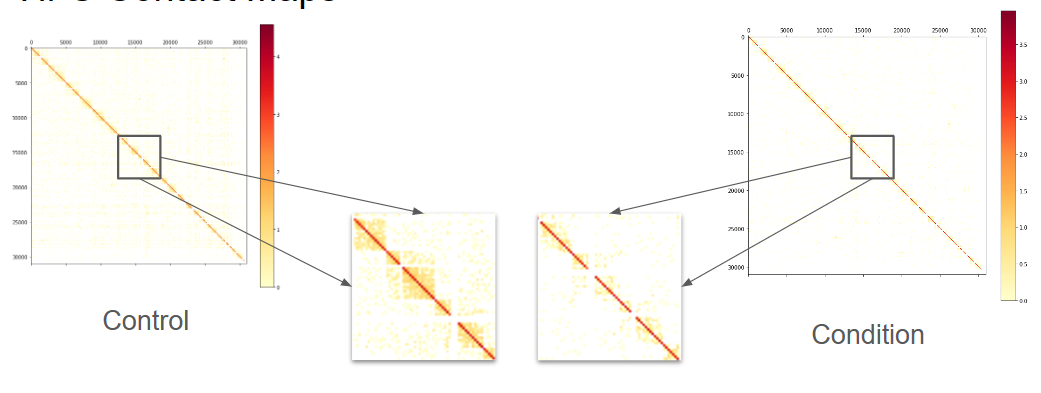
Once the p-values are computed for each TAD pair, the Benjamini-Hochberg method is used to correct the p-values via controlling the false discovery rate. From there, the standard significance threshold of a 0.05 p-value is used to determine the significantly different TAD regions between the two samples.

