Using Single-Cell HiC to Identify Cell Type and Cell Cycle Phase Distinctions using Unsupervised and Supervised Techniques

by Keya Kilachand

Advisor: Ritambhara Singh
Reader: Stephen Bach

A Thesis submitted in partial fulfillment of the requirements for Honors in the Department of Computer Science at Brown University

Department of Computer Science
Brown University
Providence, RI
May 2025

Abstract

In this investigation, we examine if it is possible to simultaneously capture cell-type and cell-cycle phase structures from single-cell HiC (scHiC) data without extracting any features or using supplemental information. Detection and analysis of these structures can uncover insights into each cell's behavior, including cellular aging, gene regulation and gene expression. These insights can enable detection of novel drug targets, how cells proliferate and differentiate, and how specific diseases develop (Van de Sande et al 496). The scHiC data modality examines one cell at a time, allowing the study of cell-to-cell variability in these structures. However, it is extremely noisy and sparse, making it challenging to extract meaningful information from this data alone. In order to overcome this, this investigation first applies preprocessing techniques to raw scHiC contact maps - gaussian filtering, bilateral filtering, singular value decomposition and Eigenvalue decomposition - to impute the data. We then use the Higashi algorithm for imputation, and try to form unsupervised clusters based on cell-type and cell-cycle. This facilitates the finding that datasets with one axis of variability (cell-type alone, *or* cell-cycle phase alone) are much more prone to forming meaningful clusters than datasets with two (cell-type *and* cell-cycle phase). We then use a number of supervised learning models to test if the imputed structures are distinct enough for cell-type and cell-cycle phase classifications to be made. This yields poor results. As such, this paper comes to two conclusions. First, unsupervised clustering is only successful for data with one axis of variability. Second, neither CNNs nor simple models can accurately extract meaningful representations from single-cell Hi-C contact matrices. This can be due to insufficient preprocessing or incorrectly designed supervised models, both of which remain areas of improvement.

1 Introduction

‘Chromatin’ refers to long chains of genetic proteins found tightly packed in the nuclei of cells. It is arranged in a 3D structure that exposes and interacts different localized parts of the chromatin chain, known as genetic loci (Liu). This chromatin structure is unique to each cell, and is intimately linked to the various phases of the cell cycle. Changes in the interactions between different regions of each chromosome decide which genes will be transcribed, and how larger molecules interact with one another in each phase of the cell cycle. This enables distinct functional phenotypes for the same genetic encoding, which in turn allows for cell differentiation and proliferation (Ma et al).

Cells spend most of their time in a state called interphase, during which they grow and replicate their chromosomes for division during the phase of the cell cycle called mitosis (National Human Genome Research Institute) For the purpose of this exploration, interphase will be further sub-categorized as follows: G1 (cell growth), Early S (beginning of DNA replication), Mid S (intermediate stages of DNA replication), and Late S (completion of DNA replication) (Wu et al). The areas of DNA on each chromosome that can be accessed change as the cell cycle progresses, thus changing the regions of genetic material that are conducive for transcription (Cooper).

In addition, the internal organisation of chromosomes changes as well - allowing consistent high-level patterns to be identified. (Naumova 948) Evidence for this changing architecture is seen in fluctuating histone activity and the differential presence of proteins in different phases of the cell cycle. These work together to change chromosomal shape, thus enabling cell cycle mechanisms and the transcription of necessary genes and proteins (Cooper).

Thus, there is strong theoretical evidence that three dimensional chromatin structures should change dynamically through the different phases of the cell cycle, and as cell differentiation occurs.

Existing literature has shown chromatin structure remodeling enables differential DNA replication, repair, and recombination that can enable mutations in cancerous tumours (Deribe et al). Furthermore, the continuation of cell cycling instead of terminal differentiation can provide strong evidence for whether a cell is cancerous or not (Caldon et al). This could potentially be identified through cell cycle phase prediction based on the observed chromatin architecture. In Tan et al, we see that studying cell-type specific gene expression could be used to discover new gene regulation circuits in diseased versus healthy biological systems, thus helping us identify diseased systems with higher accuracy, sooner (Tan et al). Thus, thorough analysis of scHi-C would enable a wide variety of practical applications such as the design of drug targets, and how to make them more effective. It can also inform the study of how cells change state, from healthy

to diseased for example, and the changes to internal chromatin structure that occur during this process.

Initially, genome wide 3D structural features were studied using the HiC protocol introduced by Liberman et al. Hi-C imaging involves gauging long range interactions between pairs of genetic loci by space constrained ligation and locus specific amplification of genetic material (PCR). At a high level, DNA that is found physically close together in cells is cross linked, the cross linked section is cut using a restriction enzyme, the cut ends are biotinylated and ligated together. The resultant sample contains ligated fragments of DNA that were originally found close to each other in the nucleus, thus providing an indication of how they could have appeared within the larger chromatin structure. These interactions are then mapped in the form of a matrix. Each data point on this map indicates an interaction between the chromosomal region on the x axis and the chromosomal region on the y axis at that point on the map. This graphical representation of chromatin structure is referred to as a contact map (Liberman et al).

However, as described by Galitsyna, et al, the limitation of Hi-C protocol in its current form is that it requires tens of thousands of cells and pools the reads from all the cells in a single contact map. Thus it produces a contact map that captures the average structure of the cell population. The problem here is that the information about chromatin structure in each individual cell is not equivalent to that averaged across a whole population (Galitsyna et al). Gene expression is highly cell specific, and sees significant cell-to-cell variation. Studying samples of cells to observe this phenomenon is therefore challenging because cells in a population are often in different phases of the cell cycle. Signals that would allow the study of individual cell dynamics get masked out when averaged across the entire sample (Raj 877).

Nagano et al overcame this restriction by limiting the assay to one cell per reaction tube, thus proposing single cell Hi-C (scHi-C) in combination with genome wide statistical analysis. This protocol looks at cells individually, thus showing the differences in chromosomal conformation from cell to cell (Nagano et al). However, this approach too has shortcomings. First, it uses an immortalized cell line, which are cells derived from a single cell that is modified to divide infinitely and are therefore homogenous. Cell type information is uniform across the entire population, making this dataset less useful to prove that cell-type specific structures can be extracted from scHiC data.

Additionally, scHiC data is extremely sparse. A large number of potential interactions between genomic loci are not captured because of procedural limitations. These contact maps are extremely expensive to generate especially for rare cell types. Only about 1% of the total interactions are captured per cell on average, resulting in contact maps that are largely populated with zeros (Liu et al). As such, it is difficult to determine whether the sparsity of these maps is due to a genuine lack of interactions between genomic loci in the cell, or because of the

procedure's inability to detect these interactions (Zheng et al). The sparsity of these contact maps also makes them extremely noisy, which further hinders analysis. Models that learn from this raw data would focus excessively on the few interactions that are actually recorded, leading to potentially biased results. This in turn might cause it to overfit training data, and lose generalisability when asked to detect similar structures in unseen scHiC data. The sparsity of the maps combined with the noisiness of the data makes it difficult to detect meaningful patterns. Therefore, it is difficult to draw statistically reliable conclusions (Wang and Cheng).

More valuable insights into structural features like compartments and loops can be revealed when the data of multiple cells is available. In order to conduct reliable analyses, a common approach is to pseudo-bulk scHi-C dataset, where cells are aggregated based on cell type and the reads of multiple similar cells are embedded together in order to obtain more substantial results (Galitsyna and Gelfand).

Most existing methods address data sparsity by employing a process known as feature extraction, where unevenly distributed signals in the data are aggregated or supplemented with additional data in order to provide prediction models with better inputs. For example, contact distribution vs distance (CDD) feature extraction involves binning interactions based on the genomic distance between the regions involved, and generating a probability distribution for the number of interactions within each bin (Wu et al). Nagano et al show chromatin contact variability in different phases of the cell cycle, and Ye et al propose that this contact probability distribution across bins changes as the cell moves through different cell cycle phases. Thus, when making predictions the model can count the observed number of interactions in unseen data, and compare this to the probability of those interactions being seen in a specific phase of the cell cycle (Wu et al).

However, this risks introducing bias into prediction models. As observed by Ye et al, CDD feature representations fail to distinguish between Late S and Mid S phase cells. The contact probability distributions don't differ much between these two cell-cycle phases, and thus the model doesn't have much to rely on when trying to make the distinction (Ye et al). This highlights an example where domain knowledge of changing chromatin structures across cell cycle phases misleads prediction models. Focussing on these expected changes could hinder the detection of underlying patterns in the data, which point to other evolutions in chromatin architecture through the cell cycle.

Feature engineering also can involve combining raw input data with other sources of information to make more informed decisions. Protein binding data (ChIP-seq) can be used in conjunction with scHiC data to point to regions that have a higher likelihood of contact based on higher protein activity. Similarly, chromatin availability data (ATAC-seq) can be used to detect regions of sparsity that have occurred because of a lack of chromatin interaction, as opposed to lack of

data (Tan et al). These techniques are intended to provide axes for machine learning or deep learning models to work along, in order to produce more meaningful and accurate results.

That said, introducing external sources of data as signals for the model to follow might bias it to look for them in unseen data. Using protein binding information to detect chromatin interactions might train a model to mistakenly overlook interactions that have less protein involvement (Meyer and Liu). Similarly, over-reliance on chromatin availability data to detect regions of sparsity might unintentionally overlook other structural features that cause the same phenomenon (Koohy et al). Thus, providing potential causes for structural features to appear in raw data might prevent the discovery of new unrelated patterns. Furthermore, batch effects or noise on the supplemental data could be amplified by the predictive model, further misleading its outputs. Additionally, as argued by Liu et al, this analysis focuses on looking for specific features of 3D genome structure as opposed to discovering features in the underlying data.

Therefore, this exploration intentionally refrains from feature extraction for fear of introducing bias into our prediction model, and addresses the sparsity of HiC data by attempting to amplify existing signals instead. This is done with the intention of maintaining the integrity of the data as much as possible, while simultaneously gleaning as much information from the contact maps as is feasible.

Zheng et al design a normalization method named ‘BandNorm’ towards reducing the noisiness of scHi-C data mentioned earlier (Zheng et al). They also created a deep generative modeling framework, scVI-3D, to correct scHi-C specific biases. In 2021, Li et al introduced a number of computational tools they refer to as ‘scHi-C Tools’. These tools can embed scHi-C data into a lower dimensional Euclidean space, allowing researchers to cluster cells based on similarities in their chromatin structure in a 2D or 3D scatter plot (Li et al). Thus, higher level patterns relating to the cell cycle and cell differentiation can be observed.

More recently, Liu et al has proposed HiRES, an experimental protocol that simultaneously profiles Hi-C data and RNA sequencing data. This allows for chromatin architecture and gene expression to be studied in parallel, which sheds lights onto complex interplay between structure and function. Their results show that chromatin interactions are closely related to transcription controls and cell function (Liu et al).

The Higashi algorithm proposed by Zhang, et al is a method of embedding the individual reads of a pseudo-bulk assay together. It uses hypergraph representation learning to identify latent correlations between single cells. The use of a hypergraph allows for signals from different modalities (chromatin versus mRNA) to be profiled simultaneously. This supplemental information is then incorporated to ‘impute’ or enhance the available scHi-C data, allowing for more robust contact maps. When vectorised in a lower dimension, this visualizes 3D structural

features such as topologically associating domains (TADs) and compartments. Both these structures refer to regions of the genome containing DNA sequences that interact more with each other than other regions, which in turn can inform how DNA folds (McArthur and Capra). If differences between these structures are observed for cells in different cell-cycle phases or of different cell types, we can glean a better understanding of how DNA changes as cells proliferate and differentiate.

Given the technical challenges of conducting scHi-C experiments, Murtaza et al has proposed deep learning framework scGrapHiC which extracts cell-type specific pseudo-bulk scHi-C contact maps from the bulk Hi-C contact maps using scRNA-seq as a guide signal. Unfortunately, pseudo-bulk scHi-C masks away crucial dynamicity in chromatin structure that manifests due to cell cycle phases. Hence, this paper extends scGrapHiC by using single cellular data directly instead of mapping pseudo bulked data to its single cellular state using scRNA-seq data.

2 Related Works

In ‘Circular Trajectory Reconstruction Uncovers Cell‐Cycle Progression and Regulatory Dynamics from Single‐Cell Hi‐C Maps’, Ye et al propose a tool to reconstruct circular trajectories of cells through the cell cycle called CIRCLET, to study changes in chromosome structure and organisation (Ye at al). The tool uses single cell HiC maps to generate four feature sets, capturing both global and local structural information. Higher-level patterns in chromosome organisation are extracted to infer structures that are characteristic of specific cell cycle phases, while at the local level specific interactions between genomic regions like topologically associated domains (TADs) and chromatin loops are identified to reconstruct cell-specific chromatin architecture. These feature sets are used as inputs to the model, which converts them into a high dimensional vector for each cell. As suggested by the Wishbone method (Setty et al), non-linear dimensionality reduction techniques are applied and the cells are organised into a k-nearest neighbours (kNN) graph, where each cell is a node and cells with more similar chromosome architecture are located closer together. This allows for major structural information to be captured while reducing noise. An arbitrary starting cell s is selected and an initial ordering of cells is determined based on structural similarity. This ordering is iteratively refined until a circular trajectory is found. The resultant trajectory shows cells in different phases of the cell cycle arranged in a circular trajectory based on the similarity of their chromosomal architecture, thus creating a model for what chromosomal changes could be observed as cells progress through the cell cycle.

The limitation of this model as described by Wu et al is that this trajectory alone is difficult to use to predict the cell cycle phase of an individual cell based on its chromatin structure. This would require the number of cells in each phase of the cell cycle to be known, such that the circular trajectory could be divided into phases based on the known number of cells in each (Wu

et al). However, the lack of datasets containing this information makes this task extremely difficult. Therefore, while CIRCLET is able to arrange cells sequentially into a pseudo-trajectory based on cell cycle information, it cannot be used alone to accurately predict which cell cycle phase a particular cell is in based on its chromatin structure.

Tan et al attempt to circumvent the use of HiC data altogether owing to how expensive and technically intensive it is. In 'Cell-type-specific prediction of 3D chromatin organization enables high-throughput in silico genetic screening' they propose a deep learning model called C.Origami, that takes in DNA sequences in conjunction with two cell-type-specific genomic features - CTCF binding and DNA accessibility - in order to predict changes in chromatin organisation in response to genetic changes. The model uses a convolutional encoder with 1D convolutional layers to extract features from the DNA sequences and the genomic features, and then condenses them together. It then uses a transformer to identify patterns in chromatin architecture that could affect gene regulation even if they are far apart in the genome. Last, it uses a decoder with 2D convolutional layers to identify spatial features and predict contact maps based on the same as output (Tan et al).

C.Origami is constructive to understand how 3D chromatin organisation can be captured from DNA sequencing data and genomic features. However, without explicit cell type or cell cycle phase labels in its dataset, it is unable to provide insight into whether structural features are observed due to cell identity or because of other factors. Furthermore, the contact maps it produces are static, and therefore are unable to capture dynamic changes in chromatin architecture with changing biological contexts. Lastly, the contact maps produced are not assigned cell cycle phase labels, thus preventing any insight into chromatin structure evolution through different phases of the cell cycle.

The most recent paper published on the subject in 2024 has an objective that is the most synonymous with that of this paper. In 'scHiCyclePred: a deep learning framework for predicting cell cycle phases from single-cell Hi-C data using multi-scale interaction information', Wu et al propose scHiCPred, a deep learning model that is able to extract features from scHiC data and use it to predict the cell cycle phase of the input cells (Wu et al). The study leverages the findings in Nagano et al to conclude that there is high cell-to-cell variability in chromosome structure, thus identifying the need for high resolution feature extraction methods. It therefore uses Small Intra-Domain Contact Probability (SICP) to detect phase-specific chromosomal interactions at an extremely refined, small scale. Furthermore, it builds on Ye et al's principle of multi scale feature extraction by combining SICPs with Bin Contact Probability (BCP) and Contact Distribution vs Distance to capture localised and globalised structural features respectively to predict cell cycle phases. Thus, after feature extraction this model employs a convolutional neural network to predict cell cycle phases of individual cells.

However, scHiCyclePred outputs only cell cycle phases from scHiC inputs, without making any inferences about cell-type-specific features. This investigation aims to build on this model to preserve cell-type-specific features while extracting those conducive to cell cycle prediction, in order to extract as much useful information from scHiC data as possible.

Furthermore, as previously mentioned, this investigation explicitly refrains from most of the feature extraction or feature engineering techniques employed by the aforementioned studies in an attempt to discover if meaningful chromosome organisation patterns can be detected while preserving the integrity of the raw input data as much as possible.

3 Methods and Materials

3.1 Data

Datasets commonly used for benchmarking purposes by single-cell HiC studies include those from Ramani et al's 'Massively multiplex single-cell Hi-C', (Ramani et al) and Liu et al's 'Linking genome structures to functions by simultaneous single-cell Hi-C and RNA-seq' (Liu et al). Ramani et al contains 4 cell lines from human cancer tissue samples, synchronized and arrested in a specific cell cycle phase, with 620 cells in total. The homogeneity of each cell line combined with the limited number of total cells sequenced makes this a relatively simple dataset. Liu et al on the other hand is significantly more complex. It contains thousands of cells from developing mouse embryos, with variance across cell type and cell cycle phase. This allows for a more holistic study of chromatin architecture and gene activity in dynamic biological systems. Therefore, the Liu et al dataset was chosen for this investigation.

3.2 Non-Algorithmic Data Amplification

3.2.1 Data Preparation

The Liu et al dataset was downloaded from NCBI GEO (accession number GSE223917) in pairs format. Each cell had its own file, containing base-pair (bp) interactions across 20 chromosomes. The total number of base pairs on each chromosome, present in the file header, was used to initialise a matrix that would serve as a contact map. For ease of computation, this number was converted to megabase pairs (Mbp) by dividing it by 1,000,000. As such, for N total base-pairs in a given chromosome, its contact map M would be of size $n \times n$ where $n = N/1000000$. Interaction data was organised into 4 columns - $chr1,\ pos1,\ chr2,\ pos2$. The $pos1$ and $pos2$ columns listed genomic loci between which interactions had been detected, and $chr1$ and $chr2$ detailed which chromosomes these loci were found on. $pos1$ and $pos2$ were also listed in base-pair format, and therefore needed to be converted to Mbps for compatibility with the contact map.

Contact maps were constructed to record inter and intrachromosomal interactions for only one chromosome at a time, due to the large size of the data. Each row of the data represented a pair of genomic loci (i, j) between which an interaction had been detected, and the number of occurrences of each unique (i, j) pair was counted. This count was used to populate the contact map A such that $A_{i,j}$ was the number of contacts detected between position i and position j on the chromosome. By this construction, the diagonal signified interactions between genomic loci and themselves. Therefore it was removed prior to further analysis. Signals providing insight into DNA folding and therefore those relevant for analysis were non-zero ones outside the diagonal. The map was finally checked for symmetry before any data imputation.

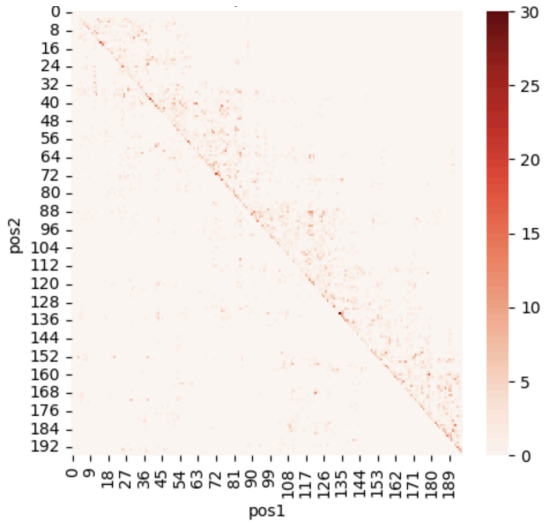

Figure 1: Contact map of a single cell in the Liu et al dataset, visualised as a heatmap

The Ramani et al dataset was downloaded from the Higashi tutorial (Zhang et al) and processed in the same manner.

3.2.2 Normalization

Three different normalization methods were used to reduce noise and decrease outlier signals. First, 0-1 normalization was used to see if any low intensity signals were being overshadowed. By forcing the interaction counts into a more contained range, lower intensity values would hopefully become more visible. Z-score normalization was used to decrease the effects of differences in sequencing depth (the number of times each genomic bin was sequenced), and to enhance the signal to noise ratio. It was also used to ensure features were on a comparable scale before dimensionality reduction techniques were applied, to moderate any deceptively large or small signals. Dimensionality reduction techniques typically look for features that account for the most variability in the dataset, and extreme values might mislead this analysis (Scikit Learn) Last, library size normalization was used for the same reasons as z-score normalization, with the added benefit of being able to conserve the distribution of the count data. The total number of

reads in each contact map was chosen as the library-size value. (Bioconductor). To enable better RGB visualization, and to amplify near 0 signals, the library-size normalized data was scaled by 255 and by 1,000,000 respectively.

As can be seen in figures 2 and 3, these normalization techniques did not visually change the data much. No distinct features were observed, nor any signals amplified enough to make a visual difference likely because of the sparsity of the data. Therefore, a more involved technique was needed to overcome this challenge.

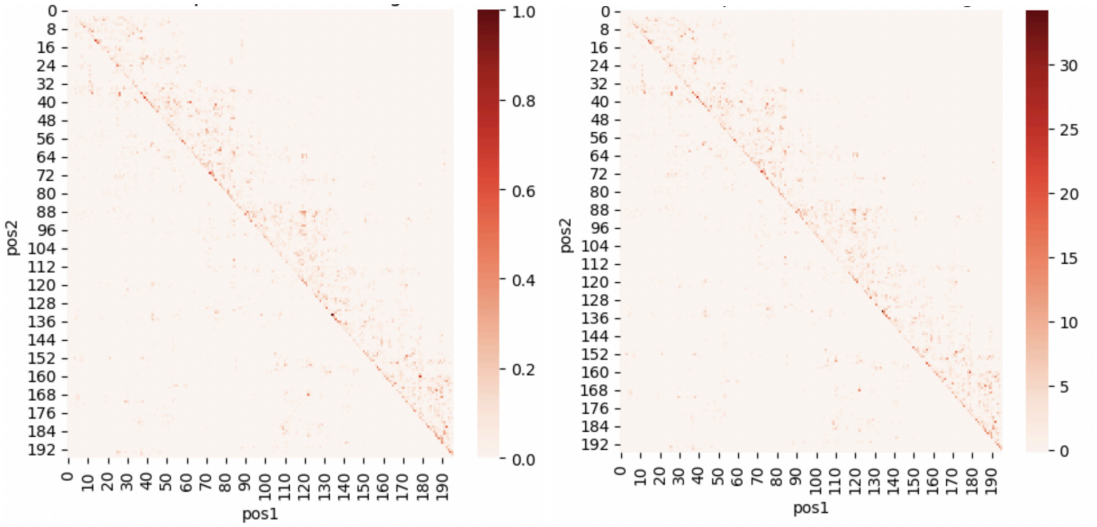

Figure 2: (Left) Contact map after 0-1 normalization (Right) Contact map after Z-score normalization