1. **Results**



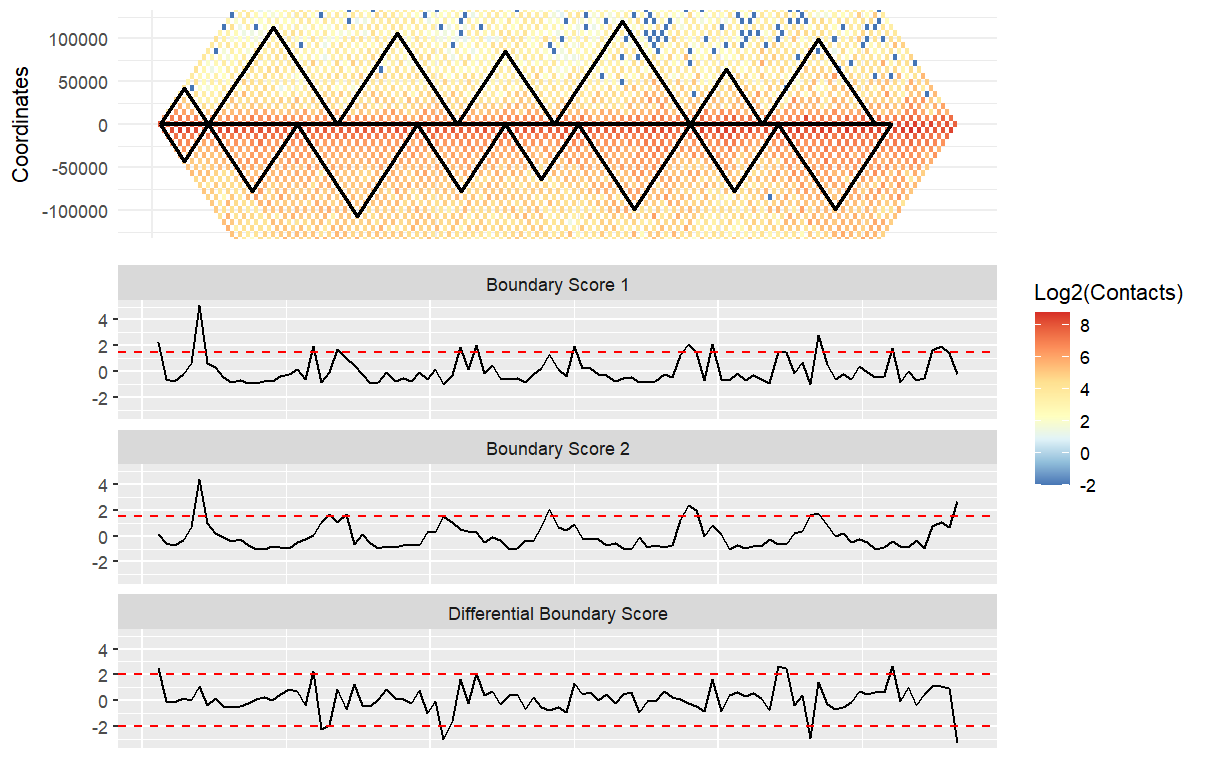
*Figure 6 - Spectral TAD*

The plots from SpectralTAD’s TAD genomic coordinates are displayed above. While many of the TADs seem to overlap, there are still specific regions where the control and condition samples exhibit differences in TAD sizes. Interestingly, these disparities are more numerous and noticeable in chromosome 6 as compared to chromosome 4, indicating that chromosome 6 may be a stronger indicator of this disease.



*Figure 7 - Cooler[6] Heatmaps*

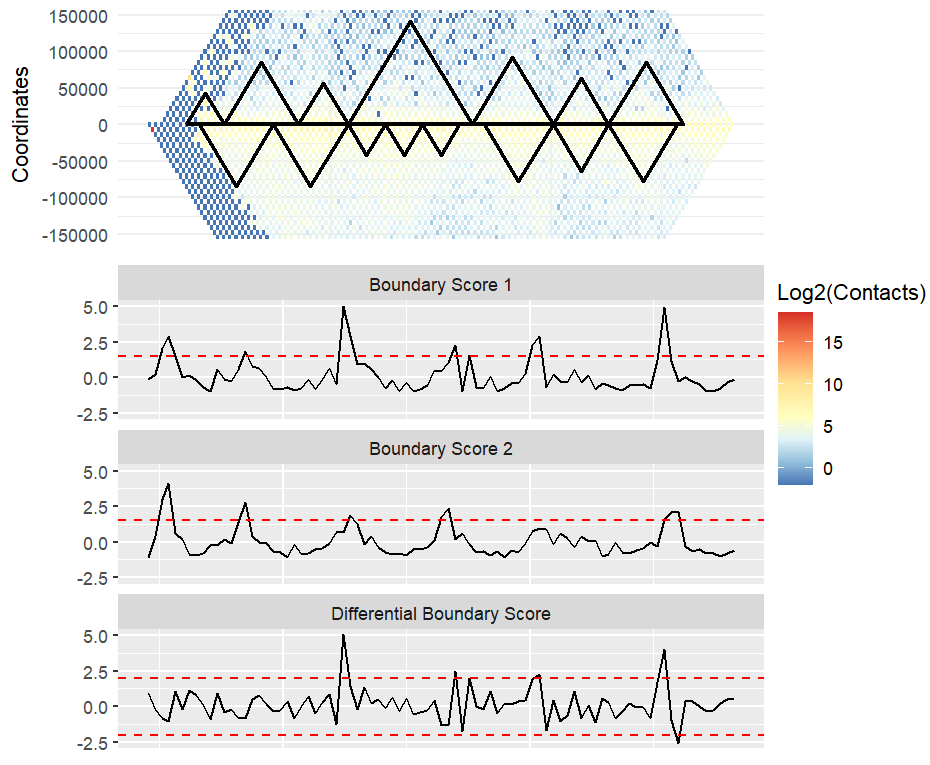
These heatmaps are generated from the Cooler Python package for the control and nasopharyngeal carcinoma cell lines. With a closer look into these heatmaps, the control genome appears to exhibit larger regions with relatively high interactivity. In other words, it is possible that the overall reduced interactions between the different chromosomal regions may play a significant factor in the identification of the cancer.