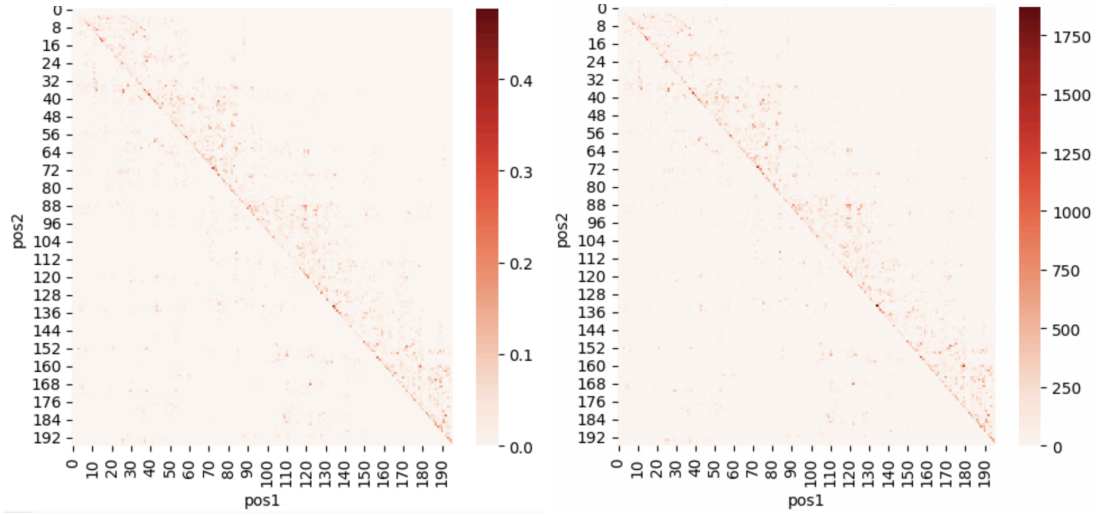

Figure 3: (Left) Contact map after library size normalization scaled by 255 (Right) Contact map after library size normalization scaled by 1000000

3.2.3 Gaussian Filtering

The sparsity of scHiC data is often due to experimental limitations that prevent certain interactions from being captured and recorded. As such, weak interactions may be present in regions that are recorded as 0s. Zhou et al propose linear convolution to resolve this. Given that the genome is linearly organized, this study hypothesizes that the interaction partners of loci with strong contact signals are likely to be neighbouring loci (Zhou et al). In other words, areas adjacent to strong Hi-C signals are expected to contain relevant structural information. As such, if sparsity exists in these areas, there is a high likelihood of it being induced by experimental limitations. Therefore, averaging out strong signals over a pre-defined neighbourhood could impute the data in a biologically sound manner, reinforcing weak signals that were lost during sequencing. An added benefit to averaging out reads across a predefined area is that random noise in the data would get smoothed out if surrounded by large regions of sparsity. Zhou et al implement this by using uniform smoothing, and weighing each neighbouring cell in their convolution window equally. However, for smoother imputation and greater spatial awareness, this investigation uses a Gaussian filter. This technique assigns a higher weight to closer neighbours, reducing the impact of distant noise and accounting for the linear organisation of the genome.

This investigation used the scipy.ndimage gaussian_filter method. For a given standard deviation σ, this method applies a filter of size $m \times m$ where $m = 6\sigma + 1$ to the contact map of size $n \times n$. This is because the Gaussian method assumes that the most relevant information falls within three standard deviations of the mean. The Gaussian kernel used is described by the equation:

$$F_{pq} = \frac{1}{2\Pi\sigma^2} e^{\frac{(p-i)^2 + (q-j)^2}{2\sigma^2}} \quad (1)$$

Where
(i, j) are the coordinates of the filter center,
$(p,\ q)$ are the coordinates of a neighboring bin in the filter window,
And σ is the standard deviation

This is applied to the matrix as follows:

$$B_{ij} = \sum_{p,q} F_{pq} A_{pq} \quad (2)$$

Where
B_{ij} is the filtered value at bin $(i,\ j)$
F_{pq} is the Gaussian kernel described above

and A_{pq} is the original value at neighbouring position $(p,\ q)$

Sigma was set to 1 and 5, to test how differences in the range of neighbouring values used to impute the data affected the final visualisation. These results are displayed in figure 4 and 5.

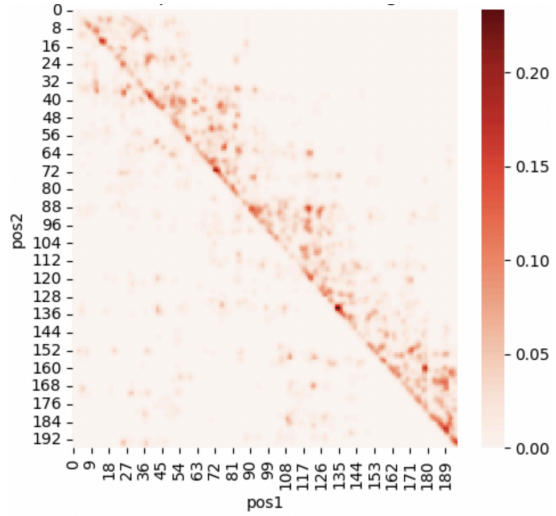

Figure 4: Contact map after Gaussian filtering with sigma = 1

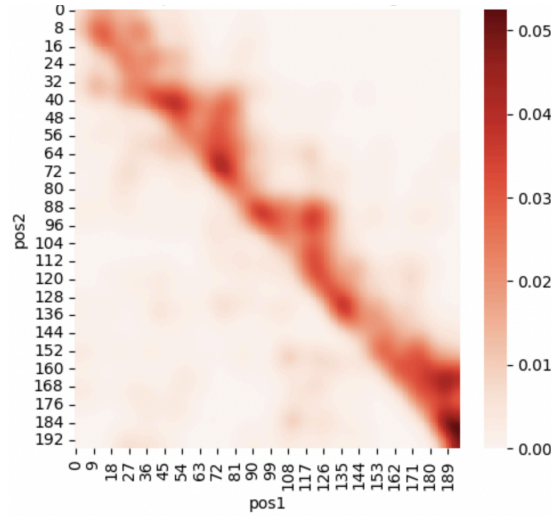

Figure 5: Contact map after Gaussian filtering with sigma = 5

As can be seen in figure 4, Gaussian filtering with sigma = 1 actually creates somewhat triangularly shaped structures in the contact map. These are typical of TADs, indicating regions of high genomic interaction. (Lajoie et al). This result is promising, but it remains to be seen whether this level of amplification creates a strong enough signal for correlations with cell-type or cell-cycle predictions to be identified. Gaussian filtering with sigma = 5 creates similar triangular shapes, though the signals are amplified much more significantly. This could be misleading.

A bilateral filter was also used as an additional layer of analysis. This technique smoothes noise while preserving edges, so as to not lose structural features like topologically-associated domains (TADs) and chromatin loops (Paris et al). Bilateral filtering weighs neighbours based on similarity in intensity in addition to proximity, while Gaussian filtering does only the latter. This allows for strong signals and potentially significant regions of genomic interaction to be preserved while noise is averaged out. The cv2 library's bilateralFilter method was used for this purpose. It is applied to the contact map as follows:

$$B_{ij} = \frac{1}{W_{ij}} \sum_{(p,q) \in N_{ij}} A_{pq} \cdot G_s(p - i, q - j) \cdot G_r(A_{pq} - A_{ij}) \quad (3)$$

Where

B_{ij} is the filtered intensity at bin (i, j)

A_{pq} is the original value at neighbour bin (p, q)

N_{ij} is the neighbourhood around bin (i, j)

$G_s(p - i, q - j)$ is the spatial Gaussian weight

$G_r(A_{pq} - A_{ij})$ is the intensity Gaussian weight

and W_{ij} is a normalization term

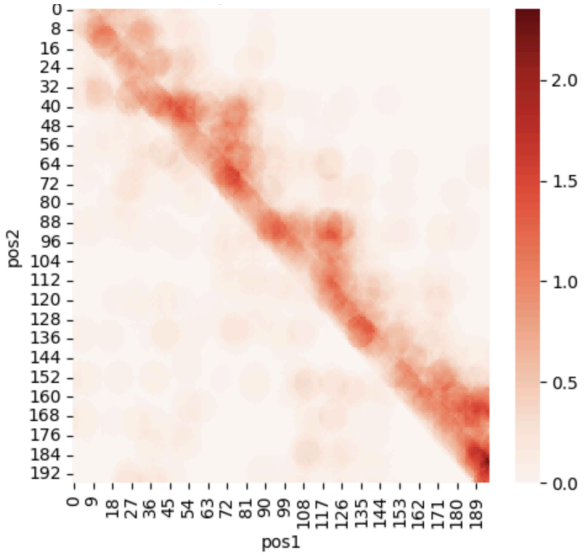

Figure 6: Contact map after Bilateral filtering

Evidenced by figure 6 bilateral filtering also produces indistinct triangular structures along the diagonal. However, an overemphasis is placed on the diagonal despite it being zero-ed out. This could mislead results as well. As such, Gaussian filtering with sigma = 1 seems to achieve the best outcome of the three techniques tried.

3.2.4 Eigenvalue Decomposition

Zhou et al propose a graph random walk with restart in addition to linear convolution to extract higher-order structures from the data. The motivation underlying both approaches is the same: to average information across a neighbourhood of bins to strengthen weak signals and smooth out noise. Convolution using Gaussian filters allows for information to be shared among linear neighbours, and graph random walks facilitate information sharing between network neighbours that are not necessarily adjacent to each other. The final product after information sharing and smoothing is then represented in a lower dimension to reveal topological structures of the chromosome for comparison and visualization. Thus, both local and global structures in the data were captured using this two-pronged approach (Zhou et al).

Graph random walk algorithms represent scHiC data as graphs where nodes are genomic bins, and edges are weighted by the genomic similarity of the bins they connect. Agents traverse from one node to the next based on which of the available edges carry the greatest weight. If edges are equally weighted, one is chosen at random. Thus, longer walks could connect genomic bins that are distantly genomically related (but related nonetheless), allowing for higher-order more complex structures to be realised. The amount of information from network neighbours used to impute each genomic bin increases proportionally with an increase in the length of the random walk.

In the same vein, this investigation uses Eigenvalue decomposition to try and detect more large-scale, global patterns in the data. This is done in the hopes of revealing structures like TADs and compartments that wouldn't be easily detectable in the noisy, sparse raw scHiC data. Eigen-decomposition reduces matrices to its orthogonal components, called eigenvectors, each with a corresponding eigenvalue. Higher eigenvalues correspond to components that capture smaller scale, local interactions that could contribute to noise. Comparatively, smaller eigenvalues correspond to components that capture more complex organisational structures, spatial features, and long range interactions between genomic bins. For the purposes of this investigation this is equivalent to longer graph random walks - interactions between genomic bins that aren't linear neighbours could be amplified revealing larger order patterns in the contact map matrices. Therefore, by averaging signals across larger network neighbourhoods, Eigen decomposition should reduce noise and amplify signals embedded in the data, providing further insight into larger-scale chromatin architecture.