**

*Figure 8a - TADCompare Plot (Chr 4)*

*Control - Boundary Score 1*

*Condition - Boundary Score 2*

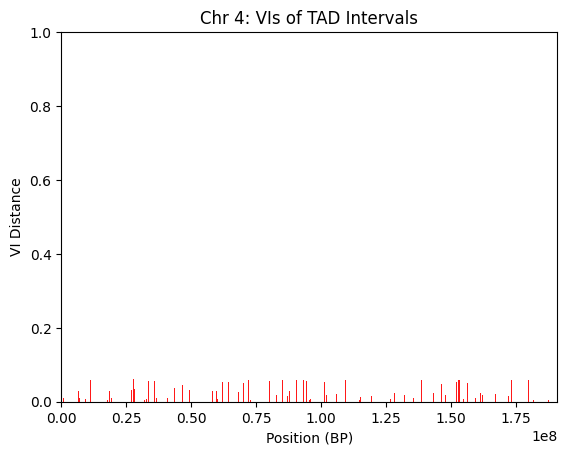


*Figure 8b - TADCompare Plot (Chr 6)*

*Control - Boundary Score 1*

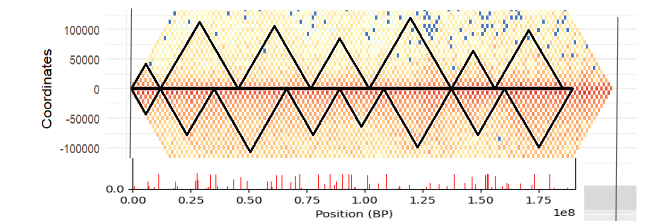
*Condition - Boundary Score 2*

These are the TAD difference plots generated from the TADCompare package. These plots provide a diverse and condensed amount of information, and are composed of three main components. The first is the heatmap that shows the interaction between the TAD regions of the chromosome. Second is the black lines along the middle of the heatmap that represent the TAD intervals for the control and condition samples, the top and bottom triangles, respectively. Lastly, the individual and differential boundary scores are provided below the heatmap. These highlight where TAD disruptions are located. Large amplitudes for the differential boundary score correspond with the TAD differences between the two samples.

****

*Figure 9 - Significant TAD Intervals Along Chromosome 4*

This figure highlights the regions along chromosome 4 (x-axis) where the significantly different TAD intervals between the two samples lie. The y-axis signifies the extent to which these TAD intervals differ between the control and condition cases (VI distance). The plot depicts numerous significant TADs along the chromosome, though there appears to be fewer of them at the starts and ends of the chromosome, as well as around the 125 million base pair position.



*Figure 10 - Quick Glance at Comparing TADCompare (top) and LocalTADSim Results (bottom)*

Although a more formal analysis directly comparing TADCompare and LocalTADSim results will be discussed in the ‘next steps’ section, here is a quick glance at comparing the two methods’ results. First, in Figure 10, it’s clear that there seem to be much fewer and larger TADs from the TADCompare plot (top) than the LocalTADSim plot (bottom); this is likely due to that plot using the top level of the TAD hierarchy, relative to the TAD intervals used in LocalTADSim. Aside from this, the start and ends of the chromosome exhibit similar TAD structures in the TADCompare plot, which is why the LocalTADSim plot has red marks at the ends of the plot (red marks represent where TADs exhibit significant difference). Notably, the middle of the chromosome -roughly base pair positions 65,000,000 to 110,000,000- appear significantly different in the TADCompare plot and is reflected by the relatively numerous red marks in the middle portion of the LocalTADSim plot.

Nonetheless, this is just a quick glance at putting these plots next to each other. More rigorous techniques and algorithms should also be explored to further understand and compare the performances of these two methods.

1. **Discussion**

Overall, the TAD comparison methods were successful in identifying differences across the TAD boundaries between the normal and condition samples.

Limitations

One limitation of the project is the comparison of significant subsets of the genome across different TAD comparison methods. Since the TAD comparison utilizes different methods, they may lead to different p-values across the genome. Specific to TAD similarity, our implementation is relatively slow, meaning the permutation test may lead to slow runtimes (note that the VI for each interval is calculated 1000 times). We could speed this up by integrating the dynamic programming formulation into the permutation test. Lastly, the original paper used the VI metric to find regions of similarity in the genome, but we are interested in TAD disruptions between healthy and cancer cells. As such, we made modifications to the permutation test, finding regions with VI greater than the average (rather than smaller). This seems intuitive, but we did not provide a proof for this.

Next Steps

We have mainly performed TAD calling and comparisons on the first few chromosomes due to high computational runtimes. As such, one clear follow-up task is to complete the TAD comparison pipeline for all the remaining chromosomes from the samples.

Following through that, we can perform our TAD comparison pipelines on other, larger Hi-C datasets to evaluate and compare their performances. We can also track and report runtimes for the different method implementations and directly evaluate their speeds relative to each other, providing us an additional point of comparison between TADCompare and LocalTADSim.

Another approach for directly comparing the two methods would be to extract the TAD regions from the TADCompare implementation where there are significant differences between the normal and condition samples. Then, the significantly different TAD regions can be aligned from both methods to produce a similarity score.

This next step of our analyses brings us to the computational problem:

| **Directly Comparing Significantly Different TAD Regions Between TADCompare and LocalTADSim Methods**  Input: Two lists of start and end positions (integers) of the significantly different TAD regions, one list from TADCompare and the other from LocalTADSim  Output: Similarity score (float) resulting from the alignment of the two input lists of start end coordinates |
| --- |

Ultimately, these next steps will solidify our understanding of comparing TADs from multiple different methods and realizing their strengths and weaknesses for future TAD studies.

1. **Conclusion**

Throughout our project, we primarily focused on implementing the LocalTADSim method for comparing TADs and producing TAD comparison visualizations from the TADCompare R package. While we were able to finish our implementation to design significant TADs, more direct testing regarding the LocalTADSim and TADCompare methods should be done to help evaluate the effectiveness of these methods in comparing TADs. Overall, our takeaways from this project are as follows:

* Cooler’s heatmaps were great for understanding the size and interactivity of the TADs across all chromosomes of the chosen sample.
* The SpectralTAD[7] package proved to be a reliable and versatile way of calling TADs, working with various Hi-C matrix formats and outputting TAD intervals into a data table for convenient downstream analysis.
* TADCompare provided a lot of information regarding boundary scores and the locations of boundary disruptions as well as TAD intervals.
* Lastly, the LocalTADSim implementation was a fast and powerful way to compare TAD sets across samples. In particular, LocalTADSim relies on the VI metric for computing similarities of TAD-TAD pairs and capitalizes on robust statistical analyses to determine whether a TAD interval has a significant difference between samples.

Ultimately, understanding the efficacy and inner workings of different TAD comparison methods suggests tools necessary to analyze 3-dimensional chromosome structure. Of importance, this kind of analysis is crucial to understanding potential epigenetic causes of complex diseases.

**References**

1. Natalie Sauerwald, Carl Kingsford, Quantifying the similarity of topological domains across normal and cancer human cell types, Bioinformatics, Volume 34, Issue 13, July 2018, Pages i475–i483, <https://doi.org/10.1093/bioinformatics/bty265>
2. Mizokami H, Okabe A, Choudhary R, Mima M, Saeda K, Fukuyo M, Rahmutulla B, Seki M, Goh BC, Kondo S, Dochi H, Moriyama-Kita M, Misawa K, Hanazawa T, Tan P, Yoshizaki T, Fullwood MJ, Kaneda A. Enhancer infestation drives tumorigenic activation of inactive B compartment in Epstein-Barr virus-positive nasopharyngeal carcinoma. EBioMedicine. 2024 Apr;102:105057. doi: 10.1016/j.ebiom.2024.105057. Epub 2024 Mar 14. PMID: 38490101; PMCID: PMC10951899.
3. Cresswell KG and Dozmorov MG (2020) TADCompare: An R Package for Differential and Temporal Analysis of Topologically Associated Domains. *Front. Genet.* 11:158. doi: 10.3389/fgene.2020.00158
4. Yang T, Zhang F, Yardımcı GG, Song F, Hardison RC, Noble WS, Yue F, Li Q. HiCRep: assessing the reproducibility of Hi-C data using a stratum-adjusted correlation coefficient. Genome Res. 2017 Nov;27(11):1939-1949. doi: 10.1101/gr.220640.117. Epub 2017 Aug 30. PMID: 28855260; PMCID: PMC5668950.
5. Hua, D., Gu, M., Zhang, X. *et al.* DiffDomain enables identification of structurally reorganized topologically associating domains. *Nat Commun* **15**, 502 (2024). <https://doi.org/10.1038/s41467-024-44782-6>
6. Nezar Abdennur, Leonid A Mirny, Cooler: scalable storage for Hi-C data and other genomically labeled arrays, *Bioinformatics*, Volume 36, Issue 1, January 2020, Pages 311–316,<https://doi.org/10.1093/bioinformatics/btz540>
7. Cresswell, K.G., Stansfield, J.C. & Dozmorov, M.G. SpectralTAD: an R package for defining a hierarchy of topologically associated domains using spectral clustering. *BMC Bioinformatics* **21**, 319 (2020). https://doi.org/10.1186/s12859-020-03652-w

Remaining Key R and Python Packages Used For TAD Analyses

* HiC2Cool: <https://pypi.org/project/hic2cool/>
* StrawR: <https://cran.r-project.org/web//packages/strawr/strawr.pdf>