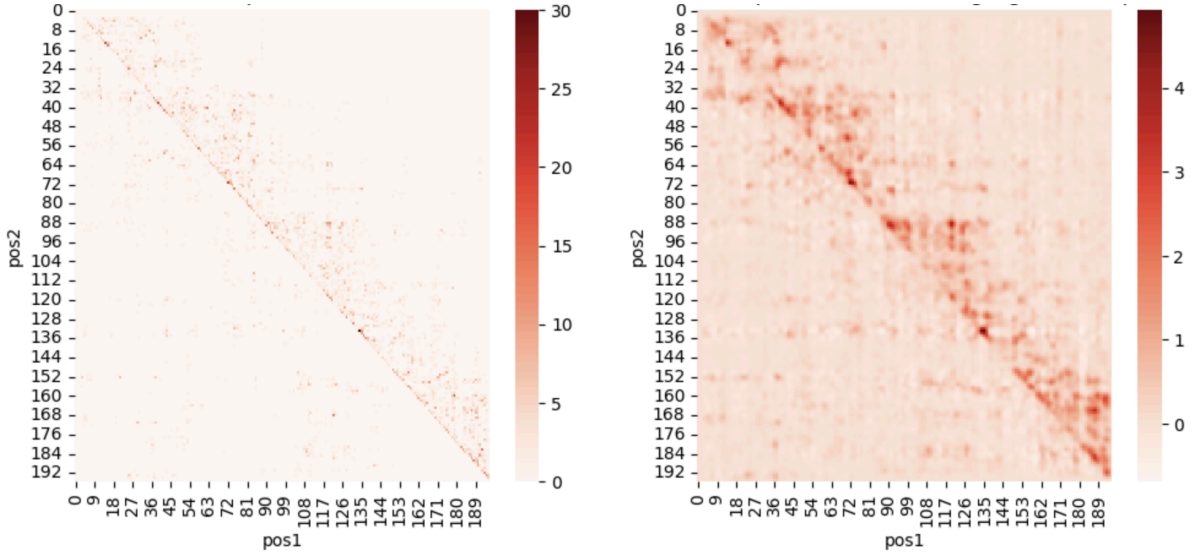

Figure 7: (Left) Contact map without any preprocessing (Right) Contact map after Gaussian filtering and Eigen decomposition

As can be seen in figure 7 above, Eigen decomposition combined with Gaussian filtering is able to identify even more distinct TADs than Guassian or bilateral filtering alone. However, this technique imputes the map to quite a large extent, as evidenced by the high frequency of bright signals. The map as a whole seems to contain more non-zero values than zeroes, evidenced by the large number of lighter-coloured areas, as compared to the extremely sparse map we began with. This could indicate that too much imputation has occurred, producing potentially misleading results. Furthermore, this heightened imputation has resulted in quite a noisy map, with no means to distinguish between relevant reads and false signals.

3.2.5 Singular Value Decomposition

Once the data has been imputed using Gaussian filtering, some form of analysis is required to visualise the larger scale features we were hoping to extract. Zhang et al use PARAFAC2, a tensor decomposition model, to break down the contact map data into a lower dimensional representation. Similar to eigenvalue decomposition, this preserves only the components of the matrix that contribute most to its variability, allowing for de-noising. This in turn could reveal larger order structures in the data, and later provide a predictive model with an axis for analysis (Zhang et al). PARAFAC2 is unique in its ability to decompose multi-dimensional tensors in different ways, thus providing multiple potential axes for analysis (Kolder and Bader). This type of data decomposition allows for data varying in one mode, or dimension, but consistent in another to be approximated. In the context of scHiC, this implies that shared structures across the genome can be modelled flexibly, without needing to fit a larger global pattern.

So far, this investigation has only dealt with scHiC data on a per-cell basis. As such, each matrix has only 2 dimensions, $bins \times bins$, and doesn't require multi-tensor decomposition methods.

A multi-dimensional tensor decomposition method like PARAFAC2 would yield the same result as Singular Value Decomposition (SVD), with less complexity. We have therefore used SVD to break the contact matrices into their components. This was implemented using numpy's linalg.svd method, where the deconstruction of contact map M was as follows:

$$M = U\Sigma V^{\top} \quad (4)$$

Where
U and V are orthogonal matrices of singular vectors capturing patterns along rows and columns respectively
and Σ is a diagonal matrix of weights given to each orthogonal vector based on its contribution to variability

The components were sorted in decreasing order of their weights, and then only the top few components that contributed to 90% of total variability were used for reconstruction. This 90% threshold can be adjusted for more or less approximation.

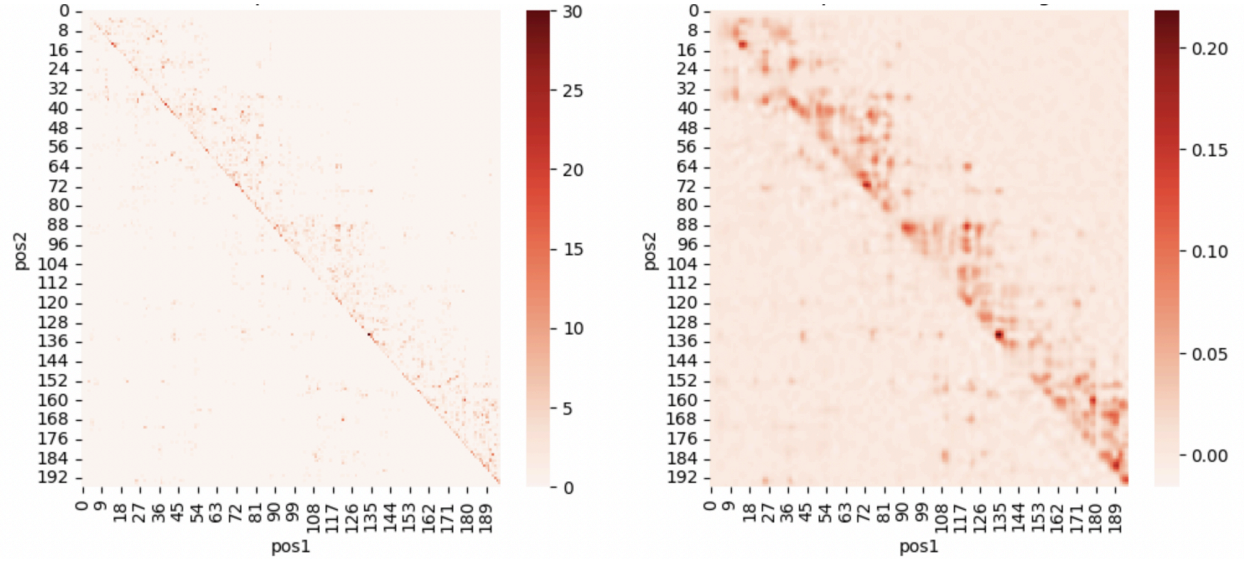

Figure 8: (Left) Contact map without any preprocessing (Right) Contact map after Gaussian filtering and Singular Value decomposition

This imputation technique seems to have achieved the best results for the data. Not only are the TADs quite distinctly visible, but the signals they are composed of have been selectively amplified. This is encouraging, because it follows biological theory - areas neighbouring strong signals are amplified and noisy reads in sparse areas are averaged out. The map produced after using this combination also is less noisy than the Eigenvalue decomposed map. As such, enough of the original sparsity is maintained to ensure the map remains interpretable, which works to our advantage. Once again however, it remains to be seen whether this level of imputation is

sufficient for cell-type and cell-cycle specific changes in structure to be observed and predicted by a machine learning model.

3.2.6 Unsupervised Clustering

Liu et al raise the issue that considering only one cell at a time while studying chromatin architecture focuses on DNA self-interaction, often overlooking larger scale features like loops and domains. In order to detect cell-to-cell variability in these features, additional layers of analysis are required. The study proposes its own imputation and embedding algorithm, scHiCluster, to analyse structures underlying the data in greater depth (Liu et al). It first emphasizes the importance of smoothing and using weighted networks among other techniques to enhance the signal-to-noise ratio of the data. They discover some success in detecting cell-cycle phase specific features, by testing existing methods like HiCRep, GenomeDISCO, and HiC-Spector. The last of the three also performs Laplacian decomposition, further reinforcing this investigation's use of SVD to visualise higher order features. HiCRep, GenomeDISCO, and HiC-Spector are used to calculate similarity measures between contact matrices, and these in turn are converted to genomic 'distances' between cells. Multidimensional Scaling (MDS) as proposed by Kruskal and Wish is used to represent the data in a lower dimension because of its ability to preserve pairwise distances (Kruskal). When used in conjunction with a denoising method like HiCRep, this has facilitated the joint embedding of cells from across multiple datasets. Once this embedding has been done, cells are ordered by cell cycle phase and clustered to test if phase distinctions can be observed. The study also develops a method to predict the cell-cycle 'angle' of each cell, to get a circular embedded graph mapping each cell to a datapoint showing its relative phase in the cycle.

Analysis into cell-to-cell variability, and identifying patterns across cells with different sequencing depths could not be done by this investigation through non-algorithmic methods alone. Liu et al's scHiCluster is incorporated into a more evolved algorithm called the Higashi algorithm. Therefore, to get a deeper understanding of underlying structures in the data we will now try to apply Higashi.

3.3 Higashi

Introduced by Zhang et al, the Higashi algorithm uses a hypergraph representation of cells to impute scHiC data, by capturing latent correlations between single cells and comparing features like TADs and compartments across cells. Its ability to simultaneously process multiple modes of data has resulted in embedding and imputation results that outperform its predecessors (Zhang et al).

Current models for imputation and embedding such as scHiCluster use techniques like graph random walks with restart, as previously discussed, to capture long-range intracellular correlations between genomic bins. However, this approach is relatively simplistic. Zhang et al

raise the issue of computational space - graph random walks for high resolution scHiC data require storage and calculations on dense contact matrices. Furthermore, this method falls short of reliably being able to compare structures like TADs and compartments (as measured by these long range correlations) between cells. Therefore, a more compact, efficient, and encompassing method is required to impute contact matrices and embed the data in a way that allows the detection of cell-to-cell variability in higher-order features.

The hypergraph proposed by Zhang et al begins with a set of nodes V and a set of hyperedges E. While each edge in a traditional graph (used for random walks as previously cited) connects only two nodes at a time, the edges in this hypergraph can connect multiple nodes together. Furthermore, while each node in the graphs used so far represents a genomic locus, nodes in the hypergraph can represent cells or genomic loci. Thus, both nodes and edges have attributes such as *node type* (locus or cell) for nodes and *genomic weight* for edges. This creates a representation of scHiC data that places regions of higher genetic similarity closer together, and is able to show bin-bin interactions through weighted edges based on the probability of that interaction occurring.

The 'hyperedge problem' this approach attempts to solve is predicting the probability of hyperedges between a set of nodes $(v_1, v_2, \ldots v_k)$. Here, as previously mentioned, hyperedges represent interactions between genomic loci. This is done three nodes at a time - one cell node representing the cell being looked at and two genomic bin nodes, between which an interaction (hyperedge) is being predicted. By looking at existing points of contact actually recorded in raw scHiC data, this embedding is able to predict 'missing' interactions between bins based on patterns observed among similar genomic loci. Furthermore, by incorporating cells as nodes in this graph, it gains the ability to look at cells that are genetically similar to the one being considered, to impute its contact map data.

This prediction task is executed using neural networks, with one fully connected layer. It takes in a (x_1, x_2, x_3) triplet as input, where x_1 represents the attributes of cell being considered, and x_2 and x_3 represent the attributes of the two genomic bins. One neural network NN1 is used to generate a static embedding s_i of each node v_i in the hypergraph, reflecting its relative importance. Here, each node is considered independent of the other nodes in the triplet. The other, NN2, generates a dynamic embedding d_i that considers the specific role of each node in the context of a particular hyperedge. The Hadamard power $d_i - s_i^{\circ 2}$, representing element-wise exponentiation of the matrix $d_i - s_i$, (Damm and Dietrich) is passed into a fully connected layer. This captures how the role of the node differs in the context of the triplet, as compared to

when it is considered independently. The output y_i of the fully connected layer then is aggregated over the three nodes $\sum_{i=1}^{3} y_i = \sum_{i=1}^{3} FC[(d_i - s_i)]$.

Separately, features such as the one hot encoding of the chromosome ID, batch ID and genomic bin distance are concatenated and passed through a multi layer perceptron. The output y_{ext} is further aggregated with the output of the fully connected layer, to integrate cell embedding information with metadata facts. As such, the final prediction of whether a hyperedge exists between the two genomic bins in the cell in the triplet is calculated as follows:

$$y_{pred} = y_{ext} + \sum_{i=1}^{3} FC[(d_i - s_i)] \quad (5)$$

The neural networks used to generate the static and dynamic embeddings, NN1 and NN2, use different loss functions for low and high sequencing depth datasets (fewer or higher reads per genomic bin). Binary cross-entropy loss is used for lower sequencing depths, because of the sparsity of the data. The loss function is used to predict whether a read exists or not, and helps distinguish between true absences and experimental errors. On the other hand, ranking loss is used for higher sequencing depths because with more meaningful data, stronger signals can be 'ranked' higher than weaker ones to make more intelligent imputation predictions.

Each cell's contact map is imputed with inferred interactions based on these predictions. By pooling information from neighboring cells to predict missing hyperedges, the model accounts for chromatin architecture in a broader context. Unlike methods like scHiCluster that rely on comparisons between cells, Higashi incorporates neighboring cells directly into the imputation process, improving the learned embeddings. If certain interactions are missing from a cell but are present in genetically similar neighbours, the hypergraph representation allows the model to infer their presence. This produces more informative contact maps, while still preserving the single-cell resolution of the original scHi-C data.

Zhang et al compare this method to the three most commonly used cell embedding techniques - HiCRep/MDS (Yang et al), scHiCluster (Liu et al) and LDA (El Hachem et al). Higashi consistently outperforms these benchmarks in its ability to create unsupervised embeddings that form accurate clusters based on cell type and cell state. It is robust to the dimension size of the visualised embeddings, and the choice of k while clustering the embedding results. It is able to clearly visualise variability in compartments and TADs across cells, allowing for cell-specific features to be identified and compared.

800 cells were compiled and imputed using the Higashi method. This number of samples was chosen based on the runtime feasibility of processing each scHiC file into a format that was compatible with the Higashi algorithm. The Higashi algorithm creates 23 imputed data files, one for each chromosome containing contact map data for 800 cells. Each imputed data file contains lists of meaningful (non-zero) signals in each cell, along with coordinates for where in each contact map these signals appeared. This information was used to create 800 (196, 196) imputed contact maps, which were stacked vertically to be used as input for the predictive models tested later.

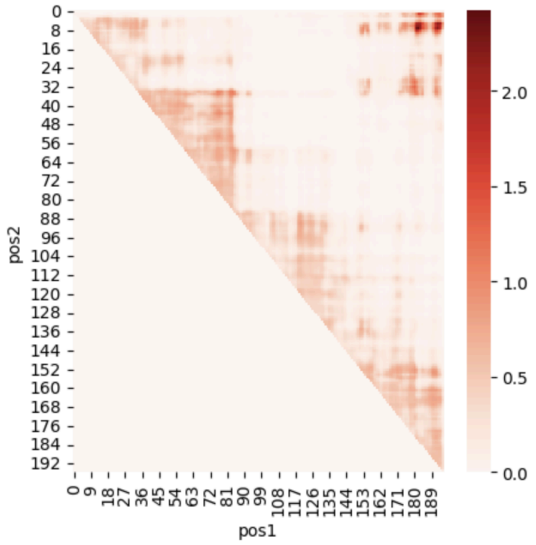

Figure 9: Contact map imputed using Higashi

As can be seen in the figure above, the Higashi algorithm is able to produce imputations of each contact map that are distinctly superior to the techniques previously used. TADs can be clearly observed and the noisiness of the data is significantly decreased.

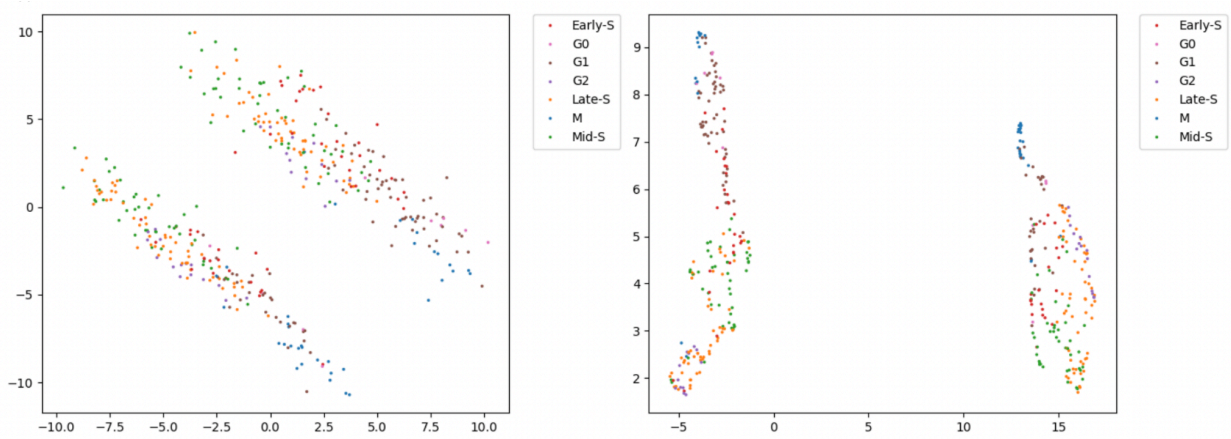

Figure 10: Embedding of imputed Liu et al dataset clustered by cell cycle phase

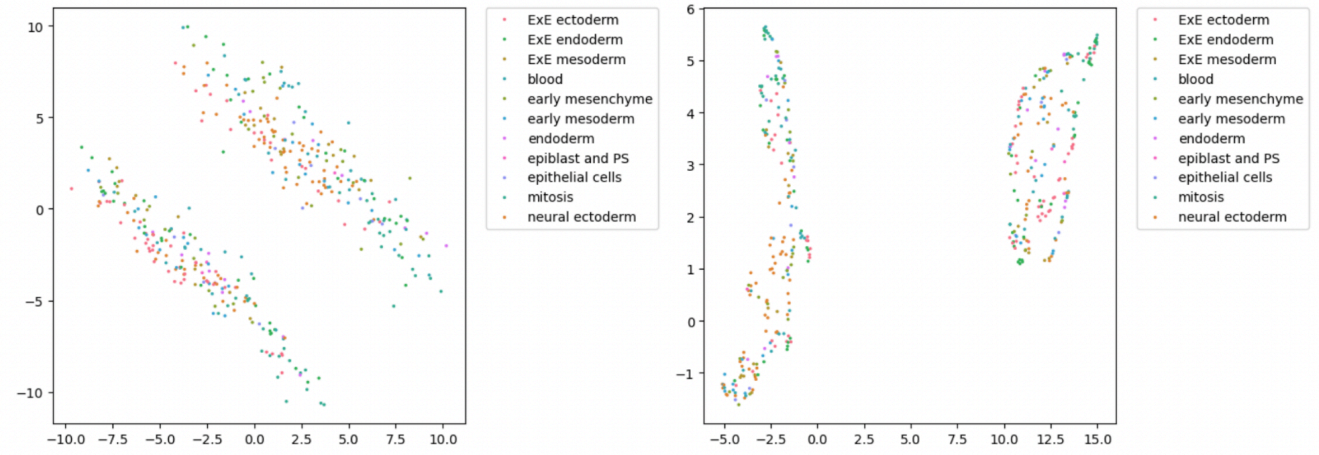

Figure 11: Embedding of imputed Liu et al dataset clustered by cell cycle type

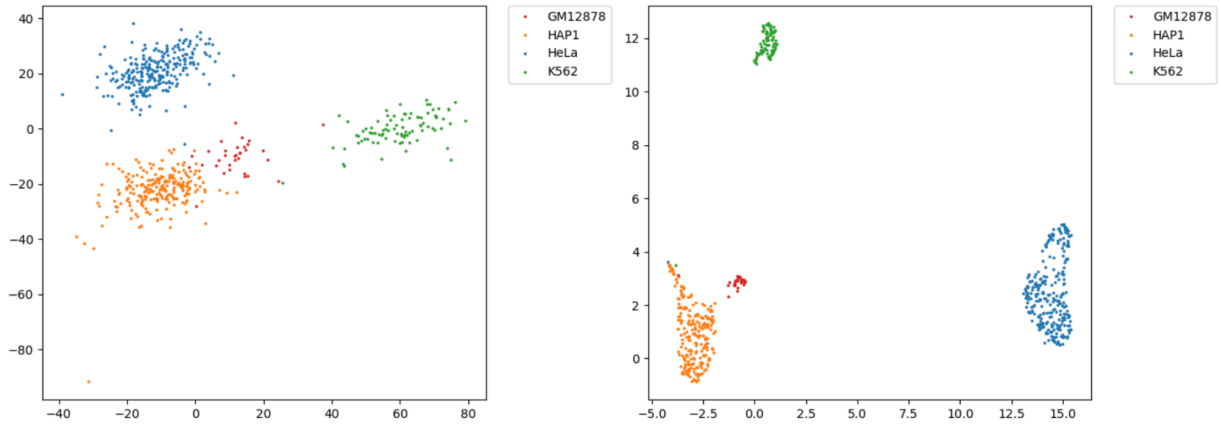

Figure 12: Embedding of imputed Ramani et al dataset clustered by cell cycle type

After imputation and embedding using the Higashi method, the Liu et al dataset didn't show any distinct clusters, neither when labelled using cell cycle phase nor cell type. Ramani et al on the other hand produced recognizable clusters based on cell type. This result could point to a confounding effect introduced by simultaneous cell type and cell cycle phase detection by the algorithm. Each contact map in the Liu et al dataset varies by cell cycle phase and cell type, while contact maps in the Ramani et al dataset vary only by cell type. Therefore, in the former case, cell type and cell cycle phase features might *both* be amplified by Higashi, thus confounding clustering. In comparison, amplifying and classifying data based on cell-type-specific features alone is a much better defined task, yielding clearer results.

3.4 Logistic Regression and Support Vector Machines

We now required a way to test the strength of the newly imputed signals, and see if meaningful cell cycle and cell type classifications could be made using them. In order to avoid introducing bias into our prediction process, or induce a model into learning information that was not truly

present in the underlying data, we began with a simple logistic regression model for cell type and cell cycle phase classification.

Cell type and cell cycle phase labels were extracted from the metadata file (Liu et al) and split into train and test sets along with the Higashi imputed dataset. We chose a 20% test 80% training split, based on common practice. The data was scaled such that local, short-range contact signals didn't dominate weaker long-range interactions, especially because logistic regression assumes that all features contribute to the cell cycle and cell type prediction label equally. Despite Higashi taking measures to temper the effect of differences in sequencing depth, standardizing the data before making predictions would help overcome this as well. K-fold cross validation with 5 folds was used to test the generalisability of the model, before evaluating its performance on unseen test data. This further step was taken to ensure that the model didn't overfit the Higashi imputed dataset, despite choosing a baseline logistic regression model for the same reason.

Two classifiers were trained in the same cross-validation loop - one to predict cell cycle labels and the other to predict cell type labels. A cross entropy loss function was used for both.

An L-BFGS (Limited-memory Broyden–Fletcher–Goldfarb–Shanno) optimizer was used in place of gradient descent, given the large number of features in the data. This allows the model to converge faster, and avoids the need to tune a learning rate. Standard BFGS uses a Hessian matrix to update model parameters instead of a learning rate as follows:

$$\theta_k = \theta_{k-1} - H_k \nabla L(\theta_{k-1}) \quad (6)$$

Where θ_t and θ_{t-1} are model parameters and $L(\theta_{t-1})$ represents the loss function. The Hessian matrix is calculated as follows:

$$H_k = V_{k-1}^{\top} H_{k-1} V_{k-1} + \rho_{k-1} s_{k-1} s_{k-1}^{\top} \quad (7)$$

Where
H_k is the updated Hessian matrix
H_{k-1} is the Hessian matrix at the previous iteration
$s_{k-1} = \theta_k - \theta_{k-1}$ or the change in parameters
$y_{k-1} = \nabla L(\theta_k) - \nabla L(\theta_{k-1})$ or the change in the gradient of the loss function
$\rho_{k-1} = \frac{1}{y_{k-1}^{\top} s_{k-1}}$, a normalization term

and $V_{k-1} = I - \rho_{k-1} s_{k-1} s_{k-1}^{\top}$ as an adjustment term to ensure that the curvature of the loss function is maintained

This allows for the model parameters to be updated in a manner that takes into account both how much they have already been changed and the change in the gradient of the loss function, allowing for faster convergence. The limited memory version of BFGS follows this in principle but instead of recalculating H_k at each step, it implicitly approximates it using the last m vectors of s_{k-1} and y_{k-1} in the interest of computational efficiency (Chen).

The linear regression model did not yield extremely high accuracy. For the Liu et al dataset with 11 cell types and 7 cell cycle phases, it yielded a 16.25% accuracy for cell type and 21.88% accuracy for cell cycle phase on the test data. This pointed to high bias in the model, reinforced by low validation accuracy while training. As such, we tried to use a support vector machine (SVM) model in the hopes of achieving higher accuracy than with a logistic regression.

Cell type and cell cycle phase labels were extracted from the same metadata file as for linear regression, and once again an 80-20 train-test split was used for the Higashi dataset with its corresponding labels. Scikit learn's SVC function was used to initialise the mode, with a radial basis function (RBF) kernel and a regularization parameter C of 1. This was done to capture non-linear relationships in the contact matrix input data, while allowing the model to generalize to unseen data. K-fold cross validation was used once again, this time with an SVM model using scikit learn's cross_val_score function.

SVC defaults to the one-versus-rest strategy using hinge loss as its loss function. It internally iterates through each class to predict the probability of a datapoint falling in that class or not. As such, it formulates class membership as a binary classification problem for each individual class. Data points in the training set are either in the class ($y_i = 1$) or they are not ($y_i = -1$)It then optimizes the problem:

$$min_{w,b} \frac{1}{2}||w||^2 + C \sum_{i=1}^{n} max(0, (1 - y_i f(x_i))) \quad (8)$$

Where

$f(x_i) = w^{\top} x_i + b$ is the equation of the decision boundary computed by the model

w is the weight vector

b is the bias

And C is the regulation parameter previously set to 1

Once again, the model performed poorly on the Higashi dataset yielding a 21.25% accuracy for cell type predictions and a 28.75% accuracy for cell cycle phase predictions on the test dataset. While these performance numbers are better than random chance - indicating that some patterns are being picked up by the model - they are far too say that any significant information can be picked up from the data using these models.

The low accuracy achieved by both the logistic regression and SVM models could potentially point to the inability of such high bias low complexity models to capture patterns in scHiC data. Even after imputing the data using the Higashi algorithm, either these models are too simple to capture complex structures in the contact maps, or the data is simply too sparse for a machine learning model to learn cell cycle phase or cell type information from it without feature manipulation or supplementation with external data.

3.5 Convolutional Neural Network

In order to test the hypothesis that high bias low complexity models like logistic regression or support vector machines are too simple to capture underlying information, this investigation uses a convolutional neural network for the same purpose. The idea behind this is that if the data truly does contain cell cycle and cell type-specific structures, a deep learning model could detect them and make accurate predictions when simpler models cannot.

The Liu et al was split into testing and training data using an 80-20 train-test split, to remain consistent with the logistic regression and SVM models used earlier. The Ramani et al data however was split into training and testing data using 70-30 split. This is because each Ramani et al contact map is (250, 250), as compared to Liu's (196, 196) contact maps, resulting in an extremely large input dataset. Any training set greater than 70% of the input data exhausted the computational resources available on Oscar, and therefore was not feasible for this exploration. This is a limitation of our model.

To prevent overfitting, the model contains only one encoder block and uses K fold cross validation to test generalizability. The encoder block itself contains one convolutional layer, one batch normalization layer, one ReLu layer and one Max Pooling layer. This is followed by a Dense layer with L2 regularization and a dropout layer to further prevent overfitting.

2D convolution is used in an attempt to capture spatial features in the data. Each contact map is treated as a (196, 196) or (250, 250) matrix and convolved using a (2,2) kernel filter with a 64 channel size, in an attempt to discover domains like TADs and compartments through differences in the compactness of the data. Batch normalization is then used to standardize the activation between layers, to enhance the detection of patterns and improve generalizability. It also allows for smooth model training by preventing gradient explosion. The ReLU layer introduces non-linearity, allowing this model to capture complex spatial structures like loops and

hierarchical patterns like cell-type specific features seen across contact maps. Last, the max pooling layer with a (2,2) pool size preserves only the strongest signals, filtering out noise and reducing computational load. It enables our model to capture key features without compromising generalizability.

The output of this is flattened and passed through a dense layer with ReLu activation to compute the final output probabilities. L2 activation is used here for better generalization. Finally, dropout is used as a final measure to prevent overfitting, and to truly learn the underlying patterns in the data as opposed to constructing them. A 20% dropout rate was selected for the Liu et al dataset, in light of the other regularization methods used, to prevent underfitting. However, this was not feasible for the Ramani et al dataset because the size of the data caused computational resources on Oscar to be exhausted for any dropout rate less than 50%. This is another limitation of our model.

This prediction task is not entirely clear cut, especially in the case of cell cycle phase. Chromatin architecture and gene expression don't vary much in cells transitioning from one phase of the cycle to another, and given its circular nature distinctions between phases are not always apparent. As such, a dense layer with softmax activation was used as the output layer. Using a probabilistic method allowed for a gradient in final predictions, accounting for differences in structure within a specific cell cycle phase or cell type. In the same vein, categorical cross entropy was used as a loss function for both prediction models. The Adam optimiser is well suited to data with sparse gradients, and therefore was chosen for this problem (Wei). 5 fold cross validation was used to test how the model would perform on unseen data. The results obtained matched those obtained for test data, indicating that the model wasn't overfitting the training samples.

In the hopes of increasing accuracy, contact maps were also pseudobulked together before being passed through the CNN. Cells of similar cell types are assumed to have similar features, and dissimilar noise signals. As such, pseudo bulking was done with the intention of enhancing the signal to noise ratio. Contact maps were filtered by cell type, and their reads were averaged across 8 cells. This was done for both imputed and raw scHiC data, to avoid errors in the imputed data from being amplified. The raw data was pseudobulked after Gaussian filtering and SVD.

As mentioned in the Higashi section of this report, the Ramani et al dataset could be imputed in a manner that enabled distinct cell-type specific clusters to form. The Liu et al dataset on the other hand showed scattered, indistinct distributions of points when labelled by cell type and by cell cycle phase. As such, we wanted to test the hypothesis that imputation using Higashi is sufficient to amplify features that vary by one characteristic, like cell-type or cell cycle phase, and is confused when used to identify features that vary along two categories. We do this by comparing

the performance of prediction models on both datasets - Ramani that only varies by cell type, and Liu that varies by both cell type and cell cycle phase. To this end, the Ramani dataset containing only cell type labels was sequenced using exactly the same methods as the Liu et al dataset, and the same models were used to make cell type predictions for both datasets. The results of this experiment would provide insight into whether two axes of variability reduced model performance.

4 Results

As an overview, this exploration occurred in two phases - data imputation and designing models to predict the cell-type and cell-cycle phase of imputed contact maps.

In order to quantify the efficacy of the preprocessing methods initially used, they were applied to raw scHiC contact maps and the result was passed through our convolutional neural network to see if accuracy predictions could be made. The higher the accuracy of the model for a specific preprocessing technique, the better its ability to extract meaningful information from sparse data. The CNN model was chosen over linear regression and SVM for this purpose because it yielded the best test performance on unseen data.

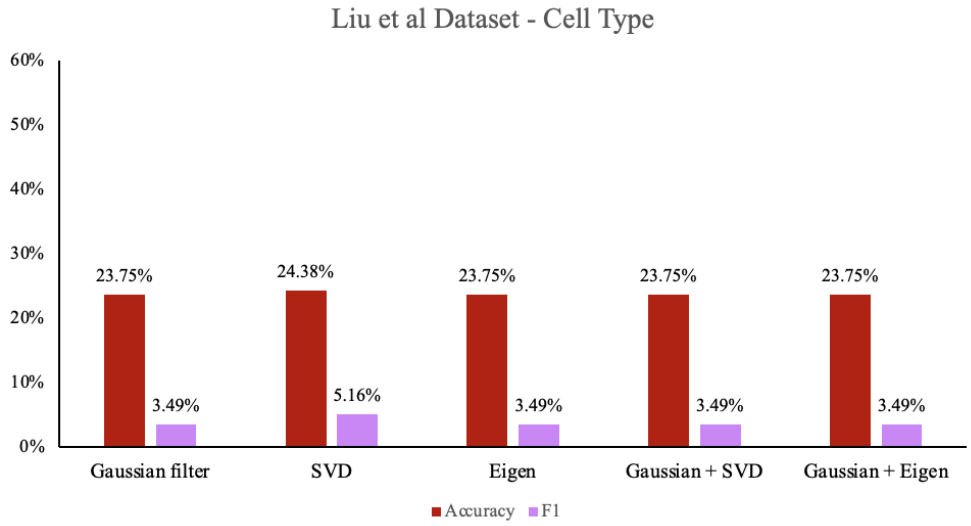

Figure 13: Accuracy and F1 scores for cell type prediction on the Liu et al dataset. Raw scHiC was data preprocessed in different ways before being used as input for the CNN. The horizontal axis shows which preprocessing technique was used, and the vertical axis quantifies the chosen metrics. Accuracy is shown in red and F1 scores in pink.

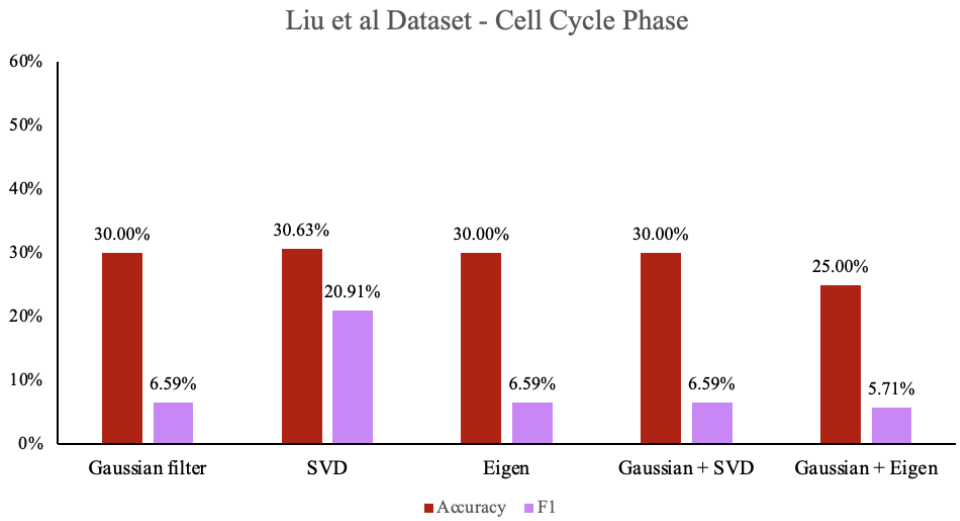

Figure 14: Accuracy and F1 scores for cell cycle phase prediction on the Liu et al dataset. Raw scHiC was data preprocessed in different ways before being used as input for the CNN. The horizontal axis shows which preprocessing technique was used, and the vertical axis quantifies the chosen metrics. Accuracy is shown in red and F1 scores in pink.

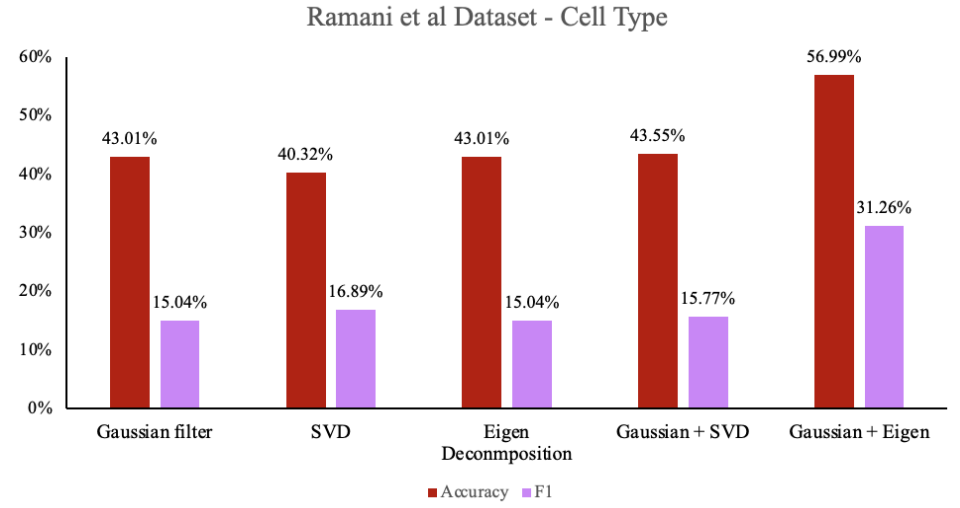

Figure 15: Accuracy and F1 scores for cell type prediction on the Ramani et al dataset. Raw scHiC was data preprocessed in different ways before being used as input for the CNN. The horizontal axis shows which preprocessing technique was used, and the vertical axis quantifies the chosen metrics. Accuracy is shown in red and F1 scores in pink.

Gaussian filtering and Eigenvalue decomposition alone yielded no performance gains over raw scHi-C input for either dataset. This could suggest that smoothing and dimensionality reduction individually are insufficient to improve the noise-to-signal ratio of the data enough to see an increase in accuracy. Interestingly, singular value decomposition alone had opposing effects on

the two datasets. Liu et al saw a slight increase in accuracy, while Ramani et al saw a decrease. However, both datasets had an increase in F1 score. This could suggest that the difference in how accuracy values changed might be due to the fact that the Ramani et al dataset is imbalanced. Another notable observation was that while Gaussian smoothing with SVD didn't improve the performance metrics for the Liu et al dataset over using raw data, it caused an increase in both performance metrics for the Ramani et al dataset. Gaussian smoothing with Eigenvalue decomposition causes a noticeable decrease in performance for Liu et al, while both accuracy and F1 score are highest for this technique with the Ramani et al dataset. As previously mentioned, the Ramani et al dataset does not contain actively dividing cells, and so the only axis of variability is cell type. The fact that Gaussian smoothing with SVD and Eigen reconstruction improves accuracy and F1 score here, could show that it is more useful for the task of predicting cell type, than for the task of predicting cell cycle phase. Furthermore, two axes of variability (cell type and cell cycle phase) could confound the amplification effects of this method, resulting in the poor model performance seen on the Liu et al dataset.

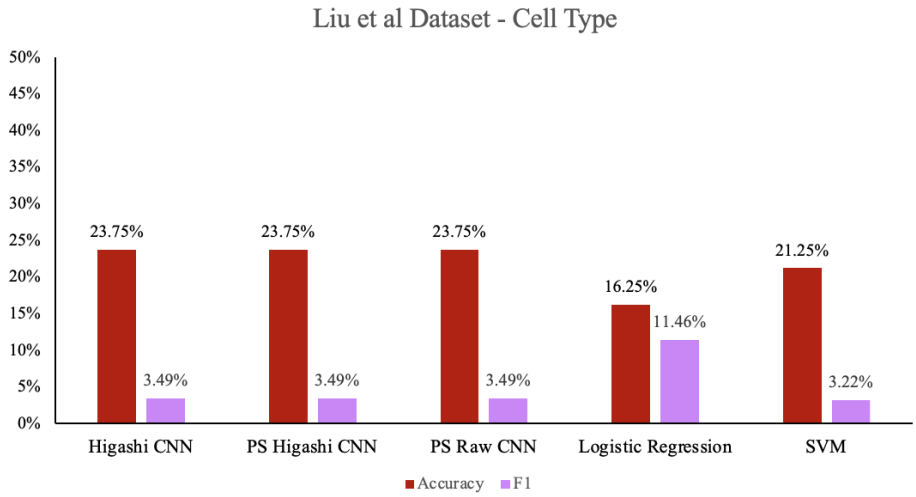

Figure 16: Accuracy and F1 scores for cell type prediction on the Liu et al dataset. The horizontal axis shows which model was used for prediction, and whether the data used was imputed using Higashi or raw. PS refers to pseudobulked data. The vertical axis quantifies the chosen metrics. Accuracy is shown in red and F1 scores in pink.

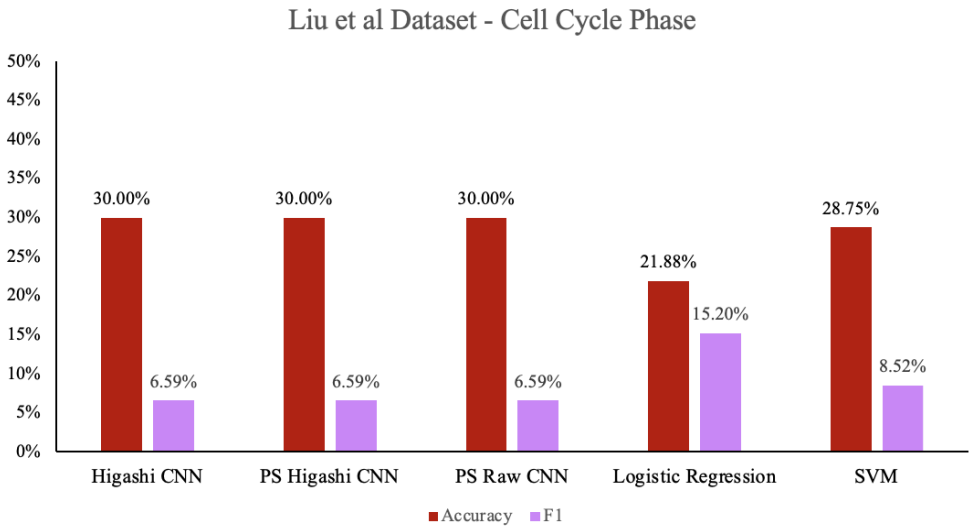

Figure 17: Accuracy and F1 scores for cell cycle phase prediction on the Liu et al dataset. The horizontal axis shows which model was used for prediction, and whether the data used was imputed using Higashi or raw. PS refers to pseudobulked data. The vertical axis quantifies the chosen metrics. Accuracy is shown in red and F1 scores in pink.

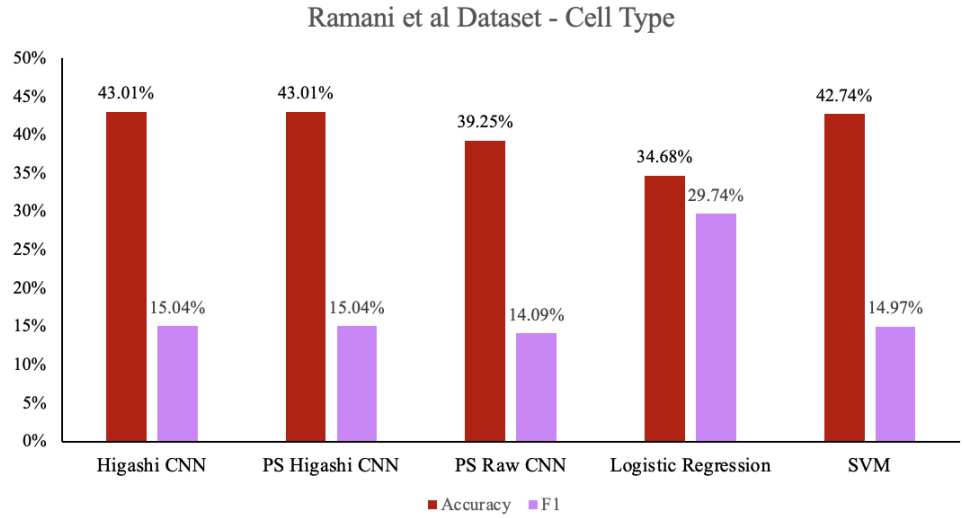

Figure 18: Accuracy and F1 scores for cell type prediction on the Ramani et al dataset. The horizontal axis shows which model was used for prediction, and whether the data used was imputed using Higashi or raw. PS refers to pseudobulked data. The vertical axis quantifies the chosen metrics. Accuracy is shown in red and F1 scores in pink.

Thus, simple preprocessing techniques alone were not able to achieve high performance metrics on either the Liu or the Ramani datasets. A more involved algorithm was required for imputation, that was able to facilitate both local and global information sharing. Therefore, the Higashi algorithm was used to impute the data before attempting to make any cell type or cell cycle phase predictions in it. Furthermore, instead of using a neural network immediately, this

investigation began with simpler models like logistic regression and support vector machines, to avoid overfitting the data.

Logistic regression was able to achieve better than average accuracy on the imputed data, but its overall performance was objectively poor. That said, it was able to achieve the highest F1 score of all the models tested. This could point to Liu and Ramani being imbalanced datasets, where simpler models like logistic regression are able to correctly predict rare or minority classes that are dismissed as noise by more complex models. Support vector machines are able to achieve better performance for both datasets, but only marginally so. These results suggest that extremely simple models may struggle to capture complex, non-linear relationships in the data, leading to high bias and underfitting. The convolutional neural network achieved the same results for imputed data, imputed pseudobulked data, and raw (non-imputed) pseudobulked data, with one exception - raw pseudobulked data from the Ramani et al dataset. This trend can be explained by the fact that the 800 cell dataset might be too small for the effects of pseudobulking to manifest in higher accuracy. The underperformance of the CNN on the pseudobulked, unimputed Ramani dataset could highlight the importance of imputation, and its conduciveness for predictions made using deep learning networks.

5 Discussion

In this investigation, we present a line of inquiry into whether cell type and cell cycle phase information can be learned from single-cell Hi-C contact matrices without any explicit feature extraction. It's an extremely challenging task due to two critical reasons. First, the single-cell Hi-C (scHi-C) data is extremely sparse and the structure features are confounded across two axes of variability: cell type and cell cycle phase.

The extreme sparsity of scHiC data severely constrains any ability of any downstream model to extract any meaningful representations from it. Specifically, bins encoded with 0 reads can either mean there is no contact between those two loci or the interaction was not observed due to the sequencing protocol biases. In order to reduce sparsity, we first explored existing methods such as, gaussian smoothing and dimensionality reduction to amplify the signal and remove noise. However, as shown in the results section, these methods couldn't sufficiently impute the data for downstream deep neural networks to identify cell-type or cell-cycle features. Next we tried Higashi, a hypergraph-based imputation approach to impute scHi-C matrices that is designed to enhance both local and global structural features. While we were able to correctly cluster the Ramani et al cells based on their cell-type, however, even Higashi struggled to learn any meaningful representations in the Liu et al dataset that contains both cell-type and cell-cycle variability. Lastly, we implemented simple machine learning and deep-learning based models to predict the cell-cycles and the cell-types from the scHi-C matrices preprocessed with a wide-variety of methods in order to test whether a supervised model can extract meaningful representations. Unfortunately, we found that these models also yielded poor performance across

these classification tasks. While CNNs achieved the best results, the scores were still not high enough to conclude that the model was able to extract meaningful representations or accurate cell-type and cell-cycle predictions.

As such, we come to the conclusion that neither CNNs nor simpler models can accurately extract meaningful representations from single-cell Hi-C contact matrices. We observe performance close to random, which suggests that even a deep learning-based model struggles to extract meaningful representations from such sparse data scHi-C. In this investigation, we tested a combination of techniques proposed by the past literature on diverse experiments to identify correct protocols to preprocess scHi-C data and how to extract representations from the extremely sparse scHi-C matrices. We conclude that existing approaches fall short of extracting meaningful representations either due to an incorrect choice of preprocessing techniques, or poor design of deep-learning based frameworks.

Higashi was similarly unable to impute the data enough for features to be reliably detected by any of the three models. Even though it was designed to capture both global and local structures in the data, these didn't appear distinctly or consistently enough across contact maps of a specific cell type or in a specific cell cycle phase. However, an interesting consideration raised by using this method was the difference in clustering results between data that varied only by cell type (Ramani et al) and data that varied by both cell type and cell cycle phase (Liu et al). When cell type was the only axis of variability, Higashi was able to impute cell-type specific features enough for distinct cell type specific clusters to form. Comparatively, the Liu et al dataset did not cluster well by cell type or cell cycle phase. DNA folding characteristics specific to certain cell types might be being conflated with those specific to cell cycle phases. As a result, they would not appear consistently across observations with a specific label, leading to low accuracy.
In order to validate this hypothesis, we ran each of the models on both sets of data to observe their relative performance. We also did so to determine which model would be most appropriate for this classification task. Unfortunately, once again, all the models tested yielded poor performance metrics as explained in the results section. Even though all the models performed better than random chance, the accuracy scores were not high enough to conclude that cell-type and cell-cycle predictions can reliably be made using this pipeline.

References

"Basics of Single-Cell Analysis with Bioconductor." *Chapter 2 Normalization*, bioconductor.org/books/3.14/OSCA.basic/normalization.html. Accessed 18 Apr. 2025.

Caldon, C. Elizabeth, et al. "Cell cycle proteins in epithelial cell differentiation: Implications for breast cancer." *Cell Cycle*, vol. 9, no. 10, 15 May 2010, pp. 1918–1928, https://doi.org/10.4161/cc.9.10.11474.

"Cell Cycle." *Genome.Gov*, National Human Genome Research Institute, 18 Apr. 2025, www.genome.gov/genetics-glossary/Cell-Cycle#:~:text=A%20cell%20spends%20most%20of,mitosis%2C%20and%20completes%20its%20division.

Chen, Yudong. "Lecture 23: Limited-Memory BFGS (L-BFGS)." *UW Madison CS*, spring 2023, pages.cs.wisc.edu/~yudongchen/cs726_sp23/Lecture_23_L-BFGS.pdf.

Cooper, Geoffrey M. "The Nucleus during Mitosis." *The Cell: A Molecular Approach. 2nd Edition.*, U.S. National Library of Medicine, 1 Jan. 1970, www.ncbi.nlm.nih.gov/books/NBK9890/#:~:text=This%20condensation%20is%20needed%20to,RNA%20synthesis%20stops%20during%20mitosis.

Damm, Tobias, and Nicolas Dietrich. "Hadamard Powers and Kernel Perceptrons." *Linear Algebra and Its Applications*, 28 Apr. 2023, pp. 1–1.

El Hachem, Elie-Julien, et al. "Latent dirichlet allocation for double clustering (LDA-DC): Discovering patients phenotypes and cell populations within a single Bayesian framework." *BMC Bioinformatics*, vol. 24, no. 1, 23 Feb. 2023, https://doi.org/10.1186/s12859-023-05177-4.

Galitsyna, Aleksandra A, and Mikhail S Gelfand. "Single-Cell hi-C data analysis: Safety in numbers." *Briefings in Bioinformatics*, vol. 22, no. 6, 18 Aug. 2021, https://doi.org/10.1093/bib/bbab316.

Galitsyna, Aleksandra, et al. *Extrusion Fountains Are Hallmarks of Chromosome Organization Emerging upon Zygotic Genome Activation*, 15 July 2023, https://doi.org/10.1101/2023.07.15.549120.

"Importance of Feature Scaling." *Scikit Learn*, scikit-learn.org/stable/auto_examples/preprocessing/plot_scaling_importance.html. Accessed 18 Apr. 2025.

Kolda, Tamara G., and Brett W. Bader. "Tensor decompositions and applications." *SIAM Review*, vol. 51, no. 3, 5 Aug. 2009, pp. 455–500, https://doi.org/10.1137/07070111x.

Koohy, Hashem, et al. "Chromatin accessibility data sets show bias due to sequence specificity of the DNase I enzyme." *PLoS ONE*, vol. 8, no. 7, 26 July 2013, https://doi.org/10.1371/journal.pone.0069853.

Kruskal, J. B. “Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis.” *Psychometrika*, vol. 29, no. 1, Mar. 1964, pp. 1–27, https://doi.org/10.1007/bf02289565.

Lajoie, Bryan R., et al. “The Hitchhiker’s Guide to hi-C analysis: Practical guidelines.” *Methods*, vol. 72, Jan. 2015, pp. 65–75, https://doi.org/10.1016/j.ymeth.2014.10.031.

Li, Xinjun, et al. “ScHiCTools: A computational toolbox for analyzing single-cell hi-C data.” *PLOS Computational Biology*, vol. 17, no. 5, 18 May 2021, https://doi.org/10.1371/journal.pcbi.1008978.

Lieberman-Aiden, Erez, et al. “Comprehensive mapping of long-range interactions reveals folding principles of the human genome.” *Science*, vol. 326, no. 5950, 9 Oct. 2009, pp. 289–293, https://doi.org/10.1126/science.1181369.

Lissanu Deribe, Yonathan, et al. “Mutations in the SWI/SNF complex induce a targetable dependence on oxidative phosphorylation in lung cancer.” *Nature Medicine*, vol. 24, no. 7, 11 June 2018, pp. 1047–1057, https://doi.org/10.1038/s41591-018-0019-5.

Liu, Jie, et al. “Unsupervised embedding of single-cell hi-C data.” *Bioinformatics*, vol. 34, no. 13, 27 June 2018, pp. i96–i104, https://doi.org/10.1093/bioinformatics/bty285.

Liu, Paul P. “Chromatin.” *Genome.Gov*, National Human Genome Research Institute, 18 Apr. 2025, www.genome.gov/genetics-glossary/Chromatin.

Liu, Weifang, et al. “Snaphic-G: Identifying long-range enhancer–promoter interactions from single-cell hi-C data via a global background model.” *Briefings in Bioinformatics*, vol. 25, no. 5, 25 July 2024, https://doi.org/10.1093/bib/bbae426.

Liu, Zhiyuan, et al. “Linking genome structures to functions by simultaneous single-cell hi-C and RNA-seq.” *Science*, vol. 380, no. 6649, 9 June 2023, pp. 1070–1076, https://doi.org/10.1126/science.adg3797.

Ma, Yiqin, et al. “How the cell cycle impacts chromatin architecture and influences cell fate.” *Frontiers in Genetics*, vol. 6, 3 Feb. 2015, https://doi.org/10.3389/fgene.2015.00019.

McArthur, Evonne, and John A. Capra. “Topologically associating domain boundaries that are stable across diverse cell types are evolutionarily constrained and enriched for heritability.” *The American Journal of Human Genetics*, vol. 108, no. 2, Feb. 2021, pp. 269–283, https://doi.org/10.1016/j.ajhg.2021.01.001.

Meyer, Clifford A., and X. Shirley Liu. “Identifying and mitigating bias in next-generation sequencing methods for chromatin biology.” *Nature Reviews Genetics*, vol. 15, no. 11, 16 Sept. 2014, pp. 709–721, https://doi.org/10.1038/nrg3788.

Nagano, Takashi, et al. “Single-Cell hi-C reveals cell-to-cell variability in chromosome structure.” *Nature*, vol. 502, no. 7469, 25 Sept. 2013, pp. 59–64, https://doi.org/10.1038/nature12593.

Naumova, Natalia, et al. “Organization of the mitotic chromosome.” *Science*, vol. 342, no. 6161, 22 Nov. 2013, pp. 948–953, https://doi.org/10.1126/science.1236083.

Paris, Sylvain, et al. “Bilateral filtering: Theory and applications.” *Foundations and Trends® in Computer Graphics and Vision*, vol. 4, no. 1, 2008, pp. 1–75, https://doi.org/10.1561/0600000020.

Raj, Arjun, et al. “Imaging individual mrna molecules using multiple singly labeled probes.” *Nature Methods*, vol. 5, no. 10, 21 Sept. 2008, pp. 877–879, https://doi.org/10.1038/nmeth.1253.

Ramani, Vijay, et al. “Massively multiplex single-cell hi-C.” *Nature Methods*, vol. 14, no. 3, 30 Jan. 2017, pp. 263–266, https://doi.org/10.1038/nmeth.4155.

Setty, Manu, et al. “Wishbone identifies bifurcating developmental trajectories from single-cell data.” *Nature Biotechnology*, vol. 34, no. 6, 2 May 2016, pp. 637–645, https://doi.org/10.1038/nbt.3569.

Tan, Jimin, et al. “Cell-type-specific prediction of 3D chromatin organization enables high-throughput in silico genetic screening.” *Nature Biotechnology*, vol. 41, no. 8, 9 Jan. 2023, pp. 1140–1150, https://doi.org/10.1038/s41587-022-01612-8.

Tan, Jimin, et al. “Cell-type-specific prediction of 3D chromatin organization enables high-throughput in silico genetic screening.” *Nature Biotechnology*, vol. 41, no. 8, 9 Jan. 2023, pp. 1140–1150, https://doi.org/10.1038/s41587-022-01612-8.

Van de Sande, Bram, et al. “Applications of single-cell RNA sequencing in drug discovery and development.” *Nature Reviews Drug Discovery*, vol. 22, no. 6, 28 Apr. 2023, pp. 496–520, https://doi.org/10.1038/s41573-023-00688-4.

Wang, Yanli, and Jianlin Cheng. “Hicdiff: Single-cell hi-C data denoising with Diffusion Models.” *Briefings in Bioinformatics*, vol. 25, no. 4, 23 May 2024, https://doi.org/10.1093/bib/bbae279.

Wei, Dagang. “Demystifying the Adam Optimizer in Machine Learning.” *Medium*, Medium, 30 Jan. 2024, medium.com/@weidagang/demystifying-the-adam-optimizer-in-machine-learning-4401d162cb9e#:~:text=Why%20Use%20Adam%3F,Stochastic%20Gradient%20Descent%20(SGD).

Wu, Yingfu, et al. “Schicyclepred: A deep learning framework for predicting cell cycle phases from single-cell hi-C data using multi-scale interaction information.” *Communications Biology*, vol. 7, no. 1, 31 July 2024, https://doi.org/10.1038/s42003-024-06626-3.

Yang, Tao, et al. “HICREP: Assessing the reproducibility of hi-C data using a stratum-adjusted correlation coefficient.” *Genome Research*, vol. 27, no. 11, 30 Aug. 2017, pp. 1939–1949, https://doi.org/10.1101/gr.220640.117.

Ye, Yusen, et al. “Circular trajectory reconstruction uncovers cell‐cycle progression and Regulatory Dynamics from single‐cell hi‐c maps.” *Advanced Science*, vol. 6, no. 23, 30 Sept. 2019, https://doi.org/10.1002/advs.201900986.

Zhang, Ruochi, et al. “Multiscale and integrative single-cell hi-C analysis with Higashi.” *Nature Biotechnology*, vol. 40, no. 2, 11 Oct. 2021, pp. 254–261, https://doi.org/10.1038/s41587-021-01034-y.

Zhang, Ruochi, et al. “Ultrafast and interpretable single-cell 3D genome analysis with fast-higashi.” *Cell Systems*, vol. 13, no. 10, Oct. 2022, https://doi.org/10.1016/j.cels.2022.09.004.

Zheng, Ye, et al. “Normalization and de-noising of single-cell hi-C data with BandNorm and SCVI-3D.” *Genome Biology*, vol. 23, no. 1, 17 Oct. 2022, https://doi.org/10.1186/s13059-022-02774-z.

Zhou, Jingtian, et al. “Robust single-cell hi-C clustering by convolution- and random-walk–based imputation.” *Proceedings of the National Academy of Sciences*, vol. 116, no. 28, 24 June 2019, pp. 14011–14018, https://doi.org/10.1073/pnas.1901423116